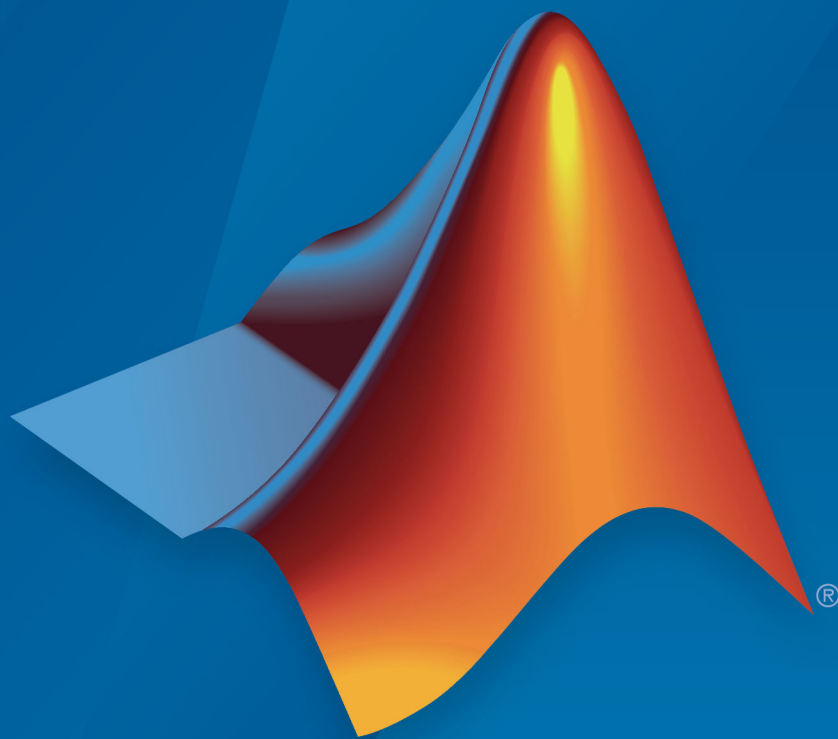**Simscape™ Driveline™**

Reference

MATLAB®&SIMULINK®

MathWorks®

How to Contact MathWorks

Latest news: www.mathworks.com

Sales and services: www.mathworks.com/sales_and_services

User community: www.mathworks.com/matlabcentral

Technical support: www.mathworks.com/support/contact_us

Phone: 508-647-7000

The MathWorks, Inc.
1 Apple Hill Drive
Natick, MA 01760-2098

Simscape™ Driveline™ Reference

Contents

Blocks — Alphabetical List

1

Technical Conventions

A

Parameter Dependencies

B

Bibliography

C

Blocks — Alphabetical List

4-Speed CR-CR

Clutch schedule for a four-speed carrier ring-carrier ring transmission
Library: Simscape / Driveline / Transmissions

Description

The 4-Speed CR-CR block consists of two planetary gear sets and five disk friction clutches. The follower shaft connects to the planet gear carrier of the output planetary gear and the ring gear of the input planetary gear. The clutches are configured to implement four ratios, one reverse ratio, and neutral. The reverse clutch and two of the clutches determine which gears the base shaft connects to. The other two clutches act as brakes, grounding various gears of the planetary sets to the transmission housing.

This diagram shows a four-speed carrier ring-carrier ring transmission. The labels for the gear components are superimposed on the input and output gears. The table lists the gear and clutch components that are labeled in the diagram.
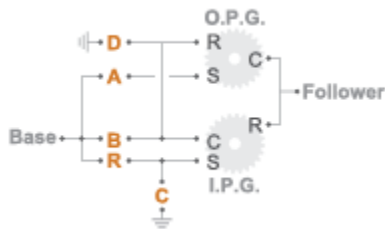
Label	Component
I.P.G.	Input planetary gear
O.P.G.	Output planetary gear
R	Ring gear

Label	Component
C	Planet gear carrier
S	Sun gear
R	Reverse clutch
A–B	Forward clutches that control the power flow path
C–D	Forward, braking clutches

Drive Ratios, Clutch Schedule, and Power Flow

The drive ratio between the transmission input and output shafts follows from the elementary gear ratios specified for the gear blocks. The elementary gear ratios are

$$g_1 = \frac{N_{RI}}{N_{SI}},$$

and

$$g_2 = \frac{N_{RO}}{N_{SO}},$$

where:

- N_{RI} is the number of teeth in the planetary ring gear on the input shaft side.
- N_{SI} is the number of teeth in the planetary sun gear on the input shaft side.
- N_{RO} is the number of teeth in the planetary ring gear on the output shaft side.
- N_{SO} is the number of teeth in the planetary sun gear on the output shaft side.

The table shows the clutch schedule, drive-ratio expressions, drive-ratio default values, and the power-flow diagrams for each gear of the 4-Speed CR-CR block.

The letters in the clutch schedule columns denote the brakes and clutches. A value of 1 denotes a locked state and a value of 0 an unlocked state. The clutch schedule generates these signals based on the Gear port input signal. The signals are scaled through a Gain block and used as actuation inputs in the clutch blocks.

The power-flow diagrams show the power flow paths between input and output shafts for each gear setting. Power flow is shown in orange. Connections to the transmission housing (a mechanical ground) are shown in black.

Gear	Clutch Schedule					Drive Ratio Equation	Default Ratio	Power Flow
	A	**B**	**C**	**D**	**R**			
4	0	**1**	**1**	0	0	$\dfrac{g_1}{1+g_1}$	0.67	
3	**1**	**1**	0	0	0	1	1	
2	**1**	0	**1**	0	0	$\dfrac{g_1+g_2}{1+g_1}$	1.67	
1	**1**	0	0	**1**	0	g_2	3	

Gear	Clutch Schedule					Drive Ratio Equation	Default Ratio	Power Flow
R	0	0	0	**1**	**1**	$-g_1$	-2	

Ports

Input

Gear — Numerical input for the gear
scalar

Specify -1 for reverse, 0 for neutral, and 1–4 for the forward gears.

Data Types: single | double

Conserving

B — Base interface
mechanical rotational

Conserving port associated with the base shaft.

F — Follower interface
mechanical rotational

Conserving port associated with the follower shaft.

Parameters

`Initial gear` — Gear state at the beginning of simulation
`Neutral` (default) | `4` | `3` | `2` | `1` | `Reverse`

Sets the initial engagement or disengagement of the clutches using the clutch schedule. When the transmission is in neutral, there is no power flow from the base shaft to the follower shaft.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also
10-Speed | 4-Speed Ravigneaux | 6-Speed Lepelletier | 7-Speed Lepelletier | 8-Speed | 9-Speed

Introduced in R2015a

4-Speed Ravigneaux

Clutch schedule for a four-speed Ravigneaux transmission
Library: Simscape / Driveline / Transmissions

Description

The 4-Speed Ravigneaux block consists of one Ravigneaux gear set and five clutches. The follower shaft connects to the ring gear of the Ravigneaux set. Three of the clutches determine the gears that the base shaft connects to for power transfer. The other two clutches serve as brakes, grounding various gears of the Ravigneaux set to the transmission housing.

This diagram shows a four-speed Ravigneaux gear transmission. The labels for the gear components are superimposed on the input and output gears. The table lists the gear and clutch components that are labeled in the diagram.
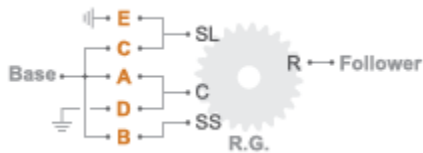
Label	Component
R.G.	Ravigneaux gear
R	Ring gear
C	Planet gear carrier
SL	Large sun gear
SS	Small sun gear
A–C	Forward clutches that control the power flow path

Label	Component
D–E	Forward, braking clutches

Drive Ratios, Clutch Schedule, and Power Flow

The drive ratio between the transmission input and output shafts follows from the elementary gear ratios specified for the gear blocks. The elementary gear ratios are

$$g_1 = \frac{N_R}{N_{SL}},$$

and

$$g_2 = \frac{N_R}{N_{SS}},$$

where:

- N_R is the number of teeth in the Ravigneaux ring gear.
- N_{SL} is the number of teeth in the Ravigneaux large sun gear.
- N_{SS} is the number of teeth in the Ravigneaux small sun gear.

The table shows the clutch schedule, drive-ratio expressions, drive-ratio default values, and the power-flow diagrams for each gear of the 4-Speed Ravigneaux block.

The letters in the clutch schedule columns denote the brakes and clutches. A value of 1 denotes a locked state and a value of 0 an unlocked state. The clutch schedule generates these signals based on the Gear port input signal. The signals are scaled through a Gain block and used as actuation inputs in the clutch blocks.

The power-flow diagrams show the power flow paths between input and output shafts for each gear setting. Power flow is shown in orange. Connections to the transmission housing (a mechanical ground) are shown in black.

Gear	Clutch Schedule					Drive Ratio Equation	Default Ratio	Power Flow
	A	B	C	D	R			

Gear	Clutch Schedule					Drive Ratio Equation	Default Ratio	Power Flow
4	**1**	0	0	0	**1**	$\dfrac{g_1}{1 + g_1}$	0.67	
3	**1**	**1**	0	0	0	1	1	
2	0	**1**	0	0	**1**	$\dfrac{g_1 + g_2}{1 + g_1}$	1.67	
1	0	**1**	0	**1**	0	g_2	3	
R	0	0	**1**	**1**	0	$-g_1$	-2	

Ports

Input

`Gear` — Numerical input for the gear
scalar

Specify -1 for reverse, 0 for neutral, and 1–4 for the forward gears.

Data Types: `single` | `double`

Conserving

B — Base interface
mechanical rotational

Conserving port associated with the base shaft.

F — Follower interface
mechanical rotational

Conserving port associated with the follower shaft.

Parameters

`Initial gear` — Gear state at the beginning of simulation
`Neutral` (default) | `4` | `3` | `2` | `1` | `Reverse`

Sets the initial engagement or disengagement of the clutches using the clutch schedule. When the transmission is in neutral, there is no power flow from the base shaft to the follower shaft.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also

10-Speed | 4-Speed CR-CR | 6-Speed Lepelletier | 7-Speed Lepelletier | 8-Speed | 9-Speed

Introduced in R2015a

6-Speed Lepelletier

Clutch schedule for a six-speed Lepelletier transmission
Library: Simscape / Driveline / Transmissions
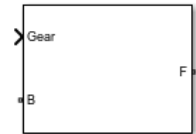
Description

The 6-Speed Lepelletier consists of one planetary gear set, one Ravigneaux gear set, and five disk friction clutches. The transmission base shaft connects to the ring gear of the planetary gear set. The follower shaft connects to the ring gear of the Ravigneaux gear set. Three of the clutches control the power flow paths between the planetary and Ravigneaux gear sets. The other two clutches serve as brakes, grounding various gears of the Ravigneaux set to the transmission housing.

This diagram shows a six-speed Lepelletier transmission. The labels for the gear components are superimposed on the input and output gears. The table lists the gear and clutch components that are labeled in the diagram.
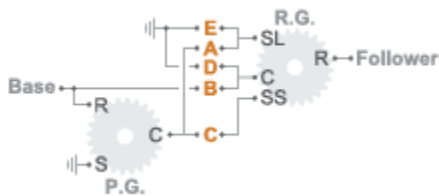
Label	Component
P.G.	Planetary gear
R.G.	Ravigneaux gear
R	Ring gear
C	Planet gear carrier

Label	Component
S	Sun gear
SL	Large sun gear
SS	Small sun gear
A–C	Forward clutches that control the power flow path
D–E	Forward, braking clutches

Drive Ratios, Clutch Schedule, and Power Flow

The drive ratio between the transmission input and output shafts follows from the elementary gear ratios specified for the gear blocks. The elementary gear ratios are

$$g_1 = \frac{N_{PR}}{N_{PS}},$$

$$g_2 = \frac{N_{RR}}{N_{RSL}},$$

and

$$g_3 = \frac{N_{RR}}{N_{RSS}},$$

where:

- N_{PR} is the number of teeth in the planetary ring gear.
- N_{PS} is the number of teeth in the planetary sun gear.
- N_{RR} is the number of teeth in the Ravigneaux ring gear.
- N_{RSL} is the number of teeth in the Ravigneaux large sun gear.
- N_{RRS} is the number of teeth in the Ravigneaux small sun gear.

The table shows the clutch schedule, drive-ratio expressions, drive-ratio default values, and the power-flow diagrams for each gear of the 6-Speed Lepelletier block.

The letters in the clutch schedule columns denote the brakes and clutches. A value of 1 denotes a locked state and a value of 0 an unlocked state. The clutch schedule generates these signals based on the Gear port input signal. The signals are scaled through a Gain block and used as actuation inputs in the clutch blocks.

The power-flow diagrams show the power flow paths between input and output shafts for each gear setting. Power flow is shown in orange. Connections to the transmission housing (a mechanical ground) are shown in black.

Gear	Clutch Schedule					Drive Ratio Equation	Default Ratio	Power Flow
	A	B	C	D	E			
6	0	1	0	0	1	$\dfrac{g_2}{1 + g_2}$	0.69	
5	1	1	0	0	0	$\dfrac{g_2}{g_2 + 1/(1 + g_1)}$	0.87	
4	0	1	1	0	0	$\dfrac{g_3(1 + g_1)}{g_3(1 + g_1) - 1}$	1.14	
3	1	0	1	0	0	$\dfrac{1 + g_1}{g_1}$	1.50	

Gear	Clutch Schedule					Drive Ratio Equation	Default Ratio	Power Flow
2	0	0	**1**	0	**1**	$\dfrac{(1 + g_1)(g_3 + g_2)}{g_1(1 + g_2)}$	2.31	
1	0	0	**1**	**1**	0	$\dfrac{g_3(1 + g_1)}{g_1}$	4.12	
R	**1**	0	0	**1**	0	$-\dfrac{g_2(1 + g_1)}{g_1}$	-3.36	

Ports

Input

Gear — Numerical input for the gear
scalar

Specify -1 for reverse, 0 for neutral, and 1–6 for the forward gears.

Data Types: `single` | `double`

Conserving

B — Base interface
mechanical rotational

Conserving port associated with the base shaft.

F — Follower interface
mechanical rotational

Conserving port associated with the follower shaft.

Parameters

`Initial gear` — Gear state at the beginning of simulation
`Neutral` (default) | `6` | `5` | `4` | `3` | `2` | `1` | `Reverse`

Sets the initial engagement or disengagement of the clutches using the clutch schedule. When the transmission is in neutral, there is no power flow from the base shaft to the follower shaft.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also
10-Speed | 4-Speed CR-CR | 4-Speed Ravigneaux | 7-Speed Lepelletier | 8-Speed | 9-Speed

Introduced in R2017a

7-Speed Lepelletier

Clutch schedule for a seven-speed Lepelletier transmission
Library: Simscape / Driveline / Transmissions
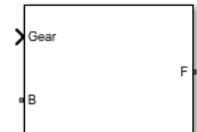
Description

The 7–Speed Lepelletier block consists of one planetary gear set, one Ravigneaux gear set, and six disk friction clutches. The transmission base shaft connects to the ring gear of the planetary gear set. The follower shaft connects to the ring gear of the Ravigneaux gear set. Three of the clutches control the power-flow paths between the planetary and Ravigneaux gear sets. The other three clutches serve as brakes, grounding various gears of the Ravigneaux set to the transmission housing.

This diagram shows a seven-speed Lepelletier transmission. The labels for the gear components are superimposed on the input and output gears. The table lists the gear and clutch components that are labeled in the diagram.
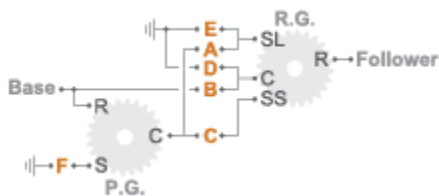
Label	Component
P.G.	Planetary gear
R.G.	Ravigneaux gear
R	Ring gear
C	Planet gear carrier

Label	Component
S	Sun gear
SL	Large sun gear
SS	Small sun gear
A–C	Forward clutches that control the power-flow path
D–F	Forward, braking clutches

Drive Ratios, Clutch Schedule, and Power Flow

The drive ratio between the transmission input and output shafts follows from the elementary gear ratios specified for the gear blocks. The elementary gear ratios are

$$g_1 = \frac{N_{PR}}{N_{PS}},$$

$$g_2 = \frac{N_{RR}}{N_{RSL}},$$

and

$$g_3 = \frac{N_{RR}}{N_{RSS}},$$

where:

- N_{PR} is the number of teeth in the planetary ring gear.
- N_{PS} is the number of teeth in the planetary sun gear.
- N_{RR} is the number of teeth in the Ravigneaux ring gear.
- N_{RSL} is the number of teeth in the Ravigneaux large sun gear.
- N_{RSS} is the number of teeth in the Ravigneaux small sun gear.

The table shows the clutch schedule, drive-ratio expressions, drive-ratio default values, and the power-flow diagrams for each gear of the 7-Speed Lepelletier block.

The letters in the clutch schedule columns denote the brakes and clutches. A value of 1 denotes a locked state and a value of 0 an unlocked state. The clutch schedule generates these signals based on the Gear port input signal. The signals are scaled through a Gain block and used as actuation inputs in the clutch blocks.

The power-flow diagrams show the power-flow paths between input and output shafts for each gear setting. Power flow is shown in orange. Connections to the transmission housing (a mechanical ground) are shown in black.

Gear	Clutch Schedule						Drive Ratio Equation	Default Ratio	Power Flow
	A	B	C	D	E	F			
7	0	**1**	0	0	**1**	**1**	$\dfrac{g_2}{1+g_2}$	0.70	
6	**1**	**1**	0	0	0	**1**	$\dfrac{g_3(1+g_1)}{g_2(1+g_1)+1}$	0.86	
5	**1**	**1**	**1**	0	0	0	1	1	
4	0	**1**	**1**	0	0	**1**	$\dfrac{g_3(1+g_1)}{g_3(1+g_1)-1}$	1.17	

Gear	Clutch Schedule						Drive Ratio Equation	Default Ratio	Power Flow
3	**1**	0	**1**	0	0	**1**	$\dfrac{1 + g_1}{g_1}$	1.63	
2	0	0	**1**	0	**1**	**1**	$\dfrac{(1 + g_1)(g_3 + g_2)}{g_1(1 + g_2)}$	2.43	
1	0	0	**1**	**1**	0	**1**	$\dfrac{g_3(1 + g_1)}{g_1}$	4.31	
R	**1**	0	0	**1**	0	**1**	$-\dfrac{g_2(1 + g_1)}{g_1}$	-3.82	

Ports

Input

Gear — Numerical input for the gear
scalar

Specify -1 for reverse, 0 for neutral, and 1–7 for the forward gears.

Data Types: `single` | `double`

Conserving

B — Base interface
mechanical rotational

Conserving port associated with the base shaft.

F — Follower interface
mechanical rotational

Conserving port associated with the follower shaft.

Parameters

Initial gear — Gear state at the beginning of simulation
`Neutral` (default) | `7` | `6` | `5` | `4` | `3` | `2` | `1` | `Reverse`

Sets the initial engagement or disengagement of the clutches using the clutch schedule. When the transmission is in neutral, there is no power flow from the base shaft to the follower shaft.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also

10-Speed | 4-Speed CR-CR | 4-Speed Ravigneaux | 6-Speed Lepelletier | 8-Speed | 9-Speed

Introduced in R2015a

8-Speed

Clutch schedule for an eight-speed transmission
Library: Simscape / Driveline / Transmissions
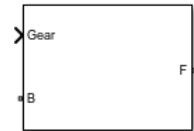
Description

The 8-Speed block consists of four planetary gear sets and five clutches. The base shaft connects to the:

- Planet gear carrier of the second planetary gear
- Other gears through three clutches that determine the power flow path.

The follower shaft connects to the planet gear carrier of the first and fourth planetary gears. The other two clutches serve as brakes, grounding various gears of planetary sets two and three to the transmission housing.

This diagram shows an eight-speed transmission. The labels for the gear components are superimposed on the input and output gears. The table lists the gear and clutch components that are labeled in the diagram.
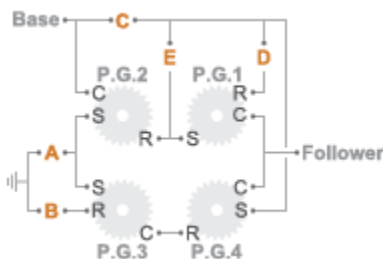
Label	Component
P.G.1–P.G.4	Planetary gears, 1–4

Label	Component
R	Ring gear
C	Planet gear carrier
S	Sun gear
C–E	Forward clutches that control the power flow path
A–B	Forward, braking clutches

Drive Ratios, Clutch Schedule, and Power Flow

The drive ratio between the transmission input and output shafts follows from the elementary gear ratios specified for the gear blocks. The elementary gear ratios are

$$g_x = \frac{N_{R_x}}{N_{S_x}},$$

where:

- N_{R_x} is the number of teeth in the planetary ring gear x, where x = 1, 2, and 3.
- N_{S_x} is the number of teeth in the planetary sun gear x, where x = 1, 2, and 3.

The table shows the clutch schedule, drive-ratio expressions, drive-ratio default values, and the power-flow diagrams for each gear of the 8-Speed block. The schedule and gear ratios are based on the manufacturer data for the GM 8L90 eight-speed automatic transmission.

The letters in the clutch schedule columns denote the brakes and clutches. A value of 1 denotes a locked state and a value of 0 an unlocked state. The clutch schedule generates these signals based on the Gear port input signal. The signals are scaled through a Gain block and used as actuation inputs in the clutch blocks.

The power-flow diagrams show the power flow paths between input and output shafts for each gear setting. Power flow is shown in orange. Connections to the transmission housing (a mechanical ground) are shown in black.

Gear	Clutch Schedule					Drive Ratio Equation	Default Ratio	Power Flow
	A	B	C	D	E			

Gear	Clutch Schedule					Drive Ratio Equation	Default Ratio	Power Flow
8	**1**	0	0	**1**	**1**	$\dfrac{g_2}{1 + g_2}$	0.68	
7	**1**	0	**1**	**1**	0	$\dfrac{g_2(1 + g_1)}{1 + g_2(1 + g_1)}$	0.85	
6	0	0	**1**	**1**	**1**	1	1	
5	0	**1**	**1**	**1**	$\dfrac{(1 + g_3)(1 + g_4)(g_2g_4)(1 + g_1)}{(1 + g_3)(g_2g_4)(1 + g_1)}$	1.26	

Gear	Clutch Schedule					Drive Ratio Equation	Default Ratio	Power Flow
4	0	**1**	0	**1**	**1**	$\dfrac{1 + g_2 + g_3}{1 + g_2}$	1.69	
3	0	**1**	**1**	0	**1**	$\dfrac{(1 + g_3)(1 + g_4)}{1 + g_3 + g_4}$	2.08	
2	**1**	**1**	0	0	**1**	$1 + g_1$	2.94	
1	**1**	**1**	**1**	0	0	$1 + g_4$	4.54	

Gear	Clutch Schedule					Drive Ratio Equation	Default Ratio	Power Flow
R	1	1	0	1	0	$-\dfrac{g_2[(1+g_1)-g_1(1+g4)]}{(1+g_2)}$	-3.85	

Ports

Input

Gear — Numerical input for the gear
scalar

Specify -1 for reverse, 0 for neutral, and 1–8 for the forward gears.

Data Types: single | double

Conserving

B — Base interface
mechanical rotational

Conserving port associated with the base shaft.

F — Follower interface
mechanical rotational

Conserving port associated with the follower shaft.

Parameters

`Initial gear` — Gear state at the beginning of simulation
`Neutral` (default) | `8` | `7` | `6` | `5` | `4` | `3` | `2` | `1` | `Reverse`

Sets the initial engagement or disengagement of the clutches using the clutch schedule. When the transmission is in neutral, there is no power flow from the base shaft to the follower shaft.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also
10-Speed | 4-Speed CR-CR | 4-Speed Ravigneaux | 6-Speed Lepelletier | 7-Speed Lepelletier | 9-Speed

Introduced in R2015a

9-Speed

Clutch schedule for a nine-speed transmission
Library: Simscape / Driveline / Transmissions
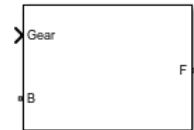
Description

The 9-Speed block consists of four planetary gear sets and six clutches. The follower shaft connects to the planet gear carrier of the fourth planetary gear. Three of the clutches determine the power flow path for the base shaft. The other three clutches serve as brakes, grounding various gears of the planetary sets to the transmission housing.

This diagram shows a nine-speed transmission. The labels for the gear components are superimposed on the input and output gears. The table lists the gear and clutch components that are labeled in the diagram.
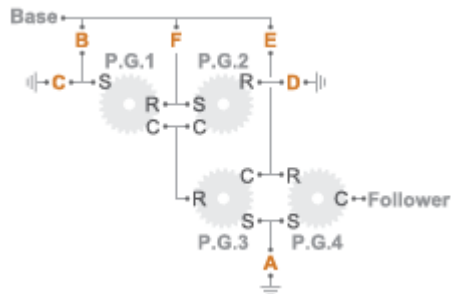
Label	Component
P.G.1–P.G.4	Planetary gears, 1–4
R	Ring gear
C	Planet gear carrier

Label	Component
S	Sun gear
B, E, F	Forward clutches that control the power flow path
A, C, D	Forward, braking clutches

Drive Ratios, Clutch Schedule, and Power Flow

The drive ratio between the transmission input and output shafts follows from the elementary gear ratios specified for the gear blocks. The elementary gear ratios are

$$g_x = \frac{N_{R_x}}{N_{S_x}},$$
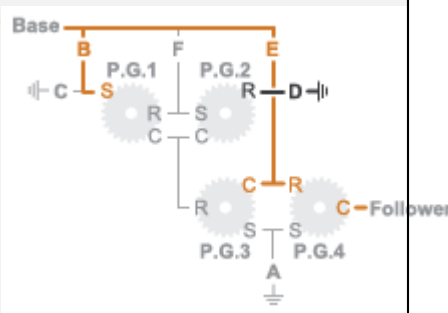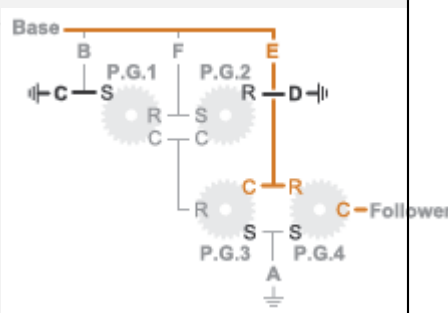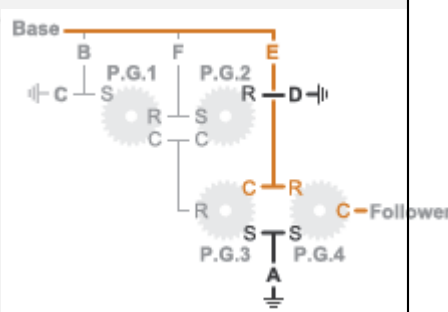
where:

- N_{R_x} is the number of teeth in the planetary ring gear x, where x = 1, 2, 3, and 4.
- N_{S_x} is the number of teeth in the planetary sun gear x, where x = 1, 2, 3, and 4

The table shows the clutch schedule, drive-ratio expressions, drive-ratio default values, and the plower-flow diagrams for each gear of the 9-Speed block. The schedule and gear ratios are based on the manufacturer data for the ZF 9HP nine-speed automatic transmission.

The letters in the clutch schedule columns denote the brakes and clutches. A value of 1 denotes a locked state and a value of 0 an unlocked state. The clutch schedule generates these signals based on the Gear port input signal. The signals are scaled through a Gain block and used as actuation inputs in the clutch blocks.

The plower-flow diagrams show the power flow paths between input and output shafts for each gear setting. Power flow is shown in orange. Connections to the transmission housing (a mechanical ground) are shown in black.

Gear	Clutch Schedule						Drive Ratio Equation	Default Ratio	Power Flow
	A	B	C	D	E	F			

Gear	Clutch Schedule						Drive Ratio Equation	Default Ratio	Power Flow
9	0	**1**	0	**1**	**1**	0	$\dfrac{1 + g_4}{1 + g_3 + g_4 - \left(\frac{g_3}{1 - g_1 g_2}\right)}$	0.4	
8	0	0	**1**	**1**	**1**	0	$\dfrac{1 + g_4}{1 + g_3 + g_4}$	0.5	
7	**1**	0	0	**1**	**1**	0	$\dfrac{1 + g_4}{1 + g_3 - \left(\frac{g_3}{1 + g_2}\right) + g_4}$	0.7	

Gear	Clutch Schedule						Drive Ratio Equation	Default Ratio	Power Flow
6	**1**	0	**1**	0	**1**	0	$\dfrac{1 + g_4}{1 + g_3 + g_4 - \left(\frac{g_1 g_3}{1 + g_1}\right)}$	0.8	
5	**1**	**1**	0	0	**1**	0	1	1	
4	**1**	0	0	0	**1**	**1**	$\dfrac{1 + g_4}{g_4}$	1.3	

Gear	Clutch Schedule						Drive Ratio Equation	Default Ratio	Power Flow
3	**1**	**1**	0	0	0	**1**	$\dfrac{(1 + g_3)(1 + g_4)}{g_3 g_4}$	1.9	
2	**1**	0	**1**	0	0	**1**	$\dfrac{(1 + g_1)(1 + g_3)(1 + g_4)}{g_1 g_3 g_4}$	2.8	
1	**1**	0	0	**1**	0	**1**	$\dfrac{(1 + g_2)(1 + g_3)(1 + g_4)}{g_3 g_4}$	4.7	

Gear	Clutch Schedule					Drive Ratio Equation	Default Ratio	Power Flow
R	0	**1**	0	**1**	0	$\underline{\mathbf{1}}\dfrac{(1 - g_1 g_2)(1 + g_3)(1 + g_4)}{g_3 g_4}$	-3.8	

Ports

Input

Gear — Numerical input for the gear
scalar

Specify -1 for reverse, 0 for neutral, and 1–9 for the forward gears.

Data Types: `single` | `double`

Conserving

B — Base interface
mechanical rotational

Conserving port associated with the base shaft.

F — Follower interface
mechanical rotational

Conserving port associated with the follower shaft.

Parameters

Initial gear — Gear state at the beginning of simulation
Neutral (default) | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | Reverse

Sets the initial engagement or disengagement of the clutches using the clutch schedule. When the transmission is in neutral, there is no power flow from the base shaft to the follower shaft.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also
10-Speed | 4-Speed CR-CR | 4-Speed Ravigneaux | 6-Speed Lepelletier | 7-Speed Lepelletier | 8-Speed

Introduced in R2015a

10-Speed

Clutch schedule for a 10-speed transmission
Library: Simscape / Driveline / Transmissions

Description

The 10-Speed transmission block consists of four planetary gear sets and six clutches. The follower shaft connects to the planet gear carrier of the fourth planetary gear. Four of the clutches determine the power flow path for the base shaft. The other two clutches serve as brakes, grounding various gears of the planetary sets to the transmission housing.

This diagram shows a 10-speed transmission. The labels for the gear components are superimposed on the input and output gears. The table lists the gear and clutch components that are labeled in the diagram. Power elements are shown in orange. Braking elements are shown in black.
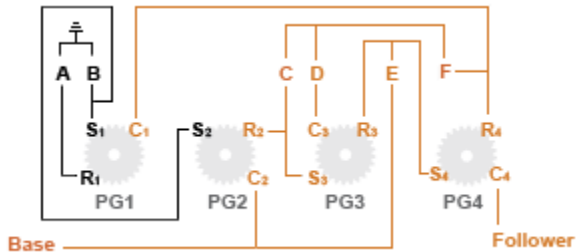
Label	Component
PG1–PG4	Planetary gears, 1–4
R_x	Ring gear of planetary gear x

Label	Component
C_x	Planet gear carrier of planetary gear x
S_x	Sun gear of planetary gear x
C, D, E, F	Clutches that control the power flow path
A, B	Braking clutches

Drive Ratios, Clutch Schedule, and Power Flow

The drive ratio between the transmission input and output shafts follows from the elementary gear ratios specified for the gear blocks. The elementary gear ratios are

$$g_x = \frac{N_{R_x}}{N_{S_x}},$$

where:

- N_{R_x} is the number of teeth in the planetary ring gear x, where $x = 1, 2, 3,$ and 4.
- N_{S_x} is the number of teeth in the planetary sun gear x, where $x = 1, 2, 3,$ and 4

The table shows the clutch schedule, drive-ratio expressions, drive-ratio default values, and the power-flow diagrams for each gear of the 10-Speed block. The schedule and gear ratios are based on the manufacturer data for a 10R80 10-speed automatic transmission.

The letters in the clutch schedule columns denote the brakes and clutches. A value of 1 denotes a locked state and a value of 0 an unlocked state. The clutch schedule generates these signals based on the Gear port input signal. The signals are scaled through a Gain block and used as actuation inputs in the clutch blocks.

The power-flow diagrams show the power flow paths between input and output shafts for each gear setting. Power flow is shown in orange. Connections to the transmission housing (a mechanical ground) are shown in black.

Gear	Clutch Schedule						Drive Ratio Equation	Default Ratio	Power Flow
	A	B	C	D	E	F			

Gear	Clutch Schedule						Drive Ratio Equation	Default Ratio	Power Flow
10	0	**1**	**1**	**1**	0	**1**	$\dfrac{g_2}{1 + g_2}$		
9	0	**1**	**1**	0	**1**	**1**	$\dfrac{g_2(1 + g_4)}{(g_2(1 + g_4)) + g_4}$		
8	0	**1**	0	**1**	**1**	**1**	$\dfrac{g_2(1 + g_3)(1 + g_4)}{(g_2(1 + g_3)(1 + g4)) + g_4}$		

Gear	Clutch Schedule						Drive Ratio Equation	Default Ratio	Power Flow
7	0	0	**1**	**1**	**1**	**1**	1		
6	**1**	0	0	**1**	**1**		$\dfrac{(1 + g_4)(1 + g_1 + g_2(1 + g_3}{(1 + g_4)(1 + g_2(1 + g_3) + g}$		
5	**1**	0	**1**	0	**1**	**1**	$\dfrac{(1 + g_1 + g_2)(1 + g_4)}{(1 + g_2)(1 + g_4) + g1}$		

1-39

Gear	Clutch Schedule						Drive Ratio Equation	Default Ratio	Power Flow
4	**1**	0	**1**	**1**	0	**1**	$\dfrac{1 + g_1 + g_2}{1 + g_2}$		
3	**1**	0	**1**	**1**	**1**	0	$\dfrac{(1 + g_1)(1 + g_4)}{1 + g_1 + g_4}$		
2	**1**	**1**	**1**	**1**	0	0	$\dfrac{g_2(1 - g_4)}{1 + g_2}$		

Gear	Clutch Schedule						Drive Ratio Equation	Default Ratio	Power Flow
1	**1**	**1**	0	**1**	**1**	0	$1 + g_4$		
R	**1**	**1**	0	**1**	0	**1**	$\dfrac{-g_2 g_3 (1+g_4)}{(1+g_2)}$		

Ports

Input

Gear — Numerical input for the gear
scalar

Specify -1 for reverse, 0 for neutral, and 1–10 for the forward gears.

Data Types: single | double

Conserving

B — Base interface
mechanical rotational

Conserving port associated with the base shaft.

F — Follower interface
mechanical rotational

Conserving port associated with the follower shaft.

Parameters

Initial gear — Gear state at the beginning of simulation
Neutral (default) | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | Reverse

Sets the initial engagement or disengagement of the clutches using the clutch schedule. When the transmission is in neutral, there is no power flow from the base shaft to the follower shaft.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also
4-Speed CR-CR | 4-Speed Ravigneaux | 6-Speed Lepelletier | 7-Speed Lepelletier | 8-Speed | 9-Speed

Introduced in R2018a

Band Brake

Frictional brake with flexible band wrapped around rotating drum
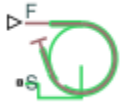
Library

Brakes & Detents/Rotational

Description

The block represents a frictional brake with a flexible band that wraps around the periphery of a rotating drum to produce a braking action. A positive actuating force causes the band to tighten around the rotating drum and it places the friction surfaces in contact. Viscous and contact friction between the drum and the flexible band surfaces cause the rotating drum to decelerate. Band brakes provide high braking torque at the cost of reduced braking precision in applications that include winch drums, chainsaws, go-karts, and mini-bikes. The model employs a simple parameterization with readily accessible brake geometry and friction parameters.
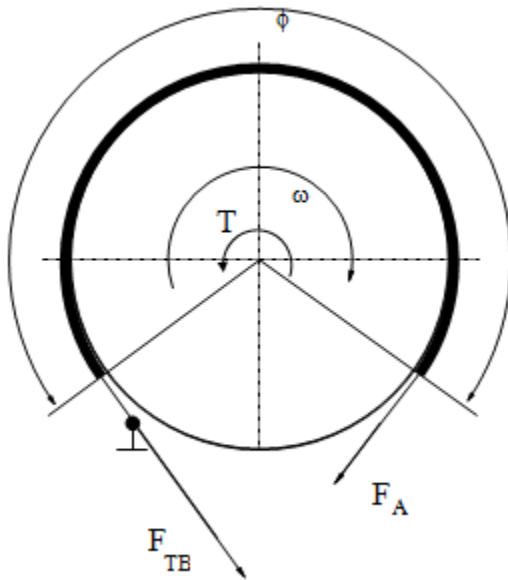
The following formula provides the braking torque as a function of the external brake actuation force that tightens the belt.

$$T = (F_{TB} - F_A) \cdot r_D + \mu_{visc} \cdot \omega,$$

In the formula, the parameters have the following meaning:

Parameter	Description
T	Braking torque
F_{TB}	Force acting on the tense branch of the band
F_A	External brake actuation force
r_D	Drum radius
μ_{visc}	Viscous friction coefficient
μ	Contact friction coefficient
ϕ	Wrap angle

Forces F_{TB} and F_A satisfy the relationship:

$$\frac{F_{TB}}{F_A} = e^{\mu\phi},$$

Replacing the relationship in the braking torque formula eliminates force F_{TB}:

$$T = F_A\left(e^{\mu\phi} - 1\right) \cdot r_D + \mu_{visc} \cdot \omega,$$

To avoid discontinuity at zero relative velocity, the model defines the actuation force F_S as a hyperbolic function:

$$F_A = F_{in} \cdot \tanh\left(\frac{4\omega}{\omega_{threshold}}\right),$$

In the previous formula, the parameters represent the following quantities:

Parameter	Description
F_{in}	Force input signal
$\omega_{threshold}$	Angular velocity threshold

Connection F is a physical signal port that represents the external tensioning force applied on the belt. Connection S is a conserving rotational port that represents the rotating drum shaft.

Variables

Use the **Variables** tab to set the priority and initial target values for the block variables before simulating. For more information, see "Set Priority and Initial Target for Block Variables" (Simscape).

Unlike block parameters, variables do not have conditional visibility. The **Variables** tab lists all the existing block variables. If a variable is not used in the set of equations corresponding to the selected block configuration, the values specified for this variable are ignored.

Assumptions and Limitations

The model does not account for actuator flow consumption.

Thermal Modeling

You can model the effects of heat flow and temperature change through an optional thermal conserving port. By default, the thermal port is hidden. To expose the thermal port, right-click the block in your model and, from the context menu, select **Simscape > Block choices**. Select a variant that includes a thermal port. Specify the associated thermal parameters for the component.

Ports

F

    Physical signal port that represents the belt tensioning force.

S

    Rotational conserving port that represents the rotating drum shaft.

H

    Thermal conserving port. The thermal port is optional and is hidden by default. To expose the port, select a variant that includes a thermal port.

Parameters

Geometry

Drum radius

    Radius of the drum contact surface. The parameter must be greater than zero. The default value is `150 mm`.

Wrap angle

    Contact angle between the flexible belt and the rotating drum. The parameter must be greater than zero. The default value is `270 deg`.

Friction

Viscous friction coefficient

Value of the viscous friction coefficient at the belt-drum contact surface. The parameter must be greater than or equal to zero. The default value is `.01 n*m/ (rad/s)`.

Temperature

Array of temperatures used to construct a 1-D temperature-efficiency lookup table. The array values must increase left to right. The default value is `[280.0, 300.0, 320.0]` K.

This parameter is visible only when you select a block variant that includes a thermal port.

Contact friction coefficient

Value of the Coulomb friction coefficient at the belt-drum contact surface. The value is greater than zero. Unless you select a block variant that includes a thermal port, the default value is `0.3`.

If you select a block variant that includes a thermal port, you specify this parameter as array. The array is the same size as the array for the **Temperature** parameter and the values increase left to right. The default value for the thermal variant is `[0.1, 0.05, 0.03]`.

Angular velocity threshold

Angular velocity at which the friction coefficient at the belt-drum contact surface practically reaches its steady-state value. The parameter must be greater than zero. The default value is `0.01 rad/s`.

Thermal Port

Thermal mass

Thermal energy required to change the component temperature by a single degree. The greater the thermal mass, the more resistant the component is to temperature change. This parameter is enabled only if you select a block variant that includes a thermal port. The default value is `50 kJ/K`.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also
Double-Shoe Brake | Loaded-Contact Rotational Friction | Rotational Detent

Topics
"Model Thermal Losses in Driveline Components"

Belt Drive

Power transmission system with taut belt connecting two pulleys
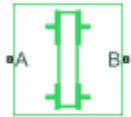
Library

Simscape / Driveline / Couplings & Drives

Description

The block represents a pair of pulleys connected with a flexible flat belt or V-shaped belt. The model accounts for friction between the flexible belt and the pulley periphery. If the friction force is not sufficient to drive the load, the model allows slip. The relationship between the tensions in the tight and loose branches conforms to the Euler equation. The model accounts for centrifugal loading in the flexible belt, pulley inertia, and bearing friction.

The schematics show the belt drive in its two possible configurations: open (left) and crossed (right). In the open belt configuration, both pulleys tend to rotate in the same direction, and the larger pulley tends to have a larger belt wrap angle. In the crossed belt configuration, the pulleys tend to rotate in opposite directions, and both pulleys have the same wrap angle.

Open Belt Configuration   Crossed Belt Configuration
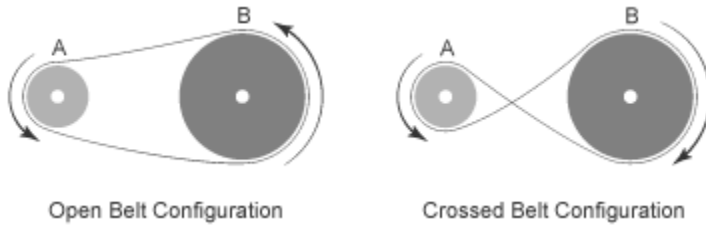
Belt Drive Schematics

In the open configuration, the wrap angle of the belt around each pulley is given by the expressions:

$$\theta_A = \pi + 2 * \sin^{-1} \frac{R_A - R_B}{C},$$

$$\theta_B = \pi - 2 * \sin^{-1} \frac{R_A - R_B}{C}$$

- θ_A is the wrap angle of pulley A.
- θ_B is the wrap angle of pulley B.
- R_A is the effective radius of pulley A.
- R_B is the effective radius of pulley B.
- C is the distance between the centers of pulley A and pulley B.

The following figure shows the wrap angles and the equation parameters used in the open-configuration calculations.
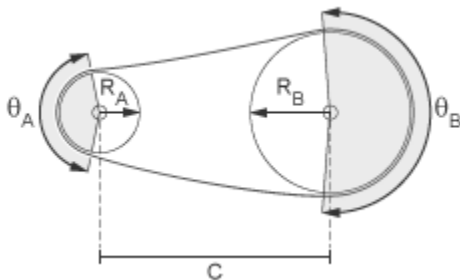
In the crossed configuration, the wrap angle of the belt around each pulley is given by the following expression. The two wrap angles are equal:

$$\theta_A = \theta_B = \pi + 2 * \sin^{-1} \frac{R_A + R_B}{C}$$

The figure shows the wrap angles calculated in the crossed-configuration expression.
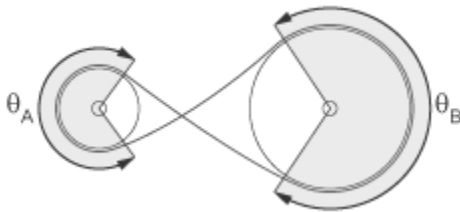
The Belt Drive block is a structural component based on the Simscape Driveline Belt Pulley block and Simscape Translational Spring and Translational Damper blocks. The Translational Spring and Translational Damper blocks model the compliance of the belt. For the equations governing the contact dynamics between the belt and the pulley, see the Belt Pulley block reference page. The following figure shows the block diagram upon which the Belt Drive block is based.
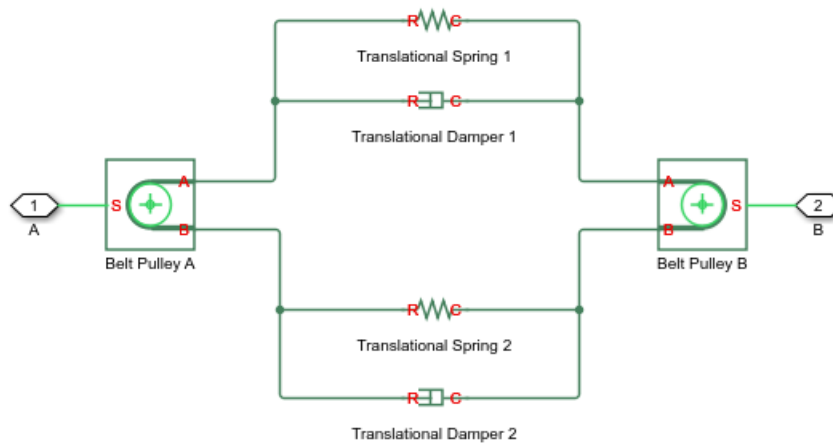
Assumptions and Limitations

- The pulleys do not translate.
- The friction coefficient and friction velocity threshold between the belt and each of the pulleys is the same.

Ports

Conserving

A — Rotational velocity
mechanical rotational

Rotational conserving port associated with the shaft of pulley A.

B — Rotational velocity
mechanical rotational

Rotational conserving port associated with the shaft of pulley B.

Parameters

Belt

Drive type

Select the type of belt drive that you want to model.

- Open belt — (Default)
- Crossed belt

For schematics of the two drive types, see "Belt Drive Schematics" on page 1-50

Belt type

Type of flexible element modeled.

- Flat belt — (Default)

- V-belt— Exposes the **V-belt sheave angle** and **Number of V-belts** parameters.

V-belt sheave angle

Sheave angle of the V-belt. Default is `30 deg`.

The **V-belt sheave angle** parameter is visible only when the **Belt type** parameter is `V-belt`.

Number of V-belts

Number of parallel V-belts. Default is `1`.

Non-integer values are rounded to the nearest integer. Increasing the number of belts increases the friction force, effective mass per unit length, and maximum allowable tension.

The **Number of V-belts** parameter is visible only when the **Belt type** parameter is `V-belt`.

Centrifugal force

Select whether to include the effects of centrifugal force. If included, centrifugal force saturates to approximately 90% of the value of the force on each belt end. Options are:

- `Do not model centrifugal force - Suitable for HIL simulation` — Default option which does not include centrifugal force contributions.
- `Model centrifugal force`— Exposes the **Belt mass per unit length** parameter for configuring the centrifugal force contribution.

Belt mass per unit length

Linear density of each belt. Default is `0.6 kg/m`.

The model uses this value to calculate the centrifugal loading. This parameter is enabled only if you select `Model centrifugal force` for the **Centrifugal force** parameter.

Longitudinal stiffness

The effective stiffness of the belt. Default is `1e+4 N/m`.

Longitudinal damping

Effective damping of the belt. Default is `1000 N/(m/s)`.

Pre-tension

Tension in the belt when the belt and pulleys are at rest. The value must be positive. The default value is `100 N`.

Maximum tension

Specifies whether the block throws an assertion when the belt tension is too large.

- `No maximum tension` — (Default)
- `Specify maximum tension`— Throws an assertion when the belt tension is too high. Exposes the **Belt maximum tension** parameter.

Belt maximum tension

Maximum allowable tension for each belt. Default is `1e+5 N`.

When the tension on either end of the belt meets or exceeds this value, the simulation ends with an error.

The **Belt maximum tension** parameter is visible only when the **Maximum tension** parameter is `Specify maximum tension`.

To specify the characteristics of pulley attached to port A, select the **Pulley A** tab. To specify the characteristics of pulley attached to port B, select the **Pulley B** tab.

Pulley A

Pulley radius

Radius of the pulley. Default is `0.15 m`.

Bearing viscous friction coefficient

Viscous friction associated with the bearings that hold the axis of the pulley. Default is `0 N*m/(rad/s)`.

Inertia

Specifies whether the block models rotational inertia of the pulley.

- `No inertia` — (Default)
- `Specify inertia and initial velocity`— Models rotational inertia. Exposes the **Pulley inertia** and **Pulley initial velocity** parameters.

Pulley inertia

Rotational inertia of the pulley. Default is `0.01 kg*m^2`.

The **Pulley inertia** parameter is visible only when the **Inertia** parameter is `Specify inertia and initial velocity`.

Pulley initial velocity

Initial rotational velocity of the pulley. Default is `0 rad/s`.

The **Pulley initial velocity** parameter is visible only when the **Inertia** parameter is `Specify inertia and initial velocity`.

To specify contact characteristics, select the **Contact** tab.

Pulley B

Pulley radius

Radius of the pulley. Default is `0.3 m`.

Bearing viscous friction coefficient

Viscous friction associated with the bearings that hold the axis of the pulley. Default is `0.001 N*m/(rad/s)`.

Inertia

Specifies whether the block models rotational inertia of the pulley.

- `No inertia` — (Default)
- `Specify inertia and initial velocity` — Models rotational inertia. Exposes the **Pulley inertia** and **Pulley initial velocity** parameters.

Pulley inertia

Rotational inertia of the pulley. Default is `0.03 kg*m^2`.

The **Pulley inertia** parameter is visible only when the **Inertia** parameter is `Specify inertia and initial velocity`.

Pulley initial velocity

Initial rotational velocity of the pulley. Default is `0 rad/s`.

The **Pulley initial velocity** parameter is visible only when the **Inertia** parameter is `Specify inertia and initial velocity`.

To specify contact characteristics, select the **Contact** tab.

Contact

Contact friction coefficient

Coulomb friction coefficient between the belt and the pulley surface. Default is `0.5`.

Wrap angle calculation

Specifies whether the block calculates the wrap angle of the belt on the pulleys using the pulley radii, center separation, and drive type or if the wrap angles are specified directly. Options are:

- `Specify pulley center separation` — (Default) The block calculates the wrap angle of the belt on the pulleys using the pulley radii, center separation, and drive type.

  Exposes the **Pulley center separation** parameter.
- `Specify wrap angles` — Uses the wrap angles that you specify.

  Exposes the **Pulley A wrap angle** and **Pulley B wrap angle** parameters.

Pulley center separation

Distance between the centers of the pulleys. Default is `0.5 m`.

The **Pulley center separation** parameter is visible only when the **Wrap angle calculation** parameter is `Specify pulley center separation`.

Pulley A wrap angle

Angle of contact between the belt and pulley attached to port A. Default is `145 deg`.

The **Pulley A wrap angle** parameter is visible only when the **Wrap angle calculation** parameter is `Specify wrap angles`.

Pulley B Wrap angle

Angle of contact between the belt and pulley attached to port B. Default is `215 deg`.

The **Pulley B wrap angle** parameter is visible only when the **Wrap angle calculation** parameter is `Specify wrap angles`.

Velocity threshold

Relative velocity required for peak kinetic friction in the contact. The friction velocity threshold improves the numerical stability of the simulation by ensuring that the force is continuous when the direction of the velocity changes. Default is `0.001 m/s`.

Real-Time Simulation

Hardware-in-the-Loop Simulation

For optimal simulation performance, set the **Belt** > **Centrifugal force** parameter to `Do not model centrifugal force - Suitable for HIL simulation`.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also
Belt Pulley | Chain Drive

Topics
"Best Practices for Modeling Pulley Networks"
"Rotational-Translational Couplings"

Belt Pulley

Power transmission element with frictional belt wrapped around pulley circumference

Library

Simscape / Driveline / Couplings & Drives

Description

This block represents a pulley wrapped in a flexible flat or V-shaped belt. The model accounts for friction between the flexible belt and the pulley periphery. If the friction force is not sufficient to drive the load, the model allows slip. The relationship between the tensions in the tight and loose branches conforms to the Euler equation. The model accounts for centrifugal loading in the flexible belt, pulley inertia, and bearing friction.
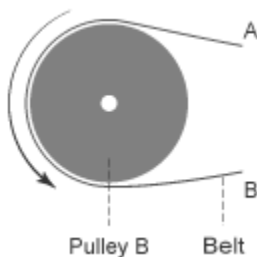
The block dialog box enables you to select the relative belt direction of motion. Depending on the chosen setting, the two belt ends can move in equal or opposite directions. The block model assumes noncompliance in the belt and no resistance to motion due to wrapping around the pulley.

The block equations model power transmission between the belt branches or to/from the pulley. The tight and loose branches use the same calculation. Without sufficient tension, the frictional force is not enough to transmit power between the pulley and belt.

The model is valid when both ends of the belt are in tension. An optional warning can display at the MATLAB® command line when either belt end loses tension. When assembling a model, ensure that tension is maintained throughout the simulation. Consider this requirement when you interpret the simulation results.

Equations

If the relative velocity between the belt and pulley is positive or zero, that is $V_{rel} \geq 0$, the Belt Pulley block calculates friction force as

$$F_{fr} = F_B - F_C = (F_A - F_C) * \exp(f * \theta).$$

If the relative velocity is negative, that is $V_{rel} < 0$, the friction force is calculated as

$$F_{fr} = F_A - F_C = (F_B - F_C) * \exp(f * \theta).$$

In both cases:

$$V_{rel} = V_A - \omega_S * R$$

$$F_C = \rho * V_B^2$$

$$V_A = -V_B$$

Where:

- V_{rel} is the relative velocity between the belt and pulley periphery.
- V_A is the Branch A linear velocity.
- V_B is the Branch B linear velocity.
- ω_S is the pulley angular velocity.
- R is the pulley radius.
- F_C is the belt centrifugal force.
- ρ is the belt linear density.
- F_{fr} is the friction force between the pulley and the belt.

- F_A is the force acting along branch A.
- F_B is the force acting along branch B.
- f is the friction coefficient.
- θ is the contact wrap angle.
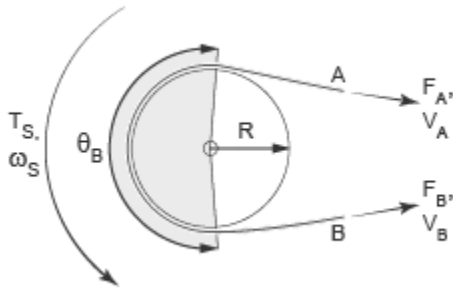
For a flat belt, specify the value of f directly in the block parameters dialog box. For a V-belt, the model calculates the value as

$$f' = \frac{f}{\sin(\phi/2)},$$

where:

- f is the effective friction coefficient for a V-belt.
- Φ is the V-belt sheave angle.

The idealization of the discontinuity at $V_{rel} = 0$ is both difficult for the solver to resolve and not physically accurate. To alleviate this issue, the friction coefficient is assumed to change its value as a function of the relative velocity such that

$$\mu = -f * \tanh\left(4 * V_{rel}/V_{thr}\right),$$

where

- μ is the instantaneous value of the friction coefficient.
- f is the steady-state value of the friction coefficient.
- V_{thr} is the friction velocity threshold.

The friction velocity threshold controls the width of the region within which the friction coefficient changes its value from zero to a steady-state maximum. The friction velocity threshold specifies the velocity at which the hyperbolic tangent equals 0.999. The smaller the value, the steeper is the change of μ.

This friction force is calculated as

$$F_{fr} = F_A - F_C = (F_B - F_C) * \exp(\mu * \theta).$$

The resulting torque on the pulley is given as

$$T_S = (F_A + F_B) * R * \tanh\left(4\frac{V_{\text{rel}}}{V_{\text{thr}}}\right) * \tanh\left(\frac{F_B}{F_{\text{thr}}}\right) - \omega_S * b.$$

where:

- T_S is the pulley torque.
- b is the bearing viscous damping.
- F_{thr} is the force threshold.

Assumptions and Limitations

- The model does not account for compliance along the length of the belt.
- Both belt ends maintain adequate tension throughout the simulation.

Ports

The sign convention is such that, when **Belt direction** is set to `Ends move in opposite direction`, a positive rotation in port **S** tends to give a negative translation for port **A** and a positive translation for port **B**.

Conserving

S — Rotational velocity
mechanical rotational

Rotational conserving port associated with the pulley shaft.

A — Velocity
mechanical translational

Translational conserving port associated with the belt end *A*.

B — Velocity
mechanical translational

Translational conserving port associated with the belt end *B*.

Parameters

Belt

Belt type

> Type of flexible element modeled.
>
> - `Flat belt` — (Default)
> - `V-belt`— Exposes the **V-belt sheave angle** and **Number of V-belts** parameters.

V-belt sheave angle

> Sheave angle of the V-belt. Default is `30 deg`.
>
> The **V-belt sheave angle** parameter is visible only when the **Belt type** parameter is `V-belt`.

Number of V-belts

> Number of parallel V-belts. The default value is `1`.
>
> Non-integer values are rounded to the nearest integer. Increasing the number of belts increases the friction force, effective mass per unit length, and maximum allowable tension.
>
> The **Number of V-belts** parameter is visible only when the **Belt type** parameter is `V-belt`.

Centrifugal force

> Select whether to include the effects of centrifugal force. If included, centrifugal force saturates to approximately 90% of the value of the force on each belt end. Options are:

- Do not model centrifugal force - Suitable for HIL simulation — Default option which does not include centrifugal force contributions.
- Model centrifugal force— Exposes the **Belt mass per unit length** parameter for configuring the centrifugal force contribution.

Belt mass per unit length

Linear density of each belt. Default is 0.6 kg/m.

The model uses this parameter to calculate the centrifugal loading. This parameter is enabled only if you select Model centrifugal force for the **Centrifugal force** parameter.

Belt direction.

Relative direction of translational motion of one belt end with respect to the other. Options include:

- Ends move in opposite direction
- Ends move in same direction

Maximum tension

Specifies whether the block throws an assertion when the belt tension is too large.

- No maximum tension — (Default)
- Specify maximum tension — Throws an assertion when the belt tension is too high. Exposes the **Belt maximum tension** parameter.

Belt maximum tension

Maximum allowable tension for each belt. Default is 1e+5 N.

When the tension on either end of the belt meets or exceeds this value, the simulation ends with an error.

The **Belt maximum tension** parameter is visible only when the **Maximum tension** parameter is Specify maximum tension.

Tension warning

Specifies whether the block writes a warning at the MATLAB command line when the tension at either end of the belt falls below zero.

- Do not check tension — (Default)

• `Warn if either end loses tension`

Select the **Pulley** tab to specify the pulley characteristics.

Pulley

Pulley radius

Radius of the pulley. Default is `0.15 m`.

Bearing viscous friction coefficient

Viscous friction associated with the bearings that hold the axis of the pulley. Default is `0.001 N*m/(rad/s)`.

Inertia

Specifies whether the block models rotational inertia of the pulley.

• `No inertia` — (Default)
• `Specify inertia and initial velocity` — Models rotational inertia. Exposes the **Pulley inertia** and **Pulley initial velocity** parameters.

Pulley inertia

Rotational inertia of the pulley. Default is `0.01 kg*m^2`.

The **Pulley inertia** parameter is visible only when the **Inertia** parameter is `Specify inertia and initial velocity`.

Pulley initial velocity

Initial rotational velocity of the pulley. Default is `0 rad/s`.

The **Pulley initial velocity** parameter is visible only when the **Inertia** parameter is `Specify inertia and initial velocity`.

to specify contact characteristics, use the **Contact** pane.

Contact

Contact friction coefficient

Coulomb friction coefficient between the belt and the pulley surface. Default is `0.5`.

Wrap angle

Angle of contact between the belt and pulley. Default is `180 deg`.

Friction velocity threshold

Relative velocity required for peak kinetic friction in the contact. The friction velocity threshold improves the numerical stability of the simulation by ensuring that the force is continuous when the direction of the velocity changes. Default is `0.001 m/s`.

Real-Time Simulation

Hardware-in-the-Loop Simulation

For optimal simulation performance, set the **Belt** > **Centrifugal force** parameter to `Do not model centrifugal force - Suitable for HIL simulation`.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also
Belt Drive | Rope Drum | Wheel and Axle

Topics
"Best Practices for Modeling Pulley Networks"
"Rotational-Translational Couplings"

Chain Drive

Power transmission system with chain and two sprockets
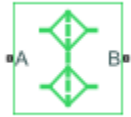
Library

Couplings & Drives

Description

This block represents a power transmission system with a chain and two sprockets. The chain meshes with the sprockets, transmitting rotary motion between the two. Power transmission can occur in reverse, that is, from driven to driver sprocket, due to external loads. This condition is known as back-driving.
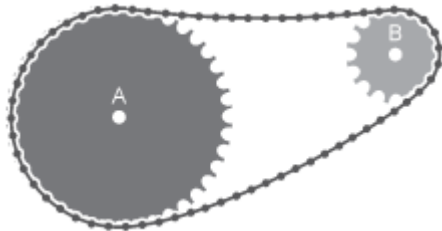
The drive chain is compliant. It can stretch under tension and slacken otherwise. The compliance model consists of a linear spring-damper set in a parallel arrangement. The spring resists tensile strain in the chain. The damper resists tensile motion between chain elements.

The spring and damper forces act directly on the sprockets that the chain connects. The spring force is present when one chain branch is taut. The damper force is present

continuously. If you specify a maximum tension value, when the net tensile force in the chain exceeds this value, simulation stops with an error.

The block accounts for viscous friction at the sprocket joint bearings. During motion, viscous friction causes power transmission losses, reducing chain-drive efficiency. These losses compound due to chain damping. Setting viscous friction and chain damping to zero eliminates power transmission losses in the chain drive.

The tensile strain rate in the chain is the difference between the sprocket tangential velocities, each the product of the angular velocity and pitch radii. Mathematically,

$$\dot{x} = \omega_A R_A - \omega_B R_B,$$

where:

- x is the tensile strain.
- ω_A, ω_B are the sprocket angular velocities.
- R_A, R_B are the sprocket pitch radii.

The figure shows the relevant equation variables.
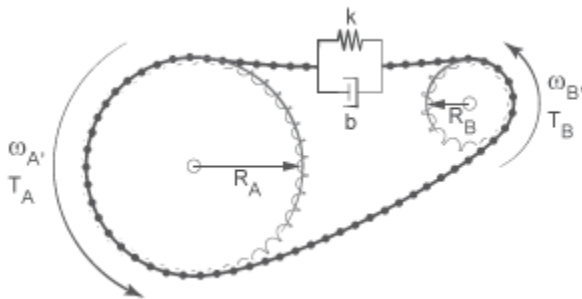
The chain tensile force is the net sum of spring and damper forces. The spring force is the product of the tensile strain and the spring stiffness constant. This force is zero whenever the tensile strain is smaller than the chain slack. The damper force is the product of the tensile strain rate and the damping coefficient. Mathematically,

$$F = \begin{cases} -\left(x - \dfrac{S}{2}\right)k - \dot{x}b, & x > \dfrac{S}{2} \\[2ex] -\dot{x}b, & \dfrac{S}{2} \geq x \geq -\dfrac{S}{2}, \\[2ex] -\left(x + \dfrac{S}{2}\right)k - \dot{x}b, & x < -\dfrac{S}{2} \end{cases}$$

where:

- S is the chain slack.
- k is the spring stiffness constant.
- b is the damper coefficient.

The chain exerts a torque on each sprocket equal to the product of the tensile force and the sprocket pitch radius. The two torques act in opposite directions according to the equations

$$T_A = -F \cdot R_A,$$

and

$$T_B = F \cdot R_B,$$

where:

- T_A is the torque that the chain applies on sprocket A.
- T_B is the torque that the chain applies on sprocket B.

Variables

Use the **Variables** tab to set the priority and initial target values for the block variables before simulating. For more information, see "Set Priority and Initial Target for Block Variables" (Simscape).

Unlike block parameters, variables do not have conditional visibility. The **Variables** tab lists all the existing block variables. If a variable is not used in the set of equations corresponding to the selected block configuration, the values specified for this variable are ignored.

Assumptions and Limitations

- The sprocket tooth ratio equals the sprocket pitch radius ratio.
- Chain inertia is negligible.

Parameters

Geometry

Specify the sprocket and chain dimensions.

Sprocket A pitch radius

Radius of the sprocket A pitch circle. The pitch circle is an imaginary circle passing through the contact point between a chain roller and a sprocket cog at full engagement. The default value is 80 mm.

Sprocket B pitch radius

Radius of the sprocket B pitch circle. The pitch circle is an imaginary circle passing through the contact point between a chain roller and a sprocket cog at full engagement. The default value is 40 mm.

Chain slack length

Maximum distance the loose branch of the drive chain can move before it is taut. This distance equals the length difference between actual and fully taut drive chains.

If one sprocket is held in place while the top chain branch is taut, then the slack length is the tangential distance that the second sprocket must rotate before the lower chain branch becomes taut. The default value is 50 mm.

Dynamics

Specify the internal forces acting in the chain and on the sprockets.

Chain stiffness

Linear spring constant in the chain compliance model. This constant describes the chain resistance to strain. The spring element accounts for elastic energy storage in the chain due to deformation. The default value is 1e+5 N/m.

Chain damping

Linear damping coefficient in the chain compliance model. This coefficient describes the chain resistance to tensile motion between adjacent chain elements. The damper element accounts for power losses in the chain due to deformation. The default value is 5 N/(m/s).

Viscous friction coefficient of sprocket A

Friction coefficient due to the rolling action of the sprocket A joint bearing in the presence of a viscous lubricant. The default value is 0.001 N*m/(rad/s).

Viscous friction coefficient of sprocket B

Friction coefficient due to the rolling action of the sprocket B joint bearing in the presence of a viscous lubricant. The default value is 0.001 N*m/(rad/s).

Maximum tension

Specify the upper tension limit in the drive chain.

Maximum tension

Select whether to constrain the maximum tensile force in the drive chain.

- No maximum tension — Chain tension can be arbitrarily large during simulation.

- Maximum tension — Chain tension must remain lower than a maximum value. Simulation stops with an error when tension exceeds this value.

Chain maximum tension

Maximum allowed value of the tensile force acting in the chain. The default value is 1e+6 N.

Ports

Port	Description
A	Conserving rotational port associated with sprocket A
B	Conserving rotational port associated with sprocket B

Example

To see how you can use the Chain Drive block in a model, open sdl_sheet_metal_feeder.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also
Simple Gear | Variable Ratio Transmission

Topics
"Gear Coupling Control Using Clutches"

Compound Planetary Gear

Planetary gear train with stepped planet gear set
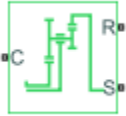
Library

Simscape / Driveline / Gears

Description

This block represents a planetary gear train with composite planet gears. Each composite planet gear is a pair of rigidly connected and longitudinally arranged gears of different radii. One of the two gears engages the centrally located sun gear while the other engages the outer ring gear.
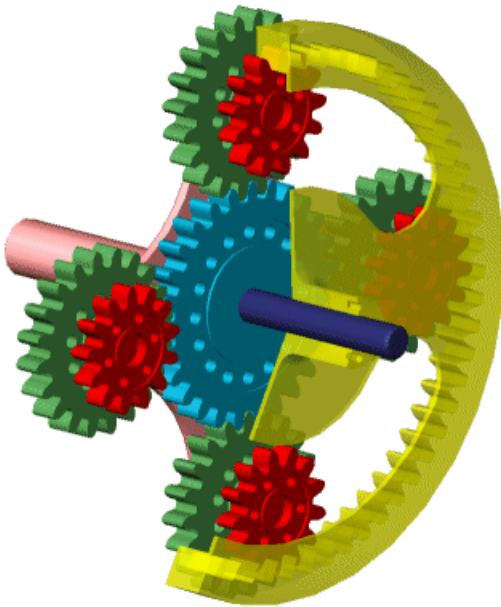
Compound Planetary Gear

The block models the compound planetary gear as a structural component based on Sun-Planet and Ring-Planet Simscape Driveline blocks. The figure shows the block diagram of this structural component.

To increase the fidelity of the gear model, you can specify properties such as gear inertia, meshing losses, and viscous losses. By default, gear inertia and viscous losses are assumed negligible. The block enables you to specify the inertias of the internal planet

gears only. To model the inertias of the carrier, sun, and ring gears, connect Simscape Inertia blocks to ports C, S, and R.

Thermal Modeling

You can model the effects of heat flow and temperature change through an optional thermal conserving port. By default, the thermal port is hidden. To expose the thermal port, right-click the block in your model and, from the context menu, select **Simscape** > **Block choices**. Select a variant that includes a thermal port. Specify the associated thermal parameters for the component.

Compound Planetary Gear Model

Ideal Gear Constraints and Gear Ratios

Compound Planetary Gear imposes two kinematic and two geometric constraints on the three connected axes and the fourth, internal wheel (planet):

$$r_C\omega_C = r_S\omega_S + r_{P1}\omega_P \quad , \quad r_C = r_S + r_{P1} \quad ,$$

$$r_R\omega_R = r_C\omega_C + r_{P2}\omega_P \quad , \quad r_R = r_C + r_{P2} \quad .$$

The ring-planet gear ratio $g_{RP} = r_R/r_{P2} = N_R/N_{P2}$ and the planet-sun gear ratio $g_{PS} = r_{P1}/r_S = N_{P1}/N_S$. N is the number of teeth on each gear. In terms of these ratios, the key kinematic constraint is:

$$(1 + g_{RP}{\cdot}g_{PS})\omega_C = \omega_S + g_{RP}{\cdot}g_{PS}\omega_R \quad .$$

The four degrees of freedom reduce to two independent degrees of freedom. The gear pairs are (1,2) = (P2,R) and (S,P1).

Warning The gear ratio g_{RP} must be strictly greater than one.

The torque transfers are:

$$g_{RP}\tau_{P2} + \tau_R - \tau_{loss}(P2,R) = 0 \quad , \quad g_{PS}\tau_S + \tau_{P1} - \tau_{loss}(S,P1) = 0 \quad ,$$

with $\tau_{loss} = 0$ in the ideal case.

Nonideal Gear Constraints and Losses

In the nonideal case, $\tau_{loss} \neq 0$. See "Model Gears with Losses".

Limitations

- Gears are assumed rigid.
- Coulomb friction slows down simulation. See "Adjust Model Fidelity".

Ports

Port	Description
C	Rotational conserving port representing the planet gear carrier
R	Rotational conserving port representing the ring gear
S	Rotational conserving port representing the sun gear
H	Thermal conserving port for thermal modeling

Parameters

Main

Ring (R) to planet (P) teeth ratio (NR/NP)

Fixed ratio g_{RP} of the ring gear to the planet gear. The gear ratio must be strictly greater than 1. The default is 2.

Planet (P) to sun (S) teeth ratio (NP/NS)

Fixed ratio g_{PS} of the planet gear to the sun gear. The gear ratio must be strictly positive. The default is 1.

Meshing Losses

Parameters for meshing losses vary with the block variant chosen—one with a thermal port for thermal modeling and one without it.

Without Thermal Port

Friction model

Select how to implement friction losses from nonideal meshing of gear teeth. The default is `No meshing losses`.

- `No meshing losses — Suitable for HIL simulation` — Gear meshing is ideal.
- `Constant efficiency` — Transfer of torque between gear wheel pairs is reduced by a constant efficiency η satisfying $0 < \eta \leq 1$. If you select this option, the panel changes from its default.

Constant Efficiency

Sun-planet and ring-planet ordinary efficiencies

Vector of torque transfer efficiencies [η_{SP} η_{RP}] for sun-planet and ring-carrier gear wheel pair meshings, respectively.

Sun-carrier and planet-carrier power thresholds

Vector of power thresholds p_{th} above which full efficiency factors apply, for sun-carrier and planet-carrier, respectively. Below these values, a hyperbolic tangent function smooths the efficiency factor, lowering the efficiency losses to zero when no power is transmitted. The default array is `[0.001 0.001]` W.

With Thermal Port

Temperature

Array of temperatures used to construct a 1-D temperature-efficiency lookup table. The array values must increase left to right. The temperature array must be the same size as each efficiency array. The default array is `[280 300 320]` K.

Sun-planet efficiency

Array of mechanical efficiencies with power flowing from the sun gear to the planet gears. Each array value is the ratio of output power to input power at one of the temperatures in the temperature array. The temperature and efficiency arrays must be the same size. The default array is `[0.95 0.9 0.85]`.

Ring-planet efficiency

Array of mechanical efficiencies with power flowing from the ring gear to the planet gears. Each array value is the ratio of output power to input power at one of the temperatures in the temperature array. The temperature and efficiency arrays must be the same size. The default array is `[0.95 0.9 0.85]`.

Sun-carrier and planet-carrier power thresholds

Array of power thresholds above which the full efficiency factors apply. Enter the thresholds for the sun and planet gears relative to the gear carrier, in this order. A hyperbolic tangent function smooths the efficiency factors between zero at rest and the values provided by the temperature-efficiency lookup tables at the power thresholds. The default array is [0.001 0.001] W.

Viscous Losses

Sun-carrier and planet-carrier viscous friction coefficients

Vector of viscous friction coefficients [μ_S μ_P] for the sun-carrier and planet-carrier gear motions, respectively. The default is [0 0].

From the drop-down list, choose units. The default is newton-meters/(radians/second) (N*m/(rad/s)).

Inertia

Planet gear inertia

Moment of inertia of the combined planet gears. This value must be positive or zero. Enter 0 to ignore gear inertia. The default value is 0 kg*m^2.

Thermal Port

Thermal mass

Thermal energy required to change the component temperature by a single degree. The greater the thermal mass, the more resistant the component is to temperature change. The default value is 50 J/K.

Initial temperature

Component temperature at the start of simulation. The initial temperature alters the component efficiency according to an efficiency vector that you specify, affecting the starting meshing or friction losses. The default value is 300 K.

Real-Time Simulation

Hardware-in-the-Loop Simulation

For optimal simulation performance, use the **Meshing Losses** > **Friction model** parameter default setting, `No meshing losses - Suitable for HIL simulation`.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also

Simscape Blocks
Planetary Gear | Ravigneaux Gear | Ring-Planet | Sun-Planet

Topics
"Basic Motion, Torque, and Force Modeling"
"Model Thermal Losses in Driveline Components"
"Prepare Simscape Driveline Models for Real-Time Simulation Using Simscape Checks"

Cone Clutch

Friction clutch with conical plates that engage when normal force exceeds threshold

Library

Clutches

Description

This block represents a friction clutch with a conical contact interface. The conical interface creates a wedging action between the clutch components, a cone and a cup, thereby reducing the normal force required for clutch engagement.

The cup component connects rigidly to the drive shaft, spinning with it as a unit. The cone component connects rigidly to the driven shaft, which sits in axial alignment with the drive shaft. The clutch engages when the cone slides toward the cup and presses tightly against its internal surface.

Friction at the cone-cup contact interface enables the clutch to transmit rotational power between the drive and driven shafts. The friction model of this block includes both static and kinetic friction contributions, the latter of which leads to power dissipation during slip between the cone and cup components.

Cone clutches find real-world application in synchromesh gearboxes, which synchronize the drive and driven shaft speeds to enable smoother engagement between transmission gears. For block model details, see "Cone Clutch Model" on page 1-80.

Cone Clutch Model

The Cone Clutch is based on the Fundamental Friction Clutch. For the complete friction clutch model, consult the Fundamental Friction Clutch block reference page. This section discusses the specialized model implemented in the Cone Clutch.

When you apply a normal force F_N, the Cone Clutch block can apply two kinds of friction to the driveline motion, kinetic and static. The clutch applies kinetic friction torque only when one driveline axis is spinning relative to the other driveline axis. The clutch applies static friction torque when the two driveline axes lock and spin together. The block iterates through multistep testing to determine when to lock and unlock the clutch.

Clutch Geometry and Variable Summary

The figure shows the cone clutch geometry and some model parameters. Refer to the table for a summary of variable descriptions.
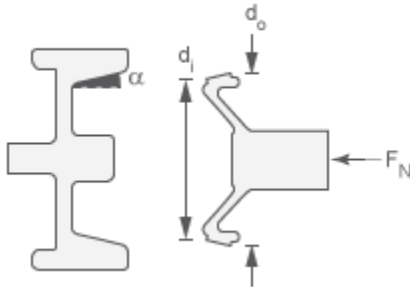
Clutch Variables

Parameter	Definition	Significance
d_o	Outer diameter of the conical contact surface	*See the preceding figure*
d_i	Inner diameter of the conical contact surface	*See the preceding figure*
α	Cone half angle	*See the preceding figure*
ω	Relative angular velocity (slip)	$\omega_F - \omega_B$
ω_{Tol}	Slip tolerance for clutch locking	*See the following model*
F_N	Normal force applied to conical surfaces	Normal force applied, if greater than threshold: $F_N > F_{th}$
α	Cone half-angle	*See the preceding figure*
r_{eff}	Effective torque radius	Effective moment arm of clutch friction force
k_K	Kinetic friction coefficient	Dimensionless coefficient of kinetic friction of conical friction surfaces. Function of ω.
k_S	Static friction coefficient	Dimensionless coefficient of static friction of conical friction surfaces.
τ_K	Kinetic friction torque	*See the following model*
τ_S	Static friction torque limit	(static friction peak factor)·(kinetic friction torque for $\omega \to 0$) (*See the following model*)

Relation to Fundamental Friction Clutch

The Cone Clutch is based on the Fundamental Friction Clutch. Instead of requiring the kinetic and static friction limit torques as input signals, the Cone Clutch calculates the kinetic and static friction from the clutch parameters and the input normal force signal F_N. See the Fundamental Friction Clutch reference page for more information about the friction clutch.

Kinetic Friction

The kinetic friction torque is the product of four factors:

$$\tau_K \qquad\qquad = \qquad\qquad k_K \cdot F_N \cdot r_{\text{eff}} \cdot \text{sgn}(\omega) \qquad\qquad .$$

The kinetic friction torque opposes the relative slip and is applied with an overall minus sign. It changes sign when ω changes sign.

You specify the *kinetic friction coefficient* k_K as either a constant or a tabulated discrete function of relative angular velocity ω. The tabulated function is assumed to be symmetric for positive and negative values of the relative angular velocity. Therefore, specify k_K for positive values of ω only.

The *effective torque radius* r_{eff} is the effective radius, measured from the driveline axis, at which the kinetic friction forces are applied at the frictional surfaces. It is related to the geometry of the conical friction surface geometry by:

$$r_{\text{eff}} = \frac{1}{3\sin\alpha} \frac{d_o{}^3 - d_i{}^3}{d_o{}^2 - d_i{}^2}$$

d_o and d_i are the contact surface maximum and minimum diameters, respectively.

Static Friction

The static friction limit is related to the kinetic friction, setting ω to zero and replacing the kinetic with the static friction coefficient:

$$\tau_S \qquad\qquad = \qquad\qquad k_S \cdot F_N \cdot r_{\text{eff}} \qquad\qquad \geq \qquad\qquad 0 \qquad\qquad .$$

$k_S > k_K$, so that the torque τ needed across the clutch to unlock it by overcoming static friction is larger than the kinetic friction at the instant of unlocking, when $\omega = 0$.

The static friction limit defines symmetric *static friction torque limits* as:

$$\tau_S \qquad\qquad \equiv \qquad\qquad \tau_S{}^+ \qquad\qquad = \qquad\qquad -\tau_S{}^- \qquad\qquad .$$

The range $[\tau_S{}^-, \tau_S{}^+]$ is used by the Fundamental Friction Clutch.

Engagement and Locking Conditions

The clutch engages (transmits torque) when the conical friction surfaces are subject to a positive normal force and generate kinetic friction: $F_N > 0$ and $\tau_K > 0$.

The clutch locks if and only if it is engaged, and the slip is less than the velocity tolerance: $|\omega| < \omega_{\text{Tol}}$.

Power Dissipated by the Clutch

The power dissipated by the clutch is $|\omega \cdot \tau_K|$. The clutch dissipates power only if it is both slipping ($\omega \neq 0$) and applying kinetic friction ($\tau_K > 0$).

Thermal Modeling

You can model the effects of heat flow and temperature change through an optional thermal conserving port. By default, the thermal port is hidden. To expose the thermal port, right-click the block in your model and, from the context menu, select **Simscape** > **Block choices**. Select a variant that includes a thermal port. Specify the associated thermal parameters for the component.

Ports

B

> Rotational conserving ports associated with the driving (base) shaft. The clutch motion is measured as the *slip* $\omega = \omega_F - \omega_B$, the angular velocity of follower relative to base.

F

> Rotational conserving ports associated with the driven (follower) shaft

N

> Physical signal terminal through which you import the normal force. This signal is positive or zero. A signal N less than zero is interpreted as zero.

H

> Thermal conserving port. The thermal port is optional and is hidden by default. To expose the port, select a variant that includes a thermal port.

Parameters

Geometry

Contact surface maximum diameter

> Outer conical diameter d_o. The default value is 150 mm.

Contact surface minimum diameter

Inner conical diameter d_i. The default value is `100 mm`.

Cone half angle

Half opening angle α of the cone geometry. The default value is `12 deg`.

Friction

Friction model

Select a parameterization method to model the kinetic friction coefficient. The options and default values for this parameter depend on the variant that you select for the block. The options are:

- `Fixed kinetic friction coefficient` — Provide a fixed value for the kinetic friction coefficient. This option:

  - Is only visible if you use the default variant of the block
  - Is the default method for parameterizing the default variant of the block
  - Affects the visibility of other parameters

  `Fixed kinetic friction coefficient`
  Kinetic friction coefficient

  The kinetic, or Coulomb, friction coefficient. The coefficient must be greater than zero. The default value is `0.3`.

  Static friction coefficient

  The static, or peak, value of the friction coefficient. The static friction coefficient must be greater than the kinetic friction coefficient. The default value is `0.35`.

- `Table lookup kinetic friction coefficient` — Define the kinetic friction coefficient by one-dimensional table lookup based on the relative angular velocity between disks. This option:

  - Is only visible if you use the default variant of the block
  - Affects the visibility of other parameters

  `Table lookup kinetic friction coefficient`
  Kinetic friction coefficient relative velocity vector

  Specify input values for the relative velocity as a vector. The values in the vector must increase from left to right. The minimum number of values

depends on the interpolation method that you select. For linear interpolation, provide at least two values per dimension. For smooth interpolation, provide at least three values per dimension. The default value is `[0.0, 100.0, 1000.0] rad/s`.

Kinetic friction coefficient vector

Specify the output values for the kinetic friction coefficient as a vector. All values must be positive. The default value is `[0.3, 0.22, 0.19]`.

Friction coefficient interpolation method

Select one of these interpolation methods to approximate the output value when the input value is between two consecutive grid points:

- `Linear` — Select this, default, option to get the best performance.
- `Smooth` — Select this option to produce a continuous curve with continuous first-order derivatives.

For more information on interpolation algorithms, see the PS Lookup Table (1D) block reference page.

Friction coefficient extrapolation method

Select one of these extrapolation methods to determine the output value when the input value is outside the range specified in the argument list:

- `Linear` — Select this, default, option to produce a curve with continuous first-order derivatives in the extrapolation region and at the boundary with the interpolation region.
- `Nearest` — Select this option to produce an extrapolation that does not go above the highest point in the data or below the lowest point in the data.

For more information on extrapolation algorithms, see the PS Lookup Table (1D) block reference page.

Static friction coefficient

The static, or peak, value of the friction coefficient. The static friction coefficient must be greater than the kinetic friction coefficient. The default value is `0.35`.

- `Temperature-dependent kinetic friction coefficient` — Define the kinetic friction coefficient by table lookup based on the temperature. This option:

  - Is only visible if you use a thermal variant of the block

- Is the default method for parameterizing the thermal variant of the block
- Affects the visibility of other parameters

`Temperature-dependent kinetic friction coefficient`

Temperature vector

Specify the input values for the temperature as a vector. The minimum number of values depends on the interpolation method that you select. For linear interpolation, provide at least two values per dimension. For smooth interpolation, provide at least three values per dimension. The values in the vector must increase from left to right. The default value is `[280.0, 300.0, 320.0]` K.

Kinetic friction coefficient vector

Specify the output values for kinetic friction coefficient as a vector. The vector must have the same number of elements as the temperature vector. All values must be greater than zero. The default value is `[0.3, 0.28, 0.25]`.

Static friction coefficient vector

Specify the static, or peak, values of the friction coefficient as a vector. The vector must have the same number of elements as the temperature vector. Each value must be greater than the value of the corresponding element in the kinetic friction coefficient vector. The default value is `[0.4, 0.38, 0.36]`.

Friction coefficient interpolation method

Select one of these interpolation methods to approximate the output value when the input value is between two consecutive grid points:

- `Linear` — Select this, default, option to get the best performance.
- `Smooth` — Select this option to produce a continuous curve with continuous first-order derivatives.

For more information on interpolation algorithms, see the PS Lookup Table (1D) block reference page.

Friction coefficient extrapolation method

Select one of these extrapolation methods to determine the output value when the input value is outside the range specified in the argument list:

- `Linear` — Select this, default, option to produce a curve with continuous first-order derivatives in the extrapolation region and at the boundary with the interpolation region.

- **Nearest** — Select this option to produce an extrapolation that does not go above the highest point in the data or below the lowest point in the data.

  For more information on extrapolation algorithms, see the PS Lookup Table (1D) block reference page.

- **Temperature and speed-dependent kinetic friction coefficient** — Define the kinetic friction coefficient by table lookup based on the temperature and the relative angular velocity between disks. This option:

  - Is only visible if you use the default variant of the block
  - Affects the visibility of other parameters

Temperature and speed-dependent kinetic friction coefficient

Temperature vector

   Specify the input values for the temperature as a vector. The minimum number of values depends on the interpolation method that you select. For linear interpolation, provide at least two values per dimension. For smooth interpolation, provide at least three values per dimension. The values in the vector must increase from left to right. The default value is `[280.0, 300.0, 320.0]` K.

Kinetic friction coefficient relative velocity vector

   Specify input values for the relative velocity as a vector. The values in the vector must increase from left to right. The minimum number of values depends on the interpolation method that you select. For linear interpolation, provide at least two values per dimension. For smooth interpolation, provide at least three values per dimension. The default value is `[0.0, 100.0, 1000.0]` rad/s.

Kinetic friction coefficient matrix

   Specify the output values for kinetic friction coefficient as a matrix. All the values must be greater than zero. The size of the matrix must equal the size of the matrix that is the result of the temperature vector × the kinetic friction coefficient relative velocity vector. The default value is `[0.34, 0.32, 0.3; 0.3, 0.28, 0.25; 0.25, 0.2, 0.15]`.

Static friction coefficient vector

   Specify the static, or peak, values of the friction coefficient as a vector. The vector must have the same number of elements as the temperature vector. Each value must be greater than the value of the corresponding element in the kinetic friction coefficient vector. The default value is `[0.4, 0.38, 0.36]`.

Friction coefficient interpolation method

Select one of these interpolation methods to approximate the output value when the input value is between two consecutive grid points:

- `Linear` — Select this, default, option to get the best performance.
- `Smooth` — Select this option to produce a continuous curve with continuous first-order derivatives.

For more information on interpolation algorithms, see the PS Lookup Table (1D) block reference page.

Friction coefficient extrapolation method

Select one of these extrapolation methods to determine the output value when the input value is outside the range specified in the argument list:

- `Linear` — Select this, default, option to produce a curve with continuous first-order derivatives in the extrapolation region and at the boundary with the interpolation region.
- `Nearest` — Select this option to produce an extrapolation that does not go above the highest point in the data or below the lowest point in the data.

For more information on extrapolation algorithms, see the PS Lookup Table (1D) block reference page.

Velocity tolerance

Relative velocity below which the two surfaces can lock. The surfaces lock if the torque across the B and F rotational ports is less than the product of the effective radius, the static friction coefficient, and the applied normal force. The default value is `0.001 rad/s`.

Threshold force

The normal force applied to the physical signal port **N** is applied to the contact only if the amount of force exceeds the value of the **Threshold force** parameter. Forces below the **Threshold force** are not applied, and there is consequently no transmitted frictional torque. The default value is `1 N`.

Initial Conditions

Initial state

Clutch state at the start of simulation. The clutch can be in one of two states, locked and unlocked. A locked clutch constrains the base and follower shafts to spin at the

same velocity, that is, as a single unit. An unlocked clutch allows the two shafts to spin at different velocities, resulting in slip between the clutch plates. The default setting is `Unlocked`.

Thermal Port

These thermal parameters are visible only when you select a block variant that includes a thermal port.

Thermal mass

Thermal energy required to change the component temperature by a single degree. The greater the thermal mass, the more resistant the component is to temperature change. The default value is 25 `kJ/K`.

Initial temperature

Component temperature at the start of simulation. The initial temperature influences the starting meshing or friction losses by altering the component efficiency according to an efficiency vector that you specify. The default value is 300 `K`.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also
Fundamental Friction Clutch

Topics
"Gear Coupling Control Using Clutches"

Cycloidal Drive

High-ratio speed reducer based on cycloidal disk motion

Library

Gears

Description

This block represents a compact, high-ratio, speed-reduction mechanism that contains four key components:

- Eccentric cam
- Cycloidal disk
- Ring-gear housing
- Pin rollers

The eccentric cam, which extends from the base shaft, sits inside the cycloidal disk. This disk meshes with the ring-gear housing. The pin rollers, which extend from the follower shaft, sit in matching holes on the cycloidal disk.
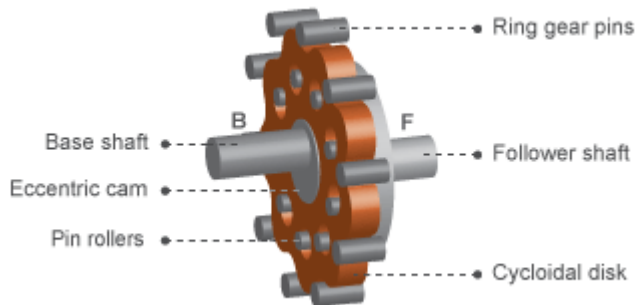
During normal operation, the base shaft drives the eccentric cam. The cam spins inside the cycloidal disk, causing it to rotate in an eccentric pattern about an offset axis. As it moves, the cycloidal disk engages the internal teeth of the ring-gear housing. The internal meshing reverses the rotational velocity direction.

Pin rollers extending from cycloidal disk holes transmit rotational motion to the follower shaft. This shaft spins counter to the base shaft at a highly reduced speed. The large reduction ratio results from the near-equal cycloidal disk and ring gear tooth numbers. The effective gear reduction ratio is

$$r = \frac{n_R - n_C}{n_C},$$

where:

- r is the gear reduction ratio.
- n_R is the number of teeth on the ring gear.
- n_C is the number of teeth on the cycloidal disk.

The gear reduction ratio constrains the angular velocities of the base and follower shafts according to the expression

$$\omega_F = -r\omega_B,$$

where:

- ω_F is the angular velocity of the follower shaft.
- ω_C is the angular velocity of the base shaft.

The gear reduction ratio also constrains the torques acting on the base and follower shafts, according to the expression

$$T_B = rT_F + T_f,$$

where:

- T_B is the net torque at the base shaft.
- T_F is the net torque at the follower shaft.
- T_f is the torque loss due to friction. For more information, see "Model Gears with Losses".

The figure shows the cycloidal drive in front and side views. The kinematics of the drive system cause a reversal of the base and follower shaft angular velocities so that the two shafts spin in opposite directions.
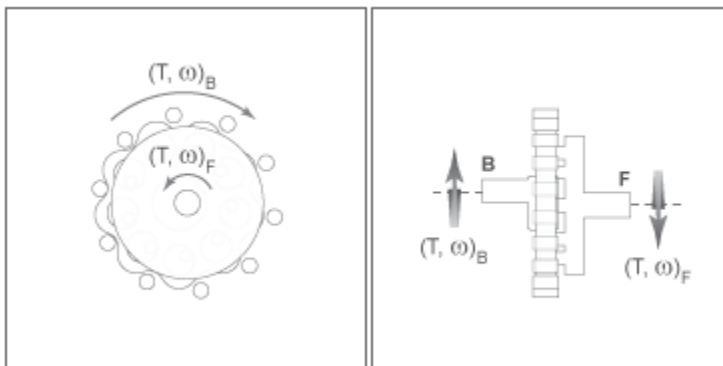
The cycloidal drive can operate in reverse mode, e.g., with power flowing from the follower shaft to the base shaft. In this mode, torque transfer efficiencies are typically negligible. You can adjust the efficiency value in the block dialog box using the **Efficiency from follower shaft to base shaft** parameter.

You can model the effects of heat flow and temperature change through an optional thermal conserving port. By default, the thermal port is hidden. To expose the thermal port, right-click the block in your model and, from the context menu, select **Simscape** > **Block choices**. Specify the associated thermal parameters for the component.

Parameters

Main

Number of teeth on cycloid disk

Total number of teeth projecting outward from the cycloidal disk perimeter. This number should be slightly smaller than the number of teeth or pins on the ring gear. The ratio of the gear tooth numbers defines the relative angular velocities of the base and follower shafts. The default value is `20`.

Number of teeth on ring gear

Total number of teeth or pins projecting inward from the ring gear housing. This number should be slightly larger than the number of teeth on the cycloidal disk. The ratio of the two gear tooth numbers defines the relative angular velocities of the base and follower shafts. The default value is `24`.

Meshing Losses

Parameters for meshing losses vary with the block variant chosen—one with a thermal port for thermal modeling and one without it.

Without Thermal Port

Efficiency from base shaft to follower shaft

Torque transfer efficiency in normal operation mode, e.g., with the base shaft driving the follower shaft. Efficiency values must fall in the interval [0, 1]. Larger efficiency values correspond to greater torque transfer between the base and follower shafts. Values approaching unity are typical. The default value is `0.90`.

Efficiency from follower shaft to base shaft

Torque transfer efficiency in reverse operation mode, e.g., with the follower shaft driving the base shaft. Efficiency values must fall in the interval [0, 1]. Larger efficiency values correspond to greater torque transfer between base and follower shafts. Values approaching zero are typical. The default value is `0.05`.

Power threshold

Absolute value of the cycloidal disk power above which the full efficiency factor applies. Below this value, a hyperbolic tangent function smooths the efficiency factor to one, lowering the efficiency losses to zero when at rest.

As a guideline, the power threshold should be lower than the expected power transmitted during simulation. Higher values might cause the block to underestimate efficiency losses. Very low values might, however, raise the computational cost of simulation.

The default value is `0.001` W.

With Thermal Port

Temperature

Array of temperatures used to construct 1-D temperature-efficiency lookup tables. The array values must increase left to right. The temperature array must be the same size as each efficiency arrays. The default array is `[280 300 320]` K.

Efficiency from base shaft to follower shaft

Array of mechanical efficiencies with power flowing from the base shaft to the follower shaft. Each array value is the ratio of output power to input power at one of the temperatures in the temperature array. The temperature and efficiency arrays must be the same size. The default array is `[0.95 0.85 0.80]`.

Efficiency from follower shaft to base shaft

Array of mechanical efficiencies with power flowing from the follower shaft to the base shaft. Each array value is the ratio of output power to input power at one of the temperatures in the temperature array. The temperature and efficiency arrays must be the same size. The default array is `[0.10 0.05 0.03]`.

Power threshold

Absolute value of the cycloidal disk power above which the full efficiency factor applies. A hyperbolic tangent function smooths the efficiency factor between zero when at rest and the value provided by the temperature-efficiency lookup table when at the power threshold. The default value is `0.001` W.

Thermal Port

Thermal mass

Thermal energy required to change the component temperature by a single degree. The greater the thermal mass, the more resistant the component is to temperature change. The default value is `50` J/K.

Initial temperature

Component temperature at the start of simulation. The initial temperature alters the component efficiency according to an efficiency vector that you specify, affecting the starting meshing or friction losses. The default value is 300 K.

Ports

Port	Description
B	Rotational conserving port representing the base shaft
F	Rotational conserving port representing the follower shaft
H	Thermal conserving port for thermal modeling

Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using MATLAB® Coder™.

See Also

Harmonic Drive | Simple Gear | Simple Gear with Variable Efficiency

Differential

Gear mechanism that allows driven shafts to spin at different speeds

Library

Simscape / Driveline / Gears

Description

This block represents a gear mechanism that allows the driven shafts to spin at different speeds. Differentials are common in automobiles, where they enable the various wheels to spin at different speeds while cornering. Ports D, S1, and S2 represent the driving and driven shafts of the differential. Any of the shafts can drive the remaining two.

The block models the differential mechanism as a structural component based on Simple Gear and Sun-Planet Bevel Simscape Driveline blocks. The figure shows the block diagram of this structural component.
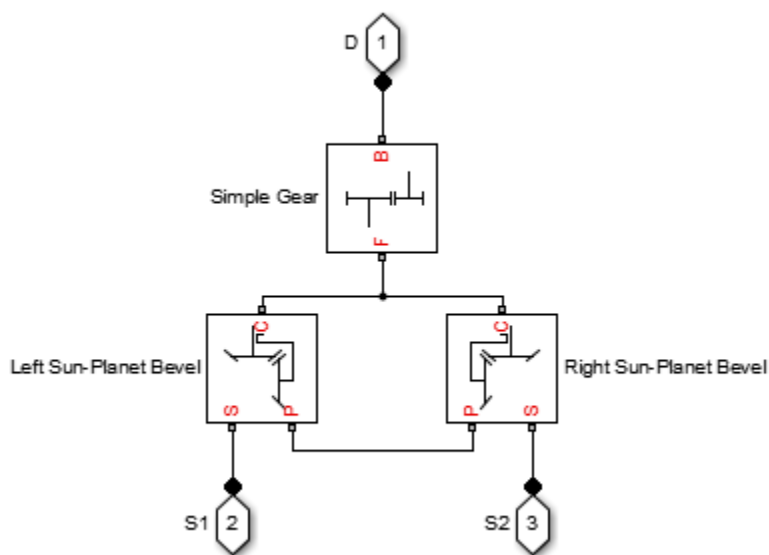
To increase the fidelity of the gear model, you can specify properties such as gear inertia, meshing losses, and viscous losses. By default, gear inertia and viscous losses are assumed negligible. The block enables you to specify the inertias of the gear carrier and internal planet gears only. To model the inertias of the outer gears, connect Simscape Inertia blocks to ports D, S1, and S2.

Thermal Modeling

You can model the effects of heat flow and temperature change through an optional thermal conserving port. By default, the thermal port is hidden. To expose the thermal port, right-click the block in your model and, from the context menu, select **Simscape > Block choices**. Select a variant that includes a thermal port. Specify the associated thermal parameters for the component.

Differential Gear Model

Ideal Gear Constraints and Gear Ratios

Differential imposes one kinematic constraint on the three connected axes:

$$\omega_D = \pm(1/2)g_D(\omega_{S1} + \omega_{S2}) ,$$

with the upper (+) or lower (–) sign valid for the differential crown to the right or left, respectively, of the centerline. The three degrees of freedom reduce to two independent degrees of freedom. The gear pairs are (1,2) = (S,S) and (C,D). C is the carrier.

The *sum* of the lateral motions is the transformed longitudinal motion. The *difference* of side motions $\omega_{S1} - \omega_{S2}$ is independent of the longitudinal motion. The general motion of the lateral shafts is a superposition of these two independent degrees of freedom, which have this physical significance:

- One degree of freedom (longitudinal) is equivalent to the two lateral shafts rotating at the same angular velocity ($\omega_{S1} = \omega_{S2}$) and at a fixed ratio with respect to the longitudinal shaft.

- The other degree of freedom (differential) is equivalent to keeping the longitudinal shaft locked ($\omega_D = 0$) while the lateral shafts rotate with respect to each other in opposite directions ($\omega_{S1} = -\omega_{S2}$).

The torques along the lateral axes, τ_{S1} and τ_{S2}, are constrained to the longitudinal torque τ_D in such a way that the power flows into and out of the gear, less any power loss P_{loss}, sum to zero:

$$\omega_{S1}\tau_{S1} + \omega_{S2}\tau_{S2} + \omega_D\tau_D - P_{loss} = 0 .$$

When the kinematic and power constraints are combined, the ideal case yields:

$$g_D\tau_D = 2(\omega_{S1}\tau_{S1} + \omega_{S2}\tau_{S2}) / (\omega_{S1} + \omega_{S2}) .$$

Fundamental Sun-Planet Bevel Gear Constraints

The effective differential constraint is composed of two subconstraints.

- The first comes from the coupling of the two sun-planet bevel gears to the carrier:

  $$(\omega_{S1} - \omega_C)/(\omega_{S2} - \omega_C) = -g_{SP2}/g_{SP1} .$$
- The second comes from the coupling of the carrier to the longitudinal driveshaft: $\omega_D = \pm g_D\omega_C$.

$g_{SP1} = r_{S1}/r_{P1}$ and $g_{SP2} = r_{S2}/r_{P2}$ are the sun-planet gear ratios of the underlying sun-planet bevel gears, in terms of the radii r of the sun and planet gears. The Differential block is implemented with $g_{SP1} = g_{SP2} = 1$, leaving g_D free to adjust.

Nonideal Gear Constraints and Losses

In the nonideal case, $\tau_{loss} \neq 0$. See "Model Gears with Losses".

Limitations

- Gears are assumed rigid.
- Coulomb friction slows down simulation. See "Adjust Model Fidelity".

Ports

Port	Description
D	Rotational conserving port representing the longitudinal driveshaft
S1	Rotational conserving port representing one of the sun gears
S2	Rotational conserving port representing one of the sun gears
H	Thermal conserving port for thermal modeling

Parameters

Main

Crown gear located

Select the placement of the bevel crown gear with respect to the centerline of the gear assembly. The default is `To the right of the centerline`.

Carrier (C) to driveshaft (D) teeth ratio (NC/ND)

Fixed ratio g_D of the carrier gear to the longitudinal driveshaft gear. The default is 4.

Meshing Losses

Parameters for meshing losses vary with the block variant chosen—one with a thermal port for thermal modeling and one without it.

Without Thermal Port

Friction model

Select how to implement friction losses from nonideal meshing of gear teeth. The default is `No meshing losses`.

- `No meshing losses — Suitable for HIL simulation` — Gear meshing is ideal.

- `Constant efficiency` — Transfer of torque between gear wheel pairs is reduced by a constant efficiency η satisfying $0 < \eta \le 1$. If you select this option, the panel changes from its default.

Constant Efficiency

Sun-sun and carrier-driveshaft ordinary efficiencies

Vector of torque transfer efficiencies $[\eta_{SS}\ \eta_D]$ for sun-sun and carrier-longitudinal driveshaft gear wheel pair meshings, respectively.

Sun-carrier and driveshaft-casing power thresholds

Vector of power thresholds p_{th} above which full efficiency loss is applied, for sun-carrier and longitudinal driveshaft-casing $[p_S\ p_D]$, respectively.

With Thermal Port

Temperature

Array of temperatures used to construct 1-D temperature-efficiency lookup tables. The array values must increase left to right. The temperature array must be the same size as each efficiency arrays. The default array is `[280 300 320]` K.

Sun-sun efficiency

Array of mechanical efficiencies with power flowing from one of the sun gears to the other. Each array value is the ratio of output power to input power at one of the temperatures in the temperature array. The temperature and efficiency arrays must be the same size. The default array is `[0.95 0.9 0.85]`.

Carrier-driveshaft efficiency

Array of mechanical efficiencies with power flowing from the gear carrier to the longitudinal driveshaft. Each array value is the ratio of output power to input power at one of the temperatures in the temperature array. The temperature and efficiency arrays must be the same size. The default array is `[0.95 0.9 0.85]`.

Sun-carrier and driveshaft-casing power thresholds

Array of power thresholds above which the full efficiency factors apply. Enter the thresholds for sun-carrier and longitudinal driveshaft-casing, in this order. A hyperbolic tangent function smooths the efficiency factors between zero when at rest and the values provided by the temperature-efficiency lookup tables at the power thresholds. The default array is `[0.001 0.001]` W.

Viscous Losses

Sun-carrier and driveshaft-casing viscous friction coefficients

Vector of viscous friction coefficients [μ_S μ_D] for the sun-carrier and longitudinal driveshaft-casing gear motions, respectively. The default is `[0 0]`.

From the drop-down list, choose units. The default is newton-meters/(radians/second) (`N*m/(rad/s)`).

Inertia

Carrier inertia

Moment of inertia of the planet gear carrier. This value must be positive or zero. Enter `0` to ignore carrier inertia. The default value is `0` kg*m^2.

Planet gear inertia

Moment of inertia of the combined planet gears. This value must be positive or zero. Enter `0` to ignore gear inertia. The default value is `0` kg*m^2.

Thermal Port

Thermal mass

Thermal energy required to change the component temperature by a single degree. The greater the thermal mass, the more resistant the component is to temperature change. The default value is `50` J/K.

Initial temperature

Component temperature at the start of simulation. The initial temperature alters the component efficiency according to an efficiency vector that you specify, affecting the starting meshing or friction losses. The default value is `300` K.

Real-Time Simulation

Hardware-in-the-Loop Simulation

For optimal simulation performance, use the **Meshing Losses** > **Friction model** parameter default setting, `No meshing losses - Suitable for HIL simulation`.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also
Disk Friction Clutch | Simple Gear | Sun-Planet Bevel

Topics
"Basic Motion, Torque, and Force Modeling"
"Limited Slip Differential with Clutches"
"Torsen Differential"
"Vehicle with Four-Wheel Drive"
"Model Thermal Losses in Driveline Components"
"Prepare Simscape Driveline Models for Real-Time Simulation Using Simscape Checks"

Disc Brake

Frictional brake with pressure-applying cylinder and pads
Library: Simscape / Driveline / Brakes & Detents / Rotational
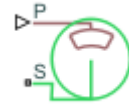
Description

The Disc Brake block represents a brake arranged as a cylinder applying pressure to one or more pads that can contact the shaft rotor. Pressure from the cylinder causes the pads to exert friction torque on the shaft. The friction torque resists shaft rotation.

Disc Brake Model

This figure shows the side and front views of a disc brake.

Pad

R_m

R_o R_i

Disc

Caliper

Piston

Pad

Backing plate

Brake fluid

Rotor/disc

Hub
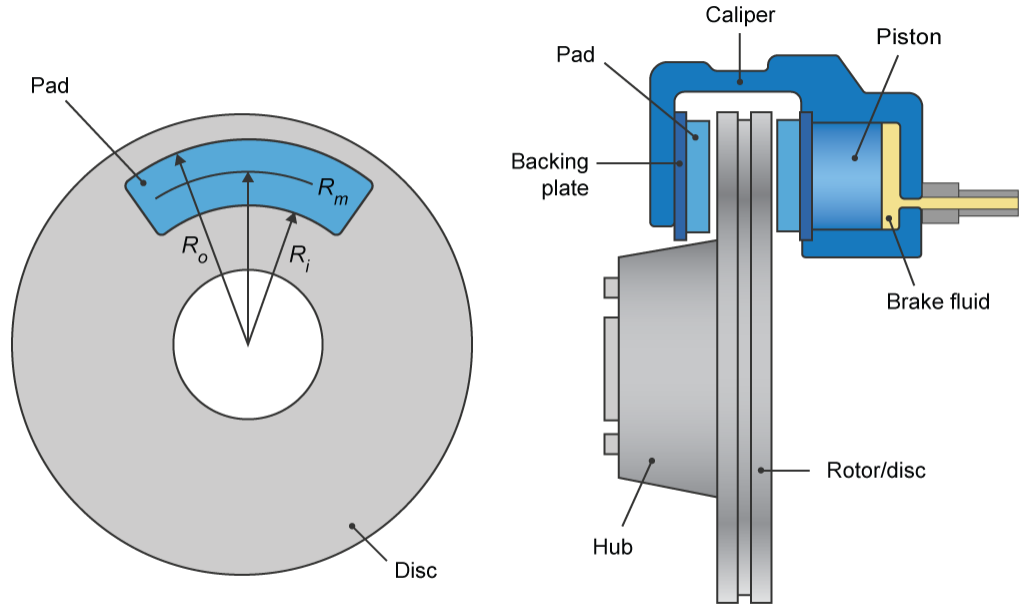
A disc brake converts brake cylinder pressure from the brake cylinder into force. The disc brake applies the force at the brake pad mean radius.

The equation that the block uses to calculate brake torque, depends on the wheel speed, Ω, such that when $\Omega \neq 0$,

$$T = \frac{\mu_k P \pi D_b{}^2 R_m N}{4}.$$

However when $\Omega = 0$, the torque applied by the brake is equal to the torque that is applied externally for wheel rotation. The maximum value of the torque that the brake can apply when $\Omega = 0$, is

$$T = \frac{\mu_s P \pi D_b{}^2 R_m N}{4}.$$

In any case, $Rm = \dfrac{Ro + Ri}{2}$.

In the equations:

- T is the brake torque.
- P is the applied brake pressure.
- Ω is the wheel speed.
- N is the number of brake pads in disc brake assembly.
- μ_s is the disc pad-rotor coefficient of static friction.
- μ_k is the disc pad-rotor coefficient of kinetic friction.
- D_b is the brake actuator bore diameter.
- R_m is the mean radius of brake pad force application on brake rotor.
- R_o is the outer radius of brake pad.
- R_i is the inner radius of brake pad.

Thermal Model

You can model the effects of heat flow and temperature change by selecting a thermal block variant. Selecting a thermal variant:

- Exposes port **H**, a conserving port in the thermal domain.
- Enables the **Thermal mass** parameter, which allows you to specify the ability of the component to resist changes in temperature.
- Enables the **Temperature** variable, which allows you to set the priority and initial target values for the block variables before simulating. For more information, see "Set Priority and Initial Target for Block Variables" (Simscape).

To select a thermal variant, right-click the block in your model and, from the context menu, select **Simscape > Block choices**. Select a variant that includes a thermal port.

Ports

Input

P — Cylinder pressure, bars
physical signal

Physical signal port associated with cylinder pressure.

Conserving

S — Shaft
rotational mechanical

Rotational mechanical conserving port associated with the shaft.

H — Heat
thermal

The thermal conserving port is optional and is hidden by default. To expose the port, select a variant that includes a thermal port.

Dependencies

Selecting a thermal variant enables thermal parameters.

Parameters

Geometry

Mean pad radius — Pad radius
150 mm (default)

Mean radius of the friction pads.

Cylinder bore — Bore diameter
10 mm (default)

Diameter of the piston.

Number of brake pads — Pad quantity
2 (default)

Number of friction pads.

Friction

Temperature — Temperature
[280, 300, 320] K (default) | increasing vector

Independent vector for determining static and kinetic frictions. The values in the vector, in K, must increase from left to right.

Dependencies

Selecting a thermal block variant for the **Block choice** parameter enables this parameter.

Static friction coefficient — Static friction
0.9 (default) | [.9, .8, .7]

Coefficient of static friction. The value that you specify for this parameter must be greater than the value that you specify for the **Coulomb friction coefficient** parameter.

Dependencies

If you select a thermal block variant for the **Block choice** parameter, specify this parameter using a vector. The vector must contain the same number of elements as the temperature vector. Each value must be greater than the value of the corresponding element in the kinetic friction coefficient vector.

Coulomb friction coefficient — Kinetic friction
0.7 (default) | [.8, .7, .6]

Coefficient of kinetic friction. The value that you specify for this parameter must be greater than 0.

Dependencies

If you select a thermal block variant for the **Block choice** parameter, specify this parameter as a vector. The vector must contain the same number of elements as the temperature vector.

Breakaway friction velocity — Friction threshold
`0.1 rad/s` (default)

Angular speed at which friction switches from static to kinetic.

Viscous friction coefficient — Viscous friction
`.001 N*m/(rad/s)` (default)

For a wet brake, the viscous friction represents the energy loss to the cooling/lubricating fluid between the clutch plates. To model a wet brake, specify a nonzero value for the coefficient of viscous friction. To model a dry brake, specify `0`.

Thermal Port

Thermal mass — Resistance to temperature change
`50 kj/K` (default)

Thermal energy required to change the component temperature by a single degree. The greater the thermal mass, the more resistant the component is to temperature change.

Dependencies

Selecting a thermal block variant for the **Block choice** parameter enables this parameter.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also
Band Brake | Double-Shoe Brake | Loaded-Contact Rotational Friction | Rotational Detent

Introduced in R2017b

Disk Friction Clutch

Friction clutch with disk plates that engage when plate pressure exceeds threshold

Library

Simscape / Driveline / Clutches

Description

This block represents a friction clutch with two flat friction plate sets that come into contact to engage. The clutch engages when the applied plate pressure exceeds an engagement threshold pressure. Once engaged, the plates experience frictional torques that enable them to transmit power between the base and follower driveshafts.

The clutch can be bidirectional or unidirectional. A bidirectional clutch can slip in the positive and negative directions. A unidirectional clutch can slip only in the positive direction. The slip direction is positive if the follower shaft spins faster than the base shaft and negative if it slips slower. The block defines the slip velocity as the difference

$$\omega = \omega_F - \omega_B,$$

where:

- ω is the slip velocity.
- ω_F is the angular velocity of the follower driveshaft.
- ω_B is the angular velocity of the base driveshaft.

The block provides a physical signal input port (P) for the applied pressure between the clutch plates. The applied pressure must be greater than or equal to zero and has units of Pascals. If the input signal falls below zero, the block treats the plate pressure as zero.

1-109

Disk Friction Clutch Model

The Disk Friction Clutch is based on the Fundamental Friction Clutch. For the complete friction clutch model, consult the Fundamental Friction Clutch block reference page. This section discusses the simplified model implemented in the Disk Friction Clutch.

When you apply a pressure signal above threshold ($P \geq P_{th}$), the Disk Friction Clutch block can apply two kinds of friction to the driveline motion, kinetic and static. The clutch applies kinetic friction torque only when one driveline axis is spinning relative to the other driveline axis. The clutch applies static friction torque when the two driveline axes lock and spin together. The block iterates through multistep testing to determine when to lock and unlock the clutch.

Clutch Variable, State, and Mode Summary

This table summarizes the clutch variables.

Clutch Variables

Symbol	Definition	Significance
ω	Relative angular velocity (slip)	$\omega_F - \omega_B$
ω_{Tol}	Slip tolerance for clutch locking	*See the following model*
P, P_{th}	Clutch pressure and threshold	Input pressure applied to clutch discs; threshold clutch pressure: P_{th}, $P > 0$.
P_{fric}	Clutch friction capacity	$\max[(P - P_{th}), 0]$
D	Clutch de-rating factor	*See the following model*
N	Number of friction surfaces	*See the following model*
A	Engagement surface area	*See the following model*
r_{eff}	Effective torque radius	Effective moment arm of clutch friction force
k_K	Kinetic friction coefficient	Dimensionless coefficient of kinetic friction of clutch discs. Function of ω.
k_S	Static friction coefficient	Dimensionless coefficient of static friction of clutch discs.
μ	Viscous drag coefficient	*See the following model*
τ_K	Kinetic friction torque	*See the following model*
τ_S	Static friction torque limit	(static friction peak factor)·(kinetic friction torque for $\omega \to 0$) (*See the following model*)

Relation to Fundamental Friction Clutch

Instead of requiring the kinetic and static friction limit torques as input signals, the Disk Friction Clutch calculates the kinetic and static friction from the clutch parameters and the input pressure signal P.

Kinetic Friction

The kinetic friction torque is the positive sum of viscous drag and surface contact friction torques:

$$\tau_K \qquad = \qquad \mu\omega \qquad + \qquad \tau_{contact} \qquad .$$

(The kinetic friction torque opposes the relative slip and is applied with an overall minus sign.) The contact friction is a product of six factors:

$$\tau_{contact} \qquad = \qquad k_K{\cdot}D{\cdot}N{\cdot}r_{eff}{\cdot}P_{fric}{\cdot}A \qquad \geq \qquad 0 \qquad .$$

You specify the *kinetic friction coefficient* k_K as either a constant or a tabulated discrete function of relative angular velocity ω. The tabulated function is assumed to be symmetric for positive and negative values of the relative angular velocity. Therefore, specify k_K for positive values of ω only.

The clutch applies a normal force from its piston as the product of the clutch friction capacity P_{fric} and engagement surface area A, on each of N friction surfaces. The pressure signal P should be nonnegative. If P is less than P_{th}, the clutch applies no friction at all.

The *effective torque radius* r_{eff} is the effective radius, measured from the driveline axis, at which the kinetic friction forces are applied at the frictional surfaces. It is related to the geometry of the friction surface by:

$$r_{eff} = \frac{2}{3}\frac{r_o{}^3 - r_i{}^3}{r_o{}^2 - r_i{}^2}$$

r_o and r_i are the outer and inner radii, respectively, of the friction surface, modeled as an annular disk.

The *clutch de-rating factor D* accounts for clutch wear. For a new clutch, D is one. For a clutch approaching a "uniform wear" state:

$$D \rightarrow \frac{3}{4}\frac{(r_o + r_i)^2}{r_o{}^2 + r_o r_i + r_i{}^2}$$

Static Friction

The static friction limit is related to the kinetic friction, setting ω to zero and replacing the kinetic with the static friction coefficient:

$$\tau_S \qquad = \qquad k_S{\cdot}D{\cdot}N{\cdot}r_{eff}{\cdot}P_{fric}{\cdot}A \qquad \geq \qquad 0 \qquad .$$

$k_S > k_K$, so that the torque τ needed across the clutch to unlock it by overcoming static friction is larger than the kinetic friction at the instant of unlocking, when $\omega = 0$.

The *static friction torque range* or limits are then defined symmetrically as:

$$\tau_S \qquad \equiv \qquad \tau_S^+ \qquad = \qquad -\tau_S^- \qquad .$$

Wait State: Locking and Unlocking

The Wait state of the Disk Friction Clutch is identical to the Wait state of the Fundamental Friction Clutch, with the replacement of the positive kinetic friction condition ($\tau_K > 0$) by the positive clutch friction capacity condition ($P \geq P_{th}$).

Power Dissipated by the Clutch

The power dissipated by the clutch is $|\omega \cdot \tau_K|$. The clutch dissipates power only if it is both slipping ($\omega \neq 0$) and applying kinetic friction ($\tau_K > 0$).

Thermal Modeling

You can model the effects of heat flow and temperature change through an optional thermal conserving port. By default, the thermal port is hidden. To expose the thermal port, right-click the block in your model and, from the context menu, select **Simscape** > **Block choices**. Select a variant that includes a thermal port. Specify the associated thermal parameters for the component.

Ports

B

    Rotational conserving port that represents the base driveshaft

F

    Rotational conserving port that represents the follower driveshaft

P

    Physical signal input port for the applied pressure between the clutch plates

H

    Thermal conserving port. The thermal port is optional and is hidden by default. To expose the port, select a variant that includes a thermal port.

Parameters

Geometry

Force action region

Select a parameterization method to model the clutch friction geometry. The default method is `Define effective radius`.

- `Define effective radius` — Model friction geometry in terms of disk radius.

  Effective Radius

  Effective torque radius

  Effective moment arm radius r_{eff} that determines the kinetic friction torque inside the clutch. The default value is `130 mm`.

  Number of friction surfaces

  Number N of friction-generating contact surfaces inside the clutch. The default value is `4`.

  Engagement piston area

  Effective area A of the clutch piston when the piston is applying pressure across the clutch. The default value is `0.001 m^2`.

  Directionality

  Slip directions the clutch allows between its plates. A bidirectional clutch allows positive and negative slip velocities. A unidirectional clutch allows only positive slip velocities. The default setting is `Bidirectional`.

  The unidirectional clutch is equivalent to a friction clutch connected in parallel to a one-way clutch that disengages only when the slip velocity becomes positive. To model a unidirectional clutch with slip in the negative direction, reverse the base and follower port connections.

- `Define annular region` — Model friction geometry in terms of annulus dimensions. If you select this option, the visible parameters change.

  Annular Region

  Friction disk outside diameter

  Diameter $2r_{\text{o}}$ across the outer edge of the friction disk annulus. The default value is `150 mm`.

Friction disk inside diameter

Diameter $2r_i$ across the inner edge of the friction disk annulus. The default value is `100` mm.

Number of friction surfaces

Number N of friction-generating contact surfaces inside the clutch. The default value is `4`.

Engagement piston area

Effective area A of the clutch piston when the piston is applying pressure across the clutch. The default value is `0.001` m^2.

Directionality

Slip directions the clutch allows between its plates. A bidirectional clutch allows positive and negative slip velocities. A unidirectional clutch allows only positive slip velocities. The default setting is `Bidirectional`.

The unidirectional clutch is equivalent to a friction clutch connected in parallel to a one-way clutch that disengages only when the slip velocity becomes positive. To model a unidirectional clutch with slip in the negative direction, reverse the base and follower port connections.

Friction

Friction model

Select a parameterization method to model the kinetic friction coefficient. The options and default values for this parameter depend on the variant that you select for the block.

The options are:

- `Fixed kinetic friction coefficient` — Provide a fixed value for the kinetic friction coefficient. This option:

  - Is only visible if you use the default variant of the block
  - Is the default method for parameterizing the default variant of the block
  - Affects the visibility of other parameters

Fixed kinetic friction coefficient

Kinetic friction coefficient

The kinetic, or Coulomb, friction coefficient. The coefficient must be greater than zero. The default value is `0.3`.

Static friction coefficient

Dimensionless Coulomb static friction coefficient k_S applied to the normal force across the clutch when the clutch is locked. k_S must be larger than k_K. The default value is `0.35`.

- `Table lookup kinetic friction coefficient` — Define the kinetic friction coefficient by one-dimensional table lookup based on the relative angular velocity between disks. This option:

  - Is only visible if you use the default variant of the block
  - Affects the visibility of other parameters

Table lookup kinetic friction coefficient

Kinetic friction coefficient relative velocity vector

Specify input values for the relative velocity as a vector. The values in the vector must increase from left to right. The minimum number of values depends on the interpolation method that you select. For linear interpolation, provide at least two values per dimension. For smooth interpolation, provide at least three values per dimension. The default value is `[0.0, 100.0, 1000.0] rad/s`.

Kinetic friction coefficient vector

Specify the output values for the kinetic friction coefficient as a vector. All values must be positive. The default value is `[0.3, 0.22, 0.19]`.

Friction coefficient interpolation method

Select one of these interpolation methods to approximate the output value when the input value is between two consecutive grid points:

- `Linear` — Select this, default, option to get the best performance.
- `Smooth` — Select this option to produce a continuous curve with continuous first-order derivatives.

For more information on interpolation algorithms, see the PS Lookup Table (1D) block reference page.

Friction coefficient extrapolation method

Select one of these extrapolation methods to determine the output value when the input value is outside the range specified in the argument list:

- `Linear` — Select this, default, option to produce a curve with continuous first-order derivatives in the extrapolation region and at the boundary with the interpolation region.
- `Nearest` — Select this option to produce an extrapolation that does not go above the highest point in the data or below the lowest point in the data.

For more information on extrapolation algorithms, see the PS Lookup Table (1D) block reference page.

Static friction coefficient

Dimensionless Coulomb static friction coefficient k_S applied to the normal force across the clutch when the clutch is locked. k_S must be larger than k_K. The default value is `0.35`.

- `Temperature-dependent kinetic friction coefficient` — Define the kinetic friction coefficient by table lookup based on the temperature. This option:

  - Is only visible if you use a thermal variant of the block
  - Is the default method for parameterizing the thermal variant of the block
  - Affects the visibility of other parameters

`Temperature-dependent kinetic friction coefficient`

Temperature vector

Specify the input values for the temperature as a vector. The minimum number of values depends on the interpolation method that you select. For linear interpolation, provide at least two values per dimension. For smooth interpolation, provide at least three values per dimension. The values in the vector must increase from left to right. The default value is `[280.0, 300.0, 320.0]` K.

Kinetic friction coefficient vector

Specify the output values for kinetic friction coefficient as a vector. The vector must have the same number of elements as the temperature vector. All values must be greater than zero. The default value is `[0.3, 0.28, 0.25]`.

Static friction coefficient vector

Specify the static, or peak, values of the friction coefficient as a vector. The vector must have the same number of elements as the temperature vector.

Each value must be greater than the value of the corresponding element in the kinetic friction coefficient vector. The default value is `[0.4, 0.38, 0.36]`.

Friction coefficient interpolation method

Select one of these interpolation methods to approximate the output value when the input value is between two consecutive grid points:

- `Linear` — Select this, default, option to get the best performance.
- `Smooth` — Select this option to produce a continuous curve with continuous first-order derivatives.

For more information on interpolation algorithms, see the PS Lookup Table (1D) block reference page.

Friction coefficient extrapolation method

Select one of these extrapolation methods to determine the output value when the input value is outside the range specified in the argument list:

- `Linear` — Select this, default, option to produce a curve with continuous first-order derivatives in the extrapolation region and at the boundary with the interpolation region.
- `Nearest` — Select this option to produce an extrapolation that does not go above the highest point in the data or below the lowest point in the data.

For more information on extrapolation algorithms, see the PS Lookup Table (1D) block reference page.

- `Temperature and speed-dependent kinetic friction coefficient` — Define the kinetic friction coefficient by table lookup based on the temperature and the relative angular velocity between disks. This option:

  - Is only visible if you use the thermal variant of the block
  - Affects the visibility of other parameters

`Temperature and speed-dependent kinetic friction coefficient`

Temperature vector

Specify the input values for the temperature as a vector. The minimum number of values depends on the interpolation method that you select. For linear interpolation, provide at least two values per dimension. For smooth interpolation, provide at least three values per dimension. The values in the vector must increase from left to right. The default value is `[280.0, 300.0, 320.0]` K.

Kinetic friction coefficient relative velocity vector

Specify input values for the relative velocity as a vector. The values in the vector must increase from left to right. The minimum number of values depends on the interpolation method that you select. For linear interpolation, provide at least two values per dimension. For smooth interpolation, provide at least three values per dimension. The default value is `[0.0, 100.0, 1000.0] rad/s`.

Kinetic friction coefficient matrix

Specify the output values for kinetic friction coefficient as a matrix. All the values must be greater than zero. The size of the matrix must equal the size of the matrix that is the result of the temperature vector × the kinetic friction coefficient relative velocity vector. The default value is `[0.34, 0.32, 0.3; 0.3, 0.28, 0.25; 0.25, 0.2, 0.15]`.

Static friction coefficient vector

Specify the static, or peak, values of the friction coefficient as a vector. The vector must have the same number of elements as the temperature vector. Each value must be greater than the value of the corresponding element in the kinetic friction coefficient vector. The default value is `[0.4, 0.38, 0.36]`.

Friction coefficient interpolation method

Select one of these interpolation methods to approximate the output value when the input value is between two consecutive grid points:

- `Linear` — Select this, default, option to get the best performance.
- `Smooth` — Select this option to produce a continuous curve with continuous first-order derivatives.

For more information on interpolation algorithms, see the PS Lookup Table (1D) block reference page.

Friction coefficient extrapolation method

Select one of these extrapolation methods to determine the output value when the input value is outside the range specified in the argument list:

- `Linear` — Select this, default, option to produce a curve with continuous first-order derivatives in the extrapolation region and at the boundary with the interpolation region.
- `Nearest` — Select this option to produce an extrapolation that does not go above the highest point in the data or below the lowest point in the data.

For more information on extrapolation algorithms, see the PS Lookup Table (1D) block reference page.

De-rating factor

Dimensionless de-rating factor D that accounts for clutch disk wear by proportionately reducing clutch friction. The default value is `1`.

Clutch velocity tolerance

Maximum slip velocity at which the clutch can lock. The slip velocity is the signed difference between the base and follower shaft angular velocities, that is, $w = w_F - w_B$. If the kinetic friction torque is nonzero and the transferred torque is within the static friction torque limits, then the clutch locks if the actual slip velocity falls below the velocity tolerance. The default value is `0.001 rad/s`.

Engagement threshold pressure

Minimum pressure P_{th} at which the clutch engages. If the pressure input signal falls below this threshold, the clutch automatically disengages. The default value is `100 Pa`.

Viscous Losses

Viscous drag torque coefficient

Viscous friction coefficient μ applied to the relative slip ω between the base and follower axes. The default value is `0 N*m/(rad/s)`.

Initial Conditions

Initial state

Clutch state at the start of simulation. The clutch can be in one of two states, locked and unlocked. A locked clutch constrains the base and follower shafts to spin at the same velocity, that is, as a single unit. An unlocked clutch allows the two shafts to spin at different velocities, resulting in slip between the clutch plates. The default setting is `Unlocked`.

Thermal Port

These thermal parameters are only visible when you select a block variant that includes a thermal port.

Thermal mass

Thermal energy required to change the component temperature by a single degree. The greater the thermal mass, the more resistant the component is to temperature change. The default value is 25 kJ/K.

Initial temperature

Component temperature at the start of simulation. The initial temperature alters the component efficiency according to an efficiency vector that you specify, affecting the starting meshing or friction losses. The default value is 300 K.

Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using MATLAB® Coder™.

See Also

Differential | Fundamental Friction Clutch | Torque Converter | Variable Ratio Transmission

Topics

"Gear Coupling Control Using Clutches"
"Clutches For Accelerating and Braking"
"CR-CR Four-Speed Transmission"
"Custom Clutch"
"Hydraulically-Actuated Driveline Clutch"
"Two-Speed Transmission"
"Vehicle with Dual Clutch Transmission"
Vehicle with Four-Speed Transmission

Dog Clutch

Clutch with toothed plates that engage when plate teeth become enmeshed
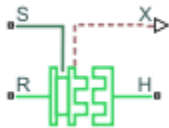
Library

Simscape / Driveline / Clutches

Description

This block represents a nonslip clutch, a mechanical device that relies on the positive engagement of interlocking teeth to transfer torque between driveline shafts. The clutch contains three key components:

- Ring
- Hub
- Shift linkage

The ring and the hub are toothed components. The ring spins with the output shaft, sliding along its longitudinal axis to engage or disengage the coaxial hub. The hub, which sits on a bearing encircling the same shaft, can spin independently until engaged.

Engagement occurs when the toothed components interlock. Once engaged, the ring and the hub spin together as a unit. To control engagement, the dog clutch contains a shift linkage that governs the position of the ring with respect to the hub.

Moving the ring towards the hub so that their teeth interlock changes the clutch state to engaged. Tooth overlap must exceed a minimum value for engagement. Moving the ring in reverse so that the two sets of teeth no longer interlock changes the clutch state back to disengaged.

The translational conserving port S specifies the shift linkage position. When the clutch is fully disengaged, the shift linkage position is zero. When the clutch is fully engaged, the shift linkage position equals the sum of the tooth height and the ring-hub clearance of the fully disengaged state:

$$z = h + z_{Gap},$$

where:

- z is the shift linkage position.
- h is the tooth height.
- z_{Gap} is the ring-hub clearance when disengaged.

The figure shows side and front views of the dog clutch and some of its relevant variables.
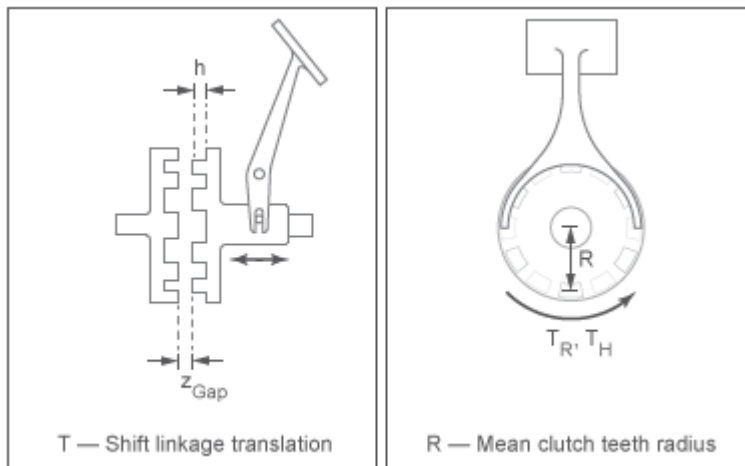
Torque Transmission Models

The Dog Clutch block provides a choice of two torque transmission models.

Friction Clutch Approximate Model

Treat clutch engagement as a friction phenomenon between the ring and the hub. This model ignores special effects such as backlash, an approximation that makes the block

better suited for linearization, fixed-step simulation, and hardware-in-loop (HIL) simulation. The Fundamental Friction Clutch block provides the foundation for the model.

In the friction approximate model, the clutch has three possible configurations: disengaged, engaged, and locked. When disengaged, the contact force between the ring and the hub is zero. This force remains zero until the shift linkage reaches the minimum position for engagement.

When the ring-hub tooth overlap (h) exceeds the minimum value for engagement, the contact force between the two components begins to increase linearly with the shift linkage position (z).

At full engagement, the contact force reaches its maximum value, and the clutch state switched to locked. In this state, the ring and the hub spin as a unit without slip. To unlock the clutch, the transmitted torque must exceed the maximum allowed value that you specify.

Dynamic Model with Backlash

Capture clutch phenomena such as backlash, torsional compliance, and contact forces between ring and hub teeth. This model provides greater accuracy than the friction clutch approximation.

In the dynamic model, the clutch has two possible configurations: disengaged and engaged. When disengaged, the contact force between the ring and the hub is zero. This force remains zero until the shift linkage reaches the minimum position for engagement.

When the ring-hub tooth overlap (h) exceeds the minimum value for engagement, a contact force kicks in between the two components. This force is the sum of torsional spring and damper components. Including backlash between the ring and hub teeth:

$$T_C = \begin{cases} -k_{RH}\left(\phi - \dfrac{\delta}{2}\right) - \mu_R \cdot \omega & \phi > \dfrac{\delta}{2} \\[2mm] 0 & -\dfrac{\delta}{2} < \phi < \dfrac{\delta}{2}, \\[2mm] -k_{RH}\left(\phi + \dfrac{\delta}{2}\right) - \mu_R \omega & \phi < -\dfrac{\delta}{2} \end{cases}$$

where:

- k_{RH} is the torsional stiffness of the ring-hub coupling.

- ϕ is the relative angle, about the common rotation axis, between the ring and the hub.
- δ is the backlash between ring and hub teeth.
- ω is the relative angular velocity between the ring and the hub. This variable describes how fast the two components slip past each other.

Compliant end stops limit the translational motion of the clutch shift linkage and the ring. The compliance model treats the end stops as linear spring-damper sets. The location of the end stops depends on the relative angle and angular velocity between the ring and hub teeth:

- If the teeth align and the relative angular velocity is smaller than the maximum value for clutch engagement, the end-stop location is the sum of the ring-hub clearance when fully disengaged and the tooth height. With the end stop at this location, the clutch can engage.
- If the teeth do not align or the relative angular velocity exceeds the maximum value for clutch engagement, the end-stop location is set to prevent the ring from engaging the hub. The clutch remains disengaged.

Translational friction opposes shift linkage and ring motion. This friction is the sum of Coulomb and viscous components:

$$F_Z = -k_K \cdot F_N \cdot \tanh\left(\frac{4v}{v_{th}}\right) - \mu_T v,$$

where:

- F_Z is the net translational friction force acting on the shift linkage and ring.
- k_K is the kinetic friction coefficient between ring and hub teeth.
- F_N is the normal force between ring and hub teeth.
- v is the translational velocity of the shift linkage and the ring.
- v_{th} is the translational velocity threshold. Below this threshold, a hyperbolic tangent function smooths the Coulomb friction force to zero as the shift linkage and ring velocity tends to zero.
- μ_T is the viscous damping coefficient acting on the shift linkage and the ring.

Clutch Engagement Conditions

The clutch engages when it satisfies a set of geometrical and dynamic conditions. These conditions specify the values that certain variables can take for clutch engagement to occur:

- The minimum position at which the ring and the hub can engage is

    $z = h_0 + z_{Gap}$,

    where h_0 is the minimum tooth overlap for clutch engagement. Adjust this parameter to minimize engagement instability, that is, the tendency of the clutch to switch rapidly between engaged and disengaged states

- The magnitude of the relative angular velocity between the ring and the hub must be smaller than the maximum engagement velocity:

    $|\omega| < |\omega_{max}|$,

    where ω_{max} is the maximum value of the relative angular velocity at which engagement can occur.

- If using the friction clutch approximate model, engagement occurs only if torque transfer between the ring and the hub remains smaller than the maximum transmitted torque that the clutch supports.

- If using the dynamic model with backlash, engagement occurs only if the relative angular position of the ring and hub teeth allows them to interlock.

Rotational Power Dissipation

When the clutch slips under an applied torque, it dissipates power. The power loss equals the product of the slip angular velocity and the contact torque between the ring and the hub:

$$P_{loss} = \omega \cdot T_C,$$

where:

- P_{loss} is the dissipated power due to slipping.
- T_C is the kinetic contact torque.

Shift Linkage and Thermal Variants

The block provides four variants:

- **Mechanical port shift linkage**
- **Mechanical port shift linkage and thermal port**
- **Physical signal position input**
- **Physical signal position input and thermal port**

One of the two shift linkage variants accepts the position input through a physical signal port, the other through a translational conserving port. To model thermal effects, you can add a thermal port to either of the shift linkage variants.

To change from the current variant, right-click the block in your model and, under **Simscape** > **Block choices**, and select the desired variant. Changing the variant changes the ports and parameters.

Ports

S

Translational conserving port or physical signal port that represents the shift linkage. The port type depends on the shift linkage variant that you select.

X

Physical signal port for sensing the clutch position

T

Thermal conserving port. The thermal port is optional and is hidden by default. To expose the port, select a variant that includes a thermal port.

R

Rotational conserving port that represents the clutch ring

H

Rotational conserving port that represents the clutch hub

Thermal Modeling

You can model the effects of heat flow and temperature change through an optional thermal conserving port. By default, the thermal port is hidden. To expose the thermal

port, right-click the block in your model and, from the context menu, select **Simscape > Block choices**. Select a variant that includes a thermal port. Specify the associated thermal parameters for the component.

Parameters

Clutch

If you select **Physical signal position input** or **Mechanical port shift linkage** as the variant for the Dog Clutch block, parameter visibility depends on the value that you select for the **Torque transmission model** parameter.

Torque transmission model

Torque transmission model

> Computational framework for modeling the dynamic behavior of the dog clutch. The default setting is `Friction clutch approximation — Suitable for HIL and linearization`.

> - `Friction clutch approximation - Suitable for HIL and linearization` — Model clutch engagement as a friction phenomenon between the ring and the hub. This model, based on the Fundamental Friction Clutch block, provides a computationally efficient approximation of the dog clutch.

>   **Friction Clutch Approximation**

>   **Maximum transmitted torque**

>   > Largest torque that the clutch can transmit, corresponding to a nonslip engaged configuration. If the torque transmitted between the ring and the hub exceeds this value, the two components begin to slip with respect to each other. This torque determines the static friction limit in the friction clutch approximation. The default is `1000 N*m`.

>   **Clutch teeth mean radius**

>   > Distance from the ring or hub center to the corresponding tooth center. The mean tooth radius determines the normal contact forces between ring and hub teeth given the transmission torque between the two components. The mean radius is greater than zero. The default value is `50 mm`.

> - `Dynamic with backlash` — Model clutch engagement in detail, accounting for such phenomena as backlash, torsional compliance, and contact forces between

ring and hub teeth. Selecting this option causes additional **Shift Linkage** and **Initial Conditions** parameters to appear.

Dynamic with Backlash

Number of teeth

Total number of teeth in the ring or the hub. The two components have equal tooth numbers. The number of teeth is greater than or equal to one. The default value is 6.

Rotational backlash

Allowable angular motion, or play, between the ring and hub teeth in the engaged clutch configuration. The backlash is greater than zero. The default value is 10 deg.

Torsional stiffness

Linear torsional stiffness coefficient at the contact interface between the ring and hub teeth. This coefficient characterizes the restoring component of the contact force between the two sets of teeth. Greater stiffness values correspond to greater contact forces. The stiffness coefficient is greater than zero. The default value is 1e+7 N*m/rad.

Torsional damping

Linear torsional damping coefficient at the contact interface between the ring and hub teeth. This coefficient characterizes the dissipative component of the contact force between the two sets of teeth. Greater damping values correspond to greater energy dissipation during contact. The damping coefficient is greater than zero. The default value is 100 N*m/(rad/s).

Selecting a thermal variant for the Dog Clutch block, makes other **Clutch** parameters visible.

Clutch Parameters for Thermal Variants

Temperature vector

Specify the input values for the temperature as a vector. The minimum number of values depends on the interpolation method that you select. For linear interpolation, provide at least two values per dimension. For smooth interpolation, provide at least three values per dimension. The values in the vector must increase from left to right. The default value is [280.0, 300.0, 320.0] K.

Maximum transmitted torque vector

Largest torque that the clutch can transmit, corresponding to a nonslip engaged configuration, specified as a vector. If the torque transmitted between the ring and the hub exceeds this value, the two components begin to slip with respect to each other. This torque determines the static friction limit in the friction clutch approximation. The vector has the same number of elements as the temperature vector. The default value is [1000.0, 1050.0, 1000.0] N*m.

Interpolation method

Select one of these interpolation methods to approximate the output value when the input value is between two consecutive grid points:

- Linear — Select this, default, option to get the best performance.
- Smooth — Select this option to produce a continuous curve with continuous first-order derivatives.

For more information on interpolation algorithms, see the PS Lookup Table (1D) block reference page.

Extrapolation method

Select one of these extrapolation methods to determine the output value when the input value is outside the range specified in the argument list:

- Linear — Select this, default, option to produce a curve with continuous first-order derivatives in the extrapolation region and at the boundary with the interpolation region.
- Nearest — Select this option to produce an extrapolation that does not go above the highest point in the data or below the lowest point in the data.

For more information on extrapolation algorithms, see the PS Lookup Table (1D) block reference page.

Clutch teeth mean radius

Distance from the ring or hub center to the corresponding tooth center. The mean tooth radius determines the normal contact forces between ring and hub teeth given the transmission torque between the two components. The mean radius is greater than zero. The default value is 50 mm.

Shift Linkage

Tooth height

Distance between the base and crest of a tooth. Ring and hub teeth share the same height. The tooth height and the ring-hub clearance when fully disengaged determine the maximum travel span of the shift linkage. The tooth height is greater than zero. The default value is `10 mm`.

Hard stop at back of shift linkage

Choice of translational motion limiting device. Select whether to stop the shift linkage when fully disengaged. The default option is `Hard stop when fully disengaged`.

This parameter is only visible if you select one of these variants for the block:

- **Mechanical port shift linkage**
- **Mechanical port shift linkage and thermal port**

Ring-hub clearance when disengaged

Maximum open gap between the ring and hub tooth crests along the shift linkage translation axis. This gap corresponds to the fully disengaged clutch state. The tooth height and the ring-hub clearance when fully disengaged determine the maximum travel span of the shift linkage. The ring-hub clearance is greater than zero. The default value is `3 mm`.

Ring stop stiffness

Linear stiffness coefficient of the ring end stop. This coefficient characterizes the restoring component of the contact force that resists translational motion past the end stops. Greater stiffness values correspond to greater contact forces and a smaller end stop compliance. The stiffness coefficient is greater than zero. The default value is `1e+6 N/m`.

This parameter is only visible if you select one of these variants for the block:

- **Mechanical port shift linkage**
- **Mechanical port shift linkage and thermal port**

Ring stop damping

Linear damping coefficient of the ring end stop. This coefficient characterizes the dissipative component of the contact force that resists translational motion past the end stops. Greater damping values correspond to greater energy dissipation during

contact. The damping coefficient is greater than or equal to zero. The default value is `1000 N/(m/s)`.

This parameter is only visible if you select one of these variants for the block:

- **Mechanical port shift linkage**
- **Mechanical port shift linkage and thermal port**

Shift linkage viscous friction coefficient

Linear damping coefficient acting on the shift linkage. This coefficient characterizes the dissipative force that resists shift linkage motion due to viscous damping. Greater coefficient values correspond to greater energy dissipation during shift linkage motion. The viscous friction coefficient is greater than zero. The default value is `100 N/(m/s)`.

This parameter is only visible if you select one of these variants for the block:

- **Mechanical port shift linkage**
- **Mechanical port shift linkage and thermal port**

Tooth-tooth friction coefficient

Kinetic friction coefficient at the contact interface between ring and hub teeth. This coefficient characterizes the dissipative force that resists shift linkage motion due to tooth-tooth contact during clutch engagement/disengagement.

Greater coefficient values correspond to greater energy dissipation during shift linkage motion. The friction coefficient is greater than zero. The default value is `0.05`.

This parameter is only visible if you select the **Mechanical port shift linkage** variant for the block and set the **Torque transmission model** parameter to `Dynamic with backlash`.

Engagement Conditions

Linkage travel direction

Direction the shift linkage must travel in to engage the clutch. Choices include positive and negative displacements. The default setting is `Positive shift linkage displacement engages clutch`.

Maximum engagement velocity

Relative angular velocity between the ring and the hub above which the clutch cannot engage. The maximum engagement velocity is greater than zero. The default value is `inf` (infinity) rad/s.

This parameter is only visible if you select one of these variants for the block:

- **Mechanical port shift linkage**
- **Mechanical port shift linkage and thermal port**

Tooth overlap to engage

Overlap length between ring and hub teeth along the common longitudinal axis above which the clutch can engage. The clutch remains disengaged until the tooth overlap by at least this length. The tooth overlap to engage is greater than zero. The default value is 3 mm.

Initial Conditions

Clutch Initial state

Clutch configuration at the start of simulation. Options include:

- `Disengaged` — Clutch transmits zero torque between the ring and the hub. `Disengaged` is the default value.
- `Engaged` — Clutch transmits torque between the ring and the hub.

If you select **Physical signal position input** for the block variant, this parameter is only visible if you set the **Torque transmission model** parameter to `Friction clutch approximation - Suitable for HIL and linearization`.

Initial shift linkage position

Shift linkage position at simulation time zero. Values between zero and the sum of the ring-hub clearance and the tooth overlap to engage are consistent with a disengaged clutch. Larger values are consistent with an engaged clutch. The default value is `0` mm.

This parameter is only visible if you select one of these variants for the block:

- **Mechanical port shift linkage**
- **Mechanical port shift linkage and thermal port**

Initial ring-hub offset angle

Rotation angle between the ring and the hub at simulation time zero. This angle determines whether the ring and hub teeth can interlock, and hence whether the clutch can engage. The initial offset angle must satisfy these conditions:

- If the clutch initial state is disengaged, the initial offset angle must fall in the range

$$-\frac{180°}{N} \le \phi_0 \le +\frac{180°}{N},$$

  where N is the number of teeth present in the ring or the hub. The two components contain the same number of teeth.

- If the clutch initial state is engaged, the initial offset angle must fall in the range

$$-\frac{\delta}{2} \le \phi_0 \le +\frac{\delta}{2},$$

  where δ is the backlash angle between the ring and hub teeth.

The default value is 0 deg.

This parameter is only visible if you satisfy both of these conditions:

- For the block variant, select **Mechanical port shift linkage** or **Physical signal position input**
- Set the **Torque transmission model** parameter to `Dynamic with backlash`.

Thermal Port

These thermal parameters are only visible when you select a block variant that includes a thermal port.

Thermal mass

Thermal energy required to change the component temperature by a single degree. The greater the thermal mass, the more resistant the component is to temperature change. The default value is 25 kJ/K.

Initial temperature

> Component temperature at the start of simulation. The initial temperature alters the component efficiency according to an efficiency vector that you specify, affecting the starting meshing or friction losses. The default value is 300 K.

Linearization and Real-Time Simulation

Linearization

To optimize your model for linearization, use the **Clutch** > **Torque transmission model** parameter default setting, `Friction clutch approximation - Suitable for HIL and linearization`.

Hardware-in-the-Loop Simulation

For optimal simulation performance, use the **Clutch** > **Torque transmission model** parameter default setting, `Friction clutch approximation - Suitable for HIL and linearization`.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using Simulink® Coder™.

See Also

Simscape Blocks
Fundamental Friction Clutch | Loaded-Contact Rotational Friction | Torsional Spring-Damper

Topics
"Gear Coupling Control Using Clutches"

"Model Thermal Losses in Driveline Components"
"Prepare Simscape Driveline Models for Real-Time Simulation Using Simscape Checks"

Double-Pinion Planetary Gear

Planetary gear train with two meshed planet gear sets
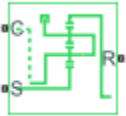
Library

Simscape / Driveline / Gears

Description

This block represents a planetary gear train with two meshed planet gear sets between the sun gear and the ring gear. A single carrier holds the two planet gear sets at different radii from the sun gear centerline, while allowing the individual gears to rotate with respect to each other. The gear model includes power losses due to friction between meshing gear teeth and viscous damping of the spinning gear shafts.
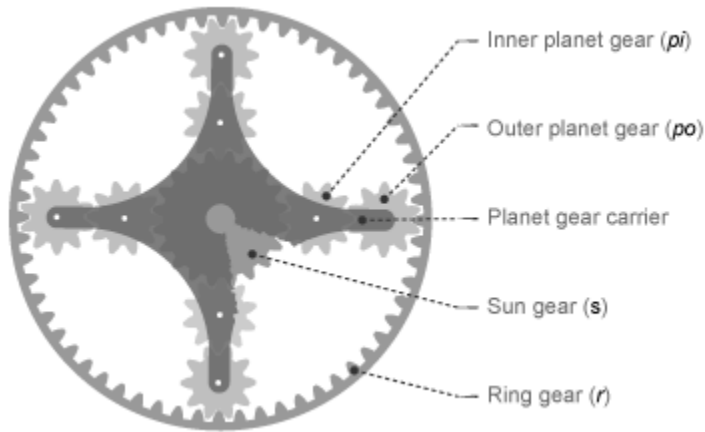
Structurally, the double-pinion planetary gear resembles a Ravigneaux gear without its second, large, sun gear. The inner planet gears mesh with the sun gear and the outer planet gears mesh with the ring gear. Because it contains two planet gear sets, the double-pinion planetary gear reverses the relative rotation directions of the ring and sun gears.

The teeth ratio of a meshed gear pair fixes the relative angular velocities of the two gears in that pair. The dialog box provides two parameters to set the ring-sun and outer planet-inner planet gear teeth ratios. A geometric constraint fixes the remaining teeth ratios—ring-outer planet and inner planet-sun. This geometric constraint requires that the ring gear radius equal the sum of the sun gear radius with the inner and outer planet gear diameters:

$$r_r = r_s + 2 \cdot r_{pi} + 2 \cdot r_{po},$$

where:

- r_r is the ring gear radius
- r_s is the sun gear radius
- r_{pi} is the inner planet gear radius
- r_{po} is the outer planet gear radius
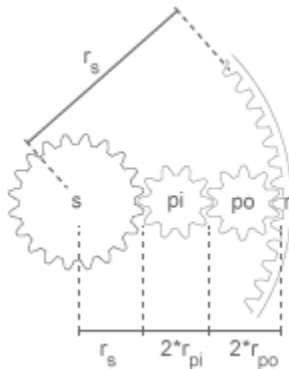
In terms of the ring-sun and outer planet-inner planet teeth ratios, the ring-outer planet teeth ratio is

$$\frac{r_r}{r_{po}} = 2 \cdot \frac{\frac{r_r}{r_s}}{\left(\frac{r_r}{r_s} - 1\right)} \cdot \frac{\left(\frac{r_{po}}{r_{pi}} + 1\right)}{\frac{r_{po}}{r_{pi}}},$$

The inner planet-sun teeth ratio is

$$\frac{r_{pi}}{r_s} = \frac{\left(\frac{r_r}{r_s} - 1\right)}{2\left(\frac{r_{po}}{r_{pi}} + 1\right)},$$

The Differential block is a composite component. It contains three underlying blocks—Ring-Planet, Planet-Planet, and Sun-Planet—connected as shown in the figure. Each block connects to a separate drive shaft through a rotational conserving port.
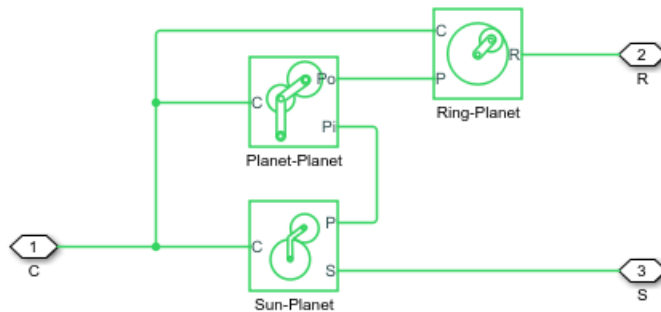
Thermal Modeling

You can model the effects of heat flow and temperature change through an optional thermal conserving port. By default, the thermal port is hidden. To expose the thermal port, right-click the block in your model and, from the context menu, select **Simscape > Block choices**. Select a variant that includes a thermal port. Specify the associated thermal parameters for the component.

Ports

Port	Description
C	Rotational conserving port representing the planet gear carrier
S	Rotational conserving port representing the sun gear
R	Rotational conserving port representing the ring gear
H	Thermal conserving port for thermal modeling

Parameters

Main

Ring (R) to sun (S) teeth ratio (NR/NS)

Teeth ratio between the ring and sun gears. This ratio is the number of teeth in the ring gear divided by the number of teeth in the sun gear. The default value is 2.

Outer planet (Po) to inner planet (Pi) teeth ratio (NPo/NPi)

Teeth ratio between the outer-planet and inner-planet gears. This ratio is the number of teeth in the outer planet divided by the number of gear teeth in the inner planet. The default value is 1.

Meshing Losses

Parameters for meshing losses vary with the block variant chosen—one with a thermal port for thermal modeling and one without it.

Without Thermal Port

Friction model

List of friction models at various precision levels for estimating power losses due to meshing.

- `No meshing losses - Suitable for HIL simulation` — Neglect friction between gear cogs. Meshing is ideal.

- `Constant efficiency` — Reduce torque transfer by a constant efficiency factor. This factor falls in the range $0 < \eta \leq 1$ and is independent of load. Selecting this option exposes additional parameters.

Sun-planet, ring-planet, and planet-planet ordinary efficiencies

Array of mechanical efficiencies between sun-planet, ring-planet, and planet-planet gear pairs, in that order. The default vector is `[0.98 0.98 0.98]`.

Sun-carrier, ring-carrier, and planet-carrier power thresholds

Array of power thresholds above which the full efficiency factors apply. Enter the thresholds in the order sun-carrier, ring-carrier, and planet-carrier. A hyperbolic tangent function smooths the efficiency factors from zero when at rest to the full efficiency values at the power thresholds.

As a guideline, the power threshold should be lower than the expected power transmitted during simulation. Higher values might cause the block to underestimate efficiency losses. Very low values might, however, raise the computational cost of simulation. The default vector is `[0.001 0.001 0.001]` W.

With Thermal Port

Temperature

Array of temperatures used to construct 1-D temperature-efficiency lookup tables. The array values must increase left to right. The temperature array must be the same size as the efficiency arrays. The default array is `[280 300 320]` K.

Sun-planet efficiency

Array of mechanical efficiencies with power flowing from the sun gear to the planet gears. Each array element is the ratio of output power to input power at one of the temperatures in the temperature array. The temperature and efficiency arrays must be the same size. The default array is `[0.75 0.65 0.60]`.

Ring-planet efficiency

Array of mechanical efficiencies with power flowing from the ring gear to the planet gears. Each array element is the ratio of output power to input power at one of the temperatures in the temperature array. The temperature and efficiency arrays must be the same size. The default array is `[0.50 0.45 0.40]`.

Planet-planet efficiency

Array of mechanical efficiencies with power flowing from the inner planet gears to the outer planet gears. Each array value is the ratio of output power to input power at

one of the temperatures in the temperature array. The temperature and efficiency arrays must be the same size. The default array is `[0.50 0.45 0.40]`.

Sun-carrier, ring-carrier and planet-carrier power thresholds

Array of power thresholds above which the full efficiency factors apply. Enter the thresholds in the order sun-carrier, ring-carrier, and planet-carrier. A hyperbolic tangent function smooths the efficiency factors between zero when at rest and the values provided by the temperature-efficiency lookup tables when at the power threshold. The default array is `[0.001 0.001 0.001]` W.

Viscous Losses

Specify the power losses due to viscous damping of the gear carriers.

Sun-carrier, ring-carrier, and planet-carrier viscous friction coefficients

Enter a vector with the viscous friction coefficients that dampen sun-carrier, ring-carrier, and planet-carrier motion, in that order. From the drop-down list, select a physical unit. The default vector is `[0 0 0]` N*m/(rad/s).

Thermal Port

Thermal mass

Thermal energy required to change the component temperature by a single degree. The greater the thermal mass, the more resistant the component is to temperature change. The default value is `50` J/K.

Initial temperature

Component temperature at the start of simulation. The initial temperature alters the component efficiency according to an efficiency vector that you specify, affecting the starting meshing or friction losses. The default value is `300` K.

Real-Time Simulation

Hardware-in-the-Loop Simulation

For optimal simulation performance, use the **Meshing Losses > Friction model** parameter default setting, `No meshing losses - Suitable for HIL simulation`.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also

Simscape Blocks
Compound Planetary Gear | Planetary Gear | Ravigneaux Gear

Topics
"Model Thermal Losses in Driveline Components"
"Prepare Simscape Driveline Models for Real-Time Simulation Using Simscape Checks"

Double-Shoe Brake

Frictional brake with two pivoted shoes diametrically positioned about a rotating drum
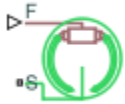
Library

Brakes & Detents/Rotational

Description

The block represents a frictional brake with two pivoted rigid shoes that press against a rotating drum to produce a braking action. The rigid shoes sit inside or outside the rotating drum in a diametrically opposed configuration. A positive actuating force causes the rigid shoes to press against the rotating drum. Viscous and contact friction between the drum and the rigid shoe surfaces cause the rotating drum to decelerate. Double-shoe brakes provide high braking torque with small actuator deflections in applications that include motor vehicles and some heavy machinery. The model employs a simple parameterization with readily accessible brake geometry and friction parameters.

(a)   (b)
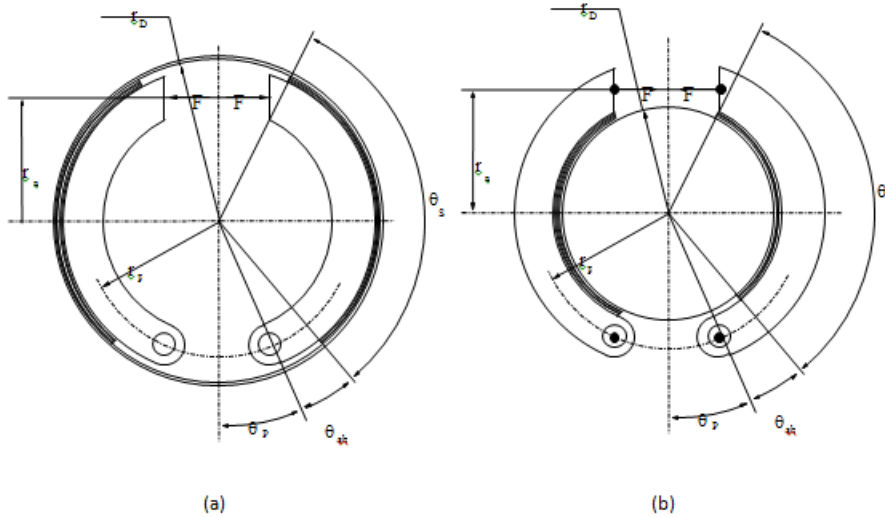
In this schematic, a) represents an internal double-shoe brake, and b) represents an external double-shoe brake. In both configurations, a positive actuation force F brings the shoe and drum friction surfaces into contact. The result is a friction torque that causes deceleration of the rotating drum. Zero and negative forces do not bring the shoe and drum friction surfaces into contact and produce zero braking torque.

The model uses the long-shoe approximation. Contact angles smaller than 45° produce less accurate results. The equations for the friction torque that the leading and trailing shoes develop are:

$$T_{LS} = \frac{c\mu p_a r_D 2 (\cos\theta_{sb} - \cos\theta_s)}{\sin\theta_a},$$

$$T_{TS} = \frac{c\mu p_b r_D 2 (\cos\theta_{sb} - \cos\theta_s)}{\sin\theta_a},$$

$$c = r_a + r_p \cos\theta_p,$$

where for $0 \le \theta_s \le \pi/2$,

$$\theta_a = \theta_s,$$

and for $\theta_s \geq \pi/2$,

$$\theta_a = \frac{\pi}{2}.$$

The parameters in the friction torque equations are:

Parameter	Description
T_{LS}	Brake torque the leading shoe develops
T_{TS}	Brake torque the trailing shoe develops
μ	Effective contact friction coefficient
p_a	Maximum linear pressure in the leading shoe-drum contact
p_b	Maximum linear pressure in the trailing shoe-drum contact
r_D	Drum radius
θ_{sb}	Shoe beginning angle
θ_s	Shoe span angle
θ_a	Angle from hinge pin to maximum pressure point
c	Arm length of the cylinder force with respect to the hinge pin
r_p	Pin location radius
θ_p	Hinge pin location angle
r_a	Actuator location radius

The model assumes that only Coulomb friction acts at the shoe-drum surface contact. Zero relative velocity between the drum and the shoes produces zero Coulomb friction. To avoid discontinuity at zero relative velocity, the friction coefficient formula employs this hyperbolic function

$$\mu = \mu_{Coulomb}\tanh\left(\frac{4\omega_{shaft}}{\omega_{threshold}}\right),$$

where:

Parameter	Description
μ	Effective contact friction coefficient
$\mu_{Coulomb}$	Contact friction coefficient
ω_{shaft}	Shaft velocity
$\omega_{threshold}$	Angular velocity threshold

Balancing the moments that act on each shoe with respect to the pin yields the pressure acting at the shoe-drum surface contact. The equations for determining the balance of moments for the leading shoe are

$$F = \frac{M_N - M_F}{c},$$

$$M_N = \frac{p_a r_p r_D}{\sin\theta_a}\left(\frac{1}{2}[\theta_s - \theta_{sb}] - \frac{1}{4}[\sin2\theta_s - \sin2\theta_{sb}]\right),$$

and

$$M_F = \frac{\mu p_a r_D}{\sin\theta_a}\left(r_D[\cos\theta_{sb} - \cos\theta_s] + \frac{r_p}{4}[\cos2\theta_s - \cos2\theta_{sb}]\right),$$

where:

Parameter	Description
F	Actuation Force
M_N	Moment acting on the leading shoe due to normal force
M_F	Moment acting on the leading shoe due to friction force
c	Arm length of the cylinder force with respect to the hinge pin
p_a	Maximum linear pressure at the shoe-drum contact surface
r_p	Pin location radius
θ_p	Hinge pin location angle

Parameter	Description
r_a	Actuator location radius

The model does not simulate self-locking brakes. If brake geometry and friction parameters cause a self-locking condition, the model produces a simulation error. A brake self-locks if the friction moment exceeds the moment due to normal forces, that is, when $M_F > M_N$.

The balance of moments for the trailing shoe is

$$F = \frac{M_N + M_F}{c}.$$

The net braking torque is

$$T = T_{LS} + T_{TS} + \mu_{visc} * \omega_{shaft},$$

where μ_{visc} is the viscous friction coefficient.

Variables

Use the **Variables** tab to set the priority and initial target values for the block variables before simulating. For more information, see "Set Priority and Initial Target for Block Variables" (Simscape).

Unlike block parameters, variables do not have conditional visibility. The **Variables** tab lists all the existing block variables. If a variable is not used in the set of equations corresponding to the selected block configuration, the values specified for this variable are ignored.

Assumptions and Limitations

- The brake uses the long-shoe approximation.
- The brake geometry does not self-lock.
- The model does not account for actuator flow consumption.

Thermal Modeling

You can model the effects of heat flow and temperature change through an optional thermal conserving port. By default, the thermal port is hidden. To expose the thermal port, right-click the block in your model and, from the context menu, select **Simscape** > **Block choices**. Select a variant that includes a thermal port. Specify the associated thermal parameters for the component.

Ports

F

    Physical signal port that represents the brake actuating force

S

    Rotational conserving port that represents the rotating drum shaft

H

    Thermal conserving port. The thermal port is optional and is hidden by default. To expose the port, select a variant that includes a thermal port.

Parameters

Geometry

Drum radius

    Radius of the drum contact surface. The parameter must be greater than zero. The default value is 150 mm.

Actuator location radius

    Distance between the drum center and the force line of action. The parameter must be greater than zero. The default value is 100 mm.

Pin location radius

    Distance between the hinge pin and drum centers. The parameter must be greater than zero. The default value is 125 mm.

1-149

Pin location angle

Angular coordinate of the hinge pin location from the brake symmetry axis. The parameter must be greater than or equal to zero. The default value is 15 deg.

Shoe beginning angle

Angle between the hinge pin and the beginning of the friction material linen of the shoe. The value of the parameter must be in the range $0 \leq \theta_{sb} \leq$ (π-pin location angle). The default value is 5 deg.

Shoe span angle

Angle between the beginning and the end of the friction material linen on the shoe. The value of the parameter must be in the range $0 < \theta_{sb} \leq$ (π -pin location angle - shoe beginning angle). The default value is 120 deg.

Friction

Viscous friction coefficient

Value of the viscous friction coefficient at the contact surface. The parameter must be greater than or equal to zero. The default value is .01 n*m/(rad/s).

Temperature

Array of temperatures used to construct a 1-D temperature-efficiency lookup table. The array values must increase left to right. The default value is [280.0, 300.0, 320.0] K.

This parameter is visible only when you select a block variant that includes a thermal port.

Contact friction coefficient

Value of the Coulomb friction coefficient at the belt-drum contact surface. The value is greater than zero. Unless you select a block variant that includes a thermal port, the default value is 0.3.

If you select a block variant that includes a thermal port, you specify this parameter as array. The array is the same size as the array for the **Temperature** parameter. The default value for the thermal variant is [0.1, 0.05, 0.03].

Angular velocity threshold

Angular velocity at which the contact friction coefficient practically reaches its steady-state value. The parameter must be greater than zero. The default value is 0.01 rad/s.

Thermal Port

Thermal mass

> Thermal energy required to change the component temperature by a single degree. The greater the thermal mass, the more resistant the component is to temperature change. This parameter is enabled only if you select a block variant that includes a thermal port. The default value is 50 kJ/K.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also
Band Brake | Loaded-Contact Rotational Friction | Rotational Detent

Topics
"Model Thermal Losses in Driveline Components"

Double-Sided Synchronizer

Back-to-back dog-cone clutch pairs assembled symmetrically about a translational detent to provide smooth gear engagement
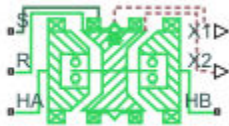
Library

Simscape / Driveline / Clutches

Description

The block represents a double-sided synchronizer that contains two back-to-back dog clutches, two back-to-back cone clutches, and one translational detent. Shift linkage translation along the negative direction causes the clutches to engage the ring with hub A. Shift linkage translation along the positive direction causes the clutches to engage the ring with hub B. When the magnitude of the shift linkage translation is smaller than the cone clutch ring-hub gap, the synchronizer is in neutral mode and does not transmit torque.
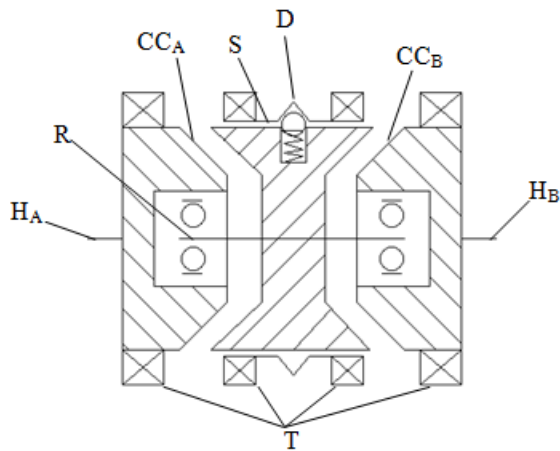
The schematic illustrates a double-sided synchronizer in the disengaged state. In this state, the ring (R) and hub (H_A and H_B) shafts can spin independently at different speeds. As the shift linkage (S) translates in the negative direction, the faces of cone clutch A (CC_A) come into contact. The friction in the cone clutch decreases the difference in rotational speed between the shafts. When the force on the shift linkage exceeds the peak force of detent (D), the dog clutch teeth (T) can engage. The detent peak force should be such that the cone clutch has enough time and normal force to bring the shafts to sufficiently similar speeds to allow engagement of the dog clutch. Similarly, translating the shift linkage along the positive direction allows the faces of cone clutch B (CC_B) to come into contact, and can allow the shaft of the ring to engage with the shaft of the hub B (H_B).
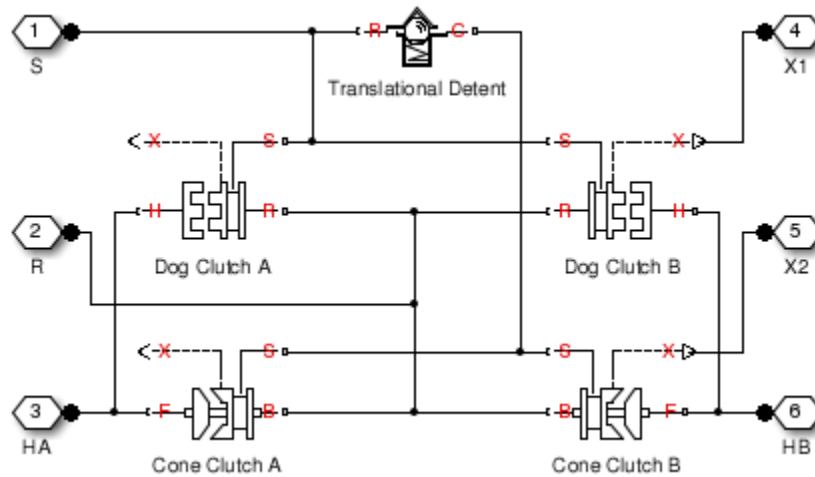
The model implements two Dog Clutch blocks, one Translational Detent, and two modified Dog Clutch blocks. Refer to each block reference page for more information on the corresponding block function.

Connections R, H_A, and H_B are mechanical rotational conserving ports that represent the ring (R), hub A (H_A), and hub B (H_B), respectively. Connection S is a mechanical translational conserving port that represents the ring shifter handle.

Connections X1 and X2 are physical signal ports that output the shift linkage positions of the dog clutches and cone clutches, respectively. The following tables provide the values of X1 and X2 in common clutch engagement cases.

Dog Clutch State	X1
Disengaged	0
Fully engaged with hub A	Negative sum of ring-hub gap and tooth height
Fully engaged with hub B	Positive sum of ring-hub gap and tooth height

Cone Clutch State	X2
Disengaged	0

Cone Clutch State	X2
Fully engaged with hub A	Negative value of ring-hub gap
Fully engaged with hub B	Positive value of ring-hub gap

The values of X1 and X2 are zero when the synchronizer is fully disengaged. When the dog clutch is fully engaged with hub A, X1 is equal to the negative sum of its ring-hub gap and tooth height. When the dog clutch is fully engaged with hub B, X1 is equal to the positive sum of its ring-hub gap and tooth height. When the cone clutch is fully engaged with hub A, X2 is equal to the negative of its ring-hub gap. When the cone clutch is fully engaged with hub B, X2 is equal to its ring-hub gap.

Thermal Modeling

You can model the effects of heat flow and temperature change through an optional thermal conserving port. By default, the thermal port is hidden. To expose the thermal port, right-click the block in your model and, from the context menu, select **Simscape** > **Block choices**. Select a variant that includes a thermal port. Specify the associated thermal parameters for the component.

Assumptions and Limitations

- The model does not account for inertia effects. You can add a Simscape Inertia block at each port to add inertia to the synchronizer model.

Ports

S

    Conserving translational port that represents the shift linkage

R

    Conserving rotational port that represents the ring shaft

HA

    Conserving rotational port that represents hub shaft A

HB

    Conserving rotational port that represents hub shaft B

X1

Physical signal output port that measures the magnitude of the dog clutch translation

X2

Physical signal output port that measures the magnitude of the cone clutch translation

T

Thermal conserving port. The thermal port is optional and is hidden by default. To expose the port, select a variant that includes a thermal port.

Parameters

Dog Clutch

The methods that are available for parameterizing the torque transmission depend on the variant of the block that you use. If you use the default variant, which has no thermal port, you parameterize the block using one of the options for the **Torque Transmission Model** parameter. If you select a variant that has a thermal port, you parameterize the block based on temperature. The models provide these common parameters:

Clutch teeth mean radius

Mean radius of the clutch teeth. The parameter must be greater than zero. The default value is 50 mm.

Maximum engagement speed

The relative angular speed between the ring and hub shafts above which the dog clutch cannot engage. The value is specific to the specific gearbox or transmission and must be minimized to avoid high dynamic impact during engagement. The default value is `inf rad/s`.

Additional Parameters for the Default Block Variant

Torque Transmission Model

Model used to represent torque transmission. The default setting is `Friction clutch approximation - Suitable for HIL and linearization`.

Friction clutch approximation - Suitable for HIL and linearization

Maximum transmitted torque

Maximum expected torque across the clutch. The default value is `1000 N*m`.

Dynamic with Backlash

Number of teeth

Number of teeth in the clutch. The parameter must be greater than or equal to one. The default value is `6`.

Rotational backlash

Angular play of the ring with respect to the hub. Greater backlash values correspond to wider tooth gaps that makes engagement easier. The parameter must be greater than zero. The default value is `10 deg`.

Torsional stiffness

Stiffness of the tooth contact. The model assumes deformation due to tooth interaction is elastic. The angular deformation at the contact surface is proportional to the applied torque and the reciprocal of the contact stiffness. The parameter must be greater than zero. The default value is `1e+7 N*m/rad`.

Torsional damping

Damping at the tooth contact surface. The value of the parameter is inversely proportional to the number of oscillations following impact. The parameter must be greater than zero. The default value is `100 N*m/(rad/s)`.

Tooth-tooth friction coefficient

Friction coefficient for the translational relative motion between the hub and ring teeth. The value of the parameter depends on the tooth material, lubrication, and quality of contacting surfaces. The parameter must be greater than zero. The default value is `0.05`.

Additional Parameters for the Thermal Block Variant

Temperature vector

Specify the input values for the temperature as a vector. The minimum number of values depends on the interpolation method that you select. For linear interpolation, provide at least two values per dimension. For smooth interpolation, provide at least three values per dimension. The values in the vector must increase from left to right. The default value is `[280.0, 300.0, 320.0]` K.

Maximum transmitted torque vector

Largest torque that the clutch can transmit, corresponding to a nonslip engaged configuration, specified as a vector. If the torque transmitted between the ring and the hub exceeds this value, the two components begin to slip with respect to each other. This torque determines the static friction limit in the friction clutch approximation. The vector has the same number of elements as the temperature vector. The default value is `[1000.0, 1050.0, 1000.0]` N*m.

Interpolation method

Select one of these interpolation methods to approximate the output value when the input value is between two consecutive grid points:

- `Linear` — Select this option, the default, to get the best performance. This method is the default.

- `Smooth` — Select this option to produce a continuous curve with continuous first-order derivatives.

For more information on interpolation algorithms, see the PS Lookup Table (1D) block reference page.

Extrapolation method

Select one of these extrapolation methods to determine the output value when the input value is outside the range specified in the argument list:

- `Linear` — Select this, default option, to produce a curve with continuous first-order derivatives in the extrapolation region and at the boundary with the interpolation region.

- `Nearest` — Select this option to produce an extrapolation that does not go above the highest point in the data or below the lowest point in the data.

For more information on extrapolation algorithms, see the PS Lookup Table (1D) block reference page.

Cone Clutch

Common parameters for the cone clutch are:

Contact surface maximum diameter

Outer diameter of the friction material linen that lines the cone clutch disks. The parameter must be greater than zero. The default value is `150` mm.

Contact surface minimum diameter

Inner diameter of the friction material linen that lines the cone clutch disks. The parameter must be greater than zero but smaller than the value of **Contact surface maximum diameter**.

Cone half angle

Half angle of the clutch cone. The parameter must be greater than zero and less than or equal to 90 degrees. The default value is `12 deg`.

Velocity tolerance

Relative velocity below which the two surfaces can lock. The surfaces lock if the torque across the B and F rotational ports is less than the product of the effective radius, the static friction coefficient, and the applied normal force. The default value is `0.001 rad/s`.

Threshold force

The normal force applied to the physical signal port **N** is applied to the contact only if the amount of force exceeds the value of the **Threshold force** parameter. Forces below the **Threshold force** are not applied, and there is consequently no transmitted frictional torque. The default value is `1 N`.

Friction model

Select a parameterization method to model the kinetic friction coefficient. The options and default values for this parameter depend on the variant that you select for the block. The options are:

- `Fixed kinetic friction coefficient` — Provide a fixed value for the kinetic friction coefficient. This option:

  - Is visible only if you use the default variant of the block
  - Is the default method for parameterizing the default variant of the block
  - Affects the visibility of other parameters

  `Fixed kinetic friction coefficient`

  Kinetic friction coefficient

    The kinetic, or Coulomb, friction coefficient. The coefficient must be greater than zero. The default value is `0.3`.

Static friction coefficient

> The static, or peak, value of the friction coefficient. The static friction coefficient must be greater than the kinetic friction coefficient. The default value is `0.35`.

- `Table lookup kinetic friction coefficient` — Define the kinetic friction coefficient by one-dimensional table lookup based on the relative angular velocity between disks. This option:

  - Is visible only if you use the default variant of the block
  - Affects the visibility of other parameters

`Table lookup kinetic friction coefficient`

Kinetic friction coefficient relative velocity vector

> Specify input values for the relative velocity as a vector. The values in the vector must increase from left to right. The minimum number of values depends on the interpolation method that you select. For linear interpolation, provide at least two values per dimension. For smooth interpolation, provide at least three values per dimension. The default value is `[0.0, 100.0, 1000.0] rad/s`.

Kinetic friction coefficient vector

> Specify the output values for the kinetic friction coefficient as a vector. All values must be positive. The default value is `[0.3, 0.22, 0.19]`.

Friction coefficient interpolation method

> Select one of these interpolation methods to approximate the output value when the input value is between two consecutive grid points:

- `Linear` — Select this, default, option to get the best performance.
- `Smooth` — Select this option to produce a continuous curve with continuous first-order derivatives.

> For more information on interpolation algorithms, see the PS Lookup Table (1D) block reference page.

Friction coefficient extrapolation method

> Select one of these extrapolation methods to determine the output value when the input value is outside the range specified in the argument list:

- `Linear` — Select this, default, option to produce a curve with continuous first-order derivatives in the extrapolation region and at the boundary with the interpolation region.

- `Nearest` — Select this option to produce an extrapolation that does not go above the highest point in the data or below the lowest point in the data.

For more information on extrapolation algorithms, see the PS Lookup Table (1D) block reference page.

Static friction coefficient

The static, or peak, value of the friction coefficient. The static friction coefficient must be greater than the kinetic friction coefficient. The default value is `0.35`.

- `Temperature-dependent kinetic friction coefficient` — Define the kinetic friction coefficient by table lookup based on the temperature. This option:

  - Is visible only if you use a thermal variant of the block

  - Is the default method for parameterizing the thermal variant of the block

  - Affects the visibility of other parameters

`Temperature-dependent kinetic friction coefficient`

Temperature vector

Specify the input values for the temperature as a vector. The minimum number of values depends on the interpolation method that you select. For linear interpolation, provide at least two values per dimension. For smooth interpolation, provide at least three values per dimension. The values in the vector must increase from left to right. The default value is `[280.0, 300.0, 320.0]` K.

Kinetic friction coefficient vector

Specify the output values for kinetic friction coefficient as a vector. The vector must have the same number of elements as the temperature vector. All values must be greater than zero. The default value is `[0.3, 0.28, 0.25]`.

Static friction coefficient vector

Specify the static, or peak, values of the friction coefficient as a vector. The vector must have the same number of elements as the temperature vector. Each value must be greater than the value of the corresponding element in the kinetic friction coefficient vector. The default value is `[0.4, 0.38, 0.36]`.

1-161

Friction coefficient interpolation method

Select one of these interpolation methods to approximate the output value when the input value is between two consecutive grid points:

- `Linear` — Select this, default, option to get the best performance.

- `Smooth` — Select this option to produce a continuous curve with continuous first-order derivatives.

For more information on interpolation algorithms, see the PS Lookup Table (1D) block reference page.

Friction coefficient extrapolation method

Select one of these extrapolation methods to determine the output value when the input value is outside the range specified in the argument list:

- `Linear` — Select this, default, option to produce a curve with continuous first-order derivatives in the extrapolation region and at the boundary with the interpolation region.

- `Nearest` — Select this option to produce an extrapolation that does not go above the highest point in the data or below the lowest point in the data.

For more information on extrapolation algorithms, see the PS Lookup Table (1D) block reference page.

- `Temperature and speed-dependent kinetic friction coefficient` — Define the kinetic friction coefficient by table lookup based on the temperature and the relative angular velocity between disks. This option:

  - Is visible only if you use the default variant of the block

  - Affects the visibility of other parameters

Temperature and speed-dependent kinetic friction coefficient

Temperature vector

Specify the input values for the temperature as a vector. The minimum number of values depends on the interpolation method that you select. For linear interpolation, provide at least two values per dimension. For smooth interpolation, provide at least three values per dimension. The values in the vector must increase from left to right. The default value is `[280.0, 300.0, 320.0]` K.

Kinetic friction coefficient relative velocity vector

Specify input values for the relative velocity as a vector. The values in the vector must increase from left to right. The minimum number of values depends on the interpolation method that you select. For linear interpolation, provide at least two values per dimension. For smooth interpolation, provide at least three values per dimension. The default value is `[0.0, 100.0, 1000.0] rad/s`.

Kinetic friction coefficient matrix

Specify the output values for kinetic friction coefficient as a matrix. All the values must be greater than zero. The size of the matrix must equal the size of the matrix that is the result of the temperature vector × the kinetic friction coefficient relative velocity vector. The default value is `[0.34, 0.32, 0.3; 0.3, 0.28, 0.25; 0.25, 0.2, 0.15]`.

Static friction coefficient vector

Specify the static, or peak, values of the friction coefficient as a vector. The vector must have the same number of elements as the temperature vector. Each value must be greater than the value of the corresponding element in the kinetic friction coefficient vector. The default value is `[0.4, 0.38, 0.36]`.

Friction coefficient interpolation method

Select one of these interpolation methods to approximate the output value when the input value is between two consecutive grid points:

- `Linear` — Select this, default, option to get the best performance.
- `Smooth` — Select this option to produce a continuous curve with continuous first-order derivatives.

For more information on interpolation algorithms, see the PS Lookup Table (1D) block reference page.

Friction coefficient extrapolation method

Select one of these extrapolation methods to determine the output value when the input value is outside the range specified in the argument list:

- `Linear` — Select this, default, option to produce a curve with continuous first-order derivatives in the extrapolation region and at the boundary with the interpolation region.
- `Nearest` — Select this option to produce an extrapolation that does not go above the highest point in the data or below the lowest point in the data.

1-163

For more information on extrapolation algorithms, see the PS Lookup Table (1D) block reference page.

Detent

Peak force

Peak shear force of the detent. The default value is `500 N`.

Notch width

Width of the region where the detent exhibits shear force. The default value is `3 mm`.

Viscous friction coefficient

Viscous friction coefficient at the contact surface of the detent. The parameter must be greater than or equal to zero. The default value is `0.1 N/(m/s)`.

Friction to peak force ratio

Ratio of the kinetic friction to the peak shear force of the detent. The parameter is used to set the value of the kinetic friction. The parameter must be greater than or equal to zero. The default value is `0.01`.

Friction velocity threshold

Velocity required for peak kinetic friction at the contact surface of the detent. The parameter ensures the force is continuous when the travel direction changes, increasing the numerical stability of the simulation. The parameter must be greater than zero. The default value is `0.05 m/s`.

Shift Linkage

Ring-hub clearance when dog clutch disengaged

Distance between ring and dog clutch hub with the ring fully retracted. The parameter specifies the maximum travel distance if engagement requirements are not met. The parameter must be greater than the value of **Ring-hub clearance when cone clutch disengaged**. The default value is `5 mm`.

Tooth overlap to engage

Minimum value of the tooth overlap beyond which the dog clutch is considered engaged. The parameter must be greater than zero. The default value is `3 mm`.

Tooth height

Height of dog clutch teeth. The parameter must be greater than zero. The default value is `10 mm`.

Ring-hub clearance when cone clutch disengaged

Distance between the ring and cone clutch hub when the ring is fully retracted. The cone clutch can begin to engage once the slider has translated this distance. The ring can travel only this distance if the engagement requirements are not satisfied. The parameter must be greater than zero. The default value is `3 mm`.

Dog clutch ring stop stiffness

Stiffness of the hard stops on both sides of the dog clutch ring. The model assumes the ring and stops behave elastically. Contact deformation is proportional to the applied force and the reciprocal of the contact stiffness. The value of the stiffness must be assigned with reference to the parameter **Tooth overlap to engage**. Too low a stiffness could cause the deformation to exceed the required overlap and initiate a false engagement. The parameter must be greater than zero. The default value is `1e +6 N/m`.

Cone clutch ring stop stiffness

Stiffness of the hard stops on both sides of the cone clutch ring. The model assumes the ring and stops behave elastically. Contact deformation is proportional to the applied force and the reciprocal of the contact stiffness. The parameter must be greater than zero. The default value is `1e+6 N/m`.

Dog clutch ring stop damping

Translational contact damping between the dog clutch ring and the hub. The value of the damping is inversely proportional to the number of oscillations that occur after impact. The parameter must be greater than zero. The default value is `1e+3 N/(m/s)`.

Cone clutch ring stop damping

Translational contact damping between the cone clutch ring and the hub. The value of damping is inversely proportional to the number of oscillations that occur after impact. The parameter must be greater than zero. The default value is `1e+3 N/(m/s)`.

Viscous friction coefficient

Viscous friction coefficient for the relative translational motion between the hub and the ring. The value of the parameter depends on lubrication state and quality of contacting surfaces. The coefficient must be greater than or equal to zero. The default value is `100 N/(m/s)`.

Initial Conditions

Initial state

Beginning configuration of cone and dog clutches. Choices include:

- `Cone clutch A and dog clutch A locked`
- `Cone clutch A locked`
- `All clutches unlocked`
- `Cone clutch B locked`
- `Cone clutch B and dog clutch B locked`

The default setting is `All clutches unlocked`.

Dog clutch initial shift linkage position

Initial position of the shift linkage section that attaches to the dog clutch. The value of the parameter has the following restrictions:

Dog Clutch State	Parameter Restriction
Dog clutch A Initially engaged	Negative of the parameter value must be greater than the sum of parameters **Ring-hub clearance when dog clutch disengaged** and **Tooth overlap to engage**
Dog clutch A Initially disengaged	Negative of the parameter value must be smaller than the sum of parameters **Ring-hub clearance when dog clutch disengaged** and **Tooth overlap to engage**
Dog clutch B Initially engaged	Parameter value must be greater than the sum of parameters **Ring-hub clearance when dog clutch disengaged** and **Tooth overlap to engage**

Dog Clutch State	Parameter Restriction
Dog clutch B Initially disengaged	Parameter value must be smaller than the sum of parameters **Ring-hub clearance when dog clutch disengaged** and **Tooth overlap to engage**

The default value is 0 mm.

Cone clutch initial shift linkage position

Initial position of the shift linkage section that attaches to the cone clutch. The value of the parameter has the following restrictions:

Dog Clutch State	Parameter Restriction
Cone clutch A initially engaged	Negative of parameter must be greater than the value of **Ring-hub clearance when cone clutch disengaged**
Cone clutch A initially disengaged	Negative of parameter must be smaller than the value of **Ring-hub clearance when cone clutch disengaged**
Cone clutch B initially engaged	Parameter must be greater than the value of **Ring-hub clearance when cone clutch disengaged**
Cone clutch B initially disengaged	Parameter must be smaller than the value of **Ring-hub clearance when cone clutch disengaged**

The default value is 0 mm.

Initial dog clutch ring-hub A offset angle

Initial angle between the ring and hub A portions of dog clutch. If the clutch is disengaged, the initial angle must lie in the range $-\pi/N \leq \theta \leq +\pi/N$, where N denotes the number of teeth in the dog clutch. If the clutch is engaged, the initial angle must be in the range $-\delta/2 \leq \theta \leq +\delta/2$, where δ is the backlash angle. The default value is 0 deg.

This parameter is only visible if you select a block variant that does not include a thermal port and set **Dog Clutch > Torque transmission model** to Dynamic with backlash.

Initial dog clutch ring-hub B offset angle

Initial angle between the ring and hub B portions of the dog clutch. If the clutch is disengaged, the initial angle must lie in the range -π/N ≤ θ ≤ +π/N, where *N* denotes the number of teeth in the dog clutch. If the clutch is engaged, the initial angle must be in the range -δ/2 ≤ θ ≤ +δ/2, where *δ* is the backlash angle. The default value is `0 deg`.

This parameter is only visible if you select a block variant that does not include a thermal port and set **Dog Clutch** > **Torque transmission model** to `Dynamic with backlash`.

Thermal Port

These thermal parameters are only visible when you select a block variant that includes a thermal port.

Thermal mass

Thermal energy required to change the component temperature by a single degree. The greater the thermal mass, the more resistant the component is to temperature change. The default value is `100 kJ/K`.

Initial temperature

Component temperature at the start of simulation. The initial temperature alters the component efficiency according to an efficiency vector that you specify, affecting the starting meshing or friction losses. The default value is `300 K`.

Linearization and Real-Time Simulation

Linearization

To optimize your model for linearization, use the **Dog Clutch** > **Torque transmission model** parameter default setting, `Friction clutch approximation - Suitable for HIL and linearization`.

Hardware-in-the-Loop Simulation

For optimal simulation performance, use the **Dog Clutch** > **Torque transmission model** parameter default setting, `Friction clutch approximation - Suitable for HIL and linearization`.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also

Simscape Blocks
Cone Clutch | Dog Clutch | Synchronizer | Translational Detent

Topics
"Model Thermal Losses in Driveline Components"
"Prepare Simscape Driveline Models for Real-Time Simulation Using Simscape Checks"

Flexible Shaft

Shaft with torsional and bending compliance
Library: Simscape / Driveline / Couplings & Drives

Description

The Flexible Shaft block represents a driveline shaft with torsional and bending compliance. The shaft consists of a flexible material that twists in response to an applied torque and bends in response to static mass unbalance. The twisting action delays power transmission between the shaft ends, altering the dynamic response of the driveline system.
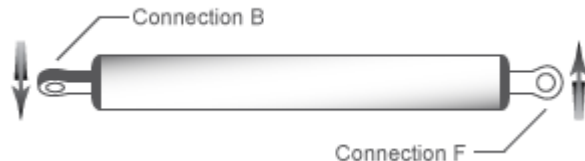
To represent the torsion-flexible shaft, the block uses a lumped parameter model. This model divides the shaft into different elements that interconnect through parallel spring damper systems. The elements provide the shaft inertia while the spring damper systems provide the shaft compliance.

You specify the shaft inertia, compliance, and number of shaft elements directly in the block dialog box. Choosing from two parameterizations, you can specify the shaft compliance by using stiffness and damping values or by using the shaft shear modulus. An additional parameter enables you to model the power losses due to viscous friction at the shaft ends. For more information, see "Torsional Model" on page 1-171.

To represent the bending-flexible shaft, the block uses your choice of a lumped parameter model or an eigenmodes model. While the lumped parameter approach is simpler to configure, the eigenmodes approach tends to simulate faster. For more information on both models, see "Bending Model" on page 1-173.

For the lumped parameter model, the number of bending shaft elements is the same as the number of torsion shaft elements, and additional parameters enable you to model linear damping proportional to shaft inertia and at the shaft ends. The model divides the shaft into different elements. The elements provide the shaft inertia, while the stiffness matrices provide the shaft compliance.

The eigenmodes model computes effective mass spring damper systems that represent the bending modes of the shaft. You can specify the number of modes to include, and additional parameters enable you to model linear damping proportional to modal mass and stiffness.

For both models, you specify the shaft bending compliance by using either bending rigidity and linear mass density or Young's modulus. Additional parameters enable you to model:

- Shaft mounting types. Options are ideal clamp, pin, free, or bearing stiffness.
- Locations and inertia of either rigid point masses or disks that are attached to the shaft.
- Locations, magnitudes, and angle offsets of static unbalances on the shaft.

Since the two modeling approaches are different, convergence between them lends confidence to their results. A recommended practice is to verify agreement for the lumped parameter and eigenmodes responses by using the parameter values from your own flexible shaft model in the "Shaft with Torsional and Transverse Flexibility" example. Then, for faster simulation of your own model, use the eigenmodes approach.

Torsional Model

For the torsional model, the Flexible Shaft block approximates the distributed, continuous properties of a shaft by a lumped parameter model. The model contains a finite number, N, of lumped inertia-damped spring elements in series, plus a final inertia. The result is a series of $N + 1$ inertias connected by N rotational springs and N rotational dampers.
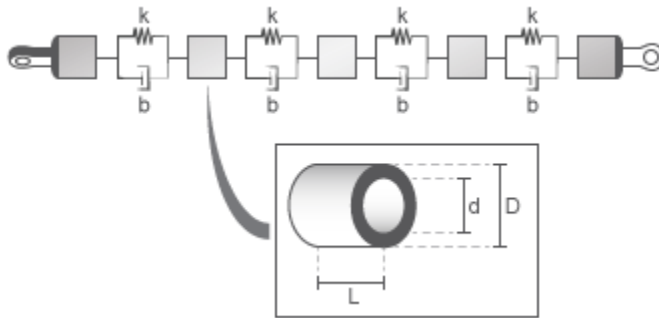
The equivalent physical network contains *N* segments, each consisting of a spring, damper, and inertia. A segment represents a short section of the driveshaft, the spring represents torsional compliance, and the damper represents material damping. The total shaft inertia is split into *N* + 1 parts, and distributed evenly along the length of the shaft.
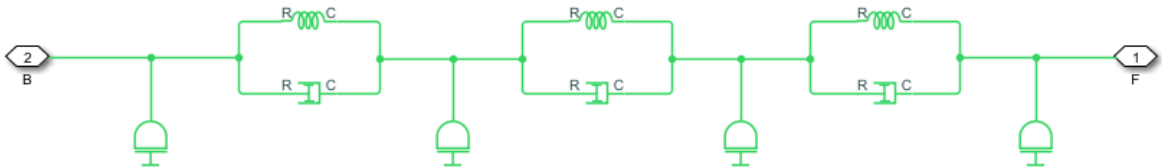
The block can also include viscous friction at the shaft ends (base and follower ports) to represent bearing losses at these points.

Note The viscous friction at the shaft ends is different from the internal material damping, which corresponds to losses arising in the shaft material itself.

You can parameterize the torsional model using either stiffness *k* and inertia *J* or the dimensions and material properties of the shaft.

Dimensions and Material Properties

The shaft stiffness and inertia are computed from the shaft dimensions and material properties by these relationships:

$$J_p = \frac{\pi}{32}\left(D^4 - d^4\right)$$

$$m = \frac{\pi}{4}\left(D^2 - d^2\right)\rho L$$

$$J = \frac{m}{8}\left(D^2 + d^2\right) = \rho L \cdot Jp$$

$$k = Jp \cdot \frac{G}{L}$$

Where:

- J_P is the polar moment of inertia of the shaft.
- D is the outer diameter of the shaft.
- d is the inner diameter of the shaft. For a solid shaft, $d = 0$. For an annular shaft, $d > 0$.
- L is the shaft length.
- m is the mass of the shaft.
- J is the moment of inertia of the shaft.
- ρ is the density of the shaft material.
- G is the shear modulus of elasticity of the shaft material.
- k is the shaft rotational stiffness.

Internal Material Damping

For either torsional parameterization, the internal material damping is defined by the damping ratio, c, for a single-segment model. In this case, the damping torque is $2\frac{ck}{\omega_N}$.

Undamped natural frequency $\omega_N = \sqrt{\frac{2k}{J}}$. The damping applied across an individual segment of a segmented model is equal to the product of the damping coefficient and the relative rotational velocity of that segment.

Bending Model

The next figure shows how to measure:

- The static unbalance offset angle, that is the angle of a static unbalance about the shaft axis relative to the x axis.
- The distances of a rigid mass and a static unbalance, relative to the base, **B**.

In the figure, the shaft has fixed supports at the base and follower. The shaft exerts forces, F, and moments, M, on the supports. The translational velocity, V, and rotational velocity, W, of the base and follower ends are relative to their respective supports. All curved arrows and sign conventions follow the right-hand rule. The signs of the physical signals that the block outputs correspond to the arrows that represent the forces, moments, and velocities of the shaft acting on the supports.

The vector signals are:

- Force, $Fr = [F_{xb}, F_{yb}, F_{xf}, F_{yf}]$
- Moment, $M = [M_{xb}, M_{yb}, M_{xf}, M_{yf}]$
- Translational velocity, $V = [V_{xb}, V_{yb}, V_{xf}, V_{yf}]$
- Rotational velocity, $W = [W_{xb}, W_{yb}, Wxf, W_{yf}]$
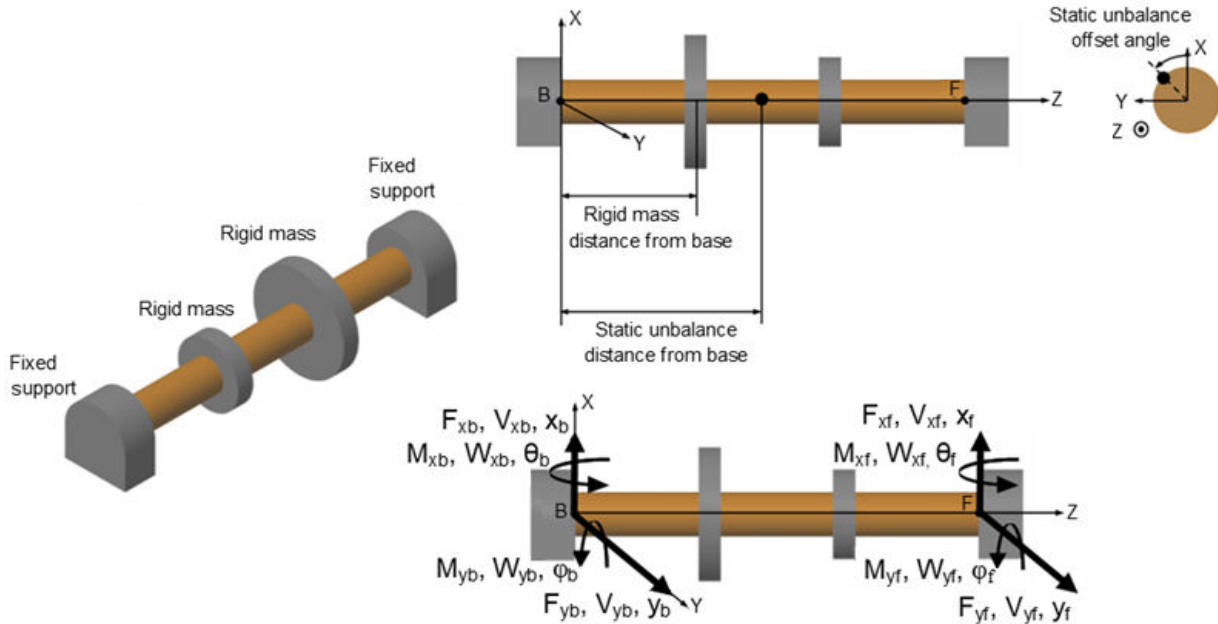
Fixed-Support Shaft Forces, Moments, and Phases

The partial differential equations governing the bending motion of a rotating shaft with points of static mass unbalance can be solved using a lumped parameter approach or an eigenmodes approach. For both models, there are two defining equations:

$$\frac{\partial^2}{\partial z^2}\left(EI\frac{\partial^2 u_x}{\partial z^2}\right) + \rho A\frac{\partial^2 u_x}{\partial t^2} = \sum_j^n m\varepsilon_j(\Omega(z))^2 cos\left(\varphi_{shaft} + \varphi_{offset,\,j}\right)$$

$$\frac{\partial^2}{\partial z^2}\left(EI\frac{\partial^2 u_y}{\partial z^2}\right) + \rho A\frac{\partial^2 u_y}{\partial t^2} = \sum_j^n m\varepsilon_j(\Omega(z))^2 sin\left(\varphi_{shaft} + \varphi_{offset,\,j}\right)$$

Where:

- EI is the shaft bending rigidity.
- ρA is the shaft linear mass density.

1-175

- $m\varepsilon_j$ is the jth static unbalance.

- $\varphi_{offset,i}$ is the initial angle offset of the jth static unbalance.

- n is the number of static unbalances.

- Ω is the shaft rotation speed.

- φ_{shaft} is the instantaneous shaft rotation angle.

- z is the coordinate along the shaft axis.

- $u_x(t,z)$ is the shaft bending deflections in the x direction perpendicular to the shaft axis.

- $u_y(t,z)$ is the shaft bending deflections in the y direction perpendicular to the shaft axis.

Lumped Parameter Approach

Like the torsion model, the lumped parameter approach for the bending model discretizes the distributed, continuous properties of the shaft into a finite number, N, of shaft segments. The N shaft segments correspond to $N + 1$ lumped inertias connected in series by damping and spring elements. However, for the bending model, each mass has four degrees of freedom: translation and rotation in both the x and y directions perpendicular to the shaft axis.
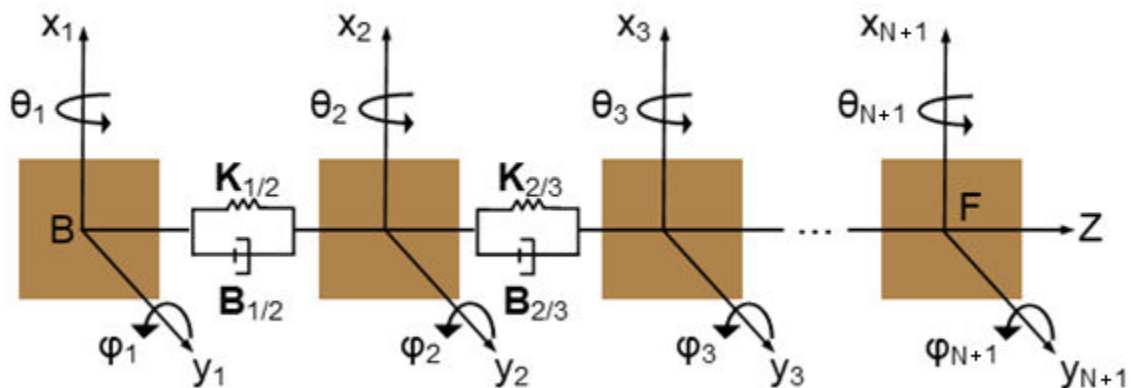
The lumped parameter equation [1] of motion is

$$M\ddot{\vec{x}} + B\dot{\vec{x}} + K\vec{x} = \vec{f}$$

Where:

- M is the $4(N + 1) \times 4(N + 1)$ matrix that represents the mass of the shaft.

- $\vec{x}$ is the $4(N + 1) \times 1$ vector that represents the degrees of freedom for all nodes, such that the degrees of freedom for the ith and the $(i + 1)^{th}$ nodes are

$$\vec{x} = \begin{bmatrix} \vdots \\ x_i \\ y_i \\ \theta_i \\ \varphi_i \\ x_{i+1} \\ y_{i+1} \\ \theta_{i+1} \\ \varphi_{i+1} \\ \vdots \end{bmatrix}$$

- K is the $4(N + 1) \times 4(N + 1)$ matrix for the spring stiffness.
- B is the $4(N + 1) \times 4(N + 1)$ matrix for the damping constant.

- $\vec{f}$ is the $4(N + 1) \times 1$ vector that represents external forces due to the application of static mass unbalance.

The equation for the mass matrix [4] is

$$M = M_{1/2} + M_{2/3} + ...M_{i/i+1} + ...M_{N/N+1} \;\; + \sum M_{disk,\ i}$$

Where:

- $M_{i/(i+1)}$ is the mass matrix for an individual shaft segment. For each shaft segment, half of the mass and moment of inertia is transferred to the nodes at both ends of the segment. The $M_{i/(i+1)}$ matrix has nonzero elements in the $(4i - 3):(4i + 4)$ rows and the $(4i - 3):(4i + 4)$ columns:

$$
M_{i/(i+1)} =
\begin{bmatrix}
0 & & & & & & & & & \\
& \ddots & & & & & & & & \\
& & \frac{1}{2}m & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
& & 0 & \frac{1}{2}m & 0 & 0 & 0 & 0 & 0 & 0 \\
& & 0 & 0 & I_d & 0 & 0 & 0 & 0 & 0 \\
& & 0 & 0 & 0 & I_d & 0 & 0 & 0 & 0 \\
& & 0 & 0 & 0 & 0 & \frac{1}{2}m & 0 & 0 & 0 \\
& & 0 & 0 & 0 & 0 & 0 & \frac{1}{2}m & 0 & 0 \\
& & 0 & 0 & 0 & 0 & 0 & 0 & I_d & 0 \\
& & 0 & 0 & 0 & 0 & 0 & 0 & 0 & I_d \\
& & & & & & & & & \ddots \\
& & & & & & & & & & 0
\end{bmatrix}
$$

Where:

- l is the segment length along the shaft axis. l depends on both the shaft length, L, and the number of segments, N, such that $l = \dfrac{L}{N+1}$.

- m is the segment mass. m depends on the outer, D, and inner, d, diameters, the density, ρ, of the shaft and the length of the segment, such that
  $m = \left(\frac{\pi}{4}\right)\left(D^2 - d^2\right)\rho l.$

- I_d, the half-segment mass moment of inertia about an axis perpendicular to the shaft axis, depends on the mass and length of the segment, such that
  $I_d = m\frac{l^2}{24} + \frac{m}{2}\left(\frac{l}{2}\right)^2.$

- $\sum M_{disk,\ i}$ is the summed mass matrices of the rigid masses concentrically attached to the shaft.

- The mass properties of each rigid mass concentrically attached to the shaft are added to the closest node, i, such that

$$M_{disk,\ i}([(4i-3):4i],[(4i-3):4i]) = \begin{bmatrix} M_{disk,\ i} & 0 & 0 & 0 \\ 0 & M_{disk,\ i} & 0 & 0 \\ 0 & 0 & I_{D,disk,\ i} & 0 \\ 0 & 0 & 0 & I_{D,disk,\ i} \end{bmatrix}$$

where $I_{D,disk,i}$ is the mass diametric moment of inertia about an axis perpendicular to the shaft for the ith disk.

The equation for the damping matrix is

$$B = \alpha M + \sum B_{disk,\ i} + \begin{bmatrix} b_B & & & & & & & & \\ & b_B & & & & & & & \\ & & \delta_B & & & & & & \\ & & & \delta_B & & & & & \\ & & & & \ddots & & & & \\ & & & & & b_F & & & \\ & & & & & & b_F & & \\ & & & & & & & \delta_F & \\ & & & & & & & & \delta_F \end{bmatrix}$$

Where:

- α is the damping constant proportional to mass.
- $[b_B, b_F]$ are the translation damping coefficients at the base, **B**, and follower, **F**.
- $[\delta_B, \delta_F]$ are the rotation damping coefficients at the base, **B**, and follower, **F**.
- $B_{disk,\ i}$ accounts for the gyroscopic effects of any concentrically attached disks, and is defined as

$$B_{disk,\ i}([(4i-3):4i],[(4i-3):4i]) = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \Omega I_{P,disk,\ i} \\ 0 & 0 & -\Omega I_{P,disk,\ i} & 0 \end{bmatrix}$$

where $I_{P,disk,i}$ is the mass polar moment of inertia about the shaft axis for the ith disk.

The equation for the bearing stiffness matrix is

$$K = K_B + K_{1/2} + K_{2/3} + ...K_{i/i+1} + ...K_{N/N+1} + K_F$$

Where:

- $K_{i/i+1}$ is the shaft segment stiffness matrix for an individual shaft segment. The stiffness matrix for the ith shaft segment, between the ith and the $(i+1)^{th}$ nodes, has nonzero elements in the $(4i-3):(4i+4)$ rows and the $(4i-3):(4i+4)$ columns, such that

$$K_{i/i+1} = \frac{2EI}{l^3}
\begin{bmatrix}
0 & & & & & & & & & \\
& \ddots & & & & & & & & \\
& & 6 & 0 & 0 & 3l & -6 & 0 & 0 & 3l \\
& & 0 & 6 & -3l & 0 & 0 & -6 & -3l & 0 \\
& & 0 & -3l & 2l^2 & 0 & 0 & 3l & l^2 & 0 \\
& & 3l & 0 & 0 & 2l^2 & -3l & 0 & 0 & l^2 \\
& & -6 & 0 & 0 & -3l & 6 & 0 & 0 & -3l \\
& & 0 & -6 & 3l & 0 & 0 & 6 & 3l & 0 \\
& & 0 & -3l & l^2 & 0 & 0 & 3l & 2l^2 & 0 \\
& & 3l & 0 & 0 & l^2 & -3l & 0 & 0 & 2l^2 \\
& & & & & & & & & \ddots \\
& & & & & & & & & & 0
\end{bmatrix}$$

Where:

- l is the segment length.
- EI is the shaft rigidity.

- K_B is the bearing stiffness matrices at the base, **B**, applied to the first node, such that

$$K_B =
\begin{bmatrix}
K_{Bxx} & 0 & 0 & 0 \\
0 & K_{Byy} & 0 & 0 \\
0 & 0 & K_{B\theta\theta} & 0 \\
0 & 0 & 0 & K_{B\varphi\varphi}
\end{bmatrix}$$

Where:

- $[K_{Bxx} \ K_{Byy}]$ is the base translational stiffness in the x and y directions.
- $[K_{B\theta\theta} \ K_{B\varphi\varphi}]$ is the base rotational stiffness about the x and y directions.

- K_F is the bearing stiffness matrices at the follower, **F**, applied to the first and N^{th} node, such that

$$K_F = \begin{bmatrix} K_{Fxx} & 0 & 0 & 0 \\ 0 & K_{Fyy} & 0 & 0 \\ 0 & 0 & K_{F\theta\theta} & 0 \\ 0 & 0 & 0 & K_{F\varphi\varphi} \end{bmatrix}$$

Where:

- $[K_{Fxx} \ K_{Fyy}]$ is the follower translational stiffness in the x and y directions.
- $[K_{F\theta\theta} \ K_{F\varphi\varphi}]$ is the follower rotational stiffness about the x and y directions

The base and follower bearing stiffness matrices, K_B and K_F, are nonzero only if you select `Bearing matrix` for the **Base (B) mounting type** parameter or **Follower (F) mounting type** parameter, respectively. If you select the `Clamped` mounting type, the kinematic conditions of zero rotation and translation are applied to the first or last four rows of $\vec{x}$ in the lumped parameter equation of motion. If you select the `Pinned` mounting type, the kinematic conditions of zero translation are applied to base of follower nodes in the lumped parameter equation of motion.

Node	Mounting Type	Boundary Condition for the Lumped Parameter Equation
Base, **B**	Clamped	$\vec{x}(1:4) = 0$
	Pinned	$\vec{x}(1:2) = 0$
	Bearing Matrix	$K_B(1:4, 1:4)$ are nontrivial.
Follower, **F**	Clamped	$\vec{x}(end-3:end) = 0$
	Pinned	$\vec{x}(end-3:end-2) = 0$

Node	Mounting Type	Boundary Condition for the Lumped Parameter Equation
	Bearing Matrix	$K_F(end-3:end, end-3$ are $:end)$ nontrivial

External forces due to each static mass unbalance are applied to the closest node. The forcing at the ith node is

$$\vec{f}_{4(i-1:i)} = \begin{bmatrix} (m\varepsilon_{j,\,offset})\Omega^2_{shaft,\,i}cos(\varphi_{shaft,\,i} + \varphi_{offset,\,j}) \\ (m\varepsilon_{j,\,offset})\Omega^2_{shaft,\,i}sin(\varphi_{shaft,\,i} + \varphi_{offset,\,j}) \\ -L_{m\varepsilon j}(m\varepsilon_{j,\,offset})\Omega^2_{shaft,\,i}sin(\varphi_{shaft,\,i} + \varphi_{offset,\,j}) \\ L_{m\varepsilon j}(m\varepsilon_{j,\,offset})\Omega^2_{shaft,\,i}cos(\varphi_{shaft,\,i} + \varphi_{offset,\,j}) \end{bmatrix}$$

Where:

- $L_{m\varepsilon j}$ is the distance of the jth static unbalance from the closest i^{th} node along the shaft axis.

- $m\varepsilon_j$ is the static unbalance.

- $\varphi_{shaft,\,i}$ is the torsion lumped parameter angle for the ith node.

Eigenmodes Approach

For the eigenmodes approach, the block computes the bending mode properties of the shaft during compilation time and then simulates the modal effective mass spring damper systems using these steps:

1   Determine the eigenmode shapes by treating the deflection of each flexible shaft segment between rigid masses as a spatial function with unknown coefficients:

   a   Generate equations for the flexible shaft segment boundary conditions.

   b   Numerically solve the system of equations for the modal frequencies.

   c   Determine the mode shape at each modal frequency using nonlinear least squares.

Note If you input the mode shapes directly, the block skips this step.

---

2 Determine these dynamic properties for each eigenmode:

- Effective modal mass
- Stiffness
- Forcing coefficients

3 Simulate the independent responses of the effective mass spring damper system for each eigenmode.

4 Obtain the total shaft response by summing the deflections of each eigenmode. Loads at the supports are proportional to the spatial derivatives of the mode shapes.

The equations for each step in the eigenmodes approach are shown. For information on the derivation of the equations, see Muszynska[5], Rao[6], and Wu[7].

The next figure shows the deflection angles and variables that the block uses to solve the eigenmodes approach boundary condition equations.
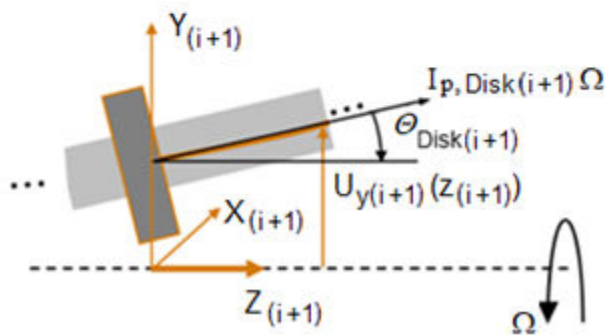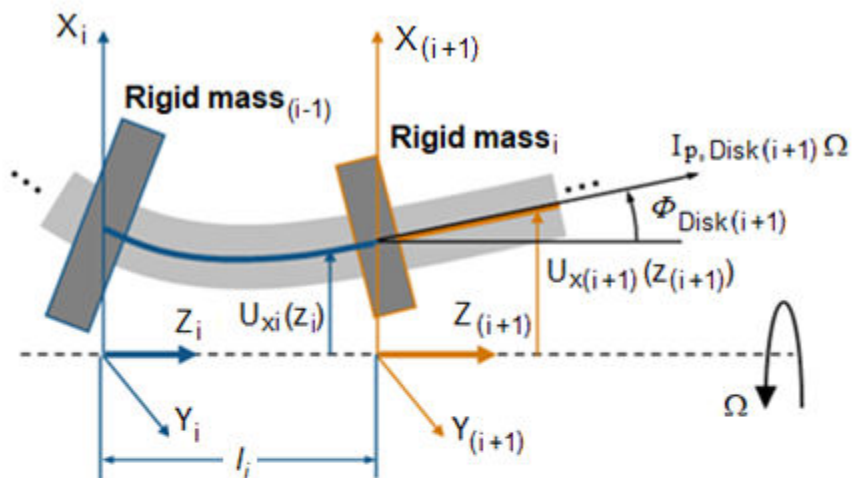
Neighboring Shaft Segments and Rigid Disk Motion

In step 1, to solve for the eigenmodes, the block considers the unforced bending equations for a beam in the two directions perpendicular to the shaft axis, using these equations:

$$\frac{\partial^2}{\partial z^2}\left(EI\frac{\partial^2 u_x}{\partial z^2}\right) + \rho A\frac{\partial^2 u_x}{\partial t^2} = 0$$

$$\frac{\partial^2}{\partial z^2}\left(EI\frac{\partial^2 u_y}{\partial z^2}\right) + \rho A\frac{\partial^2 u_y}{\partial t^2} = 0$$

The block assumes that solutions to the two equations have this form:

$$u_x = Re\{U_x(z)e^{j\omega t}\} = (A_x\sinh(\beta z) + B_x\cosh(\beta z) + C_x\sin(\beta z) + D_x cos(\beta z))Re\{e^{j\omega t}\}$$

$$u_y = Re\{U_y(z)e^{j\omega t}\} = (A_y\sinh(\beta z) + B_y\cosh(\beta z) + C_y\sin(\beta z) + D_y cos(\beta z))Re\{e^{j\omega t}\}$$

$$\beta = \left(\frac{\omega^2 \rho A}{EI}\right)^{1/4}$$

Where:

- U_x is the mode shape deflection in the x direction.
- U_y is the mode shape deflection in the y direction.
- $j = \sqrt{-1}$.
- ω is a modal frequency.
- $A_x, B_x, C_x, D_x, A_y, B_y, C_y,$ and D_y are coefficients.
- EI is the shaft bending rigidity.
- ρA is the shaft linear mass density.

In the eigenmodes approach equations, ′ indicates a derivative with respect to z, the axis along the shaft, such that:

- ′ indicates a first derivative with respect to z.
- ″ indicates a second derivative with respect to z.
- ‴ indicates a third derivative with respect to z.

In step 1a, to generate equations for the flexible shaft segment boundary conditions, the block divides the shaft into N segments between the shaft ends and $N - 1$ rigid masses attached to the shaft. For each segment, the block defines the kth mode shape in a piecewise manner along the entire shaft as $U_{kx}(z)$ and $U_{ky}(z)$.

For each individual shaft segment, i, the kth mode shapes, $U_{kxi}(z_i)$ and $U_{kyi}(z_i)$, are defined by the solutions to the unforced bending equations. For each shaft segment, i, z_i refers to an origin at the segment end closest to the shaft base, **B**, end.

Each shaft segment mode shape has eight unknown coefficients, A_x, B_x, C_x, D_x, A_y, B_y, C_y, and D_y. Mode shapes U_x and U_y satisfy boundary conditions at the disk interfaces and at both shaft ends.

The boundary conditions depend on the mounting types and are applied to both modes shapes. The boundary conditions at the base end of the shaft apply to the first segment. The boundary conditions at the follower end of the shaft apply to segment N.

Equations at Shaft B and F Ends

Node	Mounting Type	Boundary Condition Equation
Base, **B**	`Clamped` — Zero deflection and slope	• $U_x(z = 0) = U'_x(z = 0) = 0$ • $U_y(z = 0) = U'_y(z = 0) = 0$
	`Pinned` — Zero deflection and moment	• $U_x(z = 0) = U''_x(z = 0) = 0$ • $U_y(z = 0) = U''_y(z = 0) = 0$
	`Free` — Zero moment and force	• $U''_x(z = 0) = U'''_x(z = 0) = 0$ • $U''_y(z = 0) = U'''_y(z = 0) = 0$

Node	Mounting Type	Boundary Condition Equation
	`Bearing Matrix` — With bearing stiffness matrix K_B	$$K_B \vec{X}_b = K_B \begin{bmatrix} X_b \\ Y_b \\ \Theta_b \\ \Phi_b \end{bmatrix}$$ $$= K_B \begin{bmatrix} U_x \\ U_y \\ -U'_y \\ U'_x \end{bmatrix}\Bigg|_{z=0} =$$ $$EI \begin{bmatrix} -U'''_x \\ -U'''_y \\ U''_x \\ -U''_y \end{bmatrix}\Bigg|_{z=0}$$ Where: • $K_B \vec{X}_b$ is the spring load at the base end as a function of the mode shape. • X_b, Y_b, Θ_b, and Φ_b correspond to the deflections of variables in the "Fixed-Support Shaft Forces, Moments, and Phases" on page 1-175 figure.

Node	Mounting Type	Boundary Condition Equation
		• $K_B =$ $\begin{bmatrix} K_{Bxx} & 0 & 0 & 0 \\ 0 & K_{Byy} & 0 & 0 \\ 0 & 0 & K_{B\theta\theta} & 0 \\ 0 & 0 & 0 & K_{B\varphi\varphi} \end{bmatrix}$
Follower, **F**	`Clamped` — Zero deflection and slope	• $U_{xN}(z_N = l_N) = U'_{xN}(z_N = l_N) = 0$ • $U_{yN}(z_N = l_N) = U'_{yN}(z_N = l_N) = 0$
	`Pinned` — Zero deflection and moment	• $U_{xN}(z_N = l_N) = U''_{xN}(z_N = l_N) = 0$ • $U_{yN}(z_N = l_N) = U''_{yN}(z_N = l_N) = 0$
	`Free` — Zero moment and force	• $U''_{xN}(z_N = l_N) = U'''_{xN}(z_N = l_N) = 0$ • $U''_{yN}(z_N = l_N) = U'''_{yN}(z_N = l_N) = 0$

Node	Mounting Type	Boundary Condition Equation
	`Bearing Matrix` — With bearing stiffness matrix K_F	$K_F \vec{X}_f = K_F \begin{bmatrix} X_f \\ Y_f \\ \Theta_f \\ \Phi_f \end{bmatrix}$ $= K_f \begin{bmatrix} U_x \\ U_y \\ -U'_y \\ U'_x \end{bmatrix} \Bigg|_{z\,=\,L} =$ $EI \begin{bmatrix} -U'''_x \\ -U'''_y \\ U''_x \\ -U''_y \end{bmatrix} \Bigg|_{z\,=\,L}$ Where: • $K_F \vec{X}_f$ is the spring load at the follower end as a function of the mode shape • X_f, Y_f, Θ_f, and Φ_f correspond to the deflections of variables in the "Fixed-Support Shaft Forces, Moments, and Phases" on page 1-175 figure.

Node	Mounting Type	Boundary Condition Equation
		• $K_F =$ $\begin{bmatrix} K_{Fxx} & 0 & 0 & 0 \\ 0 & K_{Fyy} & 0 & 0 \\ 0 & 0 & K_{F\theta\theta} & 0 \\ 0 & 0 & 0 & K_{F\varphi\varphi} \end{bmatrix}$

At each rigid mass, the shaft segment on either side of the rigid mass must satisfy continuity in displacement and slope, and balance of forces and moments. To satisfy the boundary conditions, the block uses the equations in the table.

Equations at Rigid Masses Concentrically Attached to the Shaft

Condition	Rigid masses concentrically attached to shaft	
	Point mass	**Disk**
Displacement continuity	• $\quad U_{xi}(l_i) = U_{x(i+1)}(0)$ • $\quad U_{yi}(l_i) = U_{y(i+1)}(0)$	
Slope continuity	• $\quad U'_{xi}(l_i) = -U'_{x(i+1)}(0)$ • $\quad U'_{yi}(l_i) = -U'_{y(i+1)}(0)$	
Force balance	• $\quad -\omega^2 M_{disk,i} U_{xi}(l_i) = EIU'''_{xi}(l_i) - EIU'''_{x(i+1)}(0)$ • $\quad -\omega^2 M_{disk,i} U_{yi}(l_i) = EIU'''_{yi}(l_i) - EIU'''_{y(i+1)}(0)$	
Moment balance	• $0 = -EIU''_{xi}(l_i)$ $+ EIU''_{x(i+1)}(0)$ • $0 = -EIU''_{yi}(l_i)$ $+ EIU''_{y(i+1)}(0)$	• $-\omega^2 I_{D,disk,i}\Theta$ $- j\omega\Omega_{shaft}I_{P,disk,i}\Phi_{disk(i+1)} =$ $-EIU''_{yi}(l_i) + EIU''_{y(i+1)}(0)$ • $-\omega^2 I_{D,disk,i}\Phi$ $+ j\omega\Omega_{shaft}I_{P,disk,i}\Theta_{disk(i+1)} =$ $-EIU''_{xi}(l_i) + EIU''_{x(i+1)}(0)$

The variables for the equations in this table are:

- Ω_{shaft} is the nominal shaft rotation speed at the shaft midpoint.
- $I_{P, disk, i}$ is the rigid mass polar moment of inertia about shaft axis.
- ω is the modal frequency.
- $I_{d, disk, i}$ is the rigid mass diametric moment of inertia about shaft axis.
- *EI* is the shaft bending rigidity.
- $j = \sqrt{-1}$

In step 1b, the block numerically solves for modal frequencies by combining shaft segment boundary conditions at shaft B and F ends with the shaft segment boundary conditions at the rigid masses concentrically attached to the shaft. The resulting equation, in matrix form, is

$$H(\omega)\vec{S} = 0$$

Where:

- $\vec{S}$ contains the modal coefficients for the *N* shaft segments, such that

$$\vec{S} = \begin{bmatrix} A_{x1} \\ B_{x1} \\ C_{x1} \\ D_{x1} \\ A_{y1} \\ B_{y1} \\ C_{y1} \\ D_{y1} \\ \vdots \\ A_{xN} \\ B_{xN} \\ C_{xN} \\ D_{xN} \\ A_{yN} \\ B_{yN} \\ C_{yN} \\ D_{yN} \end{bmatrix}$$

For nontrivial modal solutions, the system of equations must satisfy $|H(\omega)| = 0$.

To determine the modal frequency solutions numerically, the block:

1   Computes the initial eigenfrequency solutions, $\omega_{kINITIAL}$, by taking the local minima of $abs(|H(\omega)|)$ over $\omega = [0:d\omega:\omega_{Max}]$, where:

   - $d\omega$ is the eigenfrequency search resolution.
   - $\omega_{\max}$ is the maximum eigenfrequency considered.

2   The block refines each $\omega_{kINITIAL}$ by replacing it with the minimum of $abs(|H(\omega)|)$ over $\omega = \omega_{kINITIAL} + d\omega([-1:d\omega_{RefineFactor}:1])$

The block then uses nonlinear least squares to determine the mode shape coefficients $\vec{S}$ at each modal frequency.

In step 2, the block determines these eigenmode dynamics properties:

- Modal mass
- Modal stiffness
- Modal damping
- Modal forcing coefficients

The block reduces the dynamics of each mode, k, to an independent system of equations:

$$M_{kx}\ddot{\eta}_{kx} + B_{kdx}\dot{\eta}_{kx} + B_{kpy}\dot{\eta}_{ky} + K_{kx}\eta_{kx} = f_{kx}$$

$$M_{ky}\ddot{\eta}_{ky} + B_{kdy}\dot{\eta}_{ky} + B_{kpx}\dot{\eta}_{kx} + K_{ky}\eta_{ky} = f_{ky}$$

Where:

- η_{kx} is the time-dependent coefficient of the kth mode in the x direction.

- η_{ky} is the time-dependent coefficient of the kth mode in the y direction.

The equations for the kth modal mass are:

$$M_{kx} = \int_0^L \rho A U_{kx}^2 \; dz \; + \sum M_{disk,i} U_{kx}^2|_{zdisk,i} \; + \sum I_{D,disk,i} U'_{kx}^2|_{zdisk,i}$$

$$M_{ky} = \int_0^L \rho A U_{ky}^2 \; dz \; + \sum M_{disk,i} U_{ky}^2|_{zdisk,i} \; + \sum I_{D,disk,i} U'_{ky}^2|_{zdisk,i}$$

The equations for the kth modal stiffness are:

$$K_{kx} = \int_0^L EIU''_{kx}{}^2 \; dz \quad + \sum U_{kx}|_{z\,=\,0} K_B([1,4],:)\vec{X}_{kb}$$

$$+ \sum U_{kx}|_{z\,=\,l} K_F([1,4],:)\vec{X}_{kf}$$

$$K_{ky} = \int_0^L EIU''_{ky}{}^2 \; dz \quad + \sum U_{ky}|_{z\,=\,0} K_B([2,3],:)\vec{X}_{kb}$$

$$+ \sum U_{ky}|_{z\,=\,l} K_F([2,3],:)\vec{X}_{kf}$$

Where:

- L is the shaft length.
- K_B and K_F are only used when the shaft boundary is defined by a bearing matrix.
- Only rows that contribute to loading in each modal direction play a role in that direction's modal stiffness expression.

The equations for modal damping due to the disk polar moment of inertias are:

$$D_{kpx} = -\Omega_{shaft} \sum I_{P,disk}(\Phi_k\Theta_k)|_{zdisk}$$

$$D_{kpy} = \Omega_{shaft} \sum I_{P,disk}(\Phi_k\Theta_k)|_{zdisk}$$

The equations for the dissipative modal damping are:

$$D_{kdx} = \alpha M_{kx} \quad + \quad \beta K_{kx}$$

$$D_{kdy} = \alpha M_{ky} \quad + \quad \beta K_{ky}$$

1-195

Where:

- α is the damping constant proportional to mass.
- β is the damping constant proportional to stiffness.

The equations for modal forcing due to the mass static unbalances are:

$$f_{kx} = \Omega_{shaft}^2 \sum_j \left(m\varepsilon_{offset,\,j} \right) U_{kx}(z_{offset,\,j}) \cos\left(\varphi_{shaft} + \varphi_{offset,\,j} \right)$$

$$f_{ky} = \Omega_{shaft}^2 \sum_j \left(m\varepsilon_{offset,\,j} \right) U_{ky}(z_{offset,\,j}) \sin\left(\varphi_{shaft} + \varphi_{offset,\,j} \right)$$

Where:

- The forces of each static unbalance, j, are summed for the total force on the kth mode.
- Ω_{shaft} is the instantaneous rotation speed at the shaft midpoint.
- φ_{shaft} is the instantaneous angle at the shaft midpoint.
- $\varphi_{offset,\,j}$ is the initial angle of the excitational j^{th} static unbalance.

In step 4, the block sums the deflections of each eigenmode to get the total shaft response. The equations for the total instantaneous deflection along the shaft are:

$$u_x(t, z) = \sum_j U_{kx}(z)\eta_{kx}(t)$$

$$u_y(t, z) = \sum_j U_{ky}(z)\eta_{ky}(t)$$

The equations for the instantaneous force acting on the supports at the shaft **B** and **F** ends in the x and y directions are:

$$F_{xb} = -EI \sum_j U_{kx}'''\big|_{z\,=\,0}\eta_{kx}(t)$$

$$F_{yb} = -EI \sum_j U_{ky}'''\big|_{z\,=\,0}\eta_{ky}(t)$$

$$F_{xf} = EI \sum U'''_{kx}|_{z = l} \eta_{kx}(t)$$

$$F_{yf} = EI \sum U'''_{ky}|_{z = l} \eta_{ky}(t)$$

The equations for the instantaneous moment acting on the supports at the shaft ends are:

$$M_{xb} = EI \sum U''_{kx}|_{z = 0} \eta_{kx}(t)$$

$$M_{yb} = -EI \sum U''_{ky}|_{z = 0} \eta_{ky}(t)$$

$$M_{xf} = -EI \sum U''_{kx}|_{z = l} \eta_{kx}(t)$$

$$M_{yf} = EI \sum U''_{ky}|_{z = l} \eta_{ky}(t)$$

Improve Simulation Speed or Accuracy

To increase simulation accuracy for the lumped parameter approach for a torsional or bending model, increase the number of segments N. However, as the number of segments increases, simulation performance, that is simulation speed, may decrease. The single-segmented torsional model exhibits an eigenfrequency that is close to the first eigenfrequency of the continuous, distributed parameter model.

For greater accuracy, you can select 2, 4, 8, or more segments. For example, the four lowest torsional eigenfrequencies are represented with an accuracy of 0.1, 1.9, 1.6, and 5.3 percent, respectively, by a 16-segmented model.

To increase simulation accuracy for the eigenmodes approach to a bending model:

- Verify that the **Nominal shaft speed for bending modes** parameter is close to the simulation shaft speed. This parameter may affect model results if you parameterize a rigid mass attached to the shaft with a large mass moment of inertia about shaft axis.

- Decrease the values of the **Eigenfrequency search resolution** and **Eigenfrequency refine factor** parameters. Reducing these values may increase the accuracy of modal frequencies and shapes.
- Decrease the damping constants. Simscape computations of the modal properties before simulation do not account for damping.
- Swap the values for the base, B and follower, F, boundary condition parameters and flip the block in the Simscape model. Eigenmodes shapes depend on sinh and cosh functions. Therefore, errors may occur near the F end.
- Check the sensitivity to the **Advanced Bending** parameters by using your parameters in the flexible shaft model in the "Shaft with Torsional and Transverse Flexibility" example. Adjust the parameters and use the links provided in the example to examine how the values affect the eigenmode frequencies and shapes. Adjust the parameter values in your model accordingly.
- Increase the values of the **Eigenfrequency upper limit** and **Limit number of modes** parameters. The highest modal frequency in the simulation must be sufficiently larger than the shaft rotation frequency.

Limitations and Assumptions

- The distributed parameter model of a continuous torsional shaft is approximated by a finite number, N, of lumped parameters.
- The flexible shaft has a constant axisymmetric cross-section along its length.
- Shaft rotation and torsion flexibility excites shaft bending, but bending does not affect shaft rotation and torsion flexibility.
- Rigid point masses or disks attached to the shaft have thin lengths parallel to the shaft axis.
- For the eigenmodes bending model, damping does not affect the eigenfrequencies.
- Bearing supports are located at the shaft endpoints.
- Shaft bending is not transmitted between Flexible Shaft blocks.
- Relative to the shaft length, the shaft outer diameter is small. The ratio of the shaft length to outer diameter is greater than 20.
- Relative to the shaft length, the bending deflection is small.
- Static mass unbalances are the only shaft-bending external exciting loads.
- Shaft supports are stationary.

- Gyroscopic effects of the rigid disks are considered; gyroscopic effects of the shaft itself are neglected.
- Static mass unbalance forcing in the eigenmodes approach uses the rotation speed at the shaft midpoint.
- To satisfy the shaft mode shape boundary conditions problem, the block uses the Levenberg-Marquardt algorithm.

Ports

Output

Fr — Force on bearing supports
physical signal

Physical signal outport associated with the force that the shaft exerts on the bearing supports.

M — Moment on bearing supports
physical signal

Physical signal outport associated with the moment that the shaft exerts on the bearing supports.

V — Shaft translational velocity
physical signal

Physical signal outport associated with the translational velocity of the shaft at the bearing supports.

W — Shaft angular velocity
physical signal

Physical signal outport associated with the angular velocity of the shaft at the bearing supports.

Conserving

B — Base
rotational mechanical

Rotational conserving port associated with the shaft base.

F — Follower
rotational mechanical

Rotational conserving port associated with the shaft follower.

Parameters

Shaft

The table shows how the visibility of some parameters depends on the option that you choose for other parameters. To learn how to read the table, see "Parameter Dependencies" on page B-2.

Shaft Parameter Dependencies

Shaft			
Model bending — Choose `Off` or `On`			
Off		**On**	
		Exposes block outports: • **Fr** • **M** • **V** • **W**	
		Exposes parameters for: • **Bending** • **Advanced Bending**	
Parameterization — Choose By `stiffness and inertia` or By `material properties`			
By stiffness and inertia	**By material properties**	**By stiffness and inertia**	**By material properties**
Number of segments			
Torsional stiffness	Shaft length	Torsional stiffness	Shaft length
Torsional inertia	Shaft geometry — Choose `Solid` or `Annular`	Torsional inertia	Shaft geometry — Choose `Solid` or `Annular`
	Solid / **Annular**	Shaft length	**Solid** / **Annular**
	Shaft outer diameter	Bending rigidity	Shaft outer diameter
	- / Shaft inner diameter	Linear density	/ Shaft inner diameter
	Material density		Material density
	Shear modulus		Shear modulus

Shaft			
			Young's modulus

Model bending — Bending model option
Off (default) | On

Option to model bending.

Dependencies

Setting the **Model bending** parameter to On exposes these ports and parameters, which are hidden by default:

- **Fr** outport
- **M** outport
- **V** outport
- **W** outport
- **Advanced Bending** parameters
- **Bending** parameters

For more information, see Shaft Parameter Dependencies.

Parameterization — Parameterization method
By stiffness and inertia (default) | By material properties

Parameterization method.

Dependencies

Each **Parameterization** option exposes related parameters and hides unrelated parameters. For more information, see Shaft Parameter Dependencies.

Number of segments — Number of shaft segments
1 (default) | positive integer

Number of segments, N, for the approximation.

A larger number of segments, N, increases the accuracy of the model, but reduces simulation performance, that is, simulation speed. The single-segmented model ($N=1$) exhibits a torsion eigenfrequency that is close to the first eigenfrequency of the continuous, distributed parameter model.

If accuracy is more important than performance, select 2, 4, 8, or more segments. For example, the four lowest torsion eigenfrequencies are represented with an accuracy of 0.1, 1.9, 1.6, and 5.3 percent, respectively, by a 16-segmented model. Accurately modeling bending dynamics generally requires more segments than torsion dynamics.

Torsional stiffness — Material stiffness
2e5 N*m/rad (default) | positive scalar

Material stiffness.

Dependencies

Setting the **Model bending** parameter to Off and the **Parameterization** parameter to By stiffness and inertia or setting the **Model bending** parameter to On exposes this parameter. For more information, see Shaft Parameter Dependencies.

Torsional Inertia — Shaft inertia
0.02 kg*m^2 (default) | positive scalar

Shaft inertia.

Dependencies

Setting the **Parameterization** parameter to By stiffness and inertia exposes this parameter. For more information, see Shaft Parameter Dependencies.

Shaft length — Shaft length
1 m (default) | positive scalar

Shaft length.

Dependencies

This parameter is visible when you set the **Model bending** parameter to Off and the **Parameterization** parameter to By stiffness and inertia or when you set the **Model bending** parameter to On. For more information, see Shaft Parameter Dependencies.

Bending rigidity — Bending rigidity
5e5 m^4*Pa (default) | positive scalar

Bending rigidity for the shaft material.

1-203

Dependencies

Setting the **Model bending** parameter to `On` and the **Parameterization** parameter to `By stiffness and inertia` exposes this parameter. For more information, see Shaft Parameter Dependencies.

`Linear density — Linear density`
19 **kg/m** (default) | positive scalar

Linear density for the shaft material.

Dependencies

Setting the **Model bending** parameter to `On` and the **Parameterization** parameter to `By stiffness and inertia` exposes this parameter. For more information, see Shaft Parameter Dependencies.

`Shaft geometry — Shaft cross-section`
`Solid` (default) | `Annular`

Shaft cross-sectional geometry.

Dependencies

Setting the **Parameterization** parameter to `By material properties` exposes this parameter. Each **Shaft geometry** option exposes other parameters. For more information, see Shaft Parameter Dependencies.

`Shaft outer diameter — Shaft outer diameter`
`0.075` m (default) | positive scalar

Shaft outer diameter.

Dependencies

Setting the **Parameterization** parameter to `By material properties` exposes this parameter. For more information, see Shaft Parameter Dependencies.

`Shaft inner diameter — Shaft inner diameter`
`0.05` m (default) | positive scalar

Shaft inner diameter. If the shaft is solid, specify `0`.

Dependencies

Setting the **Parameterization** parameter to By material properties and the **Shaft geometry** parameter to Annular exposes this parameter. For more information, see Shaft Parameter Dependencies.

Material density — Material density
7.8e3 kg/m^3 (default) | positive scalar

Material density.

Dependencies

Setting the **Parameterization** parameter to By material properties exposes this parameter. For more information, see Shaft Parameter Dependencies.

Shear modulus — Shear modulus
7.93e9 Pa (default) | positive scalar

Shear modulus for the shaft material.

Dependencies

Setting the **Parameterization** parameter to By material properties exposes this parameter. For more information, see Shaft Parameter Dependencies.

Young's modulus — Young's modulus
200e9 Pa (default) | positive scalar

Young's modulus for the material.

Dependencies

Setting the **Model bending** parameter to On and the **Parameterization** parameter to By material properties exposes this parameter. For more information, see Shaft Parameter Dependencies.

Torsion

Damping ratio from internal losses — Material damping ratio
0.01 (default) | positive scalar

Material damping ratio.

Viscous friction coefficients at base (B) and follower (F) — Viscous friction coefficients
[0, 0] N*m/(rad/s) (default) | non-negative vector

Viscous friction coefficients at base port, **B**, and the follower port, **F**.

Bending

Bending parameters are exposed only when the **Shaft** parameter **Model bending** is set to On. For more information, see Shaft Parameter Dependencies. The visibility of some **Bending** parameters also depends on the option that you choose for the **Advanced Bending** parameter **Bending vibration analysis method**.

The table shows how the visibility of some parameters depends on the option that you choose for other parameters. To learn how to read the table, see "Parameter Dependencies" on page B-2.

Bending Parameter Dependencies

Bending		
Base (B) mounting type — Choose `Clamped`, `Pinned`, `Free`, or `Bearing matrix`		
Clamped, **Pinned**, or **Free**	**Bearing matrix**	
	Base (B) translational stiffness [*x y*]	
	Base (B) rotational stiffness [*x y*]	
Follower (F) mounting type — Choose `Clamped`, `Pinned`, `Free`, or `Bearing matrix`		
Clamped, **Pinned**, or **Free**	**Bearing matrix**	
	Follower (F) translational stiffness [*x y*]	
	Follower (F) rotational stiffness [*x y*]	
Damping constant proportional to mass		
Damping constant proportional to stiffness		
Translation damping coefficient at base (B) and follower (F)*		
Rotation damping coefficient at base (B) and follower (F)*		
Damping constant proportional to stiffness**		
Rigid masses concentrically attached to shaft — Choose `None`, `Point mass`, or `Disk`		
None	**Point mass**	**Disk**
	Rigid mass locations along shaft (distance from B node)	
	Rigid masses	
	-	Rigid mass diametric moments of inertia about axis perpendicular to shaft
		Rigid mass polar moments of inertia about shaft axis
Static unbalances that excite bending		
Static unbalance distances from base (B)		
Static unbalance offset angle		

*This parameter is visible only when the **Advanced Bending** parameter **Bending vibration analysis method** is set to Lumped*

** *This parameter is visible only when the **Advanced Bending** parameter **Bending vibration analysis method** is set to* `Eigenmodes`

Base (B) mounting type — Base mounting type
Clamped (default) | Pinned | Free | Bearing matrix

Type of mounting at the base end of the shaft.

Dependencies

This parameter is visible only when the **Shaft** parameter **Model bending** is set to `On`. For more information, see Shaft Parameter Dependencies.

Setting this parameter to `Bearing matrix` exposes related parameters. For more information, see Bending Parameter Dependencies.

Base (B) translational stiffness [x y] — Base translational stiffness
[10e7 10e7] N/m (default) | non-negative vector

Translational stiffness for the mounting at the base end of the shaft, $[K_{Bxx}\ K_{Byy}]$. The first element is the support stiffness in the x direction. The second element is the support stiffness in the y direction.

Dependencies

This parameter is visible only when the **Shaft** parameter **Model bending** is set to `On` and the **Bending** parameter **Base (B) mounting type** is set to `Bearing matrix`. For more information, see Shaft Parameter Dependencies and Bending Parameter Dependencies.

Base (B) rotational stiffness [x y] — Base rotational stiffness
[10e5 10e5] N*m/rad (default) | non-negative vector

Rotational stiffness for the mounting at the base end of the shaft, $[K_{B\theta\theta}\ K_{B\varphi\varphi}]$. The first element is the support stiffness about the x direction. The second element is the support stiffness about the y direction.

Dependencies

This parameter is visible only when the **Shaft** parameter **Model bending** is set to `On` and the **Bending** parameter **Base (B) mounting type** is set to `Bearing matrix`. For more information, see Shaft Parameter Dependencies and Bending Parameter Dependencies.

Follower (F) mounting type — Follower mounting type
Clamped (default) | Pinned | Free | Bearing matrix

Type of mounting at the follower end of the shaft.

Dependencies

This parameter is visible only when the **Shaft** parameter **Model bending** is set to `On`. For more information, see Shaft Parameter Dependencies.

Setting this parameter to `Bearing matrix` exposes related parameters. For more information, see Bending Parameter Dependencies.

Follower (F) translational stiffness [x y] — Follower translational stiffness
`[10e7 10e7]` N/m (default) | non-negative vector

Translational stiffness for the mounting at the follower end of the shaft, $[K_{Fxx} \ K_{Fyy}]$. The first element is the support stiffness in the x direction. The second element is the support stiffness in the y direction.

Dependencies

This parameter is visible only when the **Shaft** parameter **Model bending** is set to `On` and the **Bending** parameter **Follower (F) mounting type** is set to `Bearing matrix`. For more information, see Shaft Parameter Dependencies and Bending Parameter Dependencies.

Follower (F) rotational stiffness [x y] — Follower rotational stiffness
`[10e5 10e5]` N*m/rad (default) | non-negative vector

Rotational stiffness for the mounting at the follower end of the shaft, $[K_{F\theta\theta} \ K_{F\varphi\varphi}]$. The first element is the support stiffness about the x direction. The second element is the support stiffness about the y direction.

Dependencies

This parameter is visible only when the **Shaft** parameter **Model bending** is set to `On` and the **Bending** parameter **Follower (F) mounting type** is set to `Bearing matrix`. For more information, see Shaft Parameter Dependencies and Bending Parameter Dependencies.

Damping constant proportional to mass — Damping constant, *a*
10 1/s (default) | positive scalar

Damping constant, *a*, proportional to mass.

When the eigenmodes bending model is enabled, a translation damper in each modal mass spring damper system has the damping coefficient aM_{Mode}, where M_{Mode} is the modal mass.

When the lumped parameter bending model is enabled, each lumped mass element has a translation damping coefficient $aM_{element}$ and rotation damping coefficient $aI_{element}$. $M_{element}$ is the combined mass of the shaft lumped mass element and a rigid point mass or disk, if a rigid point mass or disk is attached to that element. $I_{element}$ is the combined mass moment of inertia of the shaft lumped mass element and a rigid disk, if a rigid disk is attached to that element.

Dependencies

This parameter is visible only when the **Shaft** parameter **Model bending** is set to On. For more information, see Shaft Parameter Dependencies.

Translation damping coefficient at base(B) and follower(F) — Translation damping coefficient
[0, 0] N/(m/s) (default) | non-negative vector

Damping coefficients for a damper that acts on shaft translation motion at the base and follower.

The first element is for damping at the base and the second element is for damping at the follower. *X* and *Y* directions have the same damping.

Dependencies

This parameter is visible only when the **Shaft** parameter **Model bending** is set to On and the **Shaft** parameter **Bending vibration analysis method** is set to Lumped mass. For more information, see Shaft Parameter Dependencies and Bending Parameter Dependencies.

Rotation damping coefficient at base(B) and follower(F) — Rotation damping coefficient
[0, 0] N*m/(rad/s) (default) | non-negative vector

Damping coefficient for a damper that acts on shaft rotation motion at the base and follower ends in both the *X* and *Y* directions. The first element is for damping at the base and the second element is for damping at the follower. *X* and *Y* directions have the same damping.

Dependencies

This parameter is visible only when the **Shaft** parameter **Model bending** is set to `On` and the **Shaft** parameter **Bending vibration analysis method** is set to `Lumped mass`. For more information, see Shaft Parameter Dependencies and Bending Parameter Dependencies.

`Damping constant proportional to stiffness` — Damping constant, *b*
`1e-8 s` (default) | positive scalar

When the eigenmodes bending model is enabled, a translation damper in each modal mass spring damper system has the damping coefficient bK_{Mode}, where K_{Mode} is the modal stiffness.

Dependencies

This parameter is visible only when the **Shaft** parameter **Model bending** is set to `On` and the **Shaft** parameter **Bending vibration analysis method** is set to `Eigenmodes`. For more information, see Shaft Parameter Dependencies and Bending Parameter Dependencies.

`Rigid masses concentrically attached to shaft` — Type of rigid masses attached to shaft
`None` (default) | `Point mass` | `Disk`

Type, if any, of rigid masses attached to shaft.

Dependencies

This parameter is visible only when the **Shaft** parameter **Model bending** is set to `On` and the **Shaft** parameter **Bending vibration analysis method** is set to `Eigenmodes`. For more information, see Shaft Parameter Dependencies.

Setting this parameter to `Point mass` or `Disk` exposes related parameters. For more information, see Bending Parameter Dependencies.

`Rigid mass locations along shaft (distance from B node)` — Rigid mass locations along shaft in distance from base node
`0.5 m` (default) | non-negative scalar or vector

Rigid mass locations along shaft in distance from base node. For multiple masses, specify a row vector.

Dependencies

This parameter is visible only when the **Shaft** parameter **Model bending** is set to `On` and the **Bending** parameter **Rigid masses attached to shaft** is set to `Point mass`. For more information, see Shaft Parameter Dependencies and Bending Parameter Dependencies.

Rigid masses — Mass of rigid masses concentrically attached to shaft
2 kg (default) | non-negative scalar or vector

Mass of rigid masses concentrically attached to shaft. For multiple masses, specify a row vector.

Rigid mass diametric moments of inertia about axis perpendicular to shaft — Diametric moments of inertia about axis perpendicular to shaft
0.025 kg*m^2 (default) | non-negative scalar or vector

Rigid mass moments of inertia about axis perpendicular to the shaft. For multiple masses, specify a row vector.

Dependencies

This parameter is visible only when the **Shaft** parameter **Model bending** is set to `On` and the **Bending** parameter **Rigid masses attached to shaft** is set to `Disk`. For more information, see Shaft Parameter Dependencies and Bending Parameter Dependencies.

Rigid mass polar moments of inertia about shaft axis — Polar moments of inertia about shaft axis
0.050 kg*m^2 (default) | non-negative scalar or vector

Rigid polar mass moments of inertia about axis perpendicular to shaft.

Dependencies

This parameter is visible only when the **Shaft** parameter **Model bending** is set to `On` and the **Bending** parameter **Rigid masses attached to shaft** is set to `Disk`. For more information, see Shaft Parameter Dependencies and Bending Parameter Dependencies.

Static unbalances that excite bending — Excitational static unbalances
[.01, .01] m*kg (default) | non-negative vector

Static unbalances that excite bending.

Dependencies

This parameter is visible only when the **Shaft** parameter **Model bending** is set to On. For more information, see Shaft Parameter Dependencies.

`Static unbalance distances from base (B)` **— Static unbalance distances from base**
`[.33, .67] m` (default) | non-negative vector

Distance of excitational static unbalances from the base, **B** end of the shaft.

Dependencies

This parameter is visible only when the **Shaft** parameter **Model bending** is set to On. For more information, see Shaft Parameter Dependencies.

`Static unbalance offset angles` **— Static unbalance offset angles**
`[0, pi] rad` (default) | non-negative vector

Initial angle, about centerline of shaft relative to the *x*-axis, of the excitational static unbalances.

Dependencies

This parameter is visible only when the **Shaft** parameter **Model bending** is set to On. For more information, see Shaft Parameter Dependencies.

Advanced Bending

Advanced Bending parameters are exposed only when the **Shaft** parameter **Model bending** is set to On. For more information, see Shaft Parameter Dependencies.

The table shows how the visibility of some parameters depends on the option that you choose for other parameters. To learn how to read the table, see "Parameter Dependencies" on page B-2.

Advanced Bending Parameter Dependencies

Advanced Bending		
Bending vibration analysis method — Choose `Lumped mass` or `Eigenmodes` The option that you choose for this parameter affects the visibility of some **Bending** parameters. For more information, see Shaft Parameter Dependencies.		
Lumped mass	**Eigenmodes**	
	Bending mode determination — Choose `Simscape determined` or `User defined`	
	Simscape determined	**User defined**
	Limit number of modes	
	Nominal shaft speed for bending modes	
	Eigenfrequency initial search resolution	Modal frequencies
	Eigenfrequency refine factor	Shaft position
	Eigenfrequency upper limit	Mode shape matrix
	Shaft length increments for mode shape computations	

Bending vibration analysis method — Bending vibration analysis method
Lumped mass (default) | Eigenmodes

Method for analyzing the bending vibration.

Dependencies

This parameter is visible only when the **Shaft** parameter **Model bending** is set to On. For more information, see Shaft Parameter Dependencies.

Setting this parameter to Eigenmodes exposes related **Advanced Bending** parameters. For more information, see Advanced Bending Parameter Dependencies.

The option that you choose for this parameter also affects the visibility of some **Bending** parameters. For more information, see Bending Parameter Dependencies.

Bending mode determination — Bending mode determination method
Simscape determined (default) | User defined

Option to have Simscape determine the eigenmode frequencies and shapes or to define the bending mode frequencies and shapes using parameters that you specify.

Dependencies

This parameter is visible only when the **Shaft** parameter **Model bending** is set to `On` and the **Advanced Bending** parameter **Bending vibration analysis method** is set to `Eigenmodes`. Each option for this parameter exposes related parameters and hides unrelated parameters. For more information, see Shaft Parameter Dependencies and Advanced Bending Parameter Dependencies.

Limit number of modes — Maximum number of modes
6 (default) | positive integer scalar

Maximum number of modes that Simscape determines.

Dependencies

This parameter is visible only when the **Shaft** parameter **Model bending** is set to `On`, the **Advanced Bending** parameter **Bending vibration analysis method** is set to `Eigenmodes`, and the **Advanced Bending** parameter **Bending mode determination** is set to `Simscape determined`. For more information, see Shaft Parameter Dependencies and Advanced Bending Parameter Dependencies.

Nominal shaft speed for bending modes — Nominal shaft speed for bending modes
100 rpm (default) | non-negative scalar

Nominal shaft speed for bending modes.

Dependencies

This parameter is visible only when the **Shaft** parameter **Model bending** is set to `On` and the **Advanced Bending** parameter **Bending vibration analysis method** is set to `Eigenmodes`. For more information, see Shaft Parameter Dependencies and Advanced Bending Parameter Dependencies.

Eigenfrequency initial search resolution — Eigenfrequency search resolution
5 rad/s (default) | positive integer scalar

Initial resolution for the eigenfrequency search.

The algorithm for determining the eigenfrequencies considers the values $[0:d\omega:\omega_{Max}]$ as possible eigenfrequencies, where $d\omega$ is the eigenfrequency search resolution and ω_{Max} is the eigengrequency upper limit.

Dependencies

This parameter is visible only when the **Shaft** parameter **Model bending** is set to On, the **Advanced Bending** parameter **Bending vibration analysis method** is set to Eigenmodes, and the **Advanced Bending** parameter **Bending mode determination** is set to Simscape determined. For more information, see Shaft Parameter Dependencies and Advanced Bending Parameter Dependencies.

Eigenfrequency refine factor — Eigenfrequency refinement factor
1e-4 rad/s (default) | non-negative scalar

Refinement factor for the resolution for the eigenfrequency search.

For each eigenfrequency, $\omega_{kINITIAL}$, that is found using the eigenfrequency search resolution, the algorithm refines with eigenfrequency value by repeating the solution procedure when considering $\omega_{kINITIAL}+d\omega*[-1:d\omega_{RefineFactor}:1]$ as possible eigenfrequencies. This refinement procedure may be important for obtaining the correct eigenmode shape.

Dependencies

This parameter is visible only when the **Shaft** parameter **Model bending** is set to On, the **Advanced Bending** parameter **Bending vibration analysis method** is set to Eigenmodes, and the **Advanced Bending** parameter **Bending mode determination** is set to Simscape determined. For more information, see Shaft Parameter Dependencies and Advanced Bending Parameter Dependencies.

Eigenfrequency upper limit — Eigenfrequency upper limit
50000 rad/s (default) | positive integer scalar

Eigenfrequency upper limit.

Dependencies

This parameter is visible only when the **Shaft** parameter **Model bending** is set to On, the **Advanced Bending** parameter **Bending vibration analysis method** is set to Eigenmodes, and the **Advanced Bending** parameter **Bending mode determination** is set to Simscape determined. For more information, see Shaft Parameter Dependencies and Advanced Bending Parameter Dependencies.

Shaft length increments for mode shape computations — Shaft length increments
0.01 m (default) | positive scalar

Shaft length increments used for mode mass and shape computations.

The mode shapes are defined at the distances from the base, [0:*dz*:*L*], where *L* is the shaft length. These mode shape vectors are numerically integrated to determine the modal mass, stiffness, and damping.

Dependencies

This parameter is visible only when the **Shaft** parameter **Model bending** is set to On, the **Advanced Bending** parameter **Bending vibration analysis method** is set to Eigenmodes, and the **Advanced Bending** parameter **Bending mode determination** is set to Simscape determined. For more information, see Shaft Parameter Dependencies and Advanced Bending Parameter Dependencies.

Modal frequencies — Modal frequencies
[162, 162, 649, 649, 1460, 1460, 2596, 2596] rad/s (default) | vector

Modal frequencies.

Dependencies

This parameter is visible only when the **Shaft** parameter **Model bending** is set to On, the **Advanced Bending** parameter **Bending vibration analysis method** is set to Eigenmodes, and the **Advanced Bending** parameter **Bending mode determination** is set to User defined. For more information, see Shaft Parameter Dependencies and Advanced Bending Parameter Dependencies.

Shaft position — Shaft position
[0 : 0.1 : 1] m (default) | vector

Shaft position.

Dependencies

This parameter is visible only when the **Shaft** parameter **Model bending** is set to On, the **Advanced Bending** parameter **Bending vibration analysis method** is set to Eigenmodes, and the **Advanced Bending** parameter **Bending mode determination** is set to User defined. For more information, see Shaft Parameter Dependencies and Advanced Bending Parameter Dependencies.

`Mode shape matrix` **— Mode shape matrix**
[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0; 0.309, 0, 0, 0.309,
0.5878, 0, 0, 0.5878, 0.809, 0, 0, 0.809, 0.9511, 0, 0, 0.9511;
0.5878, 0, 0, 0.5878, 0.9511, 0, 0, 0.9511, 0.9511, 0, 0, 0.9511,
0.5878, 0, 0, 0.5878; 0.809, 0, 0, 0.809, 0.9511, 0, 0, 0.9511,
0.309, 0, 0, 0.309, -0.5878, 0, 0, -0.5878; 0.9511, 0, 0, 0.9511,
0.5878, 0, 0, 0.5878, -0.5878, 0, 0, -0.5878, -0.9511, 0, 0,
-0.9511; 1, 0, 0, 1, 0, 0, 0, 0, -1, 0, 0, -1, 0, 0, 0, 0; 0.9511,
0, 0, 0.9511, -.5878, 0, 0, -0.5878, -0.5878, 0, 0, -0.5878, 0.9511,
0, 0, 0.9511;0.809, 0, 0, 0.809, -.9511, 0, 0, -.9511, 0.309, 0, 0,
0.309, 0.5878, 0, 0, 0.5878; 0.5878, 0, 0, 0.5878, -0.9511, 0, 0,
-0.9511, 0.9511, 0, 0, 0.9511, -0.5878, 0, 0, -0.5878; 0.309, 0, 0,
0.309, -0.5878, 0, 0, -0.5878, 0.809, 0, 0, 0.809, -0.9511, 0, 0,
-0.9511; 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0] (default) |
vector

The mode shape matrix has dimensions n-by-$2m$, where n is the length of elements in the **Shaft position** vector and m is the length of **Modal frequencies** vector. The mode shape matrix has the form [U_{1x} U_{1y} U_{2x} U_{2y}... U_{mx} U_{my}], where the column vectors U_{ix} and U_{iy} are the mode shape deflections in the x and y directions, respectively, for the ith node. The algorithm accounts for the properties specified for the **Shaft** and **Bending** parameters that are used to compute the modal properties.

Dependencies

This parameter is visible only when the **Shaft** parameter **Model bending** is set to `On`, the **Advanced Bending** parameter **Bending vibration analysis method** is set to `Eigenmodes`, and the **Advanced Bending** parameter **Bending mode determination** is set to `User defined`. For more information, see Shaft Parameter Dependencies and [6].

Initial Conditions

`Initial shaft torsional deflection` **— Initial shaft torsional deflection**
0 rad (default) | non-negative scalar

Angular deflection of the shaft at the start of simulation.

A positive initial deflection results in a positive rotation of the base, **B**, end of the shaft relative to the follower, **F**, end of the shaft.

Initial shaft angular velocity — Initial angular velocity
0 rpm (default) | non-negative scalar

Angular velocity of the base, **B**, end of the shaft relative to the follower, **F**, end of the shaft at the start of simulation.

References

[1] Adams, M.L., *Rotating Machinery Vibration*. CRC Press, New York, 2010.

[2] Bathe, K. J., *Finite Element Procedures*. Prentice Hall, Inc, United States, 1996.

[3] Chudnovsky, V., Kennedy, D, Mukherjee, A., and Wendlandt, J., Modeling Flexible Bodies in SimMechanics and Simulink. *MATLAB Digest,* 14(3), May 2006,

[4] Miller, S., Soares, T., Van Weddingen, Y., Wendlandt, J., Modeling Flexible Bodies with Simscape Multibody. The MathWorks, 2017.

[5] Muszynska, A., *Rotordynamics,* Taylor & Francis, 2005

[6] Rao, S.S., *Vibration of Continuous Systems*. John Wiley & Sons, Hoboken, NJ, 2007.

[7] Wu, J.S., Lin, F.T., Shaw, H.J., *Analytical Solution for Whirling Speeds and Mode Shapes of a Distributed-Mass Shaft With Arbitrary Rigid Disks*. ASME Journal of Applied Mechanics, Volume 81, 2014.

See Also
Inertia | Rod | Rotational Damper | Rotational Spring | Torsional spring damper

Introduced in R2018b

Force Noise Source

Model zero-mean normally (Gaussian) distributed force
Library: Simscape / Driveline / Sources

Description

The Force Noise Source block generates zero-mean normally (Gaussian) distributed force using the Random Number source in the Simscape Foundation library. The force produced by the block depends on two factors:

- The noise power spectral density input to the block
- A random number seed that you control using the **Repeatability** and, as applicable, **Seed** parameters

The defining equation for the force that the block generates is

$$F = \sqrt{PSD/2}\frac{N(0, 1)}{\sqrt{h}},$$

where:

- F is the output force.
- PSD is the single-sided noise power spectral density.
- N is a Gaussian random number with zero mean and standard deviation of one.
- h is the sampling interval.

Ports

Input

P — Single-sided noise power spectral density
physical signal

The power spectral density (*PSD*) of noise is the average noise power per unit of bandwidth, in N^2/Hz.

Conserving

B — Base interface
mechanical translational

Mechanical translational conserving port associated with the base interface.

F — Follower interface
mechanical translational

Mechanical translational conserving port associated with the follower interface.

Parameters

Sample time — Time step period and offset
`1e-3` s (default) | [*step*, *offset*] s

The values of the time *step* period and the initial time *offset*. If you specify a scalar value for *step*, the block assumes an *offset* value of `0`. To specify a nonzero value for the initial time offset, specify the parameter values using the vector [*step*, *offset*]. The *offset* value must be less than the *step* value and greater than or equal to zero.

Repeatability — Random number seed control
`Not repeatable` (default) | `Repeatable` | `Specify seed`

The random number seed is the number that initializes the random number generator. The seed is `0` or a positive integer. To control the random number seed, set this parameter to:

- `Not repeatable` — The seed changes every time you simulate your model. The block resets the random seed using the MATLAB random number generator command:

  `seed = randi(2^32-1);`

- `Repeatable` — The seed is the same random number at the start of every simulation. The block sets the value using the same MATLAB random number generator command used by the `Not repeatable` parameter.

  When you add a Force Noise Source block to your model from the Sources library, the block generates and stores a random value for the repeated seed. When you make a copy of the Force Noise Source block from an existing block in a model, the copy generates a new random value for the repeated seed.

- `Specify seed` — The seed is a number that you specify using the **Seed** parameter. The **Seed** parameter is only available when you choose `Specify seed` for the **Repeatability** parameter.

Seed — Random number generation seed value
0 (default)

The seed must be 0 or a positive integer. This parameter is only available when you select `Specify seed` for the **Repeatability** parameter.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also

Simscape Blocks
Random Number | Rotational Velocity Noise Source | Sinusoidal Force Source | Torque Noise Source | Translational Velocity Noise Source
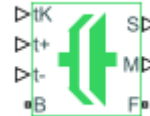
Functions
randi

Introduced in R2016b

Fundamental Friction Clutch

Friction clutch with kinetic and static-limit friction torques as inputs
Library: Simscape / Driveline / Clutches / Fundamental
            Components

Description

This block represents a friction clutch, a mechanism that transmits rotational power through friction. The clutch contains two friction plate sets, each rigidly connected to a driveshaft, that come into contact to engage. Once in contact, the plates experience frictional torques that enable power to flow between the driveshafts.

The clutch can be bidirectional or unidirectional. A bidirectional clutch can slip in the positive and negative directions. A unidirectional clutch can slip only in the positive direction. The slip direction is positive if the follower shaft spins faster than the base shaft and negative if it spins slower. The block defines the slip velocity as the difference

$$\omega = \omega_F - \omega_B,$$

where:

- ω is the slip velocity.
- ω_F is the angular velocity of the follower driveshaft.
- ω_B is the angular velocity of the base driveshaft.

Clutch States

The clutch can be in three states:

- Locked — Real clutch state in which the friction plates spin as a unit. A locked clutch has one rotational degree of freedom. It experiences no power losses due to friction.

- Unlocked — Real clutch state in which the friction plates slip with respect to each other. An unlocked clutch has two rotational degrees of freedom. It experiences power losses equal to the product of the slip velocity and the kinetic friction torque.

- Wait — Virtual clutch state that maintains the motion of the previous state while testing for locking and unlocking. The clutch degrees of freedom and power losses depend on the previous clutch state.

The schematic shows the conditions under which the clutch is locked and unlocked. The clutch is generally locked if the torque it transfers lies between its static friction torque limits and the magnitude of the slip velocity is smaller than its velocity tolerance. The clutch unlocked otherwise.
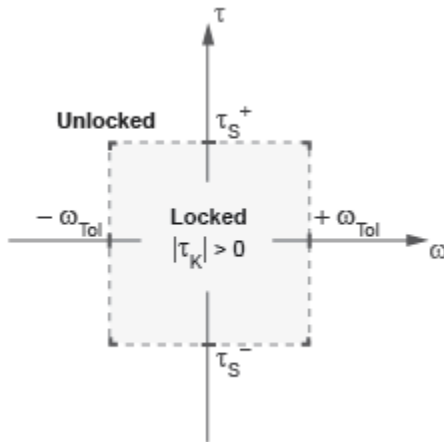
In the schematic:

- τ is the torque transferred between the clutch plates.

- τ_S^- and τ_S^+ are the static friction torque limits.

- τ_K is the kinetic friction torque between the clutch plates.

- ω_{Tol} is the slip velocity tolerance of the clutch.

- ω is the slip velocity between the clutch plates.

The block identifies the clutch state through physical signal port M using values -1, 0, and +1. The table summarizes the correspondence between the states and the output values.

State	Value
Unlocked Forward or Wait Forward	+1
Unlocked Reverse or Wait Reverse	-1
Locked or Unlocked Initial State	0

State Transitions

At the start of simulation, the clutch is in one of two states—locked or initial unlocked. The initial unlocked state is unique in that it lacks a direction of motion. The clutch remains in this state until the clutch slip velocity becomes nonzero. The clutch then transitions to the appropriate state—unlocked reverse or unlocked forward—according to the schematic.

During simulation, the clutch tests various dynamic conditions to determine the appropriate state transitions, if any. The schematics show the possible transitions, their dynamic conditions, and the resulting states. If the clutch is unidirectional, the schematic reduces to the right half. The diagram shows transitions for a one-way clutch.
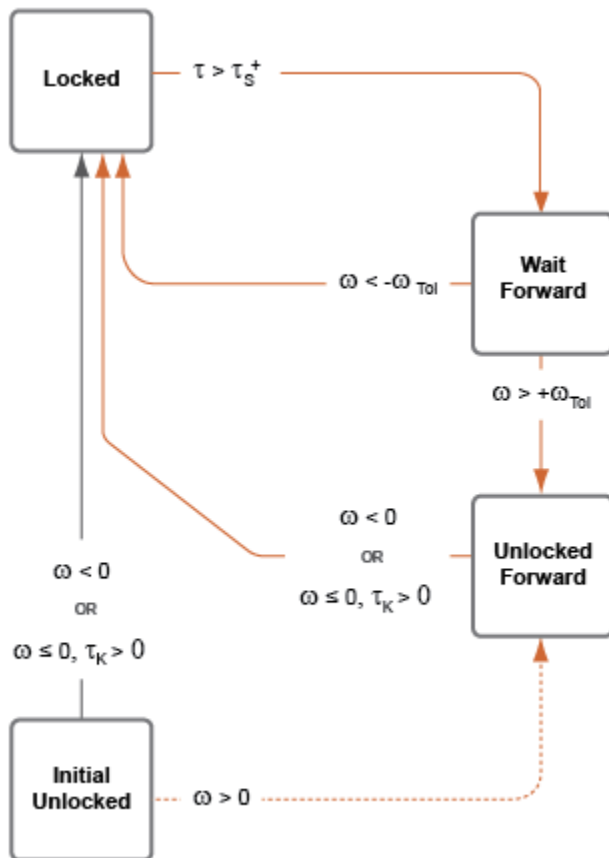
The diagram shows transitions for a two-way clutch.

Thermal Model

You can model the effects of heat flow and temperature change by selecting a thermal block variant. Selecting a thermal variant:

- Exposes port **H**, a conserving port in the thermal domain.
- Enables the **Thermal mass** parameter which allows you to specify the ability of the component to resist changes in temperature.

- Enables the **Temperature** variable which allows you to set the priority and initial target values for the block variables before simulating. For more information, see "Set Priority and Initial Target for Block Variables" (Simscape).

To select a thermal variant, right-click the block in your model and from the context menu, select **Simscape > Block choices**. Select a variant that includes a thermal port.

Ports

Input

tK — Kinetic friction torque, N*m
physical signal

Physical signal input port for the kinetic friction torque.

t+ — Static friction torque upper bound, N*m
physical signal

Physical signal input port for the upper static friction torque limit.

t- — Static friction torque lower bound, N*m
physical signal

Physical signal input port for the lower static friction torque limit.

Output

S — Clutch slip velocity, rad/sec
physical signal

Physical signal output port for the clutch slip velocity.

M — Clutch state
physical signal

Physical signal output port for the clutch state.

Conserving

B — Rotational velocity, rad/sec
mechanical rotational

Rotational conserving ports associated with the driving shaft.

F — Rotational speed, rad/sec
mechanical rotational

Rotational conserving ports associated with the driven shaft

H — Heat flow, W
thermal

The thermal conserving port is optional and is hidden by default.

Dependencies

To expose the port, select a thermal block variant.

Parameters

Directionality — Permissible inter-plate slip directions
`Bidirectional` (default) | `Unidirectional`

Slip directions the clutch allows between its plates. A bidirectional clutch allows positive and negative slip velocities. A unidirectional clutch allows only positive slip velocities.

The unidirectional clutch is equivalent to a friction clutch connected in parallel to a one-way clutch that disengages only when the slip velocity becomes positive. To model a unidirectional clutch with slip in the negative direction, reverse the base and follower port connections.

Clutch velocity tolerance — Slip velocity threshold
`0.001` rad/s (default)

Slip velocity below which the clutch can lock. The clutch locks if, after falling below the clutch velocity tolerance, the kinetic friction torque is nonzero and the transferred torque is between the static friction torque limits.

`Initial state` — Initial clutch state
`Unlocked` (default) | `Locked`

Clutch state at the start of simulation. The clutch can be in one of two states, locked and unlocked. A locked clutch constrains the base and follower shafts to spin at the same velocity, that is, as a single unit. An unlocked clutch allows the two shafts to spin at different velocities, resulting in slip between the clutch plates.

A clutch that starts simulation unlocked lacks a direction of motion. For this reason, after checking that the unlocked state is valid, the block automatically determines the appropriate direction of motion based on the driveline dynamics. Based on the direction of motion, the clutch then transitions to an unlocked-reverse or unlocked-forward state.

`Thermal mass` — Resistance to temperature change
`25 kJ/K` (default)

Thermal energy required to change the component temperature by a single degree. The greater the thermal mass, the more resistant the component is to temperature change.

Dependencies

Selecting a thermal block variant enables this parameter.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using Simulink® Coder™.

See Also
Differential | Disk Friction Clutch | Torque Converter | Variable Ratio Transmission

Topics
"Gear Coupling Control Using Clutches"

Introduced in R2011a

Generic Engine

Internal combustion engine with throttle and rotational inertia and time lag

Library

Simscape / Driveline / Engines

Description

The Generic Engine block represents a general internal combustion engine. Engine types include spark-ignition and diesel. Speed-power and speed-torque parameterizations are provided. A throttle physical signal input specifies the normalized engine torque. Optional dynamic parameters include crankshaft inertia and response time lag. A physical signal port outputs engine fuel consumption rate based on choice of fuel consumption model. Optional speed and redline controllers prevent engine stall and enable cruise control.
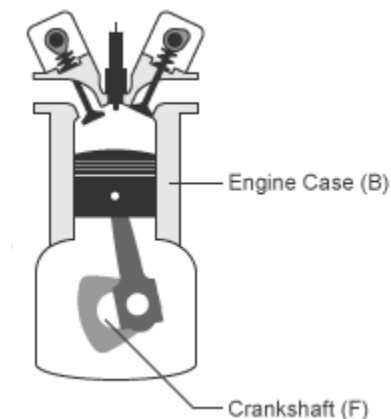
Generic Engine Model

By default, the Generic Engine model uses a programmed relationship between torque and speed, modulated by the throttle signal.

Engine Speed, Throttle, Power, and Torque

The engine model is specified by an *engine power demand* function $g(\Omega)$. The function provides the maximum power available for a given engine speed Ω. The block parameters (maximum power, speed at maximum power, and maximum speed) normalize this function to physical maximum torque and speed values.

The normalized throttle input signal T specifies the actual engine power. The power is delivered as a fraction of the maximum power possible in a steady state at a fixed engine speed. It modulates the actual power delivered, P, from the engine: $P(\Omega,T) = T \cdot g(\Omega)$. The engine torque is $\tau = P/\Omega$.

Engine Power Demand

The engine power is nonzero when the speed is limited to the operating range, $\Omega_{min} \leq \Omega \leq \Omega_{max}$. The absolute maximum engine power P_{max} defines Ω_0 such that $P_{max} = g(\Omega_0)$. Define $w \equiv \Omega/\Omega_0$ and $g(\Omega) \equiv P_{max} \cdot p(w)$. Then $p(1) = 1$ and $dp(1)/dw = 0$. The torque function is:

$\tau = (P_{max}/\Omega_0) \cdot [p(w)/w]$.

You can derive forms for $p(w)$ from engine data and models. Generic Engine uses a third-order polynomial form:
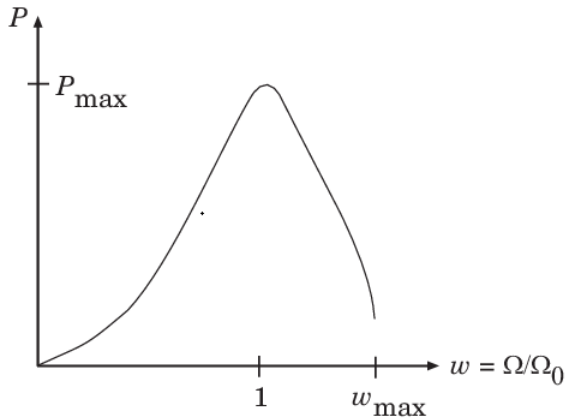
$p(w) = p_1 \cdot w + p_2 \cdot w^2 - p_3 \cdot w^3$

satisfying

$p_1 + p_2 - p_3 = 1$, $p_1 + 2p_2 - 3p_3 = 0$.

In typical engines, the p_i are positive. This polynomial has three zeros, one at $w = 0$, and a conjugate pair. One of the pair is positive and physical; the other is negative and unphysical:

$$w_{\pm} = \frac{1}{2}\left(-p_2 \pm \sqrt{p_2^2 + 4p_1p_3}\right)$$

Typical Engine Power Demand Function

Restrictions on Engine Speed and Power

- For the engine power polynomial, there are restrictions, as shown, on the polynomial coefficients p_i, to achieve a valid power-speed curve.

- If you use tabulated power or torque data, corresponding restrictions on $P(\Omega)$ remain.

Specify the speed and power as $w = \Omega/\Omega_0$ and $p = P(\Omega)/P_0$ and define the boundaries as $w_{min} = \Omega_{min}/\Omega_0$ and $w_{max} = \Omega_{max}/\Omega_0$. Then:

- The engine speed is restricted to a positive range above the minimum speed and below the maximum speed: $0 \leq w_{min} \leq w \leq w_{max}$.

- The engine power at minimum speed must be nonnegative: $p(w_{min}) \geq 0$. If you use the polynomial form, this condition is a restriction on the p_i:

  $p(w_{min}) = p_1 \cdot w_{min} + p_2 \cdot w^2_{min} - p_3 \cdot w_{min}^3 \geq 0.$

- The engine power at maximum speed must be nonnegative: $p(w_{max}) \geq 0$. If you use the polynomial form, this condition is a restriction on w_{max}: $w_{max} \leq w_+$.

Engine Power Forms for Different Engine Types

For the default parameterization, Generic Engine provides two choices of internal combustion engine types, each with different engine power demand parameters.

Power Demand Coefficient	Engine Type:	
	Spark-Ignition	Diesel
p_1	1	0.6526
p_2	1	1.6948
p_3	1	1.3474

Idle Speed Controller Model

The idle speed controller adjusts the throttle signal to increase engine rotation below a reference speed according to the following expressions:

$$\Pi = \max(\Pi_i, \Pi_c)$$

and

$$\frac{d(\Pi_c)}{dt} = \frac{0.5 \cdot \left(1 - \tanh\left(4 \cdot \frac{\omega - \omega_r}{\omega_t}\right)\right) - \Pi_c}{\tau}$$

where:

- Π — Engine throttle
- Π_i — Input throttle (port T)
- Π_c — Controller throttle
- ω — Engine speed
- ω_r — Idle speed reference
- ω_t — Controller speed threshold
- τ — Controller time constant

The controlled throttle increases with a first-order lag from zero to one when engine speed falls below the reference speed. When the engine speed rises above the reference speed, the controlled throttle decreases from one to zero. When the difference between engine velocity and reference speed is smaller than the controller speed threshold, the *tanh* function smooths the time derivative of the controlled throttle. The controlled throttle is limited to the range 0–1. The engine uses the larger of the input and controlled throttle values. If engine time lag is included, the controller changes the input *before* the lag is computed.

Redline Controller Model

While the idle speed controller determines the minimum throttle value for maintaining engine speed, the redline controller prevents excessive speed based on a maximum throttle input. To determine the maximum throttle value, the redline controller uses the idle speed controller model equation. However, for the redline controller:

- ω_r is the redline speed reference.
- ω_t is the redline speed threshold.
- τ is the redline time constant.

Performance

To increase simulation speed, use the default option, `No fuel consumption`, for the **Fuel consumption model** parameter.

If you select any other option for the **Fuel consumption model**, the block solves a nonlinear equation that is required for calculating fuel consumption. The block solves the equation even if the **FC** port, which reports the fuel consumption rate, is not connected to another block.

When the parameter is set to `No fuel consumption`, the block does not calculate fuel consumption, even if the **FC** port is connected to another block.

Limitations

This block contains an engine time lag limitation.

Engine Time Lag

Engines lag in their response to changing speed and throttle. The Generic Engine block optionally supports lag due to a changing throttle only. Time lag simulation increases model fidelity but reduces simulation performance.

Ports

Port	Description
B	Rotational conserving port representing the engine block
F	Rotational Conserving port representing the engine crankshaft
T	Physical signal input port specifying the normalized engine throttle level
P	Physical signal output port reporting the instantaneous engine power
FC	Physical signal output port reporting the fuel consumption rate

Port **T** accepts a signal with values in the range 0–1. The signal specifies the engine torque as a fraction of the maximum torque possible in steady state at fixed engine speed. The signal saturates at zero and one. Values below zero are interpreted as zero. Values above one are interpreted as one.

Port **FC** does not output data when the **Fuel consumption model** parameter is set to No fuel consumption.

Parameters

Engine Torque

The table shows how the visibility of some parameters depends on the option that you choose for other parameters. To learn how to read the table, see "Parameter Dependencies" on page B-2.

Engine Torque Parameter Dependencies

Engine Torque		
Model parameterization — Choose `Normalized 3rd-order polynomial matched to peak power`, `Tabulated torque data`, or `Tabulated power data`		
Normalized 3rd-order polynomial matched to peak power	**Tabulated torque data**	**Tabulated power data**
Engine type — Choose `Spark-ignition` or `Diesel`	Speed vector	
Maximum power	Torque vector	
Speed at maximum power	Interpolation method — Choose `Linear` or `Smooth`	
Maximum speed		
Stall speed		

Model parameterization

Select how to model the engine. Choose between these options, each of which enable other parameters:

- `Normalized 3rd-order polynomial matched to peak power` — Parametrize the engine with a power function controlled by power and speed characteristics. This is the default option.

- `Tabulated torque data` — Engine is parametrized by speed–torque table that you specify.

- `Tabulated power data` — Engine is parametrized by speed–power table that you specify.

Engine type

Choose type of internal combustion engine. Choose between `Spark-ignition`, the default option, and `Diesel`.

Selecting `Normalized 3rd-order polynomial matched to peak power` for the **Model parameterization** parameter exposes this parameter.

For more information, see the Engine Torque Parameter Dependencies table.

Maximum power

Maximum power P_{max} that the engine can output. The default is 150 kW.

Selecting `Normalized 3rd-order polynomial matched to peak power` for the **Model parameterization** parameter exposes this parameter.

For more information, see the Engine Torque Parameter Dependencies table.

Speed at maximum power

Engine speed Ω_0 at which the engine is running at maximum power. The default is 4500 rpm.

Selecting `Normalized 3rd-order polynomial matched to peak power` for the **Model parameterization** parameter exposes this parameter.

For more information, see the Engine Torque Parameter Dependencies table.

Maximum speed

Maximum speed Ω_{max} at which the engine can generate torque. The default is 6000 rpm.

During simulation, if Ω exceeds this maximum, the simulation stops with an error. The engine maximum speed Ω_{max} cannot exceed the engine speed at which the engine power becomes negative.

Selecting `Normalized 3rd-order polynomial matched to peak power` for the **Model parameterization** parameter exposes this parameter.

For more information, see the Engine Torque Parameter Dependencies table.

Stall speed

Minimum speed Ω_{min} at which the engine can generate torque. The default is 500 rpm.

During simulation, if Ω falls below this minimum, the engine torque is blended to zero.

Selecting `Normalized 3rd-order polynomial matched to peak power` for the **Model parameterization** parameter exposes this parameter.

For more information, see the Engine Torque Parameter Dependencies table.

Speed vector

Vector of values of the engine function's independent variable, the speed Ω. The default is `[500, 1000, 2000, 3000, 4000, 5000, 6000, 7000]` rpm.

The first and last speeds in the vector are interpreted as the stall speed and the maximum speed, respectively. If the speed falls below the stall speed, engine torque is blended to zero. If the speed exceeds the maximum speed, the simulation stops with an error.

Selecting `Tabulated torque data` or `Tabulated power data` for the **Model parameterization** parameter exposes this parameter.

For more information, see the Engine Torque Parameter Dependencies table.

Torque vector

Vector of values of the engine function's dependent variable, the torque τ. The default is `[380, 380, 380, 380, 350, 280, 200, 80]` N*m.

Selecting `Tabulated torque data` for the **Model parameterization** parameter exposes this parameter.

For more information, see the Engine Torque Parameter Dependencies table.

Power vector

Vector of values of the engine function's dependent variable, the power P. The default is `[20, 40, 78, 120, 145, 148, 125, 60]` kW.

Selecting `Tabulated power data` for the **Model parameterization** parameter exposes this parameter.

For more information, see the Engine Torque Parameter Dependencies table.

Interpolation method

Method to interpolate the engine speed–torque or speed–power function between discrete relative velocity values within the range of definition. Choose between `Linear`, the default choice, and `Smooth`.

Selecting `Tabulated torque data` or `Tabulated power data` for the **Model parameterization** parameter exposes this parameter.

For more information, see the Engine Torque Parameter Dependencies table.

Dynamics

The table shows how the visibility of some parameters depends on the option that you choose for other parameters. To learn how to read the table, see "Parameter Dependencies" on page B-2.

Dynamics Parameter Dependencies

Dynamics	
Inertia — Choose `No inertia` or `Specify inertia and initial velocity`	
No inertia	**Specify inertia and initial velocity**
	Engine inertia
	Initial velocity
Time constant — Choose `No lag - Suitable for HIL simulation` or `Specify time constant and initial value`	
No lag - Suitable for HIL simulation	**Specify time constant and initial value**
	Engine time constant
	Initial normalized throttle

Inertia

Select how to model the rotational inertia of the engine block. Choose between these options, each of which exposes other parameters:

- `No inertia` — Engine crankshaft is modeled with no inertia. This option is the default.
- `Specify inertia and initial velocity` — Engine crankshaft is modeled with rotational inertia and initial angular velocity.

Engine Inertia

Rotational inertia of the engine crankshaft. The default is `1 kg*m^2`.

Selecting `Specify inertia and initial velocity` for the **Inertia** parameter exposes this parameter.

For more information, see the Dynamics Parameter Dependencies table.

Initial velocity

Initial angular velocity $\Omega(0)$ of the engine crankshaft. The default is `800 rpm`.

Selecting `Specify inertia and initial velocity` for the **Inertia** parameter exposes this parameter.

For more information, see the Dynamics Parameter Dependencies table.

Time constant

Select how to model the time lag of the engine response. Choose between these options, each of which exposes other options:

- `No lag — Suitable for HIL simulation` — Engine reacts with no time lag. This option is the default.

- `Specify time constant and initial value` — Engine reacts with a time lag.

  Selecting `Specify time constant and initial value` exposes other parameters.

  For more information, see the Dynamics Parameter Dependencies table.

Engine time constant

Engine time lag. The default is `0.2 s`.

Selecting `Specify time constant and initial value` for the **Time constant** parameter exposes this parameter.

For more information, see the Dynamics Parameter Dependencies table.

Initial normalized throttle

Initial normalized engine throttle $T(0)$, ranging between zero and one. The default is `0`.

Selecting `Specify time constant and initial value` for the **Time constant** parameter exposes this parameter.

For more information, see the Dynamics Parameter Dependencies table.

Limits

Speed threshold

Width of the speed range over which the engine torque is blended to zero as Ω approaches the stall speed. The default is `100 rpm`.

Fuel Consumption

The table shows how the visibility of some parameters depends on the option that you choose for other parameters. To learn how to read the table, see "Parameter Dependencies" on page B-2.

Fuel Consumption Parameter Dependencies

Fuel Consumption				
Fuel consumption model — Choose `No fuel consumption`, `Constant per revolution`, `Fuel consumption by speed and torque`, `Brake specific fuel consumption by speed and torque`, or `Brake specific fuel consumption by speed and brake mean effective pressure`				
No fuel consumption	**Constant per revolution**	**Fuel consumption by speed and torque**	**Brake specific fuel consumption by speed and torque**	**Brake specific fuel consumption by speed and brake mean effective pressure**
	Fuel consumption per revolution			Displaced volume
				Revolutions per cycle
		Speed vector		
		Torque vector		Brake mean effective pressure vector
		Fuel consumption table	Brake specific fuel consumption table	
		Interpolation method — Choose `Linear` or `Smooth`		

Fuel consumption model

> Select a model for calculating engine-fuel consumption. Model parameterizations are compatible with standard industrial data. Choose between these options:

- No fuel consumption — The default option
- Constant per revolution
- Fuel consumption by speed and torque
- Brake specific fuel consumption by speed and torque
- Brake specific fuel consumption by speed and brake mean effective pressure

Some options expose other parameters. For more information, see the Fuel Consumption Parameter Dependencies table.

No fuel consumption

The block does not calculate fuel consumption even when the **FC** port, which reports the fuel consumption rate, is connected to another block. Selecting this option increases simulation speed.

Fuel consumption per revolution

Enter the volume of fuel consumed in one crankshaft revolution. The default is 25 mg/rev.

Selecting Constant per revolution for the **Fuel consumption model** parameter exposes this parameter.

For more information, see the Fuel Consumption Parameter Dependencies table.

Displaced volume

Enter the volume displaced by a piston stroke. The default is 400 cm^3.

Selecting Brake specific fuel consumption by speed and brake mean effective pressure for the **Fuel consumption model** parameter exposes this parameter.

For more information, see the Fuel Consumption Parameter Dependencies table.

Revolutions per cycle

Enter the number of crankshaft revolutions in one combustion cycle — e.g. 2 for a four-stroke engine, or 1 for a two-stroke engine. The default is 2.

Selecting Brake specific fuel consumption by speed and brake mean effective pressure for the **Fuel consumption model** parameter exposes this parameter.

For more information, see the Fuel Consumption Parameter Dependencies table.

Speed vector

Enter vector of engine speeds used in lookup table parameterizations. Vector size must match **Torque vector size**. The default is [1000, 2000, 3000, 4000, 5000, 6000] rpm. Selecting Fuel consumption by speed and torque, Brake specific fuel consumption by speed and torque, or Brake specific fuel consumption by speed and brake mean effective pressure for the **Fuel consumption model** parameter exposes this parameter.

For more information, see the Fuel Consumption Parameter Dependencies table.

Torque vector

Enter vector of engine torques used in the lookup table parameterizations. Vector size must match **Speed vector** size. The default is [0, 80, 160, 200, 240, 320, 360, 400] N*m. Selecting Fuel consumption by speed and torque or Brake specific fuel consumption by speed and torque for the **Fuel consumption model** parameter exposes this parameter.

For more information, see the Fuel Consumption Parameter Dependencies table.

Fuel consumption table

Enter matrix with fuel consumption rates corresponding to engine speed and torque vectors. The number of rows must equal the number of elements in the **Speed vector**. The number of columns must equal the number of elements in the **Torque vector**. The default is [.5, .9, 1.4, 1.6, 1.9, 2.7, 3.4, 4.4; 1, 1.7, 2.7, 3.1, 3.6, 5, 6, 7.4; 1.4, 2.7, 4, 4.8, 5.6, 7.5, 8.5, 10.5; 2, 3.6, 5.8, 6.7, 8, 10.4, 11.7, 13.3; 2.5, 4.8, 7.9, 9.4, 10.8, 14, 16.2, 18.6; 3.1, 6, 10.3, 11.9, 13.8, 18.4, 22, 26.5] g/s.

Selecting Fuel consumption by speed and torque for the **Fuel consumption model** parameter exposes this parameter.

For more information, see the Fuel Consumption Parameter Dependencies table.

Brake mean effective pressure vector

Enter vector of brake mean effective pressure (BMEP) values. The default is [0, 250, 500, 625, 750, 1000, 1150, 1250] kPa. The BMEP satisfies the expression:

$$BMEP = T \cdot \left(\frac{2\pi \cdot n_c}{V_d} \right)$$

where:

- *T* — Output torque
- n_c — Number of cycles per revolution
- V_d — Cylinder displaced volume

Selecting `Brake specific fuel consumption by speed and brake mean effective pressure` for the **Fuel consumption model** parameter exposes this parameter.

For more information, see the Fuel Consumption Parameter Dependencies table.

Brake specific fuel consumption table

Selecting `Brake specific fuel consumption by speed and torque` or `Brake specific fuel consumption by speed and brake mean effective pressure` for the **Fuel consumption model** parameter exposes this parameter.

For more information, see the Fuel Consumption Parameter Dependencies table.

For the `Brake specific fuel consumption by speed and torque` fuel model, enter the matrix with brake specific fuel consumption (BSFC) rates corresponding to engine speed and torque vectors. BSFC is the ratio of the fuel consumption rate to the output power. The number of rows must equal the number of elements in the **Speed vector**. The number of columns must equal the number of elements in the **Torque vector**.

For the `Brake specific fuel consumption by speed and brake mean effective pressure` fuel model, enter the matrix with brake specific fuel consumption (BSFC) rates corresponding to engine speed and brake mean effective pressure (BMEP) vectors. BSFC is the ratio of the fuel consumption rate to the output power. The number of rows must equal the number of elements in the **Speed vector**. The number of columns must equal the number of elements in the **Brake mean effective pressure vector**.

For both fuel-consumption models, the default is `[410, 380, 300, 280, 270, 290, 320, 380; 410, 370, 290, 270, 260, 270, 285, 320; 415, 380, 290, 275, 265, 270, 270, 300; 420, 390, 310, 290, 285, 280, 280, 285; 430, 410, 340, 320, 310, 300, 310, 320; 450, 430, 370, 340, 330, 330, 350, 380]` g/hr/kW.

Interpolation method

Select the interpolation method used to calculate fuel consumption at intermediate speed-torque values. Methods are `Linear` and `Smooth`. Outside the data range, fuel

consumption is held constant at the last value given in the lookup table. Selecting `Fuel consumption by speed and torque`, `Brake specific fuel consumption by speed and torque`, or `Brake specific fuel consumption by speed and brake mean effective pressure` for the **Fuel consumption model** parameter exposes this parameter.

For more information, see the Fuel Consumption Parameter Dependencies table.

Speed Control

The table shows how the visibility of some parameters depends on the option that you choose for other parameters. To learn how to read the table, see "Parameter Dependencies" on page B-2.

Speed Control Parameter Dependencies

Speed Control	
Idle speed control — Choose `Off` or `On`	
Off	**On**
	Idle speed reference
	Controller time constant
	Redline control
Redline control — Choose `Off` or `On`	
Off	**On**
	Redline speed
	Redline time constant
	Redline threshold speed

Idle speed control

Select speed control model. The options are:

- `Off` — Omit idle speed controller. Throttle input is used directly. This option is the default.
- `On` — Include idle speed controller to prevent engine stalling. This option exposes other parameters. For more information, see "Idle Speed Controller Model" on page 1-235.

1-247

Idle speed reference

Enter the value of the speed reference below which speed increases, and above which speed decreases. The default is `1000 rpm`.

Selecting `On` for the **Idle speed control** parameter exposes this parameter.

For more information, see the Speed Control Parameter Dependencies table.

Controller time constant

Enter the value of the time constant associated with an increase or decrease of the controlled throttle. The constant value must be positive. The default is `1 s`.

Selecting `On` for the **Idle speed control** parameter exposes this parameter.

For more information, see the Speed Control Parameter Dependencies table.

Controller threshold speed

Parameter used to smooth the controlled throttle value when the engine's rotational speed crosses the idle speed reference. For more information, see "Idle Speed Controller Model" on page 1-235. Large values decrease controller responsiveness. Small values increase computational cost. This parameter must be positive. The default is `1 rpm`.

Selecting `On` for the **Idle speed control** parameter exposes this parameter.

For more information, see the Speed Control Parameter Dependencies table.

Redline control

Select redline control model. Options include `Off` and `On`.

- `Off` — Omit redline controller. Throttle depends only on the idle speed controller. This option is the default.
- `On` — Include redline controller to prevent excessive speed. This option exposes other parameters.

  For more information, see the Speed Control Parameter Dependencies table.

Redline speed

Enter the value of the speed reference above which the redline control activates. The default is `5000 rpm`.

Selecting `On` for the **Redline control** parameter exposes this parameter.

For more information, see the Speed Control Parameter Dependencies table.

Redline time constant

Enter the value of the time constant associated with an increase or decrease of the controlled throttle. The constant value must be positive. The default is `1 s`.

Selecting `On` for the **Redline control** parameter exposes this parameter.

For more information, see the Speed Control Parameter Dependencies table.

Redline threshold speed

Specify the width of the region around the redline speed where the controller goes from fully enabled to not enabled. The block uses this parameter for smoothing the controlled throttle value when the engine's rotational speed crosses the redline speed reference. Large values decrease controller responsiveness. Small values increase computational cost. This parameter must be positive. The default is `1 rpm`.

Selecting `On` for the **Redline control** parameter exposes this parameter.

For more information, see the Speed Control Parameter Dependencies table.

Real-Time Simulation

Hardware-in-the-Loop Simulation

For optimal simulation performance, set the **Dynamics > Time Constant** parameter to `No time constant - Suitable for HIL simulation`.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also

Simscape Blocks
Piston | Piston Engine

Topics
"Complete Vehicle Model"
"Adjust Model Fidelity"
"Hydraulically-Actuated Driveline Clutch"
"Vehicle with Dual Clutch Transmission"
Vehicle with Four-Speed Transmission
"Vehicle with Four-Wheel Drive"
Vehicle with Manual Transmission
"Prepare Simscape Driveline Models for Real-Time Simulation Using Simscape Checks"
"Troubleshoot Engine Issues"

Harmonic Drive

High-ratio speed reducer based on elastic deformation of an elliptical gear

Library

Gears

Description

This block represents a compact, high-ratio, speed reduction mechanism that contains three key components:

- Strain wave generator
- Elliptical gear
- Circular ring gear

The strain wave generator comprises an elliptical plug mated to a raced ball bearing. It sits inside an elastic metal gear, deforming it into a slight elliptical pattern. Rotation of the elliptical pattern in the body of the gear constitutes a strain wave.

The elliptically deformed gear engages the internal teeth of a fixed circular ring gear of only slightly larger diameter. Meshing occurs concurrently at the two elongated ends of the elliptical gear. This design doubles the teeth in mesh, boosting the torque capacity of the drive system.
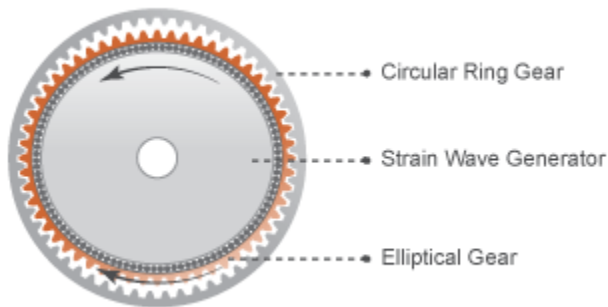
During normal operation, the base shaft drives the strain wave generator. The elliptical plug spins freely inside the elastic metal gear, propagating the strain wave about the gear rotation axis. This strain wave causes the elliptical gear teeth to engage the internal teeth of the circular ring gear progressively.

The internal meshing between the two gears causes the elliptical gear axis to spin counter to the elliptical strain wave. For every clockwise rotation that the strain wave generator completes, the elliptical gear axis rotates counterclockwise by a small amount.

Large reduction ratios arise from the near-equal gear tooth numbers. The effective gear reduction ratio is:

$$r = \frac{n_\mathrm{E}}{n_\mathrm{C} - n_\mathrm{E}},$$

where:

- r is the gear reduction ratio.
- n_C is the tooth number of the circular ring gear.
- n_E is the tooth number of the deformable elliptical gear.

Optional parameters account for power losses due to gear meshing and viscous friction. The Simple Gear block provides the foundation for this block. For more information, see Simple Gear.

Thermal Modeling

You can model the effects of heat flow and temperature change through an optional thermal conserving port. By default, the thermal port is hidden. To expose the thermal port, right-click the block in your model and, from the context menu, select **Simscape > Block choices**. Select a variant that includes a thermal port. Specify the associated thermal parameters for the component.

Ports

Port	Description
B	Conserving rotational port representing the base shaft
F	Conserving rotational port representing the follower shaft

Parameters

Main

Number of teeth on elliptical gear

Total number of teeth protruding outward from the elliptical gear perimeter. This number should be slightly smaller than the number of teeth on the circular ring gear. The ratio of the two gear tooth numbers defines the relative angular velocities of the base and follower shafts. The default value is 100.

Number of teeth on circular gear

Number of teeth protruding inward from the circular ring gear perimeter. This number should be slightly larger than the number of teeth on the elliptical gear. The ratio of the two gear tooth numbers defines the relative angular velocities of the base and follower shafts. The default value is 102.

Meshing Losses

Parameters for meshing losses vary with the block variant chosen—one with a thermal port for thermal modeling and one without it.

Without Thermal Port

Friction model

List of friction models at various precision levels for estimating power losses due to meshing.

- `No meshing losses - Suitable for HIL simulation` — Neglect friction between gear cogs. Meshing is ideal.

- `Constant efficiency` — Reduce torque transfer by a constant efficiency factor. This factor falls in the range $0 < \eta \leq 1$ and is independent of load. Selecting this option exposes additional parameters.

    Constant efficiency

    Efficiency

    Torque transfer efficiency (η) between base and follower shafts. This parameter is inversely proportional to the meshing power losses. Values can range from `0` to `1`. The default value is `0.95`.

    Follower power threshold

    Absolute value of the follower shaft power above which the full efficiency factor is in effect. A hyperbolic tangent function smooths the efficiency factor from zero when at rest to the full efficiency value at the power threshold.

    As a guideline, the power threshold should be lower than the expected power transmitted during simulation. Higher values might cause the block to underestimate efficiency losses. Very low values might, however, raise the computational cost of simulation. The default value is `0.001` W.

- `Load-dependent efficiency` — Reduce torque transfer by a variable efficiency factor. This factor falls in the range $0 < \eta < 1$ and varies with the torque load. Selecting this option exposes additional parameters.

    Load-Dependent Efficiency

    Input shaft torque at no load

    Net torque (τ_{idle}) acting on the input shaft in idle mode, e.g., when torque transfer to the output shaft equals zero. For nonzero values, the power input in idle mode completely dissipates due to meshing losses. The default value is `0.1` N*m.

Nominal output torque

Output torque (τ_F) at which to normalize the load-dependent efficiency. The default value is 5 N*m.

Efficiency at nominal output torque

Torque transfer efficiency (η) at the nominal output torque. Efficiency values must fall in the interval [0, 1]. Larger efficiency values correspond to greater torque transfer between the input and output shafts. The default value is `0.95`.

Follower angular velocity threshold

Absolute value of the follower shaft angular velocity above which the full efficiency factor is in effect (ω_F). Below this value, a hyperbolic tangent function smooths the efficiency factor to one, lowering the efficiency losses to zero when at rest.

As a guideline, the angular velocity threshold should be lower than the expected angular velocity during simulation. Higher values might cause the block to underestimate efficiency losses. Very low values might, however, raise the computational cost of simulation.

The default value is `0.01` rad/s.

With Thermal Port

Friction model

Choice of model for representing friction. The model can incorporate the temperature and load dependences of friction or just the temperature dependence. The default setting is `Temperature-dependent efficiency`.

Temperature

Array of temperatures used to construct an efficiency lookup table. The array values must increase from left to right. The temperature array must be the same size as the efficiency array in temperature-dependent models. It must be the same size as a single row of the efficiency matrix in temperature-and-load-dependent models. The default array is `[280 300 320]` K.

Load at elliptical gear

Array of elliptical-gear loads used to construct a 2-D temperature-load-efficiency lookup table for temperature-and-load-dependent efficiency models. The array values must increase from left to right. The load array must be the same size as a single column of the efficiency matrix. The default array is `[1 5 10]` N*m.

Efficiency

Array of efficiencies used to construct a 1-D temperature-efficiency lookup table for temperature-dependent efficiency models. The array elements are the efficiencies at the temperatures in the **Temperature** array. The two arrays must be the same size.

This parameter appears only when you select a temperature-dependent friction model. The default array is `[0.95 0.9 0.85]`.

Efficiency matrix

Matrix of component efficiencies used to construct a 2-D temperature-load-efficiency lookup table. The matrix elements are the efficiencies at the temperatures in the **Temperature** array and at the loads in the **Load at elliptical gear** array.

The number of rows must be the same as the number of elements in the **Temperature** array. The number of columns must be the same as the number of elements in the **Load at elliptical gear** array.

This parameter appears only when you select a temperature-and-load-dependent efficiency model. The default matrix is `[ 0.85 0.8 0.75; 0.95 0.9 0.85; 0.85 0.8 0.7 ]`.

Follower power threshold

Visible only if you select `Temperature-dependent efficiency` as the friction model. Absolute value of the follower shaft power above which the full efficiency factor is in effect. A hyperbolic tangent function smooths the efficiency factor between zero when at rest and the value provided by the temperature-efficiency lookup table when at the power threshold. The default value is `0.001` W.

Follower angular velocity threshold

Visible only if you select `Temperature and load-dependent efficiency` as the friction model. Absolute value of the follower shaft angular velocity above which the full efficiency factor is in effect. Below this value, a hyperbolic tangent function smooths the efficiency factor to one, lowering the efficiency losses to zero when at rest. The default value is `0.01` rad/s.

Viscous Losses

Viscous friction coefficients at base (B) and follower (F)

Two-element array with the viscous friction coefficients in effect at the base and follower shafts. The default array, `[0 0] N*m/(rad/s)`, corresponds to zero viscous losses.

Thermal Port

Thermal mass

Thermal energy required to change the component temperature by a single degree. The greater the thermal mass, the more resistant the component is to temperature change. The default value is 50 J/K.

Initial temperature

Component temperature at the start of simulation. The initial temperature alters the component efficiency according to an efficiency vector that you specify, affecting the starting meshing or friction losses. The default value is 300 K.

Real-Time Simulation

Hardware-in-the-Loop Simulation

For optimal simulation performance, use the **Meshing Losses** > **Friction model** parameter default setting, `No meshing losses - Suitable for HIL simulation`.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also

Simscape Blocks
Cycloidal Drive | Simple Gear | Simple Gear with Variable Efficiency

Topics
"Model Thermal Losses in Driveline Components"
"Prepare Simscape Driveline Models for Real-Time Simulation Using Simscape Checks"

Leadscrew

Leadscrew gear set of threaded rotating screw and translating nut, with adjustable thread and friction losses

Library

Simscape / Driveline / Gears / Rotational-Translational

Description

The Leadscrew block represents a threaded rotational-translational gear that constrains the two connected driveline axes, screw (S) and nut(N), to, respectively, rotate and translate together in a fixed ratio that you specify. You can choose whether the nut axis translates in a positive or negative direction, as the screw rotates in a positive right-hand direction. If the screw helix is right-hand, ω_S and v_N have the same sign. If the screw helix is left-hand, ω_S and v_N have opposite signs. For model details, see "Leadscrew Gear Model" on page 1-259.

Thermal Modeling

You can model the effects of heat flow and temperature change through an optional thermal conserving port. By default, the thermal port is hidden. To expose the thermal port, right-click the block in your model and, from the context menu, select **Simscape > Block choices**. Select a variant that includes a thermal port. Specify the associated thermal parameters for the component.

Leadscrew Gear Model

Ideal Gear Constraint and Gear Ratio

Leadscrew imposes one kinematic constraint on the two connected axes:

$$\omega_S L = 2\pi v_N \quad .$$

The transmission ratio is $R_{NS} = 2\pi/L$. L is the screw lead, the translational displacement of the nut for one turn of the screw. In terms of this ratio, the kinematic constraint is:

$$\omega_S = R_{NS} v_N \quad .$$

The two degrees of freedom are reduced to one independent degree of freedom. The forward-transfer gear pair convention is (1,2) = (S,N).

The torque-force transfer is:

$$R_{NS}\tau_S + F_N - F_{loss} = 0 \quad ,$$

with $F_{loss} = 0$ in the ideal case.

Nonideal Gear Constraint and Losses

In the nonideal case, $F_{loss} \neq 0$. For general considerations on nonideal gear modeling, see "Model Gears with Losses".

Geometric Surface Contact Friction

In the contact friction case, η_{SN} and η_{NS} are determined by:

- The screw-nut threading geometry, specified by lead angle λ and acme thread half-angle α.
- The surface contact friction coefficient k.

$$\eta_{SN} = (\cos\alpha - k\cdot\tan\alpha)/(\cos\alpha + k/\tan\lambda) \quad ,$$

$$\eta_{NS} = (\cos\alpha - k/\tan\lambda)/(\cos\alpha + k\cdot\tan\alpha) \quad .$$

1-259

Constant Efficiencies

In the constant efficiency case, you specify η_{SN} and η_{NS}, independently of geometric details.

Self-Locking and Negative Efficiency

η_{NS} has two distinct regimes, depending on lead angle λ, separated by the *self-locking point* at which $\eta_{NS} = 0$ and $\cos\alpha = k/\tan\lambda$.

- In the *overhauling regime*, $\eta_{NS} > 0$. The force acting on the nut can rotate the screw.
- In the *self-locking regime*, $\eta_{NS} < 0$. An external torque must be applied to the screw to release an otherwise locked mechanism. The more negative is η_{NS}, the larger the torque must be to release the mechanism.

η_{SN} is conventionally positive.

Meshing Efficiency

The efficiencies η of meshing between screw and nut are fully active only if the transmitted power is greater than the power threshold.

If the power is less than the threshold, the actual efficiency is automatically regularized to unity at zero velocity.

Viscous Friction Force

The viscous friction coefficient μ controls the viscous friction torque experienced by the screw from lubricated, nonideal gear threads. The viscous friction torque on a screw driveline axis is $-\mu_S\omega_S$. ω_S is the angular velocity of the screw with respect to its mounting.

Limitations

- Gear inertia is assumed negligible.
- Gears are treated as rigid components.
- Coulomb friction slows down simulation. See "Adjust Model Fidelity".

Ports

Port	Description
S	Rotational conserving port representing the screw
N	Translational conserving port representing the nut
H	Thermal conserving port for thermal modeling

Parameters

Main

Screw lead

Translational displacement *L* of the nut per revolution of the screw. The default is `0.015`.

From the drop-down list, choose units. The default is meters (`m`).

Screw helix type

Choose the directional sense of screw rotation corresponding to positive nut translation. The default is `Right-hand`. The alternate option is `Left-hand`.

Meshing Losses

Parameters for vary with the block variant chosen—one with a thermal port for thermal modeling and one without it.

Without Thermal Port

Friction model

Select how to implement from nonideal meshing of gear threads. The default is `No meshing losses — Suitable for HIL simulation`.

- `No meshing losses — Suitable for HIL simulation` — Gear meshing is ideal.

1-261

- `Constant efficiency` — Transfer of torque and force between screw and nut is reduced by friction. If you select this option, the panel expands.

Constant Efficiency

Friction parametrization

Select how to characterize the friction between gear threads. The default is `Friction coefficient and geometrical parameters`.

- `Friction coefficient and geometrical parameters` — Friction is determined by contact friction between surfaces.

Lead angle

Thread helix angle λ = arctan[$L/(\pi d)$]. d is the mean thread diameter. Must be greater than zero. The default is `15`.

From the drop-down list, choose units. The default is degrees (`deg`).

Acme thread half angle

Half-angle of the acme thread α in the normal plane. If a square thread is simulated, $\alpha = 0$. Must be greater than zero. The default is `14.5`.

From the drop-down list, choose units. The default is degrees (`deg`).

Friction coefficient

Dimensionless coefficient of kinetic friction in the thread. Must be greater than zero. The default is `0.08`.

- `Efficiencies` — Friction is determined by constant efficiencies $0 < \eta < 1$. If you select this option, the panel changes from its default.

Efficiencies

Screw-nut efficiency

Efficiency η_{SN} of the power transfer from screw to nut.

Nut-screw efficiency

Efficiency η_{NS} of the power transfer from nut to screw.

Power threshold

Power threshold above which the full efficiency factor is in effect. Below this value, a hyperbolic tangent function smooths the efficiency factor, lowering the efficiency losses to zero when no power is transmitted. The default value is `0.001` W.

With Thermal Port

Temperature

Array of temperatures used to construct a 1-D temperature-efficiency lookup table. The array values must increase from left to right. The temperature array must be the same size as the **Efficiency** array. The default array is [280 300 320] K.

Screw-nut efficiency

Array of component efficiencies with the screw as the driver—that is, with power flowing from the screw to the nut. The array values are the efficiencies at the temperatures in the **Temperature** array. The two arrays must be the same size. The default array is [0.75 0.7 0.65].

Nut-screw efficiency

Array of component efficiencies with the nut as the driver—that is, with power flowing from the nut to the screw. The array values are the efficiencies at the temperatures in the **Temperature** array. The two arrays must be the same size. The default array is [0.5 0.45 0.4].

Power threshold

Power threshold above which the full efficiency factor is in effect. A hyperbolic tangent function smooths the efficiency factor between zero at rest and the value provided by the temperature-efficiency lookup table at the power threshold. The default value is 0.001 W.

Viscous Losses

Viscous friction coefficient

Viscous friction coefficient μ_S for the screw. The default is 0.

From the drop-down list, choose units. The default is newton-meters/(radians/second) (N*m/(rad/s)).

Thermal Port

Thermal mass

Thermal energy required to change the component temperature by a single degree. The greater the thermal mass, the more resistant the component is to temperature change. The default value is 50 J/K.

Initial temperature

Component temperature at the start of simulation. The initial temperature alters the component efficiency according to an efficiency vector that you specify, affecting the starting meshing or friction losses. The default value is 300 K.

Real-Time Simulation

Hardware-in-the-Loop Simulation

For optimal simulation performance, use the **Meshing Losses** > **Friction model** parameter default setting, `No meshing losses - Suitable for HIL simulation`.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also

Simscape Blocks
Rack & Pinion | Worm Gear

Topics
"Basic Motion, Torque, and Force Modeling"
"Ratchet Leadscrew Mechanism"
"Model Thermal Losses in Driveline Components"
"Prepare Simscape Driveline Models for Real-Time Simulation Using Simscape Checks"

Limited-Slip Differential

Reduce velocity difference between two connected shafts
**Library:** Simscape / Driveline / Gears

Description

The Limited-Slip Differential block represents a limited-slip differential (LSD), which is a gear assembly that can reduce the velocity difference between two connected shafts. The block models the LSD mechanism as a structural component that combines a differential gear and a clutch.

The differential component in the LSD block is an open differential. An open differential is a gear mechanism that allows two driven shafts to spin at different speeds. In an automobile, a differential allows the inner wheels to spin more slowly than the outer wheels when the vehicle is cornering. A vehicle that has wheel shafts that are connected by an open differential can get stuck when only one of the wheels slips and then spins freely due to traction loss. This vehicle stops moving because the driveshaft supplies less power to the wheel with traction than it supplies to the spinning wheel.

In the same scenario, a vehicle that has an LSD is less likely to get stuck because it contains a clutch assembly that can transmit power to the wheel that retains traction. The clutch component in the LSD block is a friction clutch that has two sets of flat friction plates. The clutch engages when applied pressure exceeds the engagement threshold pressure. In an LSD, a spring preload that separates the sun gears presses the plates in both sets together. When the wheels experience a traction differential, the planet pinion gears exert an additional force in the direction of the high-traction wheel. If the additional pressure exceeds the engagement threshold, the clutch assembly engages. The engagement allows the driveshaft to transmit more power to the slower-spinning high-traction wheel. The additional power reduces the difference in velocity of the two shafts. Because the high-traction wheel continues to rotate, the vehicle continues to move.

The figure shows the orientation of the major components in an LSD mechanism.
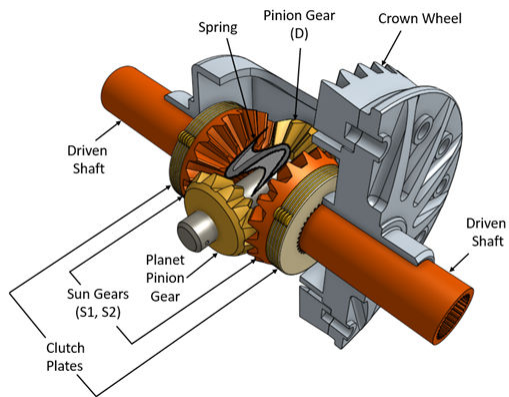
The Limited-Slip Differential block models the LSD mechanism as a structural component based on the Simscape Driveline Differential and Disk Friction Clutch blocks. The differential mechanism modeled by the Differential block is a structural component based on two other Simscape Driveline blocks, the Simple Gear and the Sun-Planet Bevel. The block diagram shows the structural components of the LSD.
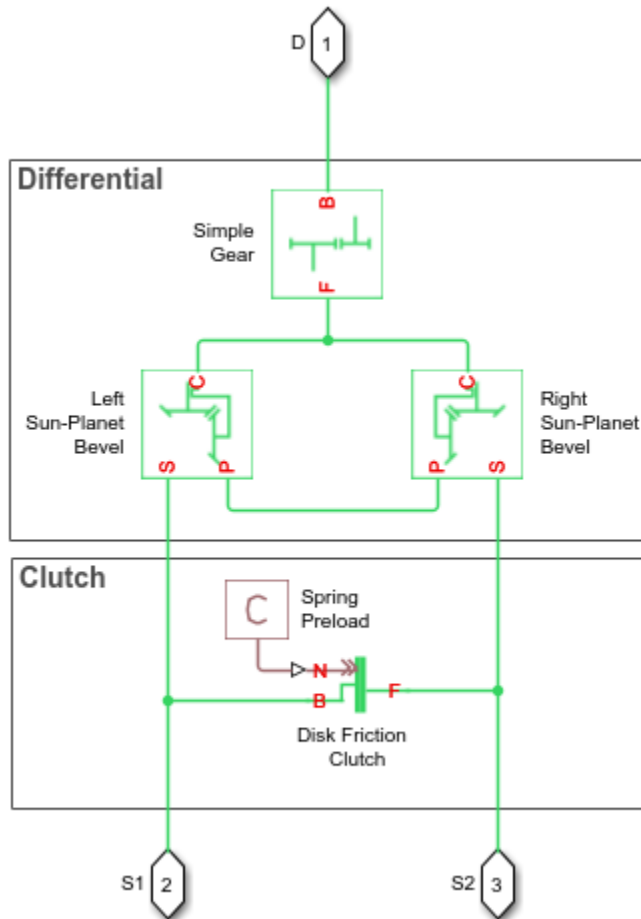
The ports of the Limited-Slip Differential block are associated with the driveshaft (port D) and the two driven shafts (ports S1 and S2), which connect the sun-gears to the wheels.

The block enables you to specify inertias only for the gear carrier and internal planet gears. By default, the inertias of the outer gears are assumed negligible. To model the inertias of the outer gears, connect Simscape Inertia blocks to the D, S1, and S2 ports.

The table shows the rotation direction of the driven shaft ports for different block parameterizations and input conditions.

Rotation Direction of the driven shaft ports (S1 and S2)	Crown Gear Location Relative to the Centerline	Rotation Direction of Driveshaft Port (D)	Relative Slippage Across the Differential
Positive	Right	Positive	0
• Positive for the nonslipping port • Negative the slipping port	Right	Positive	> 0
Negative	Right	Negative	0
• Negative for the nonslipping port • Positive the slipping port	Right	Negative	> 0
Negative	Left	Positive	0
• Negative for the nonslipping port • Positive the slipping port	Left	Positive	> 0
Positive	Left	Negative	0
• Positive for the nonslipping port • Negative the slipping port	Left	Negative	> 0

Model

To examine the mathematical models for the structural components of the Limited-Slip Differential block, see:

• Disk Friction Clutch

• Differential

• Simple Gear

• Sun-Planet Bevel

Port Options

The Limited-Slip Differential block offers these port options:

- `No thermal port` (default)
- `Show thermal port`

To model thermal losses, select `Show thermal port`. Thermal ports are conserving ports that you can connect to thermal conserving ports on Simscape Foundation Thermal Library blocks for simulating heat flow and temperature changes. For more information, see "Model Thermal Losses in Driveline Components".

To view or select a port option, use one of these methods:

- Use the block context menu:

    1 Left-click the block.

    2 From the context menu, select **Simscape** > **Block choices**.

    3 Select option from the drop-down menu.

- Use the Property Inspector:

    1 From the Simulink® Editor menu, select **View > Property Inspector**.

    2 Click the block.

    3 In the Property Inspector pane, click the **Block Choice VALUE**.

    4 Select an option from the drop-down menu.

Dependencies

These parameters are enabled only if you select `No thermal port`:

- **Friction Model**
- **Static friction coefficient**

These parameters are enabled only if you select `Show thermal port`:

- **Temperature Sun-sun efficiency**
- **Sun-sun efficiency**
- **Carrier-driveshaft efficiency**
- **Sun-carrier and driveshaft-casing power thresholds**
- **Sun-carrier and driveshaft-casing viscous friction coefficients**
- **Kinetic friction coefficient matrix**

1-269

- **Static friction coefficient vector**
- **Thermal mass**
- **Initial temperature**

Ports

Conserving

D — Driving shaft interface
mechanical rotational

Port associated with the driveshaft.

S2, S2 — Driven shaft interfaces
mechanical rotational

Ports associated with the two sun-gear shafts.

H — Heat exchange
thermal

Port associated with heat flow. The thermal port is optional and is hidden by default.

Dependencies

To enable this port, for **Block Choice**, select Show thermal port.

Parameters

Differential

Crown gear location — Driveshaft location
To the right of centerline (default) | To the left of centerline

Location of the bevel crown gear regarding the centerline of the gear assembly.

Carrier (C) to driveshaft (D) teeth ratio (NC/ND) — Gear ratio
4 (default)

Fixed ratio, g_D, of the carrier gear to the longitudinal driveshaft gear.

Friction model — Frictional loss
No meshing losses — Suitable for HIL simulation (default) | Constant
efficiency

Select whether to consider frictional losses from nonideal meshing of gear teeth:

- No meshing losses — Suitable for HIL simulation — Gear meshing is ideal
  and losses are assumed negligible.
- Constant efficiency — Transfer of torque between gear wheel pairs is reduced by
  a constant efficiency η satisfying $0 < \eta \le 1$.

You can increase the fidelity of your model by specifying meshing losses. For hardware-in-the-loop simulation, however, if your model is not real-time capable, consider ignoring the frictional losses. For more information on balancing speed and fidelity for real-time simulation using Simscape models, see "Improving Speed and Accuracy" (Simscape).

Dependencies

To enable this parameter, for **Block Choice**, select No thermal port.

Selecting Constant efficiency enables these parameters:

- **Sun-sun and carrier-driveshaft ordinary efficiencies**
- **Sun-carrier and driveshaft-casing power thresholds**
- **Sun-carrier and driveshaft-casing viscous friction coefficients**

**Sun-sun and carrier-driveshaft ordinary efficiencies — Torque transfer
efficiency**
[0.85, 0.92] (default)

Vector of torque transfer efficiencies [η_{SS} η_D] for sun-sun and carrier-longitudinal driveshaft gear wheel pair meshings, respectively.

Dependencies

To enable this parameter:

1 For **Block choice**, select No thermal port.
2 For **Friction model**, select Constant efficiency.

1-271

Temperature — Component temperature
[280.0, 300.0, 320.0] W (default)

Array of temperatures used to construct 1-D temperature-efficiency lookup tables. The array values must increase left to right. The temperature array must be the same size as each efficiency array.

Dependencies

To enable this parameter, for **Block choice**, select Show thermal port.

Sun-sun efficiency — Mechanical efficiency
[0.95, 0.9, 0.85] (default)

Array of mechanical efficiencies with power flowing from one of the sun gears to the other. Each array value is the ratio of output power to input power at one of the temperatures in the temperature array. The temperature and efficiency arrays must be the same size.

Dependencies

To enable this parameter, for **Block choice**, select Show thermal port.

Carrier-driveshaft efficiency — Mechanical efficiency
[0.95, 0.9, 0.85] (default)

Array of mechanical efficiencies with power flowing from the gear carrier to the longitudinal driveshaft. Each array value is the ratio of output power to input power at one of the temperatures in the temperature array. The temperature and efficiency arrays must be the same size.

Dependencies

To enable this parameter, for **Block choice**, select Show thermal port.

Sun-carrier and driveshaft-casing power thresholds — Efficiency boundary
[0.001, 0.001] W (default)

Vector of power thresholds p_{th} above which full efficiency loss is applied, for sun-carrier and longitudinal driveshaft-casing [p_S p_D], respectively. A hyperbolic tangent function smooths the efficiency factors between zero when at rest and the values provided by the temperature-efficiency lookup tables at the power thresholds.

Dependencies

To enable this parameter:

1 For **Block choice**, select No thermal port.

2 For **Friction model**, select Constant efficiency.

or for **Block choice**, select Show thermal port.

**Sun-carrier and driveshaft-casing viscous friction coefficients —
Component viscosity**
[0.0, 0.0] N*m/(rad/s) (default)

Vector of viscous friction coefficients [μ_S μ_D] for the sun-carrier and longitudinal driveshaft-casing gear motions, respectively.

Dependencies

To enable this parameter:

1 For **Block choice**, select No thermal port.

2 For **Friction model**, select Constant efficiency.

or for **Block choice**, select Show thermal port.

Carrier inertia — Moment of inertia
0 kg*m^2 (default)

Moment of inertia of the planet gear carrier. This value must be positive or zero. To ignore carrier ring inertia, enter 0.

Planet inertia — Moment of inertia
0 kg*m^2 (default)

Moment of inertia of the combined planet gears. This value must be positive or zero. To ignore planet gear inertia, enter 0.

Clutch

Number of friction surfaces — Number of contact surfaces
4 (default)

Number, N, of friction-generating contact surfaces inside the clutch.

Effective torque radius — Friction force application radius
130 mm (default)

Effective moment arm radius, r_{eff}, that determines the kinetic friction torque inside the clutch.

Preload force — Clutch plate engagement pressure
300 N (default)

Preload force that the spring exerts on the clutch plate assemblies. Must be greater than or equal to zero.

Kinetic friction coefficient relative velocity vector — Absolute surface velocity
[0.0, 100.0, 1000.0] rad/s (default)

Specify input values for the relative velocity as a vector. The values in the vector must increase from left to right. The minimum number of values depends on the interpolation method that you select. For linear interpolation, provide at least two values per dimension. For smooth interpolation, provide at least three values per dimension.

Kinetic friction coefficient vector — Coulomb coefficient of friction, k_K
[0.3, 0.22, 0.19] (default) | . For a block choice that shows the thermal port, the default value is [0.3, 0.28, 0.25]

Specify the output values for the kinetic friction coefficient as a vector. All values must be greater than zero.

Kinetic friction coefficient matrix — Coulomb coefficient of friction, k_K
[0.34, 0.32, 0.3; 0.3, 0.28, 0.25; 0.25, 0.2, 0.15] (default)

The coefficient of kinetic friction must be greater than zero.

Dependencies

To enable this parameter, for **Block choice**, select Show thermal port.

Static friction coefficient vector — Coulomb coefficient of friction, k_S
[0.4, 0.38, 0.36] (default)

Specify the static, or peak, values of the friction coefficient as a vector. The vector must have the same number of elements as the temperature vector. Each value must be greater than the value of the corresponding element in the kinetic friction coefficient vector.

Dependencies

To enable this parameter, for **Block choice**, select `Show thermal port`.

`Friction coefficient interpolation method` — Output value approximation method
`Linear` (default) | `Smooth`

Interpolation methods for approximating the output value when the input value is between two consecutive grid points. To optimize performance, select `Linear`. To produce a continuous curve with continuous first-order derivatives, select `Smooth`.

For more information on interpolation algorithms, see the PS Lookup Table (1D) block reference page.

`Friction coefficient extrapolation method` — Output value approximation method
`Linear` (default) | `Nearest`

Extrapolation methods for approximating the output value when the input value is outside the range specified in the argument list. To produce a curve with continuous first-order derivatives in the extrapolation region and at the boundary with the interpolation region, select `Linear`. To produce an extrapolation that does not go above the highest point in the data or below the lowest point in the data, select `Nearest`.

For more information on extrapolation algorithms, see the PS Lookup Table (1D) block reference page.

`Static friction coefficient` — Coulomb coefficient of friction, k_S
`0.35` (default)

Dimensionless Coulomb static friction coefficient, k_S, applied to the normal force across the clutch when the clutch is locked. The static friction coefficient, k_S, must be larger than the kinetic friction coefficient, k_K.

Dependencies

To enable this parameter, for **Block Choice**, select `No thermal port`.

`Velocity tolerance` — Slip velocity locking boundary
`0.001 rad/s` (default)

Maximum slip velocity at which the clutch can lock. The slip velocity is the signed difference between the base and follower shaft angular velocities, that is, $w = w_F - w_B$.

When the kinetic friction torque is nonzero and the transferred torque is within the static friction torque limits, then the clutch locks if the actual slip velocity falls below the velocity tolerance.

Initial state — Initial clutch state
Unlocked (default) | Locked

Clutch state at the start of simulation. The clutch can be in one of two states, locked and unlocked. A locked clutch constrains the base and follower shafts to spin at the same velocity, that is, as a single unit. An unlocked clutch allows the two shafts to spin at different velocities, resulting in slip between the clutch plates.

Thermal Port

Thermal mass — Component resistance to temperature change
50 J/K (default)

Thermal energy required to change the component temperature by a single degree. The greater the thermal mass, the more resistant the component is to temperature change.

Dependencies

To enable this parameter, for **Block choice**, select Show thermal port.

Initial temperature — Component initial temperature
300 K (default)

Component temperature at the start of simulation. The initial temperature alters the component efficiency according to an efficiency vector that you specify, affecting the starting meshing or friction losses.

Dependencies

To enable this parameter, for **Block choice**, select Show thermal port.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also

Simscape Blocks
Differential | Disk Friction Clutch | Simple Gear | Sun-Planet Bevel

Topics
"Model Thermal Losses in Driveline Components"
"Basic Motion, Torque, and Force Modeling"

Introduced in R2017a

Loaded-Contact Rotational Friction

Loaded-contact friction between two rotating surfaces

Library

Simscape / Driveline / Brakes & Detents / Rotational

Description

The Loaded-Contact Rotational Friction block simulates friction between two rotating surfaces loaded with a normal force.

The block is implemented as a structural component based on the Fundamental Friction Clutch block. From the locked state, the two surfaces unlock if the friction force exceeds the static friction, as defined by the static coefficient of friction and current normal force. For details on how the locking and unlocking are modeled, see the Fundamental Friction Clutch block reference page.

Friction torque is transmitted for normal forces larger than the **Threshold force** parameter.

Loaded-Contact Rotational Friction Model

The block simulates friction between two rotating surfaces loaded with a normal force. When the two rotating surfaces are not locked, the friction torque is determined with the following equations:

$$\tau_{fr} = N \cdot \mu \cdot r_{eff} \cdot \text{sign}(\omega) + \tau_{visc} ,$$

$$r_{eff} = \frac{2}{3} \cdot \frac{r_o^3 - r_i^3}{r_o^2 - r_i^2} ,$$

$$\tau_{\text{visc}} = \mu_{\text{visc}} \cdot \omega \quad,$$

where:

τ_{fr}	Friction torque
N	Normal force
μ	Friction coefficient
r_{eff}	Effective radius
r_{o}	Surface outside radius
r_{i}	Surface inside radius
ω	Relative angular velocity
τ_{visc}	Viscous drag torque
μ_{visc}	Viscous drag torque coefficient

Thermal Modeling

You can model the effects of heat flow and temperature change through an optional thermal conserving port. By default, the thermal port is hidden. To expose the thermal port, right-click the block in your model and, from the context menu, select **Simscape > Block choices**. Select a variant that includes a thermal port. Specify the associated thermal parameters for the component.

Limitations

- The model does not account for inertia. Add inertia terms externally to the **B** and **F** ports as required.
- The model computes the torque assuming a uniform distribution of the normal force.

Ports

B

  Rotational conserving ports associated with the driving surface

F

Rotational conserving ports associated with the driven surface

N

Physical signal terminal through which you import the normal force

H

Thermal conserving port. The thermal port is optional and is hidden by default. To expose the port, select a variant that includes a thermal port.

Parameters

Geometry

Force action region

Select a parameterization method to model the loaded-contact friction. The default method is `Define effective radius`.

- `Define effective radius` — Provide a value for the friction effective radius.

    Effective torque radius

    Effective radius r_{eff}. Must be greater than zero. The default value is `130 mm`.

- `Define annular region` — Define the friction effective radius in terms of the inside and outside diameters of the friction surface. If you select this option, the panel changes from its default.

    `Define annular region`

    Friction surface outside diameter

    Outside diameter r_o of the friction surfaces. Must be greater than zero. The default value is `150 mm`.

    Friction surface inside diameter

    Inside diameter r_i of the friction surfaces. Must be greater than zero, but less than the friction surface outside diameter. The default value is `100 mm`.

Friction

Friction model

Select a parameterization method to model the kinetic friction coefficient. The options and default values for this parameter depend on the variant that you select for the block. The options are:

- `Fixed kinetic friction coefficient` — Provide a fixed value for the kinetic friction coefficient. This option:

  - Is only visible if you use the default variant of the block
  - Is the default method for parameterizing the default variant of the block
  - Affects the visibility of other parameters

Fixed kinetic friction coefficient

Kinetic friction coefficient

The kinetic, or Coulomb, friction coefficient. The coefficient must be greater than zero. The default value is `0.3`.

Static friction coefficient

The static, or peak, value of the friction coefficient. The static friction coefficient must be greater than the kinetic friction coefficient. The default value is `0.35`.

- `Table lookup kinetic friction coefficient` — Define the kinetic friction coefficient by one-dimensional table lookup based on the relative velocity between the sliding surfaces. This option:

  - Is only visible if you use the default variant of the block
  - Affects the visibility of other parameters

Table lookup kinetic friction coefficient

Kinetic friction coefficient relative velocity vector

Specify input values for the relative velocity as a vector. The values in the vector must increase from left to right. The minimum number of values depends on the interpolation method that you select. For linear interpolation, provide at least two values per dimension. For smooth interpolation, provide at least three values per dimension. The default value is `[0.0, 100.0, 1000.0] rad/s`.

Kinetic friction coefficient vector

Specify the output values for the kinetic friction coefficient as a vector. All values must be positive. The default value is `[0.3, 0.22, 0.19]`.

Friction coefficient interpolation method

Select one of these interpolation methods to approximate the output value when the input value is between two consecutive grid points:

- `Linear` — Select this, default, option to get the best performance.
- `Smooth` — Select this option to produce a continuous curve with continuous first-order derivatives.

For more information on interpolation algorithms, see the PS Lookup Table (1D) block reference page.

Friction coefficient extrapolation method

Select one of these extrapolation methods to determine the output value when the input value is outside the range specified in the argument list:

- `Linear` — Select this, default, option to produce a curve with continuous first-order derivatives in the extrapolation region and at the boundary with the interpolation region.
- `Nearest` — Select this option to produce an extrapolation that does not go above the highest point in the data or below the lowest point in the data.

For more information on extrapolation algorithms, see the PS Lookup Table (1D) block reference page.

Static friction coefficient

The static, or peak, value of the friction coefficient. The static friction coefficient must be greater than the kinetic friction coefficient. The default value is `0.35`.

- `Temperature-dependent kinetic friction coefficient` — Define the kinetic friction coefficient by table lookup based on the temperature. This option:

  - Is only visible if you use a thermal variant of the block
  - Is the default method for parameterizing the thermal variant of the block
  - Affects the visibility of other parameters

Temperature-dependent kinetic friction coefficient

Temperature vector

Specify the input values for the temperature as a vector. The minimum number of values depends on the interpolation method that you select. For linear interpolation, provide at least two values per dimension. For smooth interpolation, provide at least three values per dimension. The values in the vector must increase from left to right. The default value is `[280.0, 300.0, 320.0]` K.

Kinetic friction coefficient vector

Specify the output values for kinetic friction coefficient as a vector. The vector must have the same number of elements as the temperature vector. All values must be greater than zero. The default value is `[0.3, 0.28, 0.25]`.

Static friction coefficient vector

Specify the static, or peak, values of the friction coefficient as a vector. The vector must have the same number of elements as the temperature vector. Each value must be greater than the value of the corresponding element in the kinetic friction coefficient vector. The default value is `[0.4, 0.38, 0.36]`.

Friction coefficient interpolation method

Select one of these interpolation methods to approximate the output value when the input value is between two consecutive grid points:

- `Linear` — Select this, default, option to get the best performance.
- `Smooth` — Select this option to produce a continuous curve with continuous first-order derivatives.

For more information on interpolation algorithms, see the PS Lookup Table (1D) block reference page.

Friction coefficient extrapolation method

Select one of these extrapolation methods to determine the output value when the input value is outside the range specified in the argument list:

- `Linear` — Select this, default, option to produce a curve with continuous first-order derivatives in the extrapolation region and at the boundary with the interpolation region.
- `Nearest` — Select this option to produce an extrapolation that does not go above the highest point in the data or below the lowest point in the data.

For more information on extrapolation algorithms, see the PS Lookup Table (1D) block reference page.

- `Temperature and speed-dependent kinetic friction coefficient` — Define the kinetic friction coefficient by table lookup based on the temperature and the relative velocity between the sliding surfaces. This option:

  - Is only visible if you use the thermal variant of the block
  - Affects the visibility of other parameters

`Temperature and speed-dependent kinetic friction coefficient`

Temperature vector

Specify the input values for the temperature as a vector. The minimum number of values depends on the interpolation method that you select. For linear interpolation, provide at least two values per dimension. For smooth interpolation, provide at least three values per dimension. The values in the vector must increase from left to right. The default value is `[280.0, 300.0, 320.0]` K.

Kinetic friction coefficient relative velocity vector

Specify input values for the relative velocity as a vector. The values in the vector must increase from left to right. The minimum number of values depends on the interpolation method that you select. For linear interpolation, provide at least two values per dimension. For smooth interpolation, provide at least three values per dimension. The default value is `[0.0, 100.0, 1000.0]` rad/s.

Kinetic friction coefficient matrix

Specify the output values for kinetic friction coefficient as a matrix. All the values must be greater than zero. The size of the matrix must equal the size of the matrix that is the result of the temperature vector × the kinetic friction coefficient relative velocity vector. The default value is `[0.34, 0.32, 0.3; 0.3, 0.28, 0.25; 0.25, 0.2, 0.15]`.

Static friction coefficient vector

Specify the static, or peak, values of the friction coefficient as a vector. The vector must have the same number of elements as the temperature vector. Each value must be greater than the value of the corresponding element in the kinetic friction coefficient vector. The default value is `[0.4, 0.38, 0.36]`.

Friction coefficient interpolation method

Select one of these interpolation methods to approximate the output value when the input value is between two consecutive grid points:

- `Linear` — Select this, default, option to get the best performance.
- `Smooth` — Select this option to produce a continuous curve with continuous first-order derivatives.

For more information on interpolation algorithms, see the PS Lookup Table (1D) block reference page.

Friction coefficient extrapolation method

Select one of these extrapolation methods to determine the output value when the input value is outside the range specified in the argument list:

- `Linear` — Select this, default, option to produce a curve with continuous first-order derivatives in the extrapolation region and at the boundary with the interpolation region.
- `Nearest` — Select this option to produce an extrapolation that does not go above the highest point in the data or below the lowest point in the data.

For more information on extrapolation algorithms, see the PS Lookup Table (1D) block reference page.

Velocity tolerance

Relative velocity below which the two surfaces can lock. The surfaces lock if the force across the **B** and **F** translational ports is less than the threshold force. The default value is `0.001 m/s`.

Threshold force

The normal force applied to the physical signal port **N** is applied to the contact only if the amount of force exceeds the value of the **Threshold force** parameter. Forces below the **Threshold force** are not applied, and there is consequently no transmitted frictional force. The default value is `1 N`.

1-285

Viscous Losses

Viscous drag torque coefficient

Viscous drag coefficient, μ_{visc}, for computing the drag torque. The coefficient depends on the type of operating fluid, fluid temperature, and the maximum distance between the surfaces. The default value is `0 N*m/(rad/s)`.

Initial Conditions

Initial state

Select the initial condition applied to the Fundamental Friction Clutch state inside the block:

- `Locked` — Rotational ports **B** and **F** are initially locked together. This option is the default.
- `Unlocked` — Rotational ports **B** and **F** are initially sliding relative to each other.

Thermal Port

These thermal parameters are visible only when you select a block variant that includes a thermal port.

Thermal mass

Thermal energy required to change the component temperature by a single degree. The greater the thermal mass, the more resistant the component is to temperature change. The default value is `50 kJ/K`.

Initial temperature

Component temperature at the start of simulation. The initial temperature alters the component efficiency according to an efficiency vector that you specify, affecting the starting meshing or friction losses. The default value is `300 K`.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also

Fundamental Friction Clutch | Loaded-Contact Translational Friction

Topics

"Gear Coupling Control Using Clutches"

Introduced in R2011a

Loaded-Contact Translational Friction

Loaded-contact friction between two sliding surfaces

Library

Simscape / Driveline / Brakes & Detents / Translational

Description

The Loaded-Contact Translational Friction block simulates friction between two sliding surfaces loaded with a normal force.

The block is implemented as a structural component based on the Fundamental Friction Clutch block. From the locked state, the two surfaces unlock if the friction force exceeds the static friction, as defined by the static coefficient of friction and current normal force. For details on how the locking and unlocking are modeled, see the Fundamental Friction Clutch block reference page.

Friction force is transmitted for normal forces larger than the **Threshold force** parameter.

Loaded-Contact Translational Friction Model

The block simulates friction between two sliding surfaces loaded with a normal force. When the two sliding surfaces are not locked, the friction force is determined with the following equations:

$$F_{\text{fr}} = N \cdot \mu \cdot \text{sign}(v_{\text{rel}}) + F_{\text{visc}} ,$$

$$F_{\text{visc}} = \mu_{\text{visc}} \cdot v_{rel} ,$$

where:

F_{fr}	Friction force
N	Normal force
μ	Friction coefficient
v_{rel}	Relative velocity
F_{visc}	Viscous drag force
μ_{visc}	Viscous drag force coefficient

Thermal Modeling

You can model the effects of heat flow and temperature change through an optional thermal conserving port. By default, the thermal port is hidden. To expose the thermal port, right-click the block in your model and, from the context menu, select **Simscape** > **Block choices**. Select a variant that includes a thermal port. Specify the associated thermal parameters for the component.

Limitations

- The model does not account for body mass. Add mass terms externally to the **B** and **F** ports as required.
- The model computes the friction force assuming a uniform distribution of the normal force.

Ports

B

   Translational conserving ports associated with the driving surface

F

   Translational conserving ports associated with the driven surface

N

   Physical signal terminal through which you import the normal force

H

Thermal conserving port. The thermal port is optional and is hidden by default. To expose the port, select a variant that includes a thermal port.

Parameters

Friction

Friction model

Select a parameterization method to model the kinetic friction coefficient. The options and default values for this parameter depend on the variant that you select for the block. The options are:

- `Fixed kinetic friction coefficient` — Provide a fixed value for the kinetic friction coefficient. This option:

  - Is only visible if you use the default variant of the block
  - Is the default method for parameterizing the default variant of the block
  - Affects the visibility of other parameters

  `Fixed kinetic friction coefficient`

  Kinetic friction coefficient

  The kinetic, or Coulomb, friction coefficient. The coefficient must be greater than zero. The default value is `0.3`.

  Static friction coefficient

  The static, or peak, value of the friction coefficient. The static friction coefficient must be greater than the kinetic friction coefficient. The default value is `0.35`.

- `Table lookup kinetic friction coefficient` — Define the kinetic friction coefficient by one-dimensional table lookup based on the relative velocity between the sliding surfaces. This option:

  - Is only visible if you use the default variant of the block
  - Affects the visibility of other parameters

`Table lookup kinetic friction coefficient`

Kinetic friction coefficient relative velocity vector

Specify input values for the relative velocity as a vector. The values in the vector must increase from left to right. The minimum number of values depends on the interpolation method that you select. For linear interpolation, provide at least two values per dimension. For smooth interpolation, provide at least three values per dimension. `[0.0, 0.1, 1.0]` m/s.

Kinetic friction coefficient vector

Specify the output values for the kinetic friction coefficient as a vector. All values must be positive. The default value is `[0.3, 0.22, 0.19]`.

Friction coefficient interpolation method

Select one of these interpolation methods to approximate the output value when the input value is between two consecutive grid points:

- `Linear` — Select this, default, option to get the best performance.
- `Smooth` — Select this option to produce a continuous curve with continuous first-order derivatives.

For more information on interpolation algorithms, see the PS Lookup Table (1D) block reference page.

Friction coefficient extrapolation method

Select one of these extrapolation methods to determine the output value when the input value is outside the range specified in the argument list:

- `Linear` — Select this, default, option to produce a curve with continuous first-order derivatives in the extrapolation region and at the boundary with the interpolation region.
- `Nearest` — Select this option to produce an extrapolation that does not go above the highest point in the data or below the lowest point in the data.

For more information on extrapolation algorithms, see the PS Lookup Table (1D) block reference page.

Static friction coefficient

The static, or peak, value of the friction coefficient. The static friction coefficient must be greater than the kinetic friction coefficient. The default value is `0.35`.

- `Temperature-dependent kinetic friction coefficient` — Define the kinetic friction coefficient by table lookup based on the temperature. This option:

  - Is only visible if you use a thermal variant of the block
  - Is the default method for parameterizing the thermal variant of the block
  - Affects the visibility of other parameters

`Temperature-dependent kinetic friction coefficient`

Temperature vector

   Specify the input values for the temperature as a vector. The minimum number of values depends on the interpolation method that you select. For linear interpolation, provide at least two values per dimension. For smooth interpolation, provide at least three values per dimension. The values in the vector must increase from left to right. The default value is `[280.0, 300.0, 320.0]` K.

Kinetic friction coefficient vector

   Specify the output values for kinetic friction coefficient as a vector. The vector must have the same number of elements as the temperature vector. All values must be greater than zero. The default value is `[0.3, 0.28, 0.25]`.

Static friction coefficient vector

   Specify the static, or peak, values of the friction coefficient as a vector. The vector must have the same number of elements as the temperature vector. Each value must be greater than the value of the corresponding element in the kinetic friction coefficient vector. The default value is `[0.4, 0.38, 0.36]`.

Friction coefficient interpolation method

   Select one of these interpolation methods to approximate the output value when the input value is between two consecutive grid points:

   - `Linear` — Select this, default, option to get the best performance.
   - `Smooth` — Select this option to produce a continuous curve with continuous first-order derivatives.

   For more information on interpolation algorithms, see the PS Lookup Table (1D) block reference page.

Friction coefficient extrapolation method

   Select one of these extrapolation methods to determine the output value when the input value is outside the range specified in the argument list:

- `Linear` — Select this, default, option to produce a curve with continuous first-order derivatives in the extrapolation region and at the boundary with the interpolation region.
- `Nearest` — Select this option to produce an extrapolation that does not go above the highest point in the data or below the lowest point in the data.

For more information on extrapolation algorithms, see the PS Lookup Table (1D) block reference page.

- `Temperature and speed-dependent kinetic friction coefficient` — Define the kinetic friction coefficient by table lookup based on the temperature and the relative velocity between the sliding surfaces. This option:

  - Is only visible if you use the thermal variant of the block
  - Affects the visibility of other parameters

Temperature and speed-dependent kinetic friction coefficient

Temperature vector

Specify the input values for the temperature as a vector. The minimum number of values depends on the interpolation method that you select. For linear interpolation, provide at least two values per dimension. For smooth interpolation, provide at least three values per dimension. The values in the vector must increase from left to right. The default value is `[280.0, 300.0, 320.0]` K.

Kinetic friction coefficient relative velocity vector

Specify input values for the relative velocity as a vector. The values in the vector must increase from left to right. The minimum number of values depends on the interpolation method that you select. For linear interpolation, provide at least two values per dimension. For smooth interpolation, provide at least three values per dimension. `[0.0, 0.1, 1.0]` m/s.

Kinetic friction coefficient matrix

Specify the output values for kinetic friction coefficient as a matrix. All the values must be greater than zero. The size of the matrix must equal the size of the matrix that is the result of the temperature vector × the kinetic friction coefficient relative velocity vector. The default value is `[0.34, 0.32, 0.3; 0.3, 0.28, 0.25; 0.25, 0.2, 0.15]`.

Static friction coefficient vector

Specify the static, or peak, values of the friction coefficient as a vector. The vector must have the same number of elements as the temperature vector.

1-293

Each value must be greater than the value of the corresponding element in the kinetic friction coefficient vector. The default value is `[0.4, 0.38, 0.36]`.

Friction coefficient interpolation method

Select one of these interpolation methods to approximate the output value when the input value is between two consecutive grid points:

- `Linear` — Select this, default, option to get the best performance.
- `Smooth` — Select this option to produce a continuous curve with continuous first-order derivatives.

For more information on interpolation algorithms, see the PS Lookup Table (1D) block reference page.

Friction coefficient extrapolation method

Select one of these extrapolation methods to determine the output value when the input value is outside the range specified in the argument list:

- `Linear` — Select this, default, option to produce a curve with continuous first-order derivatives in the extrapolation region and at the boundary with the interpolation region.
- `Nearest` — Select this option to produce an extrapolation that does not go above the highest point in the data or below the lowest point in the data.

For more information on extrapolation algorithms, see the PS Lookup Table (1D) block reference page.

Velocity tolerance

Relative velocity below which the two surfaces can lock. The surfaces lock if the force across the **B** and **F** translational ports is less than the threshold force. The default value is `0.001 m/s`.

Threshold force

The normal force applied to the physical signal port **N** is applied to the contact only if the amount of force exceeds the value of the **Threshold force** parameter. Forces below the **Threshold force** are not applied, and there is consequently no transmitted frictional force. The default value is `1 N`.

Viscous Losses

Viscous drag force coefficient

Viscous drag coefficient, μ_{visc}, for computing the drag force. The coefficient depends on the type of operating fluid, fluid temperature, and the maximum distance between the sliding surfaces. The default value is 0 N/(m/s).

Initial Conditions

Initial state

Select the initial condition applied to the Fundamental Friction Clutch state inside the block:

- Locked — Translational ports **B** and **F** are initially locked together. This is the default option.
- Unlocked — Translational ports **B** and **F** are initially sliding relative to each other.

Thermal Port

These thermal parameters are visible only when you select a block variant that includes a thermal port.

Thermal mass

Thermal energy required to change the component temperature by a single degree. The greater the thermal mass, the more resistant the component is to temperature change. The default value is 50 kJ/K.

Initial temperature

Component temperature at the start of simulation. The initial temperature alters the component efficiency according to an efficiency vector that you specify, affecting the starting meshing or friction losses. The default value is 300 K.

See Also

Fundamental Friction Clutch | Loaded-Contact Rotational Friction

Topics

"Gear Coupling Control Using Clutches"

Introduced in R2011a

Logic-Controlled Clutch

Friction clutch with binary input control
Library: Simscape / Driveline / Clutches

Description

The Logic-Controlled Clutch is based on the Fundamental Friction Clutch block, which transmits rotational power through friction. The Logic-Controlled Clutch block is controlled by a binary input. When the input is 1, the clutch engages, provides friction, and can lock. When the input is 0, the clutch does not engage or provide friction and it cannot lock.

If the clutch is engaging and in motion, the block applies the kinetic torque. If the clutch is engaging and not in motion, it applies the static torque.

Ports

Input

E — Control clutch engagement
physical signal

When the input is 1, the clutch engages. When the input is 0, the clutch does not engage.

Conserving

B — Base interface
mechanical rotational

Mechanical rotational conserving port associated with the base interface.

F — Follower interface
mechanical rotational

Mechanical rotational conserving port associated with the follower interface.

Parameters

Kinetic friction torque — Rotational force for a moving clutch
10 N*m (default)

Specify a value that is greater than or equal to zero for the nominal kinetic friction torque that the clutch applies when engaged and the clutch plates have relative motion.

Static friction torque — Rotational force for a nonmoving clutch
20 N*m (default)

Specify a value that is greater than or equal to the nominal kinetic friction torque. When the clutch is locked, this amount of clutch is required to break the lock and allow the clutch plates to slip.

Clutch velocity tolerance — Motion boundary condition
0.001 rad/s (default)

Specify a value that is greater than or equal to zero.

Initial State — Clutch condition at simulation start time
Unlocked (default) | Locked

Indicate whether the clutch is locked or unlocked when the simulation starts.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also
Cone Clutch | Disk Friction Clutch | Dog Clutch | Fundamental Friction Clutch

Introduced in R2016b

Nonlinear Rotational Damper

Rotational damper based on polynomial or table lookup parameterizations

Library

Simscape / Driveline / Couplings & Drives / Springs & Dampers

Description

The block represents a nonlinear rotational damper. Polynomial and table lookup parameterizations define the nonlinear relationship between damping torque and relative angular velocity. The damping torque can be symmetric or asymmetric about the zero velocity point. The block applies equal and opposite damping torques on the two rotational conserving ports.

The symmetric polynomial parameterization defines the damping torque for both positive and negative relative velocities according to the expression:

$$T = b_1\omega + sign(\omega) \cdot b_2\omega^2 + b_3\omega^3 + sign(\omega) \cdot b_4\omega^4 + b_5\omega^5,$$

where:

- T — Damping torque
- b_1, b_2, ..., b_5 — Damping coefficients
- ω — Relative angular velocity between ports R and C, $\omega = \omega_R - \omega_C$
- ω_R — Absolute angular velocity associated with port R
- ω_C — Absolute angular velocity associated with port C

To avoid zero-crossings that slow simulation, eliminate the sign function from the polynomial expression by specifying an odd polynomial ($b_2, b_4 = 0$).

The two-sided polynomial parameterization defines the damping torque for both positive and negative relative velocities according to the expression:

$$T = \begin{cases} b_{1p}\omega + b_{2p}\omega^2 + b_{3p}\omega^3 + b_{4p}\omega^4 + b_{5p}\omega^5, \ \omega \geq 0 \\ b_{1n}\omega + b_{2n}\omega^2 + b_{3n}\omega^3 + b_{4n}\omega^4 + b_{5n}\omega^5, \ \omega < 0 \end{cases}$$

where:

- b_{1p}, b_{2p}, ..., b_{5p} — Damping coefficients for positive relative velocities
- b_{1n}, b_{2n}, ..., b_{5n} — Damping coefficients for negative relative velocities

Both polynomial parameterizations use a fifth-order polynomial expression. To use a lower-order polynomial, set the unneeded higher-order coefficients to zero. To use a higher-order polynomial, fit to a lower-order polynomial or use the table lookup parameterization.

The table lookup parameterization defines damping torque based on a set of torque and angular velocity vectors. If not specified, the block automatically adds a data point at the origin (zero angular velocity and zero torque).

Assumptions and Limitations

The block assumes viscous damping. The damping force depends only on velocity.

Ports

C
    Rotational conserving port.

R
    Rotational conserving port.

Parameters

Parameterization

    Select damping parameterization. Options are `By polynomial` and `By table lookup`.

By polynomial

Symmetry

Choose between symmetric and two-sided polynomial parameterizations.

- `Symmetric` — Specify a single set of polynomial coefficients governing damping for both positive and negative relative velocities.

  Vector of damping coefficients

  Enter five-element vector with polynomial damping coefficients. Physical units are for the first coefficient.

  The default vector is `[1e-4 0 1e-5 0 1e-6]`. The default unit is `N*m/(rad/s)`.

- `Two-sided` — Specify two sets of polynomial coefficients governing damping, one for positive relative velocities, the other for negative relative velocities.

  Vector of positive rotation damping coefficients

  Enter five-element vector with polynomial damping coefficients for positive relative velocities. Physical units are for the first coefficient.

  The default vector is `[1e-4 0 1e-5 0 1e-6]`. The default unit is `N*m/(rad/s)`.

  Vector of negative rotation damping coefficients

  Enter five-element vector with polynomial damping coefficients for negative relative velocities. Physical units are for the first coefficient.

  The default vector is `[0.001 -1e-5 1e-4 0 0]`.

By table lookup

Angular velocity vector

Enter vector with relative velocity values. The vector requires a minimum number of elements, based on the selected interpolation method — two for `Linear`, and three for `Smooth`. The number of elements must match the torque vector.

The default vector is `[-1 -0.5 -0.3 −0.1 0.1 0.3 0.5 1]`. The default unit is `rad/s`.

Torque vector

Enter vector with damping torque values corresponding to velocity vector. The vector requires a minimum number of elements, based on the selected

interpolation method — two for `Linear`, three for `Smooth`. The number of elements must match the velocity vector.

The default vector is `[-0.001 -4e-4 -2e-4 -5e-5 5e-5 2e-4 4e-4 0.001]`. The unit is `N*m`.

Interpolation Method

Select method used to find velocity–torque values between specified data points.

- `Linear` — Select this option to get the best performance.
- `Smooth` — Select this option to produce a continuous curve with continuous first-order derivatives.

For more information on interpolation algorithms, see the PS Lookup Table (1D) block reference page.

Extrapolation Method

Select method used to calculate values outside the table lookup data range.

- `Linear` — Select this option to produce a curve with continuous first-order derivatives in the extrapolation region and at the boundary with the interpolation region.
- `Nearest` — Select this option to produce an extrapolation that does not go above the highest point in the data or below the lowest point in the data.

For more information on extrapolation algorithms, see the PS Lookup Table (1D) block reference page.

Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using MATLAB® Coder™.

See Also

Nonlinear Rotational Damper | Nonlinear Translational Damper | Rotational Damper | Translational Damper | Variable Translational Damper

Nonlinear Rotational Spring

Torsional spring based on polynomial or table lookup parameterizations

Library

Simscape / Driveline / Couplings & Drives / Springs & Dampers

Description

The block represents a torsional spring with nonlinear torque-displacement curve. The spring torque magnitude is a general function of displacement. It need not satisfy Hooke's law. Polynomial and table lookup parameterizations provide two ways to specify the torque-displacement relationship. The spring torque can be symmetric or asymmetric with respect to zero deformation.

The symmetric polynomial parameterization defines spring torque according to the expression:

$$T = k_1\theta + sign(\theta) \cdot k_2\theta^2 + k_3\theta^3 + sign(\theta) \cdot k_4\theta^4 + k_5\theta^5,$$

where:

- T — Spring force
- k_1, k_2, ...,k_5 — Spring coefficients
- θ — Relative displacement between ports R and C, $\theta = \theta_{init} + \theta_R - \theta_C$
- θ_{init} — Initial spring deformation
- θ_R — Absolute angular position of port R
- θ_C — Absolute angular position of port C

At simulation start (t=0), θ_R and θ_C are zero, making θ equal to θ_{init}.

To avoid zero-crossings that slow simulation, eliminate the sign function from the polynomial expression by specifying an odd polynomial (b_2, b_4 = 0).

The two-sided polynomial parameterization defines spring torque according to the expression:

$$T = \begin{cases} k_{1t}\theta + k_{2t}\theta^2 + k_{3t}\theta^3 + k_{4t}\theta^4 + k_{5t}\theta^5, & \theta \geq 0 \\ k_{1c}\theta - k_{2c}\theta^2 + k_{3c}\theta^3 - k_{4c}\theta^4 + k_{5c}\theta^5, & \theta < 0 \end{cases},$$

where:

- k_{1t}, k_{2t}, ..., k_{5t} — Spring tension coefficients
- k_{1c}, k_{2c}, ..., k_{5c} — Spring compression coefficients

Both polynomial parameterizations use a fifth-order polynomial expression. To use a lower-order polynomial, set the unneeded higher-order coefficients to zero. To use a higher-order polynomial, fit to a lower-order polynomial or use the table lookup parameterization.

The table lookup parameterization defines spring torque based on a set of torque and angular velocity vectors. If not specified, the block automatically adds a data point at the origin (zero angular velocity and zero torque).

Variables

Use the **Variables** tab to set the priority and initial target values for the block variables before simulating. For more information, see "Set Priority and Initial Target for Block Variables" (Simscape).

Unlike block parameters, variables do not have conditional visibility. The **Variables** tab lists all the existing block variables. If a variable is not used in the set of equations corresponding to the selected block configuration, the values specified for this variable are ignored.

Ports

C

Rotational conserving port

R

Rotational conserving port

Parameters

Parameterization

Select spring parameterization. Options are `By polynomial` and `By table lookup`.

By Polynomial

Specify nonlinear spring function in terms of polynomial coefficients.

Symmetry

Choose between symmetric and two-sided polynomial parameterizations.

- `Symmetric` — Specify one set of polynomial coefficients governing spring torque in both tension and compression.

    Vector of spring coefficients

    Enter five-element vector with polynomial spring coefficients. The highest non-zero order must be positive. Physical units are for the first coefficient.

    The default vector is `[1 0 0.1 0 0.01]`. The default unit is `N*m/rad`.

- `Two-sided` — Specify two sets of polynomial coefficients governing spring torque, one for positive deformation, the other for negative deformation.

    Vector of spring tension coefficients

    Enter a five-element vector containing the coefficients of the polynomial spring *tension* function. The highest order non-zero coefficient must be positive. The specified physical unit is the unit of the first polynomial coefficient.

    Vector of spring compression coefficients

    Enter a five-element vector containing the coefficients of the polynomial spring *compression* function. The highest order non-zero coefficient must be positive. The specified physical unit is the unit of the first polynomial coefficient.

By table lookup

Deformation vector

Enter a vector containing the deformation values used in the table lookup. The vector must contain a minimum number of elements based on the interpolation method: two for `Linear`, and three for `Smooth`. The vector must also contain the same number of elements as the torque vector.

Torque vector

Enter a vector containing the torque values that correspond to the deformation vector values. The vector must contain a minimum number of elements based on the interpolation method: two for `Linear`, and three for `Smooth`. The vector must also contain the same number of elements as the deformation vector. If not included in the vectors, the block automatically adds a point at the (0,0) deformation–torque coordinate.

Interpolation method

Select method used to calculate deformation-torque values between table lookup data points.

- `Linear` — Select this option to get the best performance.
- `Smooth` — Select this option to produce a continuous curve with continuous first-order derivatives.

For more information on interpolation algorithms, see the PS Lookup Table (1D) block reference page.

Extrapolation method

Select method used to calculate deformation-torque values outside the table lookup data range.

- `Linear` — Select this option to produce a curve with continuous first-order derivatives in the extrapolation region and at the boundary with the interpolation region.
- `Nearest` — Select this option to produce an extrapolation that does not go above the highest point in the data or below the lowest point in the data.

For more information on extrapolation algorithms, see the PS Lookup Table (1D) block reference page.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also
Nonlinear Translational Spring | Variable Rotational Spring | Variable Translational Spring

Introduced in R2013a

Nonlinear Translational Damper

Translational damper based on polynomial or table lookup parameterizations

Library

Simscape / Driveline / Couplings & Drives / Springs & Dampers

Description

The block represents a nonlinear translational damper. Polynomial and table lookup parameterizations define the nonlinear relationship between damping force and relative linear velocity. The damping force can be symmetric or asymmetric about the zero velocity point. The block applies equal and opposite damping forces on the two translational conserving ports.

The symmetric polynomial parameterization defines the damping force for both positive and negative relative velocities according to the expression:

$$F = b_1 v + sign(v) \cdot b_2 v^2 + b_3 v^3 + sign(v) \cdot b_4 v^4 + b_5 v^5,$$

where:

- F — Damping force
- b_1, b_2, ..., b_5 — Damping coefficients
- v — Relative linear velocity between ports R and C, $v = v_R - v_C$
- v_R — Absolute linear velocity associated with port R
- v_C — Absolute linear velocity associated with port C

To avoid zero-crossings that slow simulation, eliminate the sign function from the polynomial expression by specifying an odd polynomial ($b_2, b_4 = 0$).

1-309

The two-sided polynomial parameterization defines the damping force for both positive and negative relative velocities according to the expression:

$$F = \begin{cases} b_{1e}v + b_{2e}v^2 + b_{3e}v^3 + b_{4e}v^4 + b_{5e}v^5, \ v \geq 0 \\ b_{1c}v - b_{2c}v^2 + b_{3c}v^3 - b_{4c}v^4 + b_{5c}v^5, \ v < 0 \end{cases},$$

where:

- b_{1e}, b_{2e}, ..., b_{5e} — Damping coefficients for positive relative velocities
- b_{1c}, b_{2c}, ..., b_{5c} — Damping coefficients for negative relative velocities

Positive relative velocities correspond to damper extension (ports R and C moving away from each other). Negative relative velocities correspond to damper contraction (ports R and C moving towards each other).

Both polynomial parameterizations use a fifth-order polynomial expression. To use a lower-order polynomial, set the unneeded higher-order coefficients to zero. For polynomials of order greater than five, fit to a polynomial of order smaller than or equal to five, or use the table lookup parameterization.

The table lookup parameterization defines damping force based on a set of velocity and force vectors. If not included in the vectors, the block automatically adds a data point at the origin (zero velocity and zero force).

Assumptions and Limitations

The block assumes viscous damping. The damping force depends only on velocity.

Ports

C

    Rotational conserving port

R

    Rotational conserving port

Parameters

Parameterization

Select damping parameterization. Options are `By polynomial` and `By table lookup`.

By polynomial

Specify coefficients of polynomial damping function.

Symmetry

Choose between symmetric and two-sided polynomial parameterizations.

- `Symmetric` — Specify a single set of polynomial coefficients governing damping for both positive and negative relative velocities.

  Vector of damping coefficients

  Enter five-element vector with polynomial damping coefficients. Physical units are for the first coefficient.

  The default vector is `[10 0 1 0 0.1]`. The default unit is `N/(m/s)`.

- `Two-sided` — Specify two sets of polynomial coefficients governing damping, one for positive relative velocities, the other for negative relative velocities.

  Vector of extension damping coefficients

  Enter five-element vector with polynomial damping coefficients for positive relative velocities. Physical units are for the first coefficient.

  The default vector is `[10 0 1 0 0.1]`. The default unit is `N/(m/s)`.

  Vector of contraction damping coefficients

  Enter five-element vector with polynomial damping coefficients for negative relative velocities. Physical units are for the first coefficient.

  The default vector is `[10 0 1 0 0.1]`. The default unit is `N/(m/s)`.

By table lookup

Velocity vector

Enter vector with relative velocity values. The vector requires a minimum number of elements, based on the selected interpolation method — two for `Linear`, and three for `Smooth`. The number of elements must match the force vector.

The default vector is `[-1 -0.5 -0.3 -0.1 0.1 0.3 0.5 1]`. The default unit is `m/s`.

Force vector

Enter vector with damping force values corresponding to velocity vector. The vector requires a minimum number of elements, based on the selected interpolation method — two for `Linear`, three for `Smooth`. The number of elements must match the velocity vector.

The default vector is `[-100 -40 -20 -5 5 20 40 100]`. The default unit is `N`.

Interpolation Method

Select method used to find intermediate velocity–force values between table lookup data points.

- `Linear` — Select this option to get the best performance.
- `Smooth` — Select this option to produce a continuous curve with continuous first-order derivatives.

For more information on interpolation algorithms, see the PS Lookup Table (1D) block reference page.

Extrapolation Method

Select method used to calculate values outside the table lookup data range.

- `Linear` — Select this option to produce a curve with continuous first-order derivatives in the extrapolation region and at the boundary with the interpolation region.
- `Nearest` — Select this option to produce an extrapolation that does not go above the highest point in the data or below the lowest point in the data.

For more information on extrapolation algorithms, see the PS Lookup Table (1D) block reference page.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also

Nonlinear Rotational Damper | Rotational Damper | Translational Damper | Variable Rotational Damper | Variable Translational Damper

Nonlinear Translational Spring

Translational spring based on polynomial or table lookup parameterizations

Library

Simscape / Driveline / Couplings & Drives / Springs & Dampers

Description

The block represents a translational spring with nonlinear force-displacement curve. The spring force magnitude is a general function of displacement. It need not satisfy Hooke's law. Polynomial and table lookup parameterizations provide two ways to specify the force-displacement relationship. The spring force can be symmetric or asymmetric with respect to zero deformation.

The symmetric polynomial parameterization defines spring force according to the expression

$$F = k_1 x + sign(x) \cdot k_2 x^2 + k_3 x^3 + sign(x) \cdot k_4 x^4 + k_5 x^5,$$

where:

- F — Spring force
- $k_1, k_2, ..., k_5$ — Spring coefficients
- x — Relative displacement between ports R and C, $x = x_{init} + x_R - x_C$
- x_{init} — Initial relative displacement
- x_R — Absolute displacement associated with port R
- x_C — Absolute displacement associated with port C

At simulation start (t=0), x_R and x_C are zero, making x equal to x_{init}.

To avoid zero-crossings that slow simulation, eliminate the sign function from the polynomial expression by specifying an odd polynomial (b_2,$b_4 = 0$).

The asymmetric polynomial parameterization defines spring force according to the expression:

$$F = \begin{cases} k_{1t}x + k_{2t}x^2 + k_{3t}x^3 + k_{4t}x^4 + k_{5t}x^5, & x \geq 0 \\ k_{1c}x + k_{2c}x^2 + k_{3c}x^3 + k_{4c}x^4 + k_{5c}x^5, & x < 0 \end{cases},$$

where:

- k_1, k_2, ..., k_5 — Spring tension coefficients
- k_1, k_2, ..., k_5 — Spring compression coefficients

Both polynomial parameterizations use a fifth-order polynomial expression. To use a lower-order polynomial, set the unneeded higher-order coefficients to zero. To use a higher-order polynomial, fit to a lower-order polynomial or use the table lookup parameterization.

The table lookup parameterization defines damping torque based on a set of torque and angular velocity vectors. If not specified, the block automatically adds a data point at the origin (zero angular velocity and zero torque).

Variables

Use the **Variables** tab to set the priority and initial target values for the block variables before simulating. For more information, see "Set Priority and Initial Target for Block Variables" (Simscape).

Unlike block parameters, variables do not have conditional visibility. The **Variables** tab lists all the existing block variables. If a variable is not used in the set of equations corresponding to the selected block configuration, the values specified for this variable are ignored.

Ports

C

   Rotational conserving port

R

Rotational conserving port

Parameters

Parameterization

Select spring parameterization. Options are `By polynomial` and `By table lookup`.

By polynomial

Specify coefficients of polynomial spring function.

Symmetry

Choose between symmetric and two-sided polynomial parameterizations.

- `Symmetric` — Specify one set of polynomial coefficients governing spring force in both tension and compression.

  Vector of spring coefficients

  Enter five-element vector with polynomial spring coefficients. The highest non-zero order must be positive. Physical units are for the first coefficient.

  The default vector is `[100 0 10 0 1]`. The default unit is `N/m`.

- `Two-sided` — Specify two sets of polynomial coefficients governing spring force, one for positive relative displacements, the other for negative relative displacements.

  Vector of spring tension coefficients

  Enter five-element vector with polynomial spring tension coefficients. Physical units are for the first coefficient.

  The default vector is `[100 0 10 0 1]`. The default unit is `N/m`.

  Vector of spring compression coefficients

  Enter five-element vector with polynomial spring compression coefficients. Physical units are for the first coefficient.

  The default vector is `[1000 −10 100 0 0]`. The default unit is `N/m`.

By table lookup

Deformation vector

Enter vector with relative displacement values. the vector requires a minimum number of elements, based on the selected interpolation method — two for `Linear`, and three for `Smooth`. The number of elements must match the force vector. If not included in the vector, the block automatically adds a point at the origin, corresponding to zero displacement and zero force. If the vector includes a zero value, then the force vector must contain a corresponding zero value.

Force vector

Enter a vector containing the force values that correspond to the deformation vector values. The vector must contain a minimum number of elements based on the interpolation method: two for `Linear`, and three for `Smooth`. The vector must also contain the same number of elements as the deformation vector. If not included in the vectors, the block automatically adds a point at the (0,0) deformation–force coordinate.

Interpolation Method

Select method used to find intermediate velocity–force values between table lookup data points.

- `Linear` — Select this option to get the best performance.

- `Smooth` — Select this option to produce a continuous curve with continuous first-order derivatives.

For more information on interpolation algorithms, see the PS Lookup Table (1D) block reference page.

Extrapolation Method

Select method used to calculate values outside the table lookup data range.

- `Linear` — Select this option to produce a curve with continuous first-order derivatives in the extrapolation region and at the boundary with the interpolation region.

- `Nearest` — Select this option to produce an extrapolation that does not go above the highest point in the data or below the lowest point in the data.

For more information on extrapolation algorithms, see the PS Lookup Table (1D) block reference page.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also
Nonlinear Rotational Spring | Variable Rotational Spring | Variable Translational Spring

Piston

Piston mechanism of reciprocating combustion engine

Library

Simscape / Driveline / Engines

Description

The Piston block represents the piston mechanism of a reciprocating combustion engine. The piston model accounts for the instantaneous torque transmitted to the engine crankshaft. The instantaneous torque enables you to model vibrations in the drivetrain due to piston revolution. To model a multi-piston engine, use the Piston Engine block.
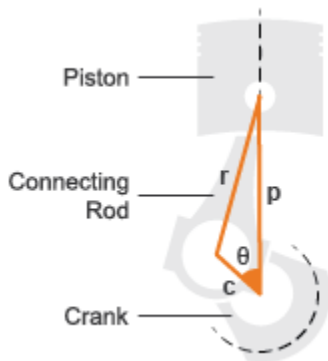
Port **B** represents the translating piston and port **F** the rotating crankshaft. The piston force follows from the cylinder pressure and cross-sectional area. The block obtains the combustion pressure from a lookup table parameterized in terms of the crank angle and, optionally, the crank angular velocity and engine throttle level.

The crank torque follows from the piston force and crank angle as well as the crank and connecting rod lengths. In terms of these inputs, the ratio of the piston force and crank torque is

$$
\frac{T_{\text{F}}}{F_{\text{B}}} = -c\left(sin(\theta) + \frac{sin(2\theta)}{2\sqrt{\left(\frac{\text{r}}{\text{c}}\right)^2 - \sin^2(\theta)}}\right),
$$

where:

- F_B is the instantaneous piston force associated with the base port.
- T_F is the instantaneous crank torque associated with the follower port.
- c is the crank length.
- θ is the instantaneous crank angle.
- r is the connecting rod length.

Piston Dimensions

Physical signal port T lets you specify the engine throttle level as a fraction between 0 and 1. This fraction corresponds to the percentage of full power generated. The block uses the physical signal input whenever the pressure lookup table in the block dialog box is parameterized only in terms of the crank angle.

Ports

B

Conserving rotational port representing the engine block.

F

Conserving rotational port representing the engine crankshaft.

T

Physical signal input port for specifying the engine throttle setting.

FC

Physical signal input port for specifying fuel consumption rate. This port is hidden by default. Selecting one of these options for the **Fuel consumption model** parameter in the **Fuel Consumption** settings exposes this port:

- `Constant per revolution`
- `Fuel consumption by speed and torque`
- `Brake specific fuel consumption by speed and torque`
- `Brake specific fuel consumption by speed and brake mean effective pressure`

Parameters

Cylinder bore

Internal diameter of the engine cylinder that the piston travels in. The default value is `0.10` m.

Piston stroke

Distance between the top dead center and bottom dead center piston positions. The default value is `0.06` m.

Piston rod length

Length of the connecting rod located between the engine piston and the crankshaft. The default value is `0.10` m.

Number of strokes per cycle

Number of strokes required to complete one engine cycle. One stroke corresponds to a full extension or retraction of the engine piston. A typical automobile engine is based on a four-stroke cycle with induction, compression, power, and exhaust stages. The default value is 4

Pressure parameterization

Engine variables that the cylinder pressure depends on. Options include `By crank angle`, `By crank angle and throttle`, `By crank angle, throttle, and crank velocity`. The default setting is `By crank angle`.

Crank angle vector

M-element vector of crank angles at which to specify the cylinder pressure. A zero angle corresponds to a piston at top dead center. The vector must range from *-S**180

to +*S**180 degrees, where *S* is the number of strokes per cycle. The default vector is an eight-element vector ranging in value from -360 to +360 deg, corresponding to a four-stroke cycle.

Throttle vector

N-element vector of engine throttle settings at which to specify the cylinder pressure. A value of `0` corresponds to no throttle and a value of `1` to full throttle. This parameter is active only when **Pressure parameterization** is set to `By crank angle and throttle` and `By crank angle, throttle, and crank velocity`. The default vector is `[0.0, 0.3, 0.8, 1.0]`.

Crank velocity vector

L-element vector of crankshaft angular velocities at which to specify the cylinder pressure. This parameter is active only when **Pressure parameterization** is set to `By crank angle, throttle, and crank velocity`. The default vector is `[0.0, 1000.0, 6000.0]` rpm.

Pressure vector

M-element vector of cylinder pressures corresponding to the crank angles specified in the **Crank angle vector** parameter. This parameter is active only when **Pressure parameterization** is set to `By crank angle`.

Pressure matrix (gauge)

M-by-*N* matrix of cylinder pressures corresponding to the crank angles and throttle settings specified in the **Crank angle vector** and **Throttle vector** parameters. This parameter is active only when **Pressure parameterization** is set to `By crank angle and throttle`. The default matrix is an 8–by-4 matrix ranging in value from 0 to 50 bar.

Pressure matrix 3D (gauge)

M-by-*N*-by-*L* matrix of cylinder pressures corresponding to the crank angles, throttle settings, and crank angular velocity values specified in the **Crank angle vector**, **Throttle vector**, and **Crank velocity vector** parameters. This parameter is active only when **Pressure parameterization** is set to `By crank angle, throttle, and crank velocity`. The default matrix is an 8–by-4–by-3 matrix ranging in value from 0 to 50 bar.

Main

Bearing viscous friction coefficient

Aggregate viscous friction coefficient of the main and rod bearings. The block uses this parameter to compute the viscous power losses between the base and follower shafts. The default value is `0.001` N*m/(rad/s).

Offset angle

Point in the engine cycle when the piston reaches top dead center. The engine cycle spans in angle from -S*180 to +S*180 degrees, where S is the number of strokes per cycle. The default value is `0` deg, corresponding to a zero offset.

Initial crank angle

Crank angle at time zero relative to a top dead center position. The default value is `90` deg.

Fuel Consumption

The table shows how the visibility of some parameters depends on the option that you choose for other parameters. To learn how to read the table, see "Parameter Dependencies" on page B-2.

Fuel Consumption Parameter Dependencies

Fuel Consumption				
Fuel consumption model — Choose `No fuel consumption`, `Constant per revolution`, `Fuel consumption by speed and torque`, `Brake specific fuel consumption by speed and torque`, or `Brake specific fuel consumption by speed and brake mean effective pressure`				
No fuel consumption	**Constant per revolution**	**Fuel consumption by speed and torque**	**Brake specific fuel consumption by speed and torque**	**Brake specific fuel consumption by speed and brake mean effective pressure**
	Fuel consumption per revolution			
		Speed vector		
		Torque vector		Brake mean effective pressure vector
		Fuel consumption table	Brake specific fuel consumption table	
		Interpolation method — Choose `Linear` or `Smooth`		

Fuel consumption model

Select a model for calculating engine-fuel consumption. Model parameterizations are compatible with standard industrial data. Choose between these options:

- `No fuel consumption` — The default option
- `Constant per revolution`
- `Fuel consumption by speed and torque`
- `Brake specific fuel consumption by speed and torque`
- `Brake specific fuel consumption by speed and brake mean effective pressure`

Selecting any option other than `No fuel consumption` exposes the **FC** port and related parameters. For more information, see the Fuel Consumption Parameter Dependencies table.

No fuel consumption

The block does not calculate fuel consumption. The **FC** port, which reports the fuel consumption rate, is not exposed. Selecting this option increases simulation speed.

Fuel consumption per revolution

Enter the volume of fuel consumed in one crankshaft revolution. The default is 25 mg/rev.

Selecting `Constant per revolution` for the **Fuel consumption model** parameter exposes this parameter.

For more information, see the Fuel Consumption Parameter Dependencies table.

Speed vector

Enter vector of engine speeds used in lookup table parameterizations. Vector size must match **Torque vector size**. The default is [1000, 2000, 3000, 4000, 5000, 6000] rpm. Selecting `Fuel consumption by speed and torque`, `Brake specific fuel consumption by speed and torque`, or `Brake specific fuel consumption by speed and brake mean effective pressure` for the **Fuel consumption model** parameter exposes this parameter.

For more information, see the Fuel Consumption Parameter Dependencies table.

Torque vector

Enter vector of engine torques used in the lookup table parameterizations. Vector size must match **Speed vector** size. The default is [0, 80, 160, 200, 240, 320, 360, 400] N*m. Selecting `Fuel consumption by speed and torque` or `Brake specific fuel consumption by speed and torque` for the **Fuel consumption model** parameter exposes this parameter.

For more information, see the Fuel Consumption Parameter Dependencies table.

Fuel consumption table

Enter matrix with fuel consumption rates corresponding to engine speed and torque vectors. The number of rows must equal the number of elements in the **Speed vector**. The number of columns must equal the number of elements in the **Torque vector**. The default is [.5, .9, 1.4, 1.6, 1.9, 2.7, 3.4, 4.4; 1, 1.7, 2.7, 3.1, 3.6, 5, 6, 7.4; 1.4, 2.7, 4, 4.8, 5.6, 7.5, 8.5, 10.5; 2, 3.6, 5.8, 6.7, 8, 10.4, 11.7, 13.3; 2.5, 4.8, 7.9, 9.4, 10.8, 14, 16.2, 18.6; 3.1, 6, 10.3, 11.9, 13.8, 18.4, 22, 26.5] g/s.

Selecting `Fuel consumption by speed and torque` for the **Fuel consumption model** parameter exposes this parameter.

For more information, see the Fuel Consumption Parameter Dependencies table.

Brake mean effective pressure vector

Enter vector of brake mean effective pressure (BMEP) values. The default is `[0, 250, 500, 625, 750, 1000, 1150, 1250]` kPa. The BMEP satisfies the expression:

$$BMEP = T \cdot \left(\frac{2\pi \cdot n_c}{V_d} \right)$$

where:

- T — Output torque
- n_c — Number of cycles per revolution
- V_d — Cylinder displaced volume

Selecting `Brake specific fuel consumption by speed and brake mean effective pressure` for the **Fuel consumption model** parameter exposes this parameter.

For more information, see the Fuel Consumption Parameter Dependencies table.

Brake specific fuel consumption table

Selecting `Brake specific fuel consumption by speed and torque` or `Brake specific fuel consumption by speed and brake mean effective pressure` for the **Fuel consumption model** parameter exposes this parameter.

For more information, see the Fuel Consumption Parameter Dependencies table.

For the `Brake specific fuel consumption by speed and torque` fuel model, enter the matrix with brake specific fuel consumption (BSFC) rates corresponding to engine speed and torque vectors. BSFC is the ratio of the fuel consumption rate to the output power. The number of rows must equal the number of elements in the **Speed vector**. The number of columns must equal the number of elements in the **Torque vector**.

For the `Brake specific fuel consumption by speed and brake mean effective pressure` fuel model, enter the matrix with brake specific fuel consumption (BSFC) rates corresponding to engine speed and brake mean effective pressure (BMEP) vectors. BSFC is the ratio of the fuel consumption rate to the output power. The number of rows must equal the number of elements in the **Speed vector**.

The number of columns must equal the number of elements in the **Brake mean effective pressure vector**.

For both fuel-consumption models, the default is [410, 380, 300, 280, 270, 290, 320, 380; 410, 370, 290, 270, 260, 270, 285, 320; 415, 380, 290, 275, 265, 270, 270, 300; 420, 390, 310, 290, 285, 280, 280, 285; 430, 410, 340, 320, 310, 300, 310, 320; 450, 430, 370, 340, 330, 330, 350, 380] g/hr/kW.

Interpolation method

Select the interpolation method used to calculate fuel consumption at intermediate speed-torque values. Methods are Linear and Smooth. Outside the data range, fuel consumption is held constant at the last value given in the lookup table. Selecting Fuel consumption by speed and torque, Brake specific fuel consumption by speed and torque, or Brake specific fuel consumption by speed and brake mean effective pressure for the **Fuel consumption model** parameter exposes this parameter.

For more information, see the Fuel Consumption Parameter Dependencies table.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also
Generic Engine | Piston Engine

Topics
"Troubleshoot Engine Issues"

Piston Engine

Reciprocating combustion engine with variable number of pistons

Library

Simscape / Driveline / Engines

Description

The Piston Engine block represents a reciprocating combustion engine with multiple cylinders. The piston model accounts for the instantaneous torque transmitted to the engine drive shaft. The instantaneous torque enables you to model vibrations in the drivetrain due to piston revolution. To model just the piston mechanism of a combustion engine, use the Piston block.
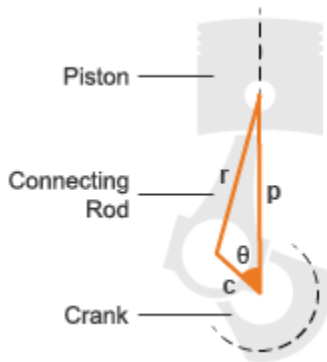
Port **B** represents the translating piston and port **F** the rotating crankshaft. The piston force follows from the cylinder pressure and cross-sectional area. The block obtains the combustion pressure from a lookup table parameterized in terms of the crank angle and, optionally, the crank angular velocity and engine throttle level.

The crank torque follows from the piston force and crank angle as well as the crank and connecting rod lengths. In terms of these inputs, the ratio of the piston force and crank torque is

$$\frac{T_{\mathrm{F}}}{F_{\mathrm{B}}} = -c\left(sin(\theta) + \frac{sin(2\theta)}{2\sqrt{\left(\frac{r}{c}\right)^2 - \sin^2(\theta)}} \right),$$

where:

- F_B is the instantaneous piston force associated with the base port.
- T_F is the instantaneous crank torque associated with the follower port.
- c is the crank length.
- θ is the instantaneous crank angle.
- r is the connecting rod length.

Piston Dimensions

Physical signal port T lets you specify the engine throttle level as a fraction between 0 and 1. This fraction corresponds to the percentage of full power generated. The block uses the physical signal input whenever the pressure lookup table in the block dialog box is parameterized only in terms of the crank angle.

Ports

B

   Conserving rotational port representing the engine block

F

   Conserving rotational port representing the engine crankshaft

T

   Physical signal input port for specifying the engine throttle setting

FC

Physical signal input port for specifying fuel consumption rate. This port is hidden by default. Selecting one of these options for the **Fuel consumption model** parameter in the **Fuel Consumption** settings exposes this port:

- `Constant per revolution`
- `Fuel consumption by speed and torque`
- `Brake specific fuel consumption by speed and torque`
- `Brake specific fuel consumption by speed and brake mean effective pressure`

Parameters

Pistons

Number of pistons

Number of pistons in the combustion engine. The default number is 4.

Offset angle vector

Vector of piston offset angles. The offset angle specifies the point in the engine cycle when the piston reaches top dead center. The engine cycle spans in angle from -S*180 to +S*180 degrees, where S is the number of strokes per cycle.

The vector size must be the same as the number of pistons. The default vector is `[0.0, 180.0, 360.0, -180.0]`, corresponding to a four-stroke, four-piston engine.

Cylinder bore

Internal diameter of the engine cylinder that the piston travels in. The default value is `0.10` m.

Piston stroke

Distance between the top dead center and bottom dead center piston positions. The default value is `0.06` m.

Piston rod length

Length of the connecting rod located between the engine piston and the crankshaft. The default value is `0.10` m.

Number of strokes per cycle

Number of strokes required to complete one engine cycle. One stroke corresponds to a full extension or retraction of the engine piston. A typical automobile engine is based on a four-stroke cycle with induction, compression, power, and exhaust stages. The default value is 4

Pressure parameterization

Engine variables that the cylinder pressure depends on. Options include By crank angle, By crank angle and throttle, By crank angle, throttle, and crank velocity. The default setting is By crank angle.

Crank angle vector

M-element vector of crank angles at which to specify the cylinder pressure. A zero angle corresponds to a piston at top dead center. The vector must range from -*S**180 to +*S**180 degrees, where *S* is the number of strokes per cycle. The default vector is an eight-element vector ranging in value from -360 to +360 deg, corresponding to a four-stroke cycle.

Throttle vector

N-element vector of engine throttle settings at which to specify the cylinder pressure. A value of 0 corresponds to no throttle and a value of 1 to full throttle. This parameter is active only when **Pressure parameterization** is set to By crank angle and throttle and By crank angle, throttle, and crank velocity. The default vector is [0.0, 0.3, 0.8, 1.0].

Crank velocity vector

L-element vector of crankshaft angular velocities at which to specify the cylinder pressure. This parameter is active only when **Pressure parameterization** is set to By crank angle, throttle, and crank velocity. The default vector is [0.0, 1000.0, 6000.0] rpm.

Pressure vector

M-element vector of cylinder pressures corresponding to the crank angles specified in the **Crank angle vector** parameter. This parameter is active only when **Pressure parameterization** is set to By crank angle.

Pressure matrix (gauge)

M-by-*N* matrix of cylinder pressures corresponding to the crank angles and throttle settings specified in the **Crank angle vector** and **Throttle vector** parameters. This parameter is active only when **Pressure parameterization** is set to By crank angle and throttle. The default matrix is an 8–by-4 matrix ranging in value from 0 to 50 bar.

Pressure matrix 3D (gauge)

M-by-*N*-by-*L* matrix of cylinder pressures corresponding to the crank angles, throttle settings, and crank angular velocity values specified in the **Crank angle vector**, **Throttle vector**, and **Crank velocity vector** parameters. This parameter is active only when **Pressure parameterization** is set to By crank angle, throttle, and crank velocity. The default matrix is an 8–by-4–by-3 matrix ranging in value from 0 to 50 bar.

Crankshaft

Shaft dynamics

Optional shaft dynamics parameters. Options include No shaft dynamics — Suitable for HIL simulation and Specify shaft stiffness, damping, and inertia

Stiffness

Stiffness coefficient of the engine crankshaft. This parameter accounts for resistance to shaft deformation. This parameter is active only when the **Shaft dynamics** parameter is set to Specify shaft stiffness, damping, and inertia. The default value is 1e6 N*m/rad.

Damping

Damping coefficients of the engine crankshaft. This parameter accounts for energy dissipation due to shaft deformation. This parameter is active only when the **Shaft dynamics** parameter is set to Specify shaft stiffness, damping, and inertia. The default value is 1000 N*m/(rad/s).

Inertia

Moment of inertia of the crankshaft about its rotational axis. This parameter accounts for resistance to sudden changes in motion. This parameter is active only when the **Shaft dynamics** parameter is set to Specify shaft stiffness, damping, and inertia. The default value is 0.02 kg*m^2.

Initial angular deflection

Deflection angle between the base and follower ends of the crankshaft at time zero. The deflection angle measures the angular deformation of the crankshaft due to torsion. This parameter is active only when the **Shaft dynamics** parameter is set to Specify shaft stiffness, damping, and inertia. The default value is 0 deg.

Initial angular velocity

Angular velocity of the crankshaft at time zero. This parameter is active only when the **Shaft dynamics** parameter is set to `Specify shaft stiffness, damping, and inertia`. The default value is `0` rpm.

Base and follower bearing viscous friction coefficients

Two-element vector with the viscous friction coefficients of the base and follower shaft bearings. The block uses viscous friction to account for energy losses between the base and follower shafts. The default vector is `[0.0, 0.0]` N*m/(rad/s), corresponding to zero damping.

Initial crank angle

Crank angle at time zero relative to a top dead center position. The default value is `90` deg.

Fuel Consumption

The table shows how the visibility of some parameters depends on the option that you choose for other parameters. To learn how to read the table, see "Parameter Dependencies" on page B-2.

Fuel Consumption Parameter Dependencies

Fuel Consumption				
Fuel consumption model — Choose `No fuel consumption`, `Constant per revolution`, `Fuel consumption by speed and torque`, `Brake specific fuel consumption by speed and torque`, or `Brake specific fuel consumption by speed and brake mean effective pressure`				
No fuel consumption	**Constant per revolution**	**Fuel consumption by speed and torque**	**Brake specific fuel consumption by speed and torque**	**Brake specific fuel consumption by speed and brake mean effective pressure**
	Fuel consumption per revolution			
		Speed vector		
		Torque vector		Brake mean effective pressure vector
		Fuel consumption table	Brake specific fuel consumption table	
		Interpolation method — Choose `Linear` or `Smooth`		

Fuel consumption model

Select a model for calculating engine-fuel consumption. Model parameterizations are compatible with standard industrial data. Choose between these options:

- `No fuel consumption` — The default option
- `Constant per revolution`
- `Fuel consumption by speed and torque`
- `Brake specific fuel consumption by speed and torque`
- `Brake specific fuel consumption by speed and brake mean effective pressure`

Selecting any option other than `No fuel consumption` exposes the **FC** port and related parameters. For more information, see the Fuel Consumption Parameter Dependencies table.

No fuel consumption

The block does not calculate fuel consumption. The **FC** port, which reports the fuel consumption rate, is not exposed. Selecting this option increases simulation speed.

Fuel consumption per revolution

Enter the volume of fuel consumed in one crankshaft revolution. The default is 25 mg/rev.

Selecting `Constant per revolution` for the **Fuel consumption model** parameter exposes this parameter.

For more information, see the Fuel Consumption Parameter Dependencies table.

Speed vector

Enter vector of engine speeds used in lookup table parameterizations. Vector size must match **Torque vector size**. The default is [1000, 2000, 3000, 4000, 5000, 6000] rpm. Selecting `Fuel consumption by speed and torque`, `Brake specific fuel consumption by speed and torque`, or `Brake specific fuel consumption by speed and brake mean effective pressure` for the **Fuel consumption model** parameter exposes this parameter.

For more information, see the Fuel Consumption Parameter Dependencies table.

Torque vector

Enter vector of engine torques used in the lookup table parameterizations. Vector size must match **Speed vector** size. The default is [0, 80, 160, 200, 240, 320, 360, 400] N*m. Selecting `Fuel consumption by speed and torque` or `Brake specific fuel consumption by speed and torque` for the **Fuel consumption model** parameter exposes this parameter.

For more information, see the Fuel Consumption Parameter Dependencies table.

Fuel consumption table

Enter matrix with fuel consumption rates corresponding to engine speed and torque vectors. The number of rows must equal the number of elements in the **Speed vector**. The number of columns must equal the number of elements in the **Torque vector**. The default is [.5, .9, 1.4, 1.6, 1.9, 2.7, 3.4, 4.4; 1, 1.7, 2.7, 3.1, 3.6, 5, 6, 7.4; 1.4, 2.7, 4, 4.8, 5.6, 7.5, 8.5, 10.5; 2, 3.6, 5.8, 6.7, 8, 10.4, 11.7, 13.3; 2.5, 4.8, 7.9, 9.4, 10.8, 14, 16.2, 18.6; 3.1, 6, 10.3, 11.9, 13.8, 18.4, 22, 26.5] g/s.

Selecting `Fuel consumption by speed and torque` for the **Fuel consumption model** parameter exposes this parameter.

For more information, see the Fuel Consumption Parameter Dependencies table.

Brake mean effective pressure vector

Enter vector of brake mean effective pressure (BMEP) values. The default is [0, 250, 500, 625, 750, 1000, 1150, 1250] kPa. The BMEP satisfies the expression:

$$BMEP = T \cdot \left(\frac{2\pi \cdot n_c}{V_d} \right)$$

where:

- T — Output torque
- n_c — Number of cycles per revolution
- V_d — Cylinder displaced volume

Selecting Brake specific fuel consumption by speed and brake mean effective pressure for the **Fuel consumption model** parameter exposes this parameter.

For more information, see the Fuel Consumption Parameter Dependencies table.

Brake specific fuel consumption table

Selecting Brake specific fuel consumption by speed and torque or Brake specific fuel consumption by speed and brake mean effective pressure for the **Fuel consumption model** parameter exposes this parameter.

For more information, see the Fuel Consumption Parameter Dependencies table.

For the Brake specific fuel consumption by speed and torque fuel model, enter the matrix with brake specific fuel consumption (BSFC) rates corresponding to engine speed and torque vectors. BSFC is the ratio of the fuel consumption rate to the output power. The number of rows must equal the number of elements in the **Speed vector**. The number of columns must equal the number of elements in the **Torque vector**.

For the Brake specific fuel consumption by speed and brake mean effective pressure fuel model, enter the matrix with brake specific fuel consumption (BSFC) rates corresponding to engine speed and brake mean effective pressure (BMEP) vectors. BSFC is the ratio of the fuel consumption rate to the output power. The number of rows must equal the number of elements in the **Speed vector**.

The number of columns must equal the number of elements in the **Brake mean effective pressure vector**.

For both fuel-consumption models, the default is [410, 380, 300, 280, 270, 290, 320, 380; 410, 370, 290, 270, 260, 270, 285, 320; 415, 380, 290, 275, 265, 270, 270, 300; 420, 390, 310, 290, 285, 280, 280, 285; 430, 410, 340, 320, 310, 300, 310, 320; 450, 430, 370, 340, 330, 330, 350, 380] g/hr/kW.

Interpolation method

Select the interpolation method used to calculate fuel consumption at intermediate speed-torque values. Methods are `Linear` and `Smooth`. Outside the data range, fuel consumption is held constant at the last value given in the lookup table. Selecting `Fuel consumption by speed and torque`, `Brake specific fuel consumption by speed and torque`, or `Brake specific fuel consumption by speed and brake mean effective pressure` for the **Fuel consumption model** parameter exposes this parameter.

For more information, see the Fuel Consumption Parameter Dependencies table.

Extended Capabilities

C/C++ Code Generation
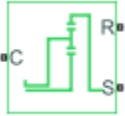Generate C and C++ code using MATLAB® Coder™.

See Also
Generic Engine | Piston

Topics
"Troubleshoot Engine Issues"

Planetary Gear

Gear train with sun, planet, and ring gears

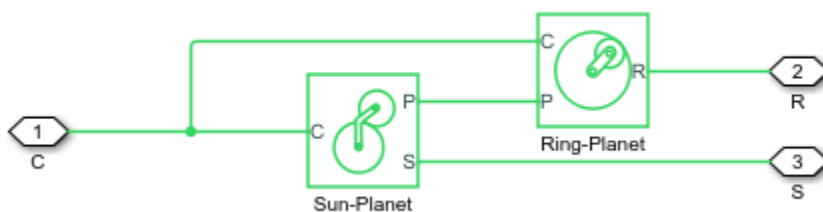
Library

Simscape / Driveline / Gears

Description

This block models a gear train with sun, planet, and ring gears. Planetary gears are common in transmission systems, where they provide high gear ratios in compact geometries. A carrier connected to a drive shaft holds the planet gears. Ports C, R, and S represent the shafts connected to the planet gear carrier, ring gear, and sun gear.
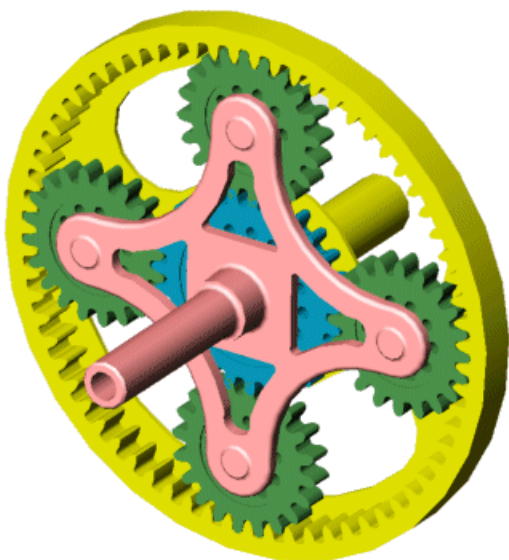
The block models the planetary gear as a structural component based on Sun-Planet and Ring-Planet Simscape Driveline blocks. The figure shows the block diagram of this structural component.

To increase the fidelity of the gear model, you can specify properties such as gear inertia, meshing losses, and viscous losses. By default, gear inertia and viscous losses are assumed negligible. The block enables you to specify the inertias of the internal planet gears only. To model the inertias of the carrier, sun, and ring gears, connect Simscape Inertia blocks to ports C, S, and R.

Thermal Modeling

You can model the effects of heat flow and temperature change through an optional thermal conserving port. By default, the thermal port is hidden. To expose the thermal port, right-click the block in your model and, from the context menu, select **Simscape** > **Block choices**. Select a variant that includes a thermal port. Specify the associated thermal parameters for the component.

Planetary Gear Model

Ideal Gear Constraints and Gear Ratios

Planetary Gear imposes two kinematic and two geometric constraints on the three connected axes and the fourth, internal gear (planet):

$$r_C \omega_C = r_S \omega_S + r_P \omega_P \;, \quad r_C = r_S + r_P \;,$$

$$r_R \omega_R = r_C \omega_C + r_P \omega_P \;, \quad r_R = r_C + r_P \;.$$

The ring-sun gear ratio $g_{RS} = r_R/r_S = N_R/N_S$. N is the number of teeth on each gear. In terms of this ratio, the key kinematic constraint is:

$$(1 + g_{RS})\omega_C = \omega_S + g_{RS}\omega_R \;.$$

The four degrees of freedom reduce to two independent degrees of freedom. The gear pairs are (1,2) = (S,P) and (P,R).

Warning The gear ratio g_{RS} must be strictly greater than one.

The torque transfer is:

$$g_{RS}\tau_S + \tau_R - \tau_{loss} = 0 \;,$$

with $\tau_{loss} = 0$ in the ideal case.

Nonideal Gear Constraints and Losses

In the nonideal case, $\tau_{loss} \neq 0$. See "Model Gears with Losses".

Limitations

- Gears are assumed rigid.
- Coulomb friction slows down simulation. See "Adjust Model Fidelity".

Ports

Port	Description
C	Rotational conserving port that represents the planet gear carrier
R	Rotational conserving port that represents the ring gear
S	Rotational conserving port that represents the sun gear
H	Thermal conserving port for thermal modeling

Parameters

Main

Ring (R) to sun (S) teeth ratio (NR/NS)

Ratio g_{RS} of the ring gear wheel radius to the sun gear wheel radius. This gear ratio must be strictly greater than 1. The default is 2.

Meshing Losses

Parameters for meshing losses vary with the block variant chosen—one with a thermal port for thermal modeling and one without it.

Without Thermal Port

Friction model

Select how to implement friction losses from nonideal meshing of gear teeth. The default is No meshing losses.

- No meshing losses — Suitable for HIL simulation — Gear meshing is ideal.

- `Constant efficiency` — Transfer of torque between gear wheel pairs is reduced by a constant efficiency η satisfying $0 < \eta \leq 1$. If you select this option, the panel changes from its default.

Constant Efficiency

Sun-planet and ring-planet ordinary efficiencies

Vector of torque transfer efficiencies [η_{SP} η_{RP}] for sun-planet and ring-planet gear wheel pair meshings, respectively.

Sun-carrier and planet-carrier power thresholds

Vector of power thresholds p_{th} above which full efficiency loss is applied, for sun-carrier and planet-carrier, respectively.

With Thermal Port

Temperature

Array of temperatures used to construct a 1-D temperature-efficiency lookup table. The array values must increase left to right. The temperature array must be the same size as each efficiency array. The default array is `[280 300 320]` K.

Sun-planet efficiency

Array of mechanical efficiencies with power flowing from the sun gear to the planet gears. Each array value is the ratio of output power to input power at one of the temperatures in the temperature array. The temperature and efficiency arrays must be the same size. The default array is `[0.95 0.9 0.85]`.

Ring-planet efficiency

Array of mechanical efficiencies with power flowing from the ring gear to the planet gears. Each array value is the ratio of output power to input power at one of the temperatures in the temperature array. The temperature and efficiency arrays must be the same size. The default array is `[0.95 0.9 0.85]`.

Sun-carrier and planet-carrier velocity thresholds

Array of power thresholds above which the full efficiency factors apply. Enter the thresholds for the sun and planet gears relative to the gear carrier, in this order. A hyperbolic tangent function smooths the efficiency factors between zero at rest and the values provided by the temperature-efficiency lookup tables at the power thresholds. The default array is `[0.001 0.001]` W.

Viscous Losses

Sun-carrier and planet-carrier viscous friction coefficients

Vector of viscous friction coefficients $[\mu_S\ \mu_P]$ for the sun-carrier and planet-carrier gear motions, respectively. The default is [0 0].

From the drop-down list, choose units. The default is newton-meters/(radians/second) (N*m/(rad/s)).

Inertia

Planet gear inertia

Moment of inertia of the combined planet gears. This value must be positive or zero. Enter 0 to ignore gear inertia. The default value is 0 kg*m^2.

Thermal Port

Thermal mass

Thermal energy required to change the component temperature by a single degree. The greater the thermal mass, the more resistant the component is to temperature change. The default value is 50 J/K.

Initial temperature

Component temperature at the start of simulation. The initial temperature alters the component efficiency according to an efficiency vector that you specify, affecting the starting meshing or friction losses. The default value is 300 K.

Real-Time Simulation

Hardware-in-the-Loop Simulation

For optimal simulation performance, use the **Meshing Losses > Friction model** parameter default setting, No meshing losses - Suitable for HIL simulation.

1-343

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also

Simscape Blocks
Compound Planetary Gear | Double-Pinion Planetary Gear | Ravigneaux Gear

Topics
"Basic Motion, Torque, and Force Modeling"
"Model Thermal Losses in Driveline Components"
"Prepare Simscape Driveline Models for Real-Time Simulation Using Simscape Checks"

Planet-Planet

Planetary gear set of carrier, inner planet, and outer planet wheels with adjustable gear ratio and friction losses

Library

Simscape / Driveline / Gears / Planetary Subcomponents

Description

The Planet-Planet gear block represents a set of carrier, inner planet, and outer planet gear wheels. Both planetary gears are connected to and rotate with respect to the carrier. The planets corotate with a fixed gear ratio that you specify. For model details, see "Planet-Planet Gear Model" on page 1-346.
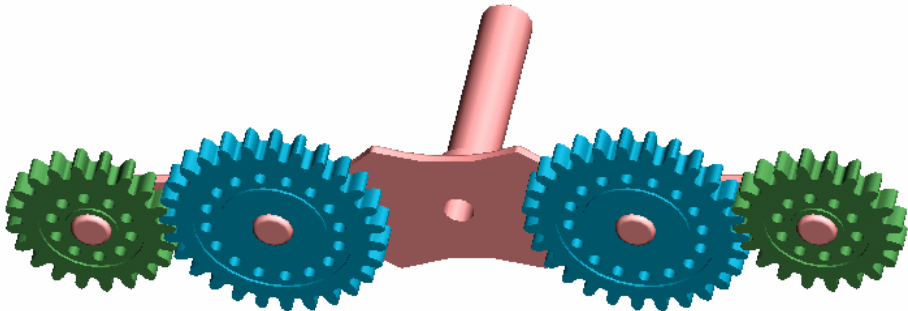
Thermal Modeling

You can model the effects of heat flow and temperature change through an optional thermal conserving port. By default, the thermal port is hidden. To expose the thermal

port, right-click the block in your model and, from the context menu, select **Simscape > Block choices**. Select a variant that includes a thermal port. Specify the associated thermal parameters for the component.

Planet-Planet Gear Model

Ideal Gear Constraints and Gear Ratios

Planet-Planet imposes one kinematic and one geometric constraint on the three connected axes:

$$r_{\mathrm{C}}\omega_{\mathrm{C}} = r_{\mathrm{Po}}\omega_{\mathrm{Po}} + r_{\mathrm{Pi}}\omega_{\mathrm{Pi}} \quad , \quad r_{\mathrm{C}} = r_{\mathrm{Po}} + r_{\mathrm{Pi}} \quad .$$

The outer planet-to-inner planet gear ratio $g_{\mathrm{oi}} = r_{\mathrm{Po}}/r_{\mathrm{Pi}} = N_{\mathrm{Po}}/N_{\mathrm{Pi}}$. N is the number of teeth on each gear. In terms of this ratio, the key kinematic constraint is:

$$(1 + g_{\mathrm{oi}})\omega_{\mathrm{C}} = \omega_{\mathrm{Pi}} + g_{\mathrm{oi}}\omega_{\mathrm{Po}} \quad .$$

The three degrees of freedom reduce to two independent degrees of freedom. The gear pair is (1,2) = (Pi,Po).

The torque transfer is:

$$g_{\mathrm{oi}}\tau_{\mathrm{Pi}} + \tau_{\mathrm{Po}} - \tau_{\mathrm{loss}} = 0 \quad ,$$

with $\tau_{\mathrm{loss}} = 0$ in the ideal case.

Nonideal Gear Constraints and Losses

In the nonideal case, $\tau_{loss} \neq 0$. See "Model Gears with Losses".

Limitations

- Gear inertia is assumed negligible.
- Gears are treated as rigid components.
- Coulomb friction slows down simulation. See "Adjust Model Fidelity".

Ports

Port	Description
C	Rotational conserving port representing the planet carrier
Po	Rotational conserving port representing the outer planet gear
Pi	Rotational conserving port representing the inner planet gear
H	Thermal conserving port for thermal modeling

Parameters

Main

Outer planet (Po) to inner planet (Pi) teeth ratio (NPo/NPi)

Ratio g_{oi} of the outer planet gear radius wheel to the inner planet gear wheel radius. This gear ratio must be strictly positive. The default is 2.

Meshing Losses

Parameters for meshing losses vary with the block variant chosen—one with a thermal port for thermal modeling and one without it.

Without Thermal Port

Friction model

Select how to implement friction losses from nonideal meshing of gear teeth. The default is `No meshing losses`.

- `No meshing losses — Suitable for HIL simulation` — Gear meshing is ideal.
- `Constant efficiency` — Transfer of torque between gear wheel pairs is reduced by a constant efficiency η satisfying $0 < \eta \le 1$. If you select this option, the panel changes from its default.

Constant Efficiency

Ordinary efficiency

Torque transfer efficiency η_{PP} for planet-planet gear wheel pair meshing.

Inner planet-carrier power threshold

Power threshold above which the full efficiency factor is in effect. Below this value, a hyperbolic tangent function smooths the efficiency factor, lowering the efficiency losses to zero when no power is transmitted. The default value is `0.001` W.

With Thermal Port

Temperature

Array of temperatures used to construct a 1-D temperature-efficiency lookup table. The array values must increase left to right. The temperature array must be the same size as the **Efficiency** array. The default array is `[280 300 320]` K.

Efficiency

Array of component efficiencies used to construct a 1-D temperature-efficiency lookup table. The array values are the efficiencies at the temperatures in the **Temperature** array. The two arrays must be the same size. The default array is `[0.95 0.9 0.85]`.

Inner planet-carrier power threshold

Power threshold above which the full efficiency factor is in effect. A hyperbolic tangent function smooths the efficiency factor between zero at rest and the value provided by the temperature-efficiency lookup table at the power threshold. The default value is `0.001` W.

Viscous Losses

Inner planet-carrier viscous friction coefficient

Viscous friction coefficient μ_{Pi} for the inner planet-carrier gear motion. The default is `0`.

From the drop-down list, choose units. The default is newton-meters/(radians/second) (`N*m/(rad/s)`).

Thermal Port

Thermal mass

Thermal energy required to change the component temperature by a single degree. The greater the thermal mass, the more resistant the component is to temperature change. The default value is 50 J/K.

Initial temperature

Component temperature at the start of simulation. The initial temperature alters the component efficiency according to an efficiency vector that you specify, affecting the starting meshing or friction losses. The default value is 300 K.

Real-Time Simulation

Hardware-in-the-Loop Simulation

For optimal simulation performance, use the **Meshing Losses** > **Friction model** parameter default setting, No meshing losses - Suitable for HIL simulation.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also

Simscape Blocks
Planetary Gear | Ravigneaux Gear | Ring-Planet | Sun-Planet | Sun-Planet Bevel

Topics
"Basic Motion, Torque, and Force Modeling"
"Model Thermal Losses in Driveline Components"

"Prepare Simscape Driveline Models for Real-Time Simulation Using Simscape Checks"

Rack & Pinion

Rack and pinion gear coupling translational and rotational motion, with adjustable pinion radius and friction losses

Library

Simscape / Driveline / Gears / Rotational-Translational

Description

The Rack & Pinion block represents rack and pinion gear that converts between translational and rotational motion. The rotational-translational gear constrains the pinion (P) and rack (R) to, respectively, rotate and translate together in a fixed ratio that you specify. You can choose whether the rack axis translates in a positive or negative direction, as the pinion rotates in a positive direction, by using the **Rack direction** parameter.

Thermal Modeling

You can model the effects of heat flow and temperature change through an optional thermal conserving port. By default, the thermal port is hidden. To expose the thermal port, right-click the block in your model and, from the context menu, select **Simscape** > **Block choices**. Select a variant that includes a thermal port. Specify the associated thermal parameters for the component.

Gear Model

Model Variables

R_{RP}	Rack-pinion gear ratio
ω_P	Angular velocity of the pinion shaft
v_R	Translational velocity of the rack
r_P	Effective radius of the pinion
N_P	Number of teeth on the pinion
x_R	Rack tooth spacing
τ_P	Pinion shaft torque
F_R	Rack force
F_{loss}	Total loss force
F_{Coul}	Friction force
η	Torque transfer efficiency
p_{th}	Power threshold
μ_P	Viscous friction coefficient for the pinion shaft
μ_R	Viscous friction coefficient for the rack motion

Ideal Gear Constraint and Gear Ratio

Rack & Pinion imposes one kinematic constraint on the two connected axes:

$$\omega_P = R_{RP}v_R .$$

The transmission ratio is:

$$R_{RP} = 1 / r_P = \omega_P / v_N = \pm 2\pi / N_P v_R .$$

The two degrees of freedom are reduced to one independent degree of freedom. The forward-transfer gear pair convention is (1,2) = (P,R).

The torque-force transfer is:

$$R_{RP}\tau_P + F_R - F_{loss} = 0 ,$$

with $F_{\text{loss}} = 0$ in the ideal case.

Nonideal Gear Constraint

In the nonideal case, $F_{\text{loss}} \neq 0$. For general considerations on nonideal gear modeling, see "Model Gears with Losses".

In a nonideal pinion-rack pair (P,R), the angular velocity and geometric constraints are unchanged. But the transferred torque, force, and power are reduced by:

- Coulomb friction between teeth surfaces on P and R, characterized by constant efficiency η
- Viscous coupling of driveshafts with bearings, parametrized by viscous friction coefficients μ

Meshing Efficiency

The efficiency η of meshing between pinion and rack is fully active only if the transmitted power is greater than the power threshold.

If the power is less than the threshold, the actual efficiency is automatically regularized to unity at zero velocity.

Efficiency is assumed equal for both the forward and reverse power flow.

Viscous Friction Force

The viscous friction coefficients μ_P and μ_R control the viscous friction torque and force experienced by the rack and pinion from lubricated, nonideal bearings. The viscous friction torque on the pinion axis is $-\mu_P \omega_P$. The viscous friction force on the rack motion is $-\mu_R v_R$.

Limitations

- Gear inertia is assumed negligible.
- Gears are treated as rigid components.
- Coulomb friction slows down simulation. See "Adjust Model Fidelity".

Ports

Port	Description
P	Rotational conserving port representing the pinion
R	Translational conserving port representing the rack
H	Thermal conserving port for modeling heat transfer

P is a rotational conserving port. R is a translational conserving port. They represent the pinion and the rack, respectively.

Parameters

Main

Parameterize by

Select how to parameterize the rack and pinion gear. The default is `Pinion radius`.

- `Pinion radius` — Gear ratio is defined by the effective radius of the pinion.

  Pinion radius

  Effective radius of the pinion r_P. Must be greater than zero. The default is `100`.

  From the drop-down list, choose units. The default is millimeters (`mm`).

- `Tooth parameters` — Gear ratio is defined by the number of teeth on the pinion gear and the rack tooth spacing. If you select this option, the panel changes from its default.

  `Tooth parameters`

  Number of pinion teeth

  Number of teeth on the pinion N_P. Must be greater than zero. The default is `20`.

Rack tooth spacing

Spacing between the teeth on the rack x_R. Must be greater than zero. The default is 5.

From the drop-down list, choose units. The default is millimeters (mm).

Rack direction

Choose whether the rack axis translates in a positive or negative direction when the pinion rotates in a positive direction. The default is `Positive for positive pinion rotation`.

Meshing Losses

Parameters for meshing and friction losses vary with the block variant chosen—one with a thermal port for thermal modeling and one without it.

Without Thermal Port

Friction model

Select how to implement friction losses from nonideal meshing of gear teeth. The default is `No meshing losses`.

- `No meshing losses — Suitable for HIL simulation` — Gear meshing is ideal.

- `Constant efficiency` — Transfer of torque and force between pinion and rack is reduced by constant efficiency $0 < \eta \leq 1$. If you select this option, the panel expands.

Constant Efficiency

Efficiency

Torque transfer efficiency η for the rack-pinion gear meshing, which is the same for the forward and reverse power flows. Must be greater than zero but less than, or equal to, one. The default is `0.8`.

Rack power threshold

Power threshold above which the full efficiency factor is in effect. Below this value, a hyperbolic tangent function smooths the efficiency factor, lowering the efficiency losses to zero when no power is transmitted. The default value is `0.001` W.

1-355

With Thermal Port

Temperature

Array of temperatures used to construct a 1-D temperature-efficiency lookup table. The array values must increase left to right. The temperature array must be the same size as the **Efficiency** array. The default array is `[280 300 320]` K.

Efficiency

Array of component efficiencies used to construct a 1-D temperature-efficiency lookup table. The array values are the efficiencies at the temperatures in the **Temperature** array. The two arrays must be the same size. The default array is `[0.95 0.9 0.85]`.

Rack power threshold

Power threshold above which the full efficiency factor is in effect. A hyperbolic tangent function smooths the efficiency factor between zero at rest and the value provided by the temperature-efficiency lookup table at the power threshold. The default value is `0.001` W.

Viscous Losses

Pinion rotational viscous friction coefficient

Viscous friction coefficient μ_P for the pinion shaft. The default is `0`.

From the drop-down list, choose units. The default is newton-meters/(radians/second) (`N*m/(rad/s)`).

Rack translational viscous friction coefficient

Viscous friction coefficient μ_R for the rack motion. The default is `0`.

From the drop-down list, choose units. The default is newton/(meters/second) (`N/(m/s)`).

Thermal Port

Thermal mass

Thermal energy required to change the component temperature by a single degree. The greater the thermal mass, the more resistant the component is to temperature change. The default value is `50` J/K.

Initial temperature

Component temperature at the start of simulation. The initial temperature alters the component efficiency according to an efficiency vector that you specify, affecting the starting meshing or friction losses. The default value is 300 K.

Real-Time Simulation

Hardware-in-the-Loop Simulation

For optimal simulation performance, use the **Meshing Losses** > **Friction model** parameter default setting, `No meshing losses - Suitable for HIL simulation`.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also
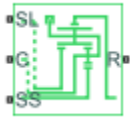
Simscape Blocks
Leadscrew | Simple Gear

Topics
"Basic Motion, Torque, and Force Modeling"
"Model Thermal Losses in Driveline Components"
"Prepare Simscape Driveline Models for Real-Time Simulation Using Simscape Checks"

Ravigneaux Gear

Planetary gear with two sun gears and two planet gear sets


## Library

Simscape / Driveline / Gears

Description

This block represents a planetary gear train with dual sun and planet gear sets. The two sun gears are centrally located and separated longitudinally along a common rotation axis. The smaller of these gears engages an inner planet gear set, which in turn engages an outer planet gear set. The outer planet gear set, whose length spans the distance between the two sun gears, engages both the larger sun gear and the ring gear.

A carrier holds the planet gear sets in place at different radii. The carrier, which rigidly connects to a drive shaft, can spin as a unit with respect to the sun and ring gears. Revolute joints, each located between a planet gear and the carrier, enable the gears to spin about their individual longitudinal axes.

The relative angular velocities of the sun, planet, and ring gears follow from the kinematic constraints between them. For more information, see "Ravigneaux Gear Model" on page 1-361.
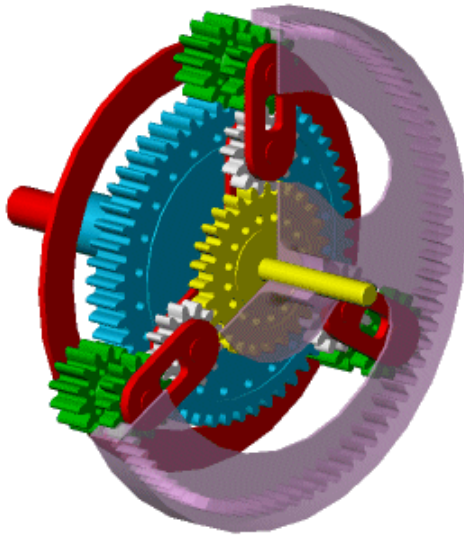
The block models the Ravigneaux gear as a structural component based on Sun-Plant, Planet-Planet, and Ring-Planet Simscape Driveline blocks. The figure shows the block diagram of this structural component.
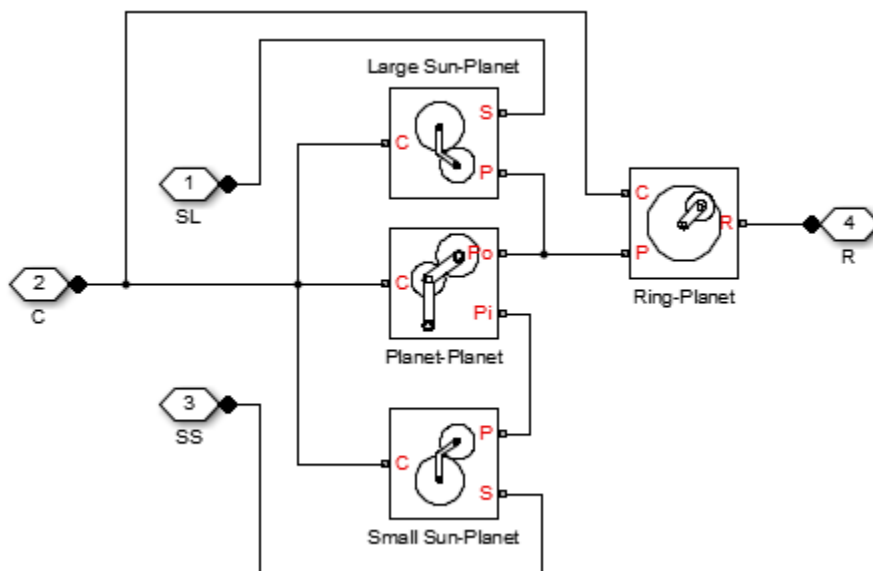
To increase the fidelity of the gear model, you can specify properties such as gear inertia, meshing losses, and viscous losses. By default, gear inertia and viscous losses are assumed negligible. The block enables you to specify the inertias of the internal planet gears only. To model the inertias of the carrier, large sun, small sun, and ring gears, connect Simscape Inertia blocks to ports C, SL, SS, and R.

Thermal Modeling

You can model the effects of heat flow and temperature change through an optional thermal conserving port. By default, the thermal port is hidden. To expose the thermal port, right-click the block in your model and, from the context menu, select **Simscape > Block choices**. Select a variant that includes a thermal port. Specify the associated thermal parameters for the component.

Ravigneaux Gear Model

Ideal Gear Constraints and Gear Ratios

Ravigneaux imposes four kinematic and four geometric constraints on the four connected axes and the two internal wheels (inner and outer planets):

$$r_{Ci}\omega_C = r_{SS}\omega_{SS} + r_{Pi}\omega_{Pi} \quad , \quad r_{Ci} = r_{SS} + r_{Pi} \quad ,$$

$$r_{Co}\omega_C = r_{SL}\omega_{SL} + r_{Po}\omega_{Po} \quad , \quad r_{Co} = r_{SL} + r_{Po} \quad ,$$

$$(r_{Co} - r_{Ci})\omega_C = r_{Pi}\omega_{Pi} + r_{Po}\omega_{Po} \quad , \quad r_{Co} - r_{Ci} = r_{Po} + r_{Pi} \quad ,$$

$$r_R\omega_R = r_{Co}\omega_C + r_{Po}\omega_{Po} \quad , \quad r_R = r_{Co} + r_{Po} \quad .$$

The ring-small sun ratio $g_{RSS} = r_R/r_{SS} = N_R/N_{SS}$ and ring-large sun gear ratio $g_{RSL} = r_R/r_{SL} = N_R/N_{SL}$. N is the number of teeth on each gear. In terms of these ratios, the key kinematic constraints are:

$$(g_{RSS} - 1)\omega_C = g_{RSS}\omega_R - \omega_{SS} \quad ,$$

$$(g_{RSL} + 1)\omega_C = g_{RSL}\omega_R + \omega_{SL} \quad .$$

The six degrees of freedom reduce to two independent degrees of freedom. The gear pairs are (1,2) = (LS,P), (SS,P), (P,R), and (P,P).

Warning The gear ratio g_{RSS} must be strictly greater than the gear ratio g_{RSL}. The gear ratio g_{RSL} must be strictly greater than one.

The torque transfers are:

$$g_{RSS}\tau_{SS} + \tau_R - \tau_{loss}(SS,R) = 0 \quad , \quad g_{RSL}\tau_{SL} + \tau_R - \tau_{loss}(SL,R) = 0 \quad ,$$

with $\tau_{loss} = 0$ in the ideal case.

Nonideal Gear Constraints and Losses

In the nonideal case, $\tau_{loss} \neq 0$. See "Model Gears with Losses".

Limitations

- Gears are assumed rigid.
- Coulomb friction slows down simulation. See "Adjust Model Fidelity".

Ports

Port	Description
C	Rotational conserving port representing the planet gear carrier
R	Rotational conserving port representing the ring gear
SL	Rotational conserving port representing the large sun gear
SS	Rotational conserving port representing the small sun gear
H	Thermal conserving port for thermal modeling

Parameters

Main

Ring (R) to large sun (SL) teeth ratio (NR/NSL)

Ratio g_{RSL} of the ring gear wheel radius to the large sun gear wheel radius. This gear ratio must be strictly greater than 1. The default is 2.

Ring (R) to small sun (SS) teeth ratio (NR/NSS)

Ratio g_{RSS} of the ring gear wheel radius to the small sun gear wheel radius. This gear ratio must be strictly greater than the ring-large sun gear ratio. The default is 3.

Meshing Losses

Parameters for meshing losses vary with the block variant chosen—one with a thermal port for thermal modeling and one without it.

Without Thermal Port

Friction model

Select how to implement friction losses from nonideal meshing of gear teeth. The default is No meshing losses.

- No meshing losses — Suitable for HIL simulation — Gear meshing is ideal.
- Constant efficiency — Transfer of torque between gear wheel pairs is reduced by a constant efficiency η satisfying $0 < \eta \leq 1$. If you select this option, the panel changes from its default.

Constant Efficiency

Large sun-planet, small sun-planet, ring-planet, and planet-planet ordinary efficiencies

Vector of torque transfer efficiencies [η_{LSP} η_{SSP} η_{RP} η_{PP}] for large sun-planet, small sun-planet, ring-planet, and planet-planets gear wheel pair meshings, respectively.

Large sun-carrier, small sun-carrier, large sun planet-carrier, and small sun planet-carrier power thresholds

Vector of power thresholds p_{th} above which full efficiency loss is applied, for large sun-carrier, small sun-carrier, large sun planet-carrier, and small sun planet-carrier velocities, respectively. The default is [0.001 0.001 0.001 0.001] W.

With Thermal Port

Temperature

Array of temperatures used to construct a 1-D temperature-efficiency lookup table. The array values must increase left to right. The temperature array must be the same size as each efficiency array. The default array is [280 300 320] K.

Large sun-planet efficiency

Array of mechanical efficiencies with power flowing from the large sun gear to the large sun planet gears. Each array value is the ratio of output power to input power at

1-363

one of the temperatures in the temperature array. The temperature and efficiency arrays must be the same size. The default array is `[0.95 0.9 0.85]`.

Small sun-planet efficiency

Array of mechanical efficiencies with power flowing from the small sun gear to the small sun planet gears. Each array value is the ratio of output power to input power at one of the temperatures in the temperature array. The temperature and efficiency arrays must be the same size. The default array is `[0.95 0.9 0.85]`.

Ring-planet efficiency

Array of mechanical efficiencies with power flowing from the ring gear to the large sun planet gears. Each array value is the ratio of output power to input power at one of the temperatures in the temperature array. The temperature and efficiency arrays must be the same size. The default array is `[0.95 0.9 0.85]`.

Planet-planet efficiency

Array of mechanical efficiencies with power flowing from the small sun planet gears to the large sun planet gears. Each array value is the ratio of output power to input power at one of the temperatures in the temperature array. The temperature and efficiency arrays must be the same size. The default array is `[0.95 0.9 0.85]`.

Large sun-carrier, small sun-carrier, large sun planet-carrier and small sun planet-carrier power thresholds

Array of power thresholds above which the full efficiency factors apply. Enter the thresholds for the large sun gear, small sun gear, large sun planet gears, and small sun planet gears, all relative to the gear carrier, in this order. A hyperbolic tangent function smooths the efficiency factors between zero at rest and the values provided by the temperature-efficiency lookup tables at the power thresholds. The default array is `[0.001 0.001 0.001 0.001]` W.

Viscous Losses

Large sun-carrier, small sun-carrier, large sun planet-carrier, and small sun planet-carrier viscous friction coefficients

Vector of viscous friction coefficients $[\mu_{LS} \ \mu_{SS} \ \mu_{LSP} \ \mu_{SSP}]$ for the large sun-carrier, small sun-carrier, large sun planet-carrier, and small sun planet-carrier gear motions, respectively. The default is `[0 0 0 0]`.

From the drop-down list, choose units. The default is newton-meters/(radians/second) (`N*m/(rad/s)`).

Inertia

Inner planet gear inertia

Moment of inertia of the planet gear carrier. This value must be positive or zero. Enter `0` to ignore carrier inertia. The default value is `0` kg*m^2.

Outer planet gear inertia

Moment of inertia of the combined planet gears. This value must be positive or zero. Enter `0` to ignore gear inertia. The default value is `0` kg*m^2.

Thermal Port

Thermal mass

Thermal energy required to change the component temperature by a single degree. The greater the thermal mass, the more resistant the component is to temperature change. The default value is `50` J/K.

Initial temperature

Component temperature at the start of simulation. The initial temperature alters the component efficiency according to an efficiency vector that you specify, affecting the starting meshing or friction losses. The default value is `300` K.

Real-Time Simulation

Hardware-in-the-Loop Simulation

For optimal simulation performance, use the **Meshing Losses** > **Friction model** parameter default setting, `No meshing losses - Suitable for HIL simulation`.

Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using MATLAB® Coder™.

See Also

Simscape Blocks

Compound Planetary Gear | Planet-Planet | Planetary Gear | Ring-Planet | Sun-Planet
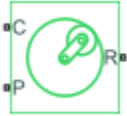
Topics

"Basic Motion, Torque, and Force Modeling"
"Ravigneaux Four-Speed Transmission"
"Model Thermal Losses in Driveline Components"
"Prepare Simscape Driveline Models for Real-Time Simulation Using Simscape Checks"

Ring-Planet

Planetary gear set of carrier, planet, and ring wheels with adjustable gear ratio and friction losses

Library

Simscape / Driveline / Gears / Planetary Subcomponents

Description

The Ring-Planet gear block represents a set of carrier, planet, and ring gear wheels. The planet is connected to and rotates with respect to the carrier. The planet and ring corotate with a fixed gear ratio that you specify. A ring-planet and a sun-planet gear are basic elements of a planetary gear set. For model details, see "Ring-Planet Gear Model" on page 1-368.

Thermal Modeling

You can model the effects of heat flow and temperature change through an optional thermal conserving port. By default, the thermal port is hidden. To expose the thermal port, right-click the block in your model and, from the context menu, select **Simscape** > **Block choices**. Select a variant that includes a thermal port. Specify the associated thermal parameters for the component.

Ring-Planet Gear Model

Ideal Gear Constraints and Gear Ratios

Ring-Planet imposes one kinematic and one geometric constraint on the three connected axes:

$$r_R\omega_R \quad = \quad r_C\omega_C \quad + \quad r_P\omega_P \quad , \quad r_R \quad = \quad r_C \quad + \quad r_P \quad .$$

The ring-planet gear ratio $g_{RP} = r_R/r_P = N_R/N_P$. N is the number of teeth on each gear. In terms of this ratio, the key kinematic constraint is:

$$g_{RP}\omega_R \qquad = \qquad \omega_P \qquad + \qquad (g_{RP} \qquad - \qquad 1)\omega_C \qquad .$$

The three degrees of freedom reduce to two independent degrees of freedom. The gear pair is $(1,2) = (P,R)$.

Warning The ring-planet gear ratio g_{RP} must be strictly greater than one.

The torque transfer is:

$$g_{RP}\tau_P \qquad + \qquad \tau_R \qquad - \qquad \tau_{loss} \qquad = \qquad 0 \qquad ,$$

with $\tau_{loss} = 0$ in the ideal case.

Nonideal Gear Constraints and Losses

In the nonideal case, $\tau_{loss} \neq 0$. See "Model Gears with Losses".

Limitations

- Gear inertia is assumed negligible.
- Gears are treated as rigid components.
- Coulomb friction slows down simulation. See "Adjust Model Fidelity".

Ports

Port	Description
C	Rotational conserving port representing the planet carrier
P	Rotational conserving port representing the planet gear
R	Rotational conserving port representing the ring gear

Port	Description
H	Thermal conserving port for thermal modeling

Parameters

Main

Ring (R) to planet (P) teeth ratio (NR/NP)

Ratio g_{RP} of the ring gear wheel radius to the planet gear wheel radius. This gear ratio must be strictly greater than 1. The default value is 2.

Meshing Losses

Parameters for meshing losses vary with the block variant chosen—one with a thermal port for thermal modeling and one without it.

Without Thermal Port

Friction model

Select how to implement friction losses from nonideal meshing of gear teeth. The default is `No meshing losses`.

- `No meshing losses — Suitable for HIL simulation` — Gear meshing is ideal.
- `Constant efficiency` — Transfer of torque between gear wheel pairs is reduced by a constant efficiency η satisfying $0 < \eta \leq 1$. If you select this option, the panel changes from its default.

  Constant Efficiency

  ### Ordinary efficiency

  Torque transfer efficiency η_{RP} for ring-planet gear wheel pair meshing.

  ### Planet-carrier power threshold

  Power threshold above which the full efficiency factor is in effect. Below this value, a hyperbolic tangent function smooths the efficiency factor, lowering the

efficiency losses to zero when no power is transmitted. The default value is
`0.001` W.

With Thermal Port

Temperature

Array of temperatures used to construct a 1-D temperature-efficiency lookup table.
The array values must increase left to right. The temperature array must be the same
size as the **Efficiency** array. The default array is `[280 300 320]` K.

Efficiency

Array of component efficiencies used to construct a 1-D temperature-efficiency lookup
table. The array values are the efficiencies at the temperatures in the **Temperature**
array. The two arrays must be the same size. The default array is `[0.95 0.9 0.85]`.

Planet-carrier power threshold

Power threshold above which the full efficiency factor is in effect. A hyperbolic
tangent function smooths the efficiency factor between zero at rest and the value
provided by the temperature-efficiency lookup table at the power threshold. The
default value is `0.001` W.

Viscous Losses

Planet-carrier viscous friction coefficient

Viscous friction coefficient μ_P for the planet-carrier gear motion. The default is `0`.

From the drop-down list, choose units. The default is newton-meters/(radians/second)
(`N*m/(rad/s)`).

Thermal Port

Thermal mass

Thermal energy required to change the component temperature by a single degree.
The greater the thermal mass, the more resistant the component is to temperature
change. The default value is `50` J/K.

Initial temperature

Component temperature at the start of simulation. The initial temperature alters the
component efficiency according to an efficiency vector that you specify, affecting the
starting meshing or friction losses. The default value is `300` K.

Real-Time Simulation

Hardware-in-the-Loop Simulation

For optimal simulation performance, use the **Meshing Losses** > **Friction model** parameter default setting, `No meshing losses - Suitable for HIL simulation`.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also

Simscape Blocks
Compound Planetary Gear | Planet-Planet | Planetary Gear | Sun-Planet | Sun-Planet Bevel

Topics
"Basic Motion, Torque, and Force Modeling"
"Adjust Model Fidelity"
"Epicyclic Gear Efficiency Measurement"
"Model Thermal Losses in Driveline Components"
"Prepare Simscape Driveline Models for Real-Time Simulation Using Simscape Checks"

Road Profile

Road profile with varying elevation and friction properties
Library: Simscape / Driveline / Tires & Vehicles

Description

The Road Profile block represents a road profile with varying elevation and friction properties. As a vehicle travels, the axle parameters and the position of the center of gravity (CG) determine the position of the front and rear axles. The block uses the axle positions to compute vehicle angle and, optionally, tire friction coefficients. The block models friction using either Magic Formula or kinetic and static friction coefficients.

Variables

Use the **Variables** tab to set the priority and initial target values for the block variables before simulating. For more information, see "Set Priority and Initial Target for Block Variables" (Simscape).

Unlike block parameters, variables do not have conditional visibility. The **Variables** tab lists all the existing block variables. If a variable is not used in the set of equations corresponding to the selected block configuration, the values specified for this variable are ignored.

Ports

Conserving

H — Horizontal motion
mechanical translational

Conserving port associated with the horizontal motion. Connect to the **H** port of the Vehicle Body block.

Output

MF — Front axle friction coefficients
vector

Physical signal output port for the friction coefficients of the front axle.

The vector contains four elements for the Magic Formula friction model or two elements for the friction coefficients friction model. Connect to the **M** port of a front axle Tire (Friction Parameterized) or Tire (Magic Formula) block.

This port is visible only when the **Friction output** parameter is set to one of these values:

- **Physical signal Magic Formula coefficients**
- **Physical signal friction coefficients**

MR — Rear axle friction coefficients
vector

Physical signal output port for the friction coefficients of the rear axle.

The vector contains four elements for the Magic Formula friction model or two elements for the friction coefficients friction model. Connect to the **M** port of a rear axle Tire (Friction Parameterized) or Tire (Magic Formula) block.

Dependencies

This port is visible only when the **Friction** parameter **Friction output** is set to one of these values:

- **Physical signal Magic Formula coefficients**
- **Physical signal friction coefficients**

beta — Vehicle angle
scalar

Physical signal output port for the vehicle angle.

Parameters

Main

Horizontal distance from CG to front axle — Distance between CG and front axle
`1.4` m (default) | positive scalar

Horizontal distance from the center of gravity (CG) to the front axle. Specify the same value as is specified for the **Horizontal distance from CG to front axle** parameter of the connected Vehicle Body block.

### Horizontal distance from CG to rear axle — Distance between CG and rear axle
`1.6` m (default) | positive scalar

Horizontal distance from the center of gravity (CG) to the rear axle. Specify the same value as is specified for the **Horizontal distance from CG to rear axle** parameter of the connected Vehicle Body block.

Horizontal distance for vertical profile — Vertical profile horizontal distance
`[-10, 0, 10]` m (default) | vector

Table lookup corresponding to the distance that the CG of the vehicle moves relative to the reference frame x-axis. For example the figure shows a horizontal profile of [-10, 0, 10]. Table lookup extrapolation is nearest.
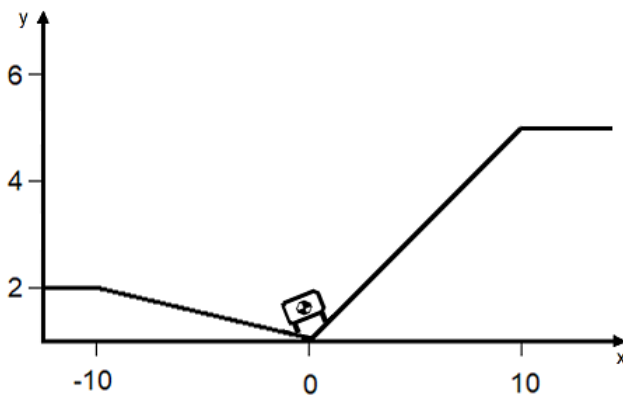
Vertical profile — Vertical profile
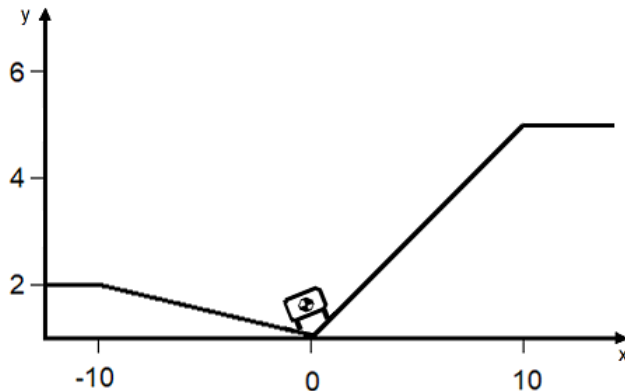`[0, 0, .25]` m (default) | vector

Table lookup corresponding to the distance that the CG of the vehicle moves relative to the reference frame *y*-axis. For example the figure shows a vertical profile of [2, 0, 5]. Table lookup extrapolation is nearest.

Friction

Friction output — Friction coefficient output type
None (default) | Physical signal Magic Formula coefficientsPhysical signal friction coefficients

Type of friction coefficient that the block outputs. Select `Physical signal Magic Formula coefficients` to vary friction for Tire (Magic Formula) blocks. Select `Physical signal friction coefficients` to vary friction for Tire (Friction Parameterized) blocks.

Dependencies

Setting this parameter to **Physical signal Magic Formula coefficients** or **Physical signal friction coefficients** makes the **MF** and **MR** ports and associated **Friction** parameters visible.

Horizontal distance for friction profile — Horizontal distance for friction profile
`[-10, 2, 2.1, 6, 6.1, 10]` m (default) | vector

Friction profile horizontal distance.

Dependencies

This parameter is visible only when the **Friction** parameter **Friction output** is set to one of these values:

- **Physical signal Magic Formula coefficients**
- **Physical signal friction coefficients**

Magic Formula coefficients for front axle — Front axle Magic Formula coefficients
[10, 2, 1, 1; 10, 2, 1, 1; 8, 3, 1, 1; 8, 3, 1, 1; 9, 4, 1, 1; 9, 4, 1, 1] (default) | matrix

Magic Formula coefficients specified as an *n*-by-*4* matrix, where *n* is the number of points in the vector specified by the **Horizontal distance for friction profile** parameter.

Dependencies

This parameter is visible only when the **Friction** parameter **Friction output** is set to **Physical signal Magic Formula coefficients**.

Magic Formula coefficients for rear axle — Rear axle Magic Formula coefficients
[10, 2, 1, 1; 10, 2, 1, 1; 8, 3, 1, 1; 8, 3, 1, 1; 9, 4, 1, 1; 9, 4, 1, 1] (default) | matrix

Magic Formula coefficients specified as an *n*-by-*4* matrix, where *n* is the number of points in the vector specified by the **Horizontal distance for friction profile** parameter.

Dependencies

This parameter is visible only when the **Friction** parameter **Friction output** is set to **Physical signal Magic Formula coefficients**.

Static and kinetic friction coefficients for front axle — Front axle friction coefficients
[.9, .7; .9, .7; .7, .5; .7, .5; .8, .6; .8, .6] (default) | matrix

Front axle static and kinetic friction coefficients specified as an *n*-by-*2* matrix, where *n* is the number of points in the vector specified by the **Horizontal distance for friction profile** parameter.

Dependencies

This parameter is visible only when the **Friction** parameter **Friction output** is set to **Physical signal friction coefficients**.

`Static and kinetic friction coefficients for rear axle — Rear axle friction coefficients`
`[.9, .7; .9, .7; .7, .5; .7, .5; .8, .6; .8, .6]` (default) | matrix

Rear axle static and kinetic friction coefficients specified as an *n*-by-*2* matrix, where *n* is the number of points in the vector specified by the **Horizontal distance for friction profile** parameter.

Dependencies

This parameter is visible only when the **Friction** parameter **Friction output** is set to **Physical signal friction coefficients**.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also
Tire (Friction Parameterized) | Tire (Magic Formula) | Tire-Road Interaction (Magic Formula) | Vehicle Body

Topics
"Model a Road Profile with Varying Elevation and Friction"

Introduced in R2019a

Rod

Axially flexible bar or cable
Library: Simscape / Driveline / Couplings & Drives

Description

The Rod block represents an axially flexible bar or cable in tension or compression.

Tension

Compression
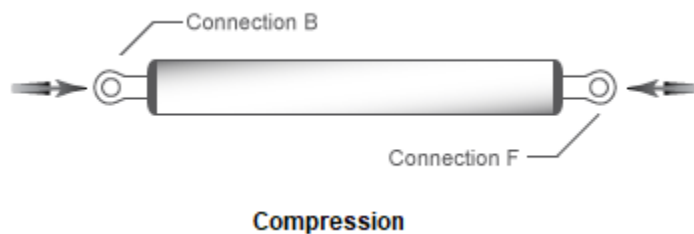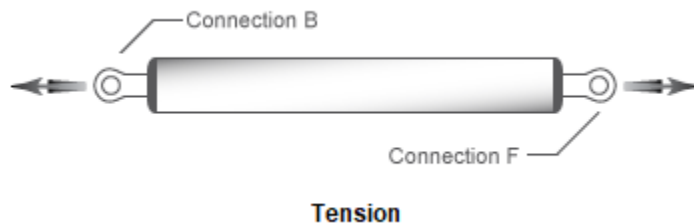
Model

To represent the bar or cable, the block uses a lumped parameter model. The model is composed of $N+1$ lumped masses that are connected in series by N sets of parallel-connected spring and damper circuits. The spring represents elasticity. The damper represents material damping.
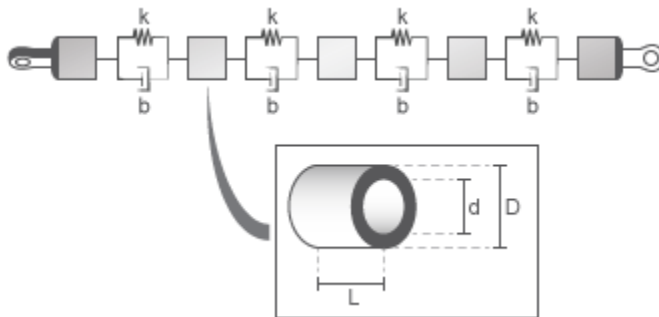
A single segment model exhibits an eigenfrequency that is close to the first eigenfrequency of the distributed parameter model. For more accurate analysis, select 2, 4, 8, or more segments.

The equivalent physical network contains *N* spring and damper circuits and *N*+1 mass blocks. The total mass of the rod is distributed evenly over the mass blocks. The stiffness of the spring in each spring and damper circuit is equal to the *N* times the stiffness of the rod material.
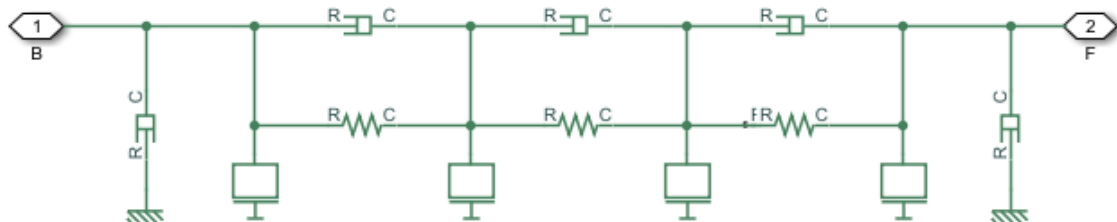
Equations

The defining equations for the model are:

$$\eta = 2\zeta\sqrt{\frac{K \cdot m}{2}}$$

$$A = \frac{\pi(D^2 - d^2)}{4}$$

$$K = \frac{E \cdot A}{L}$$

$$m = A \cdot L \cdot \rho$$

Where:

- η is the damping of the rod.
- ς is the damping ratio of the rod material.
- K is the stiffness of the rod.
- A is the cross-sectional area of the rod.
- D is the outer diameter of the rod.
- d is the inner diameter of the rod, where

    - $d = 0$ for a solid rod.
    - $d > 0$ for an annular, that is, hollow, rod.
- E is Young's modulus, that is, the modulus of elasticity, of the rod material.
- L is the length of the rod.
- m is the mass of the rod.
- ρ is the density of the rod material.

Assumptions and Limitations

- The rod does not buckle, in the case of a bar, or go slack, in the case of a cable, in tension.
- The rod has a constant cross-section along its length.
- The distributed parameter model is approximated as a finite number of segments, N.

Ports

Conserving

B — Base
translational mechanical

Translational conserving port associated with rod base or input.

F — Follower
translational mechanical

Translational conserving port associated with rod follower or output.

Parameters

Rod

The table shows how the visibility of some **Rod** parameters depends on choices that you make for other **Rod** parameters. To learn how to read the table, see "Parameter Dependencies" on page B-2.

Rod Parameter Dependencies Table

Rod		
Parameterization — Choose By `stiffness and inertia` or By `material properties`		
By stiffness and inertia	**By material properties**	
Stiffness	Geometry — Choose `Solid` or `Annular`	
Mass	**Solid**	**Annular**
	Length	
	Outer diameter	
	-	Inner diameter
	Density	
	Young's modulus	
Damping ratio		
Number of segments		

Parameterization — Parameterization method
By `stiffness and inertia` (default) | By `material properties`

Parameterization method.

Each **Parameterization** option exposes other parameters. For more information, see Rod Parameter Dependencies Table.

Stiffness — Material stiffness
`5e8 N/m` (default) | non-negative scalar

Material stiffness.

Setting the **Parameterization** parameter to By `stiffness and inertia` exposes this parameter. For more information, see Rod Parameter Dependencies Table.

Mass — Rod mass
`20 kg` (default) | positive scalar

Rod mass.

Dependencies

Setting the **Parameterization** parameter to By stiffness and inertia exposes this parameter. For more information, see Rod Parameter Dependencies Table.

Geometry — Rod cross-section
Solid (default) | Annular

Rod cross-sectional geometry.

Dependencies

Setting the **Parameterization** parameter to By material properties exposes this parameter. Each **Geometry** option exposes other parameters. For more information, see Rod Parameter Dependencies Table.

Length — Rod length
1 m (default) | positive scalar

Rod length.

Dependencies

Setting the **Parameterization** parameter to By material properties exposes this parameter. For more information, see Rod Parameter Dependencies Table.

Outer diameter — Rod outer diameter
0.075 m (default) | positive scalar

Rod outer diameter.

Dependencies

Setting the **Parameterization** parameter to By material properties exposes this parameter. For more information, see Rod Parameter Dependencies Table.

Inner diameter — Rod inner diameter
0.050 m (default) | positive scalar

Rod inner diameter. If the rod is solid, specify 0.

Dependencies

Setting the **Parameterization** parameter to `By material properties` and the **Geometry** parameter to `Annular` exposes this parameter. For more information, see Rod Parameter Dependencies Table.

Density — Material density
7.8e3 kg/m^3 (default) | positive scalar

Material density.

Dependencies

Setting the **Parameterization** parameter to `By material properties` exposes this parameter. For more information, see Rod Parameter Dependencies Table.

Young's modulus — Young's modulus
200e9 Pa (default) | positive scalar

Young's modulus for the rod material.

Dependencies

Setting the **Parameterization** parameter to `By material properties` exposes this parameter. For more information, see Rod Parameter Dependencies Table.

Damping ratio — Material damping ratio
0.01 (default) | non-negative scalar

Material damping ratio.

Number of segments — *N*
1 (default) | positive integer scalar

Number of segments, *N*, for the approximation.

A larger number of segments, *N*, increases the accuracy of the model but reduces simulation performance, that is, simulation speed. The single-segmented model (*N*=1) exhibits an eigenfrequency that is close to the first eigenfrequency of the continuous, distributed parameter model.

If accuracy is more important than performance, select 2, 4, 8, or more segments. For example, the four lowest eigenfrequencies are represented with an accuracy of 0.1, 1.9, 1.6, and 5.3 percent, respectively, by a 16-segmented model.

Viscous Losses

Viscous friction coefficients at base (B) and follower (F) — Viscous friction coefficients
[0, 0] N/(m/s) (default) | non-negative vector

Viscous friction coefficients at base port, **B**, and the follower port, **F**.

Initial Conditions

Initial deflection — Initial deflection
0 m (default) | non-negative scalar

Deflection of the rod at the start of simulation.

A positive initial deflection results in a positive translation of the base, **B**, end of the rod relative to the follower, **F**, end of the rod.

Initial velocity — Initial velocity
0 m/s (default) | non-negative scalar

Longitudinal velocity of the base, **B**, end of the rod relative to the follower, **F**, end of the rod at the start of simulation.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using Simulink® Coder™.

See Also
Flexible Shaft | Mass | Translational Damper | Translational Spring

Introduced in R2018b

Rolling Resistance

Model rolling resistance

Library

Tires & Vehicles/Tire Subcomponents

Description

The block models the resistance force that acts on the wheel hub due to the rolling resistance at the road-wheel contact surface. The model can use a constant resistance coefficient or the pressure and velocity dependence of the SAE J2452 standard. The resistance force is zero when the normal force acting at the wheel-road surface is less than or equal to zero.

Constant Resistance Coefficient Model

In the constant coefficient model, the rolling resistance is directly proportional to the resistance coefficient:

$F = N\mu$

where parameters represent the following quantities:

- F — Rolling resistance force
- N — Normal Force
- μ — Rolling resistance coefficient

The rolling resistance coefficient has a hyperbolic form that eliminates discontinuity at $v_{hub} = 0$:

$\mu = \mu_0 \tanh(4v_{hub}/v_{threshold})$

where parameters represent the following quantities:

- μ_0 — Asymptotic rolling resistance coefficient
- v_{hub} — Hub velocity
- $v_{threshold}$ — Threshold velocity

Pressure and Velocity Dependent Model

The pressure- and velocity-dependent model uses the following formula:

$$F = \left(\frac{P}{P_0}\right)^{\alpha}\left(\frac{N}{N_0}\right)^{\beta} N_0 \cdot (A + B|v_{hub}| + Cv_{hub}^2)$$

where parameters represent the following quantities:

- P — Tire pressure
- v_{hub} — Hub velocity
- α, β, A, B, C — Approximating coefficients
- P_0 — 1 Pascal (Pa)
- N_0 — 1 Newton (N)

In the previous equation, parameters P_0 and N_0 remove the physical units from each exponential expression base.

Connection H is a mechanical translational conserving port that represents the hub of the tire. Connection N is a physical signal input port that represents the normal force acting on the tire. Normal force is positive if it points downward.

Parameters

Resistance Model

Select the model used to compute the rolling resistance on a wheel hub. The parameter provides two options:

- Constant coefficient
- Pressure and velocity dependent

The default value is Constant coefficient.

Constant Coefficient

Selecting the Constant coefficient option exposes two model parameters: **Constant coefficient** and **Velocity threshold**.

Constant Coefficient

Coefficient that sets the proportionality between the normal force and the rolling resistance force. The parameter must be greater than zero. The default value is 0.015.

Velocity Threshold

Velocity at which the full rolling resistance force is transmitted to the rolling hub. The parameter ensures the force remains continuous during velocity direction changes, which increases the numerical stability of the simulation. The parameter must be greater than zero. The default value is 0.001 m/s.

Pressure and Velocity Dependent

Tire pressure

Inflation pressure of the tire. The parameter must be greater than zero. The default value is 250e+3 Pa.

Alpha

Exponent of the tire pressure in the model equation. See "Pressure and Velocity Dependent Model" on page 1-388. The default value is -0.003.

Beta

Exponent of the normal force model equation. The default value is 0.97.

Coefficient A

Velocity independent force component in the model equation. The parameter must be greater than zero. The default value is 84e-4.

Coefficient B

Velocity dependent force component in the model equation. The parameter must be greater than zero. The default value is 6.2e-4 s/m.

Coefficient C

Force component that depends on the square of the velocity term in the model equation. The parameter must be greater than zero. The default value is `1.6e-4 s^2/m^2`.

Velocity Threshold

Velocity at which the full rolling resistance force is transmitted to the rolling hub. The parameter ensures the force remains continuous during velocity direction changes, which increases the numerical stability of the simulation. The parameter must be greater than zero. The default value is `1e-3 m/s`.

Ports

Port	Description
N	Physical signal input port that represents the normal force
H	Conserving translational port that represents the wheel hub

Extended Capabilities

C/C++ Code Generation

Generate C and C++ code using MATLAB® Coder™.

See Also

Tire (Friction Parameterized) | Tire (Magic Formula) | Tire-Road Interaction (Magic Formula)

Rope Drum

Power transmission system with tightly wound rope around cylindrical drum
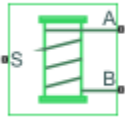
Library

Simscape / Driveline / Couplings & Drives

Description

This block represents a power transmission system with two key components:

- Rope
- Drum

The rope is tightly wound around the cylindrical drum, at a sufficient tension level to prevent slipping between the two components. The rope can be wound such that its ends point in equal or opposite directions. Depending on the rope configuration, the two rope ends can move in equal or opposite directions. The rope windup types that you specify in the block dialog box are:

- If the rope ends point in the same direction, they translate in opposite directions.
- If the rope ends point in opposite directions, they translate in the same direction.

Rope ends move
in opposite directions

Rope ends move
in the same direction
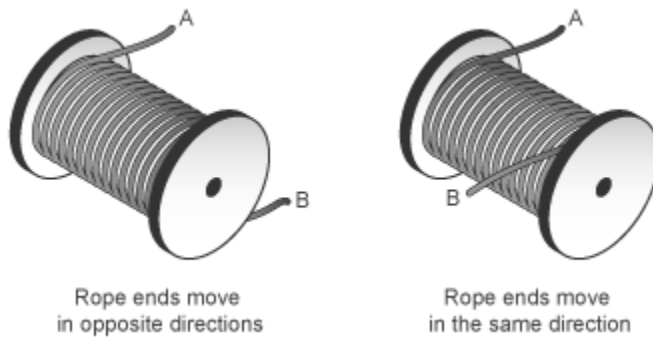
During normal operation, a driving torque causes the cylindrical drum to rotate about its symmetry axis. The rotating drum transmits a tensile force to the rope ends, which translate with respect to the drum centerline. The relative direction of translation between the two rope ends depends on the rope windup type. However, a positive drum rotation always corresponds to a positive translation at rope end A.

The rope drum model assumes that each rope end remains taut during simulation. This assumption is important since slack rope ends do not transmit force. An optional tension warning indicates the failure of this assumption. The warning, which appears at the MATLAB command prompt, prompts careful analysis of the simulation results.

Optional effects in the rope drum model include:

- Drum inertia
- Viscous friction at the drum bearing

The net torque acting on the cylinder satisfies the force balance equation

$$T = F_B \cdot R \cdot \delta - F_A \cdot R + \mu \cdot \omega,$$

where:

- T is the net torque acting on the drum.
- F_A and F_B are the external forces pulling on rope ends A and B.
- R is the drum radius.
- μ is the viscous friction coefficient at the drum bearings.
- ω is the drum angular velocity.

- δ is the rope windup type according to the table.

Rope Windup Type	δ
Ends move in the same direction	-1
Ends move in opposite directions	+1

The figure shows the equation variables.
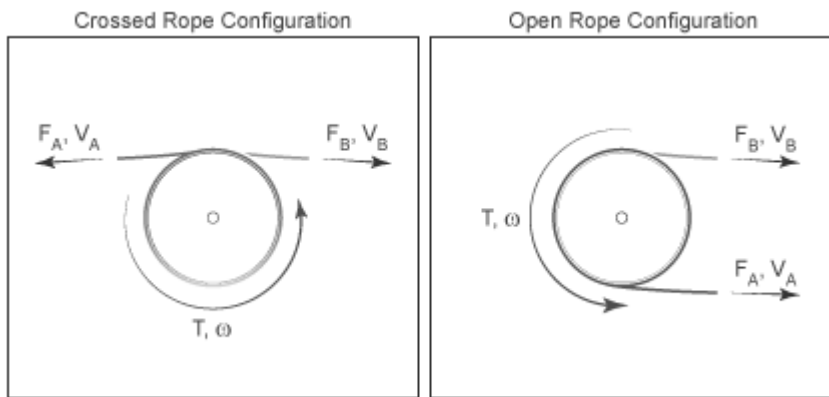
The translational velocities of the two rope ends are functions of the drum radius and angular velocity. Each translational velocity is equal to the tangential velocity of a point at the drum periphery according to the expressions:

$$V_A = -\omega \cdot R,$$
$$V_B = +\omega \cdot R,$$

where:

- V_A is the translational velocity of rope end A.
- V_B is the translational velocity of rope end B.

Assumptions and Limitations

- Slip does not occur at the rope-drum contact surface.

- Rope compliance is ignored.

Ports

Port	Description
S	Conserving rotational port associated with the drum shaft
A	Conserving translational port associated with rope end A
B	Conserving translational port associated with rope end B

Parameters

Rope Winding

Drum radius

Distance between the drum center and its periphery. The drum periphery coincides with the contact surface between the drum and the rope. The default value is `0.1` m.

Rope windup type

Relative direction of the translation motion of the two rope ends, A and B. Options include:

- `Ends move in the same direction`
- `Ends move in opposite direction`

Tension warning

Optional warning at the MATLAB command line indicating that at least one rope end has become slack. A rope end becomes slack when the net tensile forces responsible for keeping it taut fall below zero. Since slack cables do not transmit force, you must interpret simulation results carefully if a tension warning appears at the MATLAB command line. Warning options include:

- `Do not check tension`
- `Warn if either end loses tension`

Drum Dynamics

Bearing viscous friction coefficient

Linear damping coefficient in effect at the drum bearing. At a given drum angular velocity, this coefficient determines the power losses due to viscous friction. The default value is `0.001` (N*m)/(rad/s).

Inertia

Optional parameter accounting for drum inertia, its resistance to angular acceleration. Options include:

- `No inertia` — Default mode. Keep this mode if drum inertia has a negligible impact on driveline dynamics. Selecting this mode sets drum inertia to zero.
- `Specify inertia and initial velocity` — Select this mode if drum inertia has a significant impact on driveline dynamics. Selecting this mode exposes additional parameters enabling you to specify the drum inertia and initial angular velocity.

Drum inertia

Moment of inertia of the drum about its rotation axis. In the simple case of a solid cylindrical drum, the moment of inertia is

$$I = \frac{1}{2}MR^2,$$

where M is the drum mass and R is the drum radius.

The default value of the drum inertia is `0.01` kg*m^2.

Drum initial velocity

Angular velocity of the drum at simulation time zero. The default value is `0` rad/s.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also

Wheel and Axle

Topics

"Basic Motion, Torque, and Force Modeling"
"Power Window System"

Rotational Damper

Faultable linear rotational damper
Library: Simscape / Driveline / Couplings & Drives
                      Simscape / Foundation Library / Mechanical
                      Simscape / Foundation Library / Mechanical /
                      Rotational Elements

Description

The Rotational Damper block models a faultable linear rotational viscous damper. From the start of the simulation, the block uses the nominal damping coefficient. If you enable faults for the block, the damping coefficient changes in response to one or both of these triggers:

*   Simulation time — A fault occurs at a specified time.

*   Simulation behavior — If the magnitude of the rotational acceleration exceeds a specified maximum value, the block increments the number of shocks by one. A fault occurs if the number of shocks exceeds a specified maximum number of shocks.

If a fault trigger event occurs, the block uses the faulted damping coefficient for the remainder of the simulation. You can program the block to issue a fault report as a warning or error message.

The defining equations are

$$T = D\omega$$

and

$$\omega = \omega_R - \omega_C,$$

where

* T is the torque transmitted through the damper.
* D is the damping (viscous friction) coefficient.
* ω is the relative velocity.

- ω_R and ω_C are the absolute velocities of terminals R and C, respectively.

The positive block direction is from port R to port C. Therefore, the torque is positive if it acts in the direction from R to C.

Ports

Conserving

R — Rod interface
mechanical rotational

Mechanical rotational conserving port associated with the rod interface.

C — Case interface
mechanical rotational

Mechanical rotational conserving port associated with the case interface.

Parameters

Main

Damping coefficient — Damping coefficient for the no-fault condition
`1e-3 N*m/(rad/s)` (default)

The oscillation-reducing torque due to viscous friction that the block applies if no trigger event occurs. Specify a value that is greater than or equal to zero.

Faults

Enable faults — Fault mode capability
`Off` (default) | `On`

To enable the block to use a different damping coefficient based on the simulation time or when the magnitude of the rotational acceleration exceeds the specified number of shocks, select `On`.

Dependencies

Selecting On makes these parameters visible:

- **Faulted damping coefficient**
- **Enable temporal fault trigger**
- **Enable behavioral fault trigger**
- **Reporting when fault occurs**

Faulted damping coefficient — Damping coefficient for the fault condition
100 N*m/(rad/s) (default)

The oscillation-reducing torque due to viscous friction that the block applies if a trigger event occurs. Specify a value that is greater than or equal to zero.

Dependencies

This parameter is only visible if you select On for the **Enable faults** parameter.

Enable temporal fault trigger — Fault trigger capability
Off (default) | On

To enable the block to use the faulted damping coefficient starting at a specified time in the simulation, select On.

Dependencies

This parameter is only visible if you select On for the **Enable faults** parameter.

Selecting On makes the **Simulation time for fault event** parameter visible.

Simulation time for fault event — Temporal fault trigger
5 s (default)

Specify the time during the simulation that you want the block to start using the faulted damping coefficient.

Dependencies

This parameter is only visible if you select On for the **Enable faults** parameter and On for the **Enable temporal fault trigger** parameter.

Enable behavioral fault trigger — Fault trigger capability
Off (default) | On

To use the faulted damping coefficient when the number of sustained shocks exceeds a specified maximum number of shocks, select `On`.

Dependencies

This parameter is only visible if you select `On` for the **Enable faults** parameter.

Selecting `On` makes these parameters visible:

- **Maximum permissible acceleration**
- **Maximum number of shocks**

Maximum permissible acceleration — Behavioral fault trigger
`100 rad/s^2` (default)

The maximum value that the magnitude of the rotational acceleration can reach without causing the block to increment the number of sustained shocks.

Dependencies

This parameter is only visible if you select `On` for the **Enable faults** parameter and `On` for the **Enable behavioral fault trigger** parameter.

Maximum number of shocks — Behavioral fault trigger
`1` (default)

The maximum number of shocks that can be sustained without triggering the block to use the faulted damping coefficient.

Dependencies

This parameter is only visible if you select `On` for the **Enable faults** parameter and `On` for the **Enable behavioral fault trigger** parameter.

Reporting when fault occurs — Fault reporting
`None` (default) | `Warning` | `Error`

Select your reporting preference for the fault condition.

Dependencies

This parameter is only visible if you select `On` for the **Enable faults** parameter.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also
Nonlinear Rotational Damper | Nonlinear Translational Damper | Translational Damper | Variable Rotational Damper | Variable Translational Damper

Introduced in R2016b

Rotational Detent

Double-sided, spring-loaded rotational detent

Library

Simscape / Driveline / Brakes & Detents / Rotational

Description

The Rotational Detent block represents a double-sided, spring-loaded rotational detent.

The detent rod rotates over the detent case. The case has notches or magnets that create a net relative torque with the rod as the rod rotates. With the block, you can model rotational detents with peak torque and notch width, or with a table lookup. The model also includes viscous damping and kinetic friction between the rod and case. You can set the friction to zero.

Ports

R and **C** are rotational conserving ports associated with the rod and case, respectively.

Parameters

Detent Characteristics

Parameterization

Select how to specify the detent characteristics. The default is `By peak torque and notch width`.

- By `peak torque and notch width` — Specify detent characteristics by the peak torque and the notch width.

Peak Torque and Notch Width

Peak torque

    The peak torque developed inside the detent. The default is `0.25`.

    From the drop-down list, choose units. The default is newton-meters (`N*m`).

Notch width

    The width of the region where the torque develops. The default is `5`.

    From the drop-down list, choose units. The default is degrees (`deg`).

- By `table lookup` — Define the detent characteristics by one-dimensional table lookup based on the relative displacement between the slider and case. If you select this option, the panel changes from its default option.

Table Lookup

Relative rotation vector

    Specify the vector of input relative rotations that correspond to the values in **Torque vector**. These two vectors must be of the same size. The angles must be specified in strictly increasing order. The default values are `[-5.0 -4.0 -3.0 -1.0 1.0 3.0 4.0 5.0]`.

    From the drop-down list, choose units. The default is degrees (`deg`).

Torque vector

    Specify the vector of input values for the developed torque that correspond to the values in **Relative rotation vector**. The default values are `[0 0 -0.25 -0.10 0.10 0.25 0 0]`.

    From the drop-down list, choose units. The default is newton-meters (`N*m`).

Initial relative rotation

    The relative angle of the rod and case when simulation starts. The default is `0`.

    From the drop-down list, choose units. The default is degrees (`deg`).

Interpolation method

    Select one of these interpolation methods for approximating the output value when the input value is between two consecutive grid points:

- `Linear` — Select this, default, option to get the best performance.
- `Smooth` — Select this option to produce a continuous curve with continuous first-order derivatives.

For more information on interpolation algorithms, see the PS Lookup Table (1D) block reference page.

Spacing

Detent spacing

Select how to specify the detent spacing. The default is `Regularly spaced`.

- `Regularly spaced` — Specify detent spacing by regular intervals.

   Regular spacing

   Number of detents

   Specify the number of identical, regularly spaced detents. Must be greater than or equal to 1. The default is `1`.

   Creates a number of detents evenly spaced from –180 to +180 degrees, with one detent center at 0 degrees. Non-integer values are rounded to the nearest integer.

- `By angle vector` — Specify detent spacing by custom intervals. If you select this option, the panel changes from its default option.

   Custom intervals

   Vector of detent centers

   Specify the locations of the centers of identical detents as a vector of angles. These values must lie between –180 and +180 degrees. The minimum spacing between the detents must be greater than the **Notch width** parameter. The default values are `[-40 0 30 85]`.

   From the drop-down list, choose units. The default is degrees (`deg`).

Friction

Viscous friction coefficient

Specify the viscous friction coefficient for the rod-case contact. Must be greater than or equal to 0. The default is `0.1`.

From the drop-down list, choose units. The default is newton-meters/(radians/second) (`N*m/(rad/s)`).

Friction to peak torque ratio

Specify the kinetic friction coefficient for the rod-case contact. Must be greater than or equal to 0. The default is `0.01`.

The kinetic friction is this ratio multiplied by the peak torque.

Friction velocity threshold

Specify the relative angular velocity required for peak kinetic friction in the detent. Must be greater than 0. The default is `10`.

From the drop-down list, choose units. The default is revolutions/minute (`rpm`).

Rotational Detent Model

As the rod rotates over the case, the relative rotation $\phi = \phi_R - \phi_C$ causes a torque τ to develop. Depending on your choice of parameterization, the torque model is defined by the peak torque and notch width, or by a table lookup specifying relative rotation versus torque.

Regardless of the parameterization, if you specify multiple detents, all detents have the same torque-relative rotation function.

Peak Torque and Notch Width

If you choose the peak torque and notch width parameterization, the block ensures that the torque-relative rotation curve provides a continuous torque and torque derivative over the detent region. The peak torques are halfway between the detent center and detent edge, as shown in the following figure.
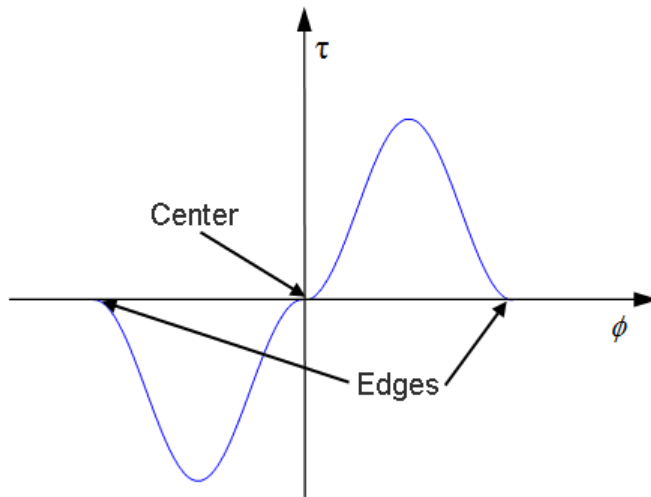
By Table Lookup

With the table lookup parameterization, you can create an arbitrary function relating torque to relative rotation. If you create such a function, consider the following best practices.

Energy Conservation

If you want to ensure that the detent conserves energy, the total integral of the torque-relative rotation curve (area under the curve) must be zero.

Simulation Performance

To stabilize simulation of the detent, avoid discontinuities in the torque-relative rotation function.

Limitations

- The model does not account for inertia. Add inertia terms externally to the R and C ports as required.
- If you use the peak torque-notch width or the table lookup parameterization, the kinetic friction is independent of the detent normal force.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.
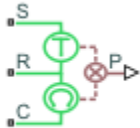
See Also
Loaded-Contact Rotational Friction | Tire (Friction Parameterized) | Translational Detent

Topics
"Clutches, Clutch-Like Elements, and Coulomb Friction"

Rotational Power Sensor

Mechanical sensor used to measure average or instantaneous rotational power

Library

Sensors

Description

The block represents a sensor that measures the rotational power change due to one or more driveline components connected between ports R and C. The power output can be an instantaneous or average value. Three averaging options are available: period, vibration, and revolution averaging.

The instantaneous power is the product of the current angular velocity and torque. Use the instantaneous power to analyze peak power transmission and identify heavily loaded components.

The time-averaged power is the integral of the instantaneous power over one sample period, divided by that period. For simulation times smaller than the sample period, the average power is zero. The power output is constant for each sample period.

The vibration-averaged power is the integral of the power over one vibration period, divided by that period. The vibration period is the time between two consecutive crossings from negative to positive velocity values. If the vibration has a variable frequency, the averaging period can change. For simulation times smaller than the vibration period, the average power is zero. The power output is constant for each vibration period.

The revolution-averaged power is the integral of the instantaneous power over one revolution period, divided by that period. The revolution period is defined as the time between two consecutive crossings of zero angular position. If the revolution has a variable frequency, the averaging period can change. For simulation times smaller than the revolution period, the average power is zero. The power output is constant for each revolution period.

Parameters

Measurement Type

Select either instantaneous or average power measurement. The default type is `Instantaneous Power`.

- `Instantaneous power` — Output the rotational power measured for the current simulation time step.

- `Average power over a specified period` — Output the rotational power averaged over a fixed time period.

  Averaging period

  Enter the time period used to compute the average power. Typical values include the cycle length of a rotating device.

  The default value is `1`. The default unit is `s`.

- `Average power over a vibration` — Output the rotational power averaged over one vibration period.

  Maximum vibration frequency

  Enter the maximum vibration frequency over which to average the rotational power. Very high values can reduce simulation speed. Very low values can result in lower power measurements.

  The default value is `1000`. The default unit is `Hz`.

- `Average power over a revolution` — Output the rotational power averaged over one revolution period.

  Maximum rotational speed

  Enter the maximum speed over which to average the rotational power. Very high values can reduce simulation speed. Very low values can result in lower power measurements.

The default value is `1000`. The default unit is `rpm`.

Ports

Port	Description
P	Physical signal output port for rotational power
S	Rotational conserving port associated with driving source
R	Rotational conserving port associated with loaded driveline components
C	Rotational conserving port associated with case holding loaded driveline components

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also
Translational Power Sensor

Introduced in R2013a

Rotational Velocity Noise Source

Produce zero-mean normally (Gaussian) distributed rotational velocity
Library: Simscape / Driveline / Sources

Description

The Rotational Velocity Source block generates zero-mean normally (Gaussian) distributed rotational velocity using the Random Number source in the Simscape Foundation library. The rotational velocity produced by the block depends on two factors:

- The noise power spectral density input to the block
- A random number seed that you control using the **Repeatability** and, as applicable, **Seed** parameters

The defining equation for the rotational velocity that the block generates is

$$v = \sqrt{PSD/2}\frac{N(0, 1)}{\sqrt{h}},$$

where:

- v is the output rotational velocity.
- PSD is the single-sided noise power spectral density.
- N is a Gaussian random number with zero mean and standard deviation of one.
- h is the sampling interval.

Ports

Input

P — Single-sided noise power spectral density
physical signal

The power spectral density (*PSD*) of noise is the average noise power per unit of bandwidth, in $(rad/s)^2/Hz$.

Conserving

B — Base interface
mechanical rotational

Mechanical rotational conserving port associated with the base interface.

F — Follower interface
mechanical rotational

Mechanical rotational conserving port associated with the follower interface.

Parameters

Sample time — Time step period and offset
1e-3 s (default) | [*step, offset*] s

The values of the time *step* period and the initial time *offset*. If you specify a scalar value for *step,* the block assumes an *offset* value of 0. To specify a nonzero value for the initial time offset, specify the parameter values using the vector [*step, offset*]. The *offset* value must be less than the *step* value and greater than or equal to zero.

Repeatability — Random number seed control
Not repeatable (default) | Repeatable | Specify seed

The random number seed is the number that initializes the random number generator. The seed is 0 or a positive integer. To control the random number seed, set this parameter to:

- `Not repeatable` — The seed changes every time you simulate your model. The block resets the random seed using the MATLAB random number generator command:

  `seed = randi(2^32-1);`

- `Repeatable` — The seed is the same random number at the start of every simulation. The block sets the value using the same MATLAB random number generator command used by the `Not repeatable` parameter.

  When you add a Rotational Velocity Noise Source block to your model from the Sources library, the block generates and stores a random value for the repeated seed. When you make a copy of the Rotational Velocity Noise Source block from an existing block in a model, the copy generates a new random value for the repeated seed.

- `Specify seed` — The seed is a number that you specify using the **Seed** parameter. The **Seed** parameter is only available when you choose `Specify seed` for the **Repeatability** parameter.

Seed — Random number generation seed value
0 (default)

The seed must be 0 or a positive integer. This parameter is only available when you select `Specify seed` for the **Repeatability** parameter.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also

Simscape Blocks
Force Noise Source | Random Number | Sinusoidal Rotational Velocity Source | Torque Noise Source | Translational Velocity Noise Source

Functions
randi

Introduced in R2016b

Shock Absorber

Mechanism for damping translational vibrations

Library

Simscape / Driveline / Couplings & Drives

Description

The Shock Absorber block represents a spring-damper system commonly used to dampen vibration in mechanical systems. An internal force acts between ports R and C. This force is the sum of spring stiffness, damping, Coulomb friction, and hard-stop contributions. All force contributions are optional.
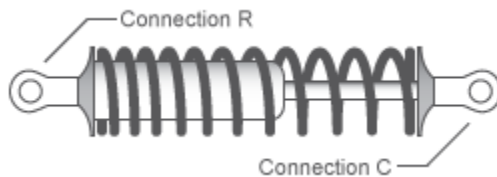
The Shock Absorber block uses the models of these blocks:

Block	Contribution	Library
Loaded-Contact Translational Friction	Coulomb friction	**Simscape** / **Driveline** / **Brakes & Detents** / **Translational**
Rotational Damper	Damping	**Simscape** / **Foundation Library** / **Mechanical** / **Translational Elements**
Rotational Spring	Spring	

Block	Contribution	Library
Rotational Hard Stop	Hard stop	

Assumptions and Limitations

- Including hard-stop and Coulomb friction enhances model fidelity, but reduces simulation speed. For more information, see "Driveline Simulation Performance".

Ports

Conserving

C

Mechanical translational port associated with the slider that travels between stops installed on the case.

R

Mechanical translational port associated with the rod.

Parameters

Spring-Damper

Restoring spring stiffness

Enter the value of the viscous spring stiffness constant, k. The default value is `1e4 N/m`. The value must be greater than or equal to zero.

Viscous friction coefficient

Enter the value of the viscous damping constant, b. The default value is `1e2 m/s`. The value must be greater than or equal to zero.

Coulomb friction force

Enter the value of the Coulomb friction force. Setting the value to `0` eliminates the Coulomb friction force contribution. This enhances simulation speed, making the model more suitable for HIL testing. The default value is `0 N`. The value must be greater than or equal to zero.

Ratio of static to kinetic friction

Enter the value of the static/kinetic friction ratio, that is F_s/F_k. The value must be greater than one. The default value is `1.1`. The value must be greater than or equal to one.

Velocity tolerance

Enter the value of the relative velocity below which ports R and C lock and translate together. The default value is `0.001 m/s`. The value must be greater than zero.

Hard Stops

Hard stop

Include or exclude hard-stop force by selecting one of these options:

- `No hard stops — Suitable for HIL simulation` — To enhance simulation speed by excluding the hard-stop force contribution, select this default option.

- `Compliant hard stops` — To enhance model fidelity by including the hard-stop force contribution, select this option. Selecting this option enables other parameters.

Upper bound

Upper hard-stop position, *UB*. Positive displacement beyond the upper bound activates the hard-stop contact force. The default value is `0.1 m`. The value must be greater than the lower bound.

Selecting `Compliant hard stops` for the **Hard stop** parameter enables this parameter.

Lower bound

Lower hard-stop position, *LB*. Negative displacement beyond the lower bound activates the hard-stop contact force. The default value is `-0.1 m`. The value must be smaller than the upper bound.

Selecting `Compliant hard stops` for the **Hard stop** parameter enables this parameter.

Contact stiffness

Enter the value of the hard-stop stiffness constant, k_{HS}. The default value is `1e6 N/m`. The value must be greater than or equal to zero.

Selecting `Compliant hard stops` for the **Hard stop** parameter enables this parameter.

Contact damping

Enter the value of the hard-stop damping constant, b_{HS}. This parameter specifies dissipating property of colliding bodies. The greater the value of the parameter, the more energy dissipates during an interaction. The default value is `1e4 N/(m/s)`. The value must be greater than zero.

Selecting `Compliant hard stops` for the **Hard stop** parameter enables this parameter.

Hard stop model

Select the hard-stop model:

- `Stiffness and damping applied smoothly through transition region, damped rebound` — Specify a transition region, in which the force is scaled from zero. At the end of the transition region, the full stiffness and damping are applied. This model has damping applied on the rebound, but it is limited to the value of the stiffness force. In this sense, damping can reduce or eliminate the force provided by the stiffness, but never exceed it. All equations are smooth and produce no zero crossings.

  Selecting this option enables the `Transition region` parameter.

- `Full stiffness and damping applied at bounds, undamped rebound` — This model has full stiffness and damping applied with impact at upper and lower bounds, with no damping on the rebound. Equations produce no zero crossings when velocity changes sign, but there is a position-based zero crossing at the bounds. Having no damping on rebound helps to push the slider past this position quickly. This model has nonlinear equations.

- `Full stiffness and damping applied at bounds, damped rebound` — This model has full stiffness and damping applied with impact at upper and lower bounds, with damping applied on the rebound as well. Equations are switched linear, but produce position-based zero crossings. Use this hard-stop model if `simscape.findNonlinearBlocks` indicates that this is the block that prevents the whole network from being switched linear.

Selecting `Compliant hard stops` for the **Hard stop** parameter enables this parameter.

Transition region

Region where the force is ramped up from zero to the full value. At the end of the transition region, the full stiffness and damping are applied. The default value is `0.1 mm`.

Selecting `Stiffness and damping applied smoothly through transition region, damped rebound` for the **Hard stop model** enables this parameter.

Initial Conditions

Initial deformation

Enter the initial value of the spring deformation. The default value is `0 m`.

Real-Time Simulation

Hardware-in-the-Loop Simulation

For optimal simulation performance, use the **Hard Stops** > **Hard stop** parameter default setting, `No hard stops - Suitable for HIL simulation`.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also

Simscape Blocks
Nonlinear Translational Damper | Nonlinear Translational Spring | Rotational Hard Stop | Variable Translational Damper | Variable Translational Spring

Topics

"Prepare Simscape Driveline Models for Real-Time Simulation Using Simscape Checks"

Simple Gear

Simple gear of base and follower wheels with adjustable gear ratio, friction losses, and triggered faults
Library:　　　Simscape / Driveline / Gears
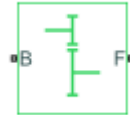
Description

The Simple Gear block represents a gearbox that constrains the connected driveline axes of the base gear, *B*, and the follower gear, *F*, to corotate with a fixed ratio that you specify. You choose whether the follower axis rotates in the same or opposite direction as the base axis. If they rotate in the same direction, the angular velocity of the follower, ω_F, and the angular velocity of the base, ω_B, have the same sign. If they rotate in opposite directions, ω_F and ω_B have opposite signs.

Ideal Gear Constraint and Gear Ratio

The kinematic constraint that the Simple Gear block imposes on the two connected axes is

$$r_F \omega_F = r_B \omega_B$$

where:

- r_F is the radius of the follower gear.
- ω_F is the angular velocity of the follower gear.
- r_B is the radius of the base gear.
- ω_B is the angular velocity of the base gear.

The follower-base gear ratio is

$$g_{FB} = \frac{r_F}{r_B} = \frac{N_F}{N_B}$$

1-421

where:

- N_B is the number of teeth in the base gear.
- N_{BF} is the number of teeth in the follower gear.

Reducing the two degrees of freedom to one independent degree of freedom yields the torque transfer equation

$$g_{FB}\tau_B + \tau_F - \tau_{loss} = 0$$

where:

- τ_B is the input torque.
- τ_F is the output torque.
- τ_{loss} is the torque loss due to friction.

For the ideal case, $\tau_{loss} = 0$.

Nonideal Gear Constraint and Losses

In the nonideal case, $\tau_{loss} \neq 0$. For general considerations on nonideal gear modeling, see "Model Gears with Losses".

In a nonideal gear pair (B,F), the angular velocity, gear radii, and gear teeth constraints are unchanged. But the transferred torque and power are reduced by:

- Coulomb friction between teeth surfaces on gears *B* and *F*, characterized by efficiency, η
- Viscous coupling of driveshafts with bearings, parametrized by viscous friction coefficients, μ

Constant Efficiency

In the constant efficiency case, η is constant, independent of load or power transferred.

Load-Dependent Efficiency

In the load-dependent efficiency case, η depends on the load or power transferred across the gears. For either power flow,

$$\tau_{Coul} = g_{FB}\tau_{idle} + k\tau_F$$

where:

- τ_{Coul} is the Coulomb friction dependent torque.
- k is a proportionality constant.
- τ_{idle} is the net torque acting on the input shaft in idle mode.

Efficiency, η, is related to τ_{Coul} in the standard, preceding form but becomes dependent on load:

$$\eta = \frac{\tau_F}{g_{FB}\tau_{idle} + (k + 1)\tau_F}$$

Faults

If you enable faults for the block, the efficiency changes in response to one or both of these triggers:

- Simulation time — A fault occurs at a specified time.
- Simulation behavior — A fault occurs in response to an external trigger. Enabling an external fault trigger exposes port **T**.

If a fault trigger occurs, for the remainder of the simulation, the block uses the faulted efficiency in one of these ways:

- Throughout rotation
- When the rotation angle is within a faulted range that you specify

You can program the block to issue a fault report as a warning or error message.

Thermal Model

You can model the effects of heat flow and temperature change by selecting a thermal block variant. Selecting a thermal variant:

- Exposes port **H**, a conserving port in the thermal domain.
- Enables the **Thermal mass** parameter, which allows you to specify the ability of the component to resist changes in temperature.
- Enables the **Initial Temperature** parameter, which allows you to set the initial temperature.

To select a thermal variant, right-click the block in your model and, from the context menu, select **Simscape > Block choices**. Select a variant that includes a thermal port.

Assumptions

- Gear inertia is assumed negligible.
- Gears are treated as rigid components.
- Coulomb friction slows down simulation. See "Adjust Model Fidelity".

Ports

Input

T — External fault trigger
physical signal

Physical signal input port for an external fault trigger.

Dependencies

To expose the **T** port:

1   For the **Meshing Losses Friction model** parameter, select `Constant efficiency` or `Load-dependent efficiency`.
2   For the **Faults Enable faults** parameter, select `On`.
3   For the **Faults Enable external fault trigger** parameter, select `On`.
4   Click **OK** or **Apply**.

For information on related dependencies, see "Parameter Dependencies Table" on page 1-425.

Conserving

B — Base
rotational mechanical

Rotational mechanical conserving port associated with the base, or input, shaft.

F — Follower
rotational mechanical

Rotational mechanical conserving port associated with the follower, or output, shaft.

H — Heat
thermal

Thermal port associated with heat flow. Heat flow affects gear temperature, and therefore, power transmission efficiency.

Dependencies

The thermal conserving port is optional and is hidden by default. To expose the port, select a variant that includes a thermal port.

Selecting a thermal variant enables thermal parameters. For more information, see "Parameter Dependencies Table" on page 1-425.

Parameters

Parameter Dependencies Table

The table shows how the visibility of some **Meshing Losses** parameters and **Faults** parameters depend on the thermal model and the option that you choose for other parameters. To learn how to read the table, see "Parameter Dependencies" on page B-2.

Default Model — For nonthermal models, thermal port H is not visible.			Thermal Model — For thermal models, thermal port H is visible.	
Meshing Losses			Meshing Losses	
Friction model — Choose `No meshing losses - Suitable for HIL simulation`, `Constant efficiency`, or `Load-dependent efficiency`			Friction model — Choose `Temperature-dependent efficiency` or `Temperature and load-dependent efficiency`	
No meshing	Constant efficiency	Load-dependent efficiency	Temperature-dependent efficiency	Temperature and load-dependent efficiency

Default Model — For nonthermal models, thermal port H is not visible.			**Thermal Model — For thermal models, thermal port H is visible.**	
Meshing Losses			**Meshing Losses**	
Friction model — Choose `No meshing losses - Suitable for HIL simulation`, `Constant efficiency`, or `Load-dependent efficiency`			**Friction model — Choose `Temperature-dependent efficiency` or `Temperature and load-dependent efficiency`**	
losses - Suitable for HIL simulation	Efficiency	Input shaft torque at no load	Temperature	Temperature
	Follower power threshold	Nominal output torque	Efficiency	Load at base gear
		Efficiency at nominal output torque	Follower power threshold	Efficiency matrix
		Follower angular velocity threshold		Follower angular velocity threshold
Faults	**Faults**			
	Enable faults — Choose `Off` or `On`			
	Off	**On**		
		Faulted efficiency		
		Enable external fault trigger — Choose `Off` or `On`. Selecting `On` makes thermal port **T** visible.		
		Enable temporal fault trigger — Choose `Off` or `On`		
		Off	**On**	
			Simulation time for fault event	
		Faulted angle range		
		Reporting when fault occurs — Choose `None`, or `Warning`, or `Error`		

Main

Follower (F) to base (B) teeth ratio (NF/NB) — Gear ratio
2 (default) | positive scalar

Fixed ratio g_{FB} of the follower axis to the base axis. The gear ratio must be strictly positive.

Output shaft rotates — Motion direction
In opposite direction to input shaft (default) | In same direction as input shaft

Direction of motion of the follower (output) driveshaft relative to the motion of the base (input) driveshaft.

Meshing Losses

Meshing losses parameters depend on the thermal model. For more information, see "Parameter Dependencies Table" on page 1-425.

Default (Nonthermal) Meshing Losses Parameters

Friction model — Default friction model
No meshing losses - Suitable for HIL simulation (default) | Constant efficiency | Load-dependent efficiency

Friction models at various precision levels for estimating power losses due to meshing.

- No meshing losses - Suitable for HIL simulation — Neglect friction between gear cogs. Meshing is ideal.
- Constant efficiency — Reduce torque transfer by a constant efficiency factor. This factor falls in the range $0 < \eta \leq 1$ and is independent from load.
- Load-dependent efficiency — Reduce torque transfer by a variable efficiency factor. This factor falls in the range $0 < \eta < 1$ and varies with the torque load.

Dependencies

This parameter is visible when you choose a thermal model. This parameter affects the visibility of other **Meshing Losses** parameters and **Faults** parameters.

For more information, see "Parameter Dependencies Table" on page 1-425.

Efficiency — Torque transfer efficiency, *η*
`0.8` (default) | $0 \leq \eta < 1$

Torque transfer efficiency, *η*, between base and follower shafts. Efficiency is inversely proportional to the meshing power losses.

Dependencies

This parameter is visible when you choose a nonthermal model and set **Friction model** to `Constant efficiency`.

For more information, see "Parameter Dependencies Table" on page 1-425.

Follower power threshold — Efficiency factor power basis
`0.001` W (default) | positive scalar

Absolute value of the follower shaft power above which the full efficiency factor is in effect. Below this value, a hyperbolic tangent function smooths the efficiency factor to 1, lowering the efficiency losses to 0 when no power is transmitted.

As a guideline, the power threshold should be lower than the expected power transmitted during simulation. Higher values might cause the block to underestimate efficiency losses. Very low values tend to raise the computational cost of simulation.

Dependencies

This parameter is visible when you choose a nonthermal model and set **Friction model** to `Constant efficiency`.

For more information, see "Parameter Dependencies Table" on page 1-425.

Input shaft torque at no load — No-load torque
`0.1` N*m (default) | positive scalar

Net torque, τ_{idle}, acting on the input shaft in idle mode, that is, when torque transfer to the output shaft equals zero. For nonzero values, the power input in idle mode completely dissipates due to meshing losses.

Dependencies

This parameter is visible when you choose a nonthermal model and set **Friction model** to `Load-dependent efficiency`.

For more information, see "Parameter Dependencies Table" on page 1-425.

Nominal output torque — Nominal torque
5 N*m (default) | positive scalar

Output torque, τ_F, at which to normalize the load-dependent efficiency.

Dependencies

This parameter is visible when you choose a nonthermal model and set **Friction model** to Load-dependent efficiency.

For more information, see "Parameter Dependencies Table" on page 1-425.

Efficiency at nominal output torque — Nominal efficiency, η
0.8 (default) | $0 \leq \eta < 1$ | scalar

Torque transfer efficiency, η, at the nominal output torque. Larger efficiency values correspond to greater torque transfer between the input and output shafts.

Dependencies

This parameter is visible when you choose a nonthermal model and set **Friction model** to Load-dependent efficiency.

For more information, see "Parameter Dependencies Table" on page 1-425.

Follower angular velocity threshold — Efficiency factor angular velocity basis
0.01 rad/s (default) | positive scalar

Absolute value of the follower shaft angular velocity above which the full efficiency factor is in effect, ω_F. Below this value, a hyperbolic tangent function smooths the efficiency factor to one, lowering the efficiency losses to zero when at rest.

As a guideline, the angular velocity threshold should be lower than the expected angular velocity during simulation. Higher values might cause the block to underestimate efficiency losses. Very low values tend to raise the computational cost of simulation.

Dependencies

This parameter is visible when you choose a nonthermal model and set **Friction model** to Load-dependent efficiency.

For more information, see "Parameter Dependencies Table" on page 1-425.

1-429

Thermal Model Meshing Losses Parameters

Friction model — Thermal friction model
Temperature-dependent efficiency (default) | Temperature and load-dependent efficiency

Friction models at various precision levels for estimating power losses due to meshing. The block incorporates temperature dependencies and can incorporate load dependencies.

Friction models at various precision levels for estimating power losses due to meshing.

- Temperature-dependent efficiency — Reduce torque transfer by a constant efficiency factor that is dependent on temperature but does not consider the gear load. This factor falls in the range $0 < \eta \leq 1$ and is independent from load.
- Load-dependent efficiency — Reduce torque transfer by a variable efficiency factor that is dependent on temperature and load. This factor falls in the range $0 < \eta < 1$ and varies with the torque load.

Dependencies

This parameter is visible when you choose a thermal model. This parameter affects the visibility of other **Meshing Losses** parameters.

For more information, see "Parameter Dependencies Table" on page 1-425.

Temperature — Temperature
[280, 300, 320] K (default) | positive array

Array of temperatures used to construct an efficiency lookup table. The array values must increase from left to right. The temperature array must be the same size as the efficiency array in temperature-dependent models. The array must be the same size as a single row of the efficiency matrix in temperature and load dependent models.

Dependencies

This parameter is visible for thermal models.

For more information, see "Parameter Dependencies Table" on page 1-425.

Efficiency — Efficiency, *η*
[.95, .9, .85] (default) | $0 \leq \eta < 1$ | array

Array of efficiencies used to construct a 1-D temperature-efficiency lookup table for temperature-dependent efficiency models. The array values are the efficiencies at the temperatures in the **Temperature** array. The number of elements must be the same as the number of elements in the **Temperature** array.

Dependencies

This parameter is visible for thermal models when you set **Friction model** to `Temperature-dependent efficiency`.

For more information, see "Parameter Dependencies Table" on page 1-425.

Follower power threshold — Efficiency factor power basis
`0.001` W (default) | positive array

Absolute value of the follower shaft power above which the full efficiency factor is in effect, p_F. Below this value, a hyperbolic tangent function smooths the efficiency factor to 1, lowering the efficiency losses to 0 when no power is transmitted.

As a guideline, the power threshold should be lower than the expected power transmitted during simulation. Higher values might cause the block to underestimate efficiency losses. Very low values tend to raise the computational cost of simulation.

Dependencies

This parameter is visible for thermal models when you set **Friction model** to `Temperature-dependent efficiency`.

For more information, see "Parameter Dependencies Table" on page 1-425.

Load at base gear — Base gear load
`[1, 5, 10]` N*m (default) | positive array

Array of base-gear loads used to construct a 2-D temperature load efficiency lookup table for temperature and load dependent efficiency models. The array values must increase left to right. The load array must be the same size as a single column of the efficiency matrix.

Dependencies

This parameter is visible when you choose a thermal model and set **Friction model** to `Temperature and load-dependent efficiency`.

For more information, see "Parameter Dependencies Table" on page 1-425.

Efficiency matrix — Efficiency
[.85, .8, .75; .95, .9, .85; .85, .8, .7] (default) | $0 \leq \eta < 1$ | positive array

Matrix of component efficiencies used to construct a 2-D temperature load efficiency lookup table. The matrix elements are the efficiencies at the temperatures given by the **Temperature** array and at the loads given by the **Load at base gear** array.

The number of rows must be the same as the number of elements in the **Temperature** array. The number of columns must be the same as the number of elements in the **Load at base gear** array.

Dependencies

This parameter is visible when you choose a thermal model and set **Friction model** to Temperature and load-dependent efficiency.

For more information, see "Parameter Dependencies Table" on page 1-425.

Follower angular velocity threshold — Efficiency factor angular velocity basis
0.01 rad/s (default) | positive scalar

Absolute value of the follower shaft angular velocity above which the full efficiency factor is in effect, ω_F. Below this value, a hyperbolic tangent function smooths the efficiency factor to one, lowering the efficiency losses to zero when at rest.

As a guideline, the angular velocity threshold should be lower than the expected angular velocity during simulation. Higher values might cause the block to underestimate efficiency losses. Very low values tend to raise the computational cost of simulation.

Dependencies

This parameter is visible when you choose a thermal model and set **Friction model** to Temperature and load-dependent efficiency.

For more information, see "Parameter Dependencies Table" on page 1-425.

Viscous Losses

Viscous friction coefficients at base (B) and follower (F) — Coefficients of viscous friction
[0, 0] N*m/(rad/s) (default) | positive array

Two-element array with the viscous friction coefficients in effect at the base and follower shafts. To neglect viscous losses, use the default setting, `[0, 0]`.

Faults

For nonthermal models, **Faults** parameters are not visible when you set the **Meshing Losses Friction model** parameter to `No meshing losses - Suitable for HIL simulation`.

Enable faults — Fault option
on (default) | off

Enable externally or temporally triggered faults.

Dependencies

This parameter is not visible when you choose a nonthermal model and set the **Meshing Losses Friction model** parameter to `No meshing losses - Suitable for HIL simulation`. This parameter affects the visibility of other **Faults** parameters.

For more information, see "Parameter Dependencies Table" on page 1-425.

Faulted efficiency — Faulted efficiency
0.05 (default) | $0 \leq \eta < 1$ | scalar

Efficiency when a fault is triggered.

Dependencies

This parameter is not visible when you choose a nonthermal model and set the **Meshing Losses Friction model** parameter to `No meshing losses - Suitable for HIL simulation` or when you set the **Enable faults** parameter to `off`.

For more information, see "Parameter Dependencies Table" on page 1-425.

Enable external fault trigger — External fault trigger option
on (default) | off

Option to enable an externally triggered fault.

Dependencies

This parameter is not visible when you choose a nonthermal model and set the **Meshing Losses Friction model** parameter to `No meshing losses - Suitable for HIL`

simulation or when you set the **Enable faults** parameter to off. When you select on for this parameter, the **T** port is exposed.

For more information, see "Parameter Dependencies Table" on page 1-425.

`Enable temporal fault trigger` — Temporal fault trigger option
on (default) | off

Option to enable a temporally triggered fault.

Dependencies

This parameter is not visible when you choose a nonthermal model and set the **Meshing Losses Friction model** parameter to No meshing losses - Suitable for HIL simulation or when you set the **Enable faults** parameter to off. When you select on for this parameter, the **Simulation time for fault event** parameter becomes visible.

For more information, see "Parameter Dependencies Table" on page 1-425.

`Simulation time for fault event` — Fault time trigger
5 s (default) | positive scalar

Simulation time that triggers a temporal fault.

Dependencies

This parameter is not visible when you choose a nonthermal model and set the **Meshing Losses Friction model** parameter to No meshing losses - Suitable for HIL simulation, or when you set the **Enable faults** parameter to on and set the **Enable temporal fault trigger** parameter to off.

For more information, see "Parameter Dependencies Table" on page 1-425.

`Faulted angle range` — Fault angle range
pi / 12 rad (default)

Rotational angle range for the faulted efficiency. For a value or multiples of 2π rad, the faulted efficiency is applicable throughout rotation.

Dependencies

This parameter is not visible when you choose a nonthermal model and set the **Meshing Losses Friction model** parameter to No meshing losses - Suitable for HIL simulation or when you set the **Enable faults** parameter to off.

For more information, see "Parameter Dependencies Table" on page 1-425.

`Reporting when fault occurs` — Fault report
`None` (default) | `Warning` | `Error`

Reporting preference for the fault condition.

Dependencies

This parameter is not visible when you choose a nonthermal model and set the **Meshing Losses Friction model** parameter to `No meshing losses - Suitable for HIL simulation` or when you set the **Enable faults** parameter to `off`.

For more information, see "Parameter Dependencies Table" on page 1-425.

Thermal Port

`Thermal mass` — Thermal mass
`50 J/K` (default) | positive scalar

Thermal energy required to change the component temperature by a single degree. The greater the thermal mass, the more resistant the component is to temperature change.

Dependencies

Selecting a thermal block variant for the **Block choice** parameter makes this parameter visible.

`Initial temperature` — Temperature
`300 K` (default) | positive scalar

Temperature at simulation start.

Dependencies

Selecting a thermal block variant for the **Block choice** parameter makes this parameter visible.

Extended Capabilities

Real-Time and Hardware-in-the-Loop Simulation

For optimal simulation performance, use the **Meshing Losses** > **Friction model** parameter default setting, `No meshing losses - Suitable for HIL simulation`.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also
Variable Ratio Transmission

Topics
"Basic Motion, Torque, and Force Modeling"
"Gearbox Efficiency Measurement"
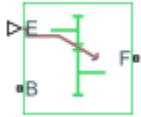"Simple Gear"
"Gear with Backlash"
"Model Thermal Losses in Driveline Components"
"Prepare Simscape Driveline Models for Real-Time Simulation Using Simscape Checks"

Introduced in R2011a

Simple Gear with Variable Efficiency

Gear set with parallel-axis rotation and variable meshing efficiency

Library

Gears

Description

The block represents a simple gear train with variable meshing efficiency. The gear train transmits torque at a specified ratio between base and follower shafts arranged in a parallel configuration. Shaft rotation can occur in equal or opposite directions. Gear losses are optional. They include meshing and viscous bearing losses. To specify the variable meshing efficiency, the block contains a physical signal port that you can use to input a general time-varying signal. Inertia and compliance effects are ignored.

You can model the effects of heat flow and temperature change through an optional thermal conserving port. By default, the thermal port is hidden. To expose the thermal port, right-click the block in your model and, from the context menu, select **Simscape** > **Block choices**. Specify the associated thermal parameters for the component.

Parameters

Main

Follower (F) to base (B) teeth ratio (NF/NB)

Enter the gear ratio. This is the fraction of follower over base gear teeth numbers, NF/NB. The ratio must be positive. The default value is 2.

Output shaft rotates

Select the relative rotation between shafts. This is the rotation direction of the output shaft with respect to the input shaft. Options include equal or opposite directions. The default setting is `In opposite direction to input shaft`.

Meshing Losses

Minimum efficiency

Enter the smallest efficiency value allowed for the gear. The efficiency is the power ratio between output and input shafts. The physical signal input saturates for values below the minimum efficiency or above 1. The minimum efficiency must be positive. The default value is `0.01`.

Follower power threshold

Enter the follower shaft power above which full efficiency factor is in effect. A hyperbolic tangent function smooths the efficiency factor between zero when at rest and the value provided by the temperature-efficiency lookup table when at the power threshold. The default value is `0.001` W.

Viscous Losses

Viscous friction coefficients at base (B) and follower (F)

Enter a two-element vector with the viscous friction coefficients of the base and follower gears. Coefficients must be positive. The default vector is `[0 0]`. The default unit is `N*m/(rad/s)`.

Thermal Port

Thermal mass

Thermal energy required to change the component temperature by a single degree. The greater the thermal mass, the more resistant the component is to temperature change. The default value is `50` J/K.

Initial temperature

Component temperature at the start of simulation. The initial temperature alters the component efficiency according to an efficiency vector that you specify, affecting the starting meshing or friction losses. The default value is `300` K.

Simple Gear Model

Ideal Gear Constraint and Gear Ratio

Simple Gear imposes one kinematic constraint on the two connected axes:

$$r_F \omega_F = r_B \omega_B .$$

The follower-base gear ratio $g_{FB} = r_F/r_B = N_F/N_B$. N is the number of teeth on each gear. The two degrees of freedom reduce to one independent degree of freedom.

The torque transfer is:

$$g_{FB} \tau_B + \tau_F - \tau_{loss} = 0 ,$$

with $\tau_{loss} = 0$ in the ideal case.

Nonideal Gear Constraint and Losses

In the nonideal case, $\tau_{loss} \neq 0$. For general considerations on nonideal gear modeling, see "Model Gears with Losses".

Limitations

- Gear inertia is assumed negligible.
- Gears are treated as rigid components.
- Coulomb friction slows down simulation. See "Adjust Model Fidelity".

Ports

Port	Description
B	Rotational conserving port representing the base shaft
F	Rotational conserving port representing the follower shaft
H	Thermal conserving port for thermal modeling

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also
Simple Gear | Variable Ratio Transmission

Topics
"Basic Motion, Torque, and Force Modeling"

Sinusoidal Force Source

Produce sinusoidal force
Library: Simscape / Driveline / Sources

Description

The Sinusoidal Force Source block produces sinusoidal force with a constant offset. A positive force value generates force acting from port B to port F.

The defining equation for the sinusoidal force that the block generates is

$$F = F_o + A * \sin(2\pi f t + \varphi),$$

where:

- F is the output force.
- F_o is the offset force of the output.
- A is the peak amplitude of the sinusoid.
- f is frequency of the sinusoid in Hz.
- t is time.
- φ is phase shift of the sinusoid.

Ports

Conserving

B — Base interface
mechanical translational

Mechanical translational conserving port associated with the base interface.

F — Follower interface
mechanical translational

Mechanical translational conserving port associated with the follower interface.

Parameters

Offset — Output average value
0 N (default)

Average value that the output sinusoidal force oscillates about.

Sinusoid peak amplitude — Output peak amplitude
1 N (default)

Peak amplitude of the output sinusoidal force.

Phase shift — Output phase shift
0 deg (default)

Phase shift of the output sinusoidal force.

Frequency — Output frequency
1 Hz (default)

Frequency of the output sinusoidal force. The frequency must be greater than or equal to 0.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also
Force Noise Source | Sinusoidal Rotational Velocity Source | Sinusoidal Torque Source | Sinusoidal Translational Velocity Source

Introduced in R2016b

Sinusoidal Rotational Velocity Source

Produce sinusoidal rotational velocity
Library: Simscape / Driveline / Sources

Description

The Sinusoidal Rotational Velocity Source block produces sinusoidal rotational velocity with a constant offset.

The relative rotational velocity between the base and follower interfaces is

$$v_{FB} = v_F - v_B,$$

where:

- v_{FB} is the relative rotational velocity.
- v_F is the absolute rotational velocity at the follower interface.
- v_B is the absolute rotational velocity at the base interface.

The defining equation for the sinusoidal rotational velocity that the block generates is

$$v = v_o + A * \sin(2\pi f t + \varphi),$$

where:

- v is the output rotational velocity.
- v_o is the offset rotational velocity of the output.
- A is the peak amplitude of the sinusoid.
- f is frequency of the sinusoid in Hz.
- t is time.
- φ is phase shift of the sinusoid.

Ports

Conserving

B — Base interface
mechanical rotational

Mechanical rotational conserving port associated with the base interface.

F — Follower interface
mechanical rotational

Mechanical rotational conserving port associated with the follower interface.

Parameters

Offset — Output average value
0 rad/s (default)

Average value that the output sinusoidal rotational velocity oscillates about.

Sinusoid peak amplitude — Output peak amplitude
1 rad/s (default)

Peak amplitude of the output sinusoidal rotational velocity.

Phase shift — Output phase shift
0 deg (default)

Phase shift of the output sinusoidal rotational velocity.

Frequency — Output frequency
1 Hz (default)

Frequency of the output sinusoidal rotational velocity. The frequency must be greater than or equal to 0.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also
Rotational Velocity Noise Source | Sinusoidal Force Source | Sinusoidal Torque Source | Sinusoidal Translational Velocity Source

Introduced in R2016b

Sinusoidal Torque Source

Produce sinusoidal torque
Library: Simscape / Driveline / Sources

Description

The Sinusoidal Torque Source block produces sinusoidal torque with a constant offset. A positive torque value generates torque acting from port B to port F.

The defining equation for the sinusoidal torque that the block generates is

$$T = T_o + A*\sin(2\pi f t + \varphi),$$

where:

- T is the output torque.
- T_o is the offset torque of the output.
- A is the peak amplitude of the sinusoid.
- f is frequency of the sinusoid in Hz.
- t is time.
- φ is phase shift of the sinusoid.

Ports

Conserving

B — Base interface
mechanical rotational

Mechanical rotational conserving port associated with the base interface.

F — Follower interface
mechanical rotational

Mechanical rotational conserving port associated with the follower interface.

Parameters

Offset — Output average value
0 N*m (default)

Average value that the sinusoidal output torque oscillates about.

Sinusoid peak amplitude — Output peak amplitude
1 N*m (default)

Peak amplitude of the sinusoidal output torque.

Phase shift — Output phase shift
0 deg (default)

Peak amplitude of the sinusoidal output torque.

Frequency — Output frequency
1 Hz (default)

Frequency of the sinusoidal output torque. The frequency must be greater than or equal to 0.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also
Sinusoidal Force Source | Sinusoidal Rotational Velocity Source | Sinusoidal Translational Velocity Source | Torque Noise Source

Introduced in R2016b

Sinusoidal Translational Velocity Source

Produce sinusoidal translational velocity
Library: Simscape / Driveline / Sources

Description

The Sinusoidal Translational Velocity Source block produces sinusoidal translational velocity with a constant offset.

The relative translational velocity between the base and follower interfaces is

$$v_{FB} = v_F - v_B,$$

where:

- v_{FB} is the relative translational velocity.
- v_F is the absolute translational velocity at the follower interface.
- v_B is the absolute translational velocity at the base interface.

The defining equation for the sinusoidal translational velocity that the block generates is

$$v = v_o + A * \sin(2\pi f t + \varphi),$$

where:

- v is the output translational velocity.
- v_o is the offset translational velocity of the output.
- A is the peak amplitude of the sinusoid.
- f is frequency of the sinusoid in Hz.
- t is time.
- φ is phase shift of the sinusoid.

Ports

Conserving

B — Base interface
mechanical translational

Mechanical translational conserving port associated with the base interface.

F — Follower interface
mechanical translational

Mechanical translational conserving port associated with the follower interface.

Parameters

Offset — Output average value
0 m/s (default)

Average value that the output sinusoidal translational velocity oscillates about.

Sinusoid peak amplitude — Output peak amplitude
1 m/s (default)

Peak amplitude of the output sinusoidal translational velocity.

Phase shift — Output phase shift
0 deg (default)

Phase shift of the output sinusoidal translational velocity

Frequency — Output frequency
1 Hz (default)

Frequency of the output sinusoidal translational velocity. The frequency must be greater than or equal to 0.

Extended Capabilities

C/C++ Code Generation
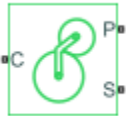Generate C and C++ code using MATLAB® Coder™.

See Also
Sinusoidal Force Source | Sinusoidal Rotational Velocity Source | Sinusoidal Torque Source | Translational Velocity Noise Source

Introduced in R2016b

Sun-Planet

Planetary gear set of carrier, planet, and sun wheels with adjustable gear ratio and friction losses

Library

Simscape / Driveline / Gears / Planetary Subcomponents

Description

The Sun-Planet gear block represents a set of carrier, planet, and sun gear wheels. The planet is connected to and rotates with respect to the carrier. The planet and sun corotate with a fixed gear ratio that you specify and in the same direction with respect to the carrier. A sun-planet and a ring-planet gear are basic elements of a planetary gear set. For model details, see "Sun-Planet Gear Model" on page 1-454.

Thermal Modeling

You can model the effects of heat flow and temperature change through an optional thermal conserving port. By default, the thermal port is hidden. To expose the thermal port, right-click the block in your model and, from the context menu, select **Simscape > Block choices**. Select a variant that includes a thermal port. Specify the associated thermal parameters for the component.

Sun-Planet Gear Model

Ideal Gear Constraints and Gear Ratios

Sun-Planet imposes one kinematic and one geometric constraint on the three connected axes:

$$r_C \omega_C \quad = \quad r_S \omega_S \quad + \quad r_P \omega_P \quad , \quad r_C \quad = \quad r_P \quad + \quad r_S \quad .$$

The planet-sun gear ratio $g_{PS} = r_P/r_S = N_P/N_S$. N is the number of teeth on each gear. In terms of this ratio, the key kinematic constraint is:

$$\omega_S \quad = \quad -g_{PS} \omega_P \quad + \quad (1 \quad + \quad g_{PS}) \omega_C \quad .$$

The three degrees of freedom reduce to two independent degrees of freedom. The gear pair is (1,2) = (S,P).

Warning   The planet-sun gear ratio g_{PS} must be strictly greater than one.

The torque transfer is:

$$g_{PS} \tau_S \quad + \quad \tau_P \quad - \quad \tau_{loss} \quad = \quad 0 \quad ,$$

with $\tau_{loss} = 0$ in the ideal case.

Nonideal Gear Constraints and Losses

In the nonideal case, $\tau_{loss} \neq 0$. See "Model Gears with Losses".

Limitations

- Gear inertia is assumed negligible.
- Gears are treated as rigid components.
- Coulomb friction slows down simulation. See "Adjust Model Fidelity".

Ports

Port	Description
C	Rotational conserving port representing the planet carrier
P	Rotational conserving port representing the planet gear
S	Rotational conserving port representing the sun gear
H	Thermal conserving port for thermal modeling

Parameters

Main

Ring (R) to sun (S) teeth ratio (NP/NS)

Ratio g_{RS} of the ring gear wheel radius to the sun gear wheel radius. This gear ratio must be strictly greater than 1. The default value is 2.

Meshing Losses

Parameters for meshing losses vary with the block variant chosen—one with a thermal port for thermal modeling and one without it.

Without Thermal Port

Friction model

Select how to implement friction losses from nonideal meshing of gear teeth. The default is `No meshing losses`.

- `No meshing losses — Suitable for HIL simulation` — Gear meshing is ideal.
- `Constant efficiency` — Transfer of torque between gear wheel pairs is reduced by a constant efficiency η satisfying $0 < \eta \leq 1$. If you select this option, the panel changes from its default.

Constant Efficiency

Ordinary efficiency

Torque transfer efficiency η_{SP} for sun-planet gear wheel pair meshing.

Sun-carrier power threshold

Power threshold above which the full efficiency factor is in effect. Below this value, a hyperbolic tangent function smooths the efficiency factor, lowering the efficiency losses to zero when no power is transmitted. The default value is `0.001` W.

With Thermal Port

Temperature

Array of temperatures used to construct a 1-D temperature-efficiency lookup table. The array values must increase left to right. The temperature array must be the same size as the **Efficiency** array. The default array is `[280 300 320]` K.

Efficiency

Array of component efficiencies used to construct a 1-D temperature-efficiency lookup table. The array values are the efficiencies at the temperatures in the **Temperature** array. The two arrays must be the same size. The default array is `[0.95 0.9 0.85]`.

Sun-carrier power threshold

Power threshold above which the full efficiency factor is in effect. A hyperbolic tangent function smooths the efficiency factor between zero at rest and the value provided by the temperature-efficiency lookup table at the power threshold. The default value is `0.001` W.

Viscous Losses

Sun-carrier viscous friction coefficient

Viscous friction coefficient μ_S for the sun-carrier gear motion. The default is `0`.

From the drop-down list, choose units. The default is newton-meters/(radians/second) (`N*m/(rad/s)`).

Thermal Port

Thermal mass

Thermal energy required to change the component temperature by a single degree. The greater the thermal mass, the more resistant the component is to temperature change. The default value is 50 J/K.

Initial temperature

Component temperature at the start of simulation. The initial temperature alters the component efficiency according to an efficiency vector that you specify, affecting the starting meshing or friction losses. The default value is 300 K.

Real-Time Simulation

Hardware-in-the-Loop Simulation

For optimal simulation performance, use the **Meshing Losses** > **Friction model** parameter default setting, `No meshing losses - Suitable for HIL simulation`.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also

Simscape Blocks
Compound Planetary Gear | Planet-Planet | Planetary Gear | Ring-Planet | Sun-Planet Bevel

Topics
"Basic Motion, Torque, and Force Modeling"
"Model Thermal Losses in Driveline Components"

"Prepare Simscape Driveline Models for Real-Time Simulation Using Simscape Checks"

Sun-Planet Bevel

Planetary gear set of carrier, beveled planet, and sun wheels with adjustable gear ratio, assembly orientation, and friction losses

Library

Simscape / Driveline / Gears / Planetary Subcomponents

Description

The Sun-Planet Bevel gear block represents a set of carrier, planet, and sun gear wheels. The planet is connected to and rotates with respect to the carrier. The planet and sun corotate with a fixed gear ratio. You control the direction of rotation by setting the assembly orientation, left or right. A sun-planet and a ring-planet gear are basic elements of a planetary gear set. For model details, see "Sun-Planet Bevel Gear Model" on page 1-460.

Thermal Modeling

You can model the effects of heat flow and temperature change through an optional thermal conserving port. By default, the thermal port is hidden. To expose the thermal port, right-click the block in your model and, from the context menu, select **Simscape > Block choices**. Select a variant that includes a thermal port. Specify the associated thermal parameters for the component.

Sun-Planet Bevel Gear Model

Ideal Gear Constraints and Gear Ratios

Sun-Planet Bevel imposes one kinematic and one geometric constraint on the three connected axes.

$$r_C \omega_C = r_S \omega_S \pm r_P \omega_P$$

$$r_C = r_S \pm r_P$$

Where:

- r_C is the radius of the carrier gear.
- ω_C is the angular velocity of the carrier gear.
- r_S is the radius of the sun gear.
- ω_S is the angular velocity of the sun gear.
- r_P is the radius of the planet gear.
- ω_P is the angular velocity of the planet gear.

The planet-sun gear ratio is defined as

$$g_{PS} = \frac{r_P}{r_S} = \frac{N_P}{N_S},$$

where:

- g_{PS} is the planet-sun gear ratio. As $r_P > r_S$, $g_{PS} > 1$.

- N_P is the number of teeth in the planet gear.
- N_S is the angular velocity of the sun gear.

In terms of this ratio, the key kinematic constraint is:

- $\omega_S = g_{PS}\omega_P - \omega_C$ for a left-oriented bevel assembly
- $\omega_S = g_{PS}\omega_P + \omega_C$ for a right-oriented bevel assembly

The three degrees of freedom reduce to two independent degrees of freedom. The gear pair is $(1,2) = (S,P)$.

Warning The planet-sun gear ratio, g_{PS}, must be strictly greater than one.

The torque transfer is defined as

$$\tau_P = \tau_{loss} - g_{PS}\tau_S,$$

where:

- τ_{loss} is the torque loss.
- τ_s is the torque for the sun gear.
- τ_p is the torque for the planet gear.

For the ideal case, there is no torque loss, that is $\tau_{loss} = 0$. Then the torque transfer equation is $\tau_P = g_{PS}\tau_S$.

Nonideal Gear Constraints and Losses

In the nonideal case, $\tau_{loss} \neq 0$. See "Model Gears with Losses".

Limitations

- Gear inertia is assumed negligible.
- Gears are treated as rigid components.
- Coulomb friction slows down simulation. See "Adjust Model Fidelity".

Ports

Port	Description
C	Rotational conserving port representing the gear carrier
P	Rotational conserving port representing the planet gear
S	Rotational conserving port representing the sun gear

Port	Description
H	Thermal conserving port for thermal modeling

Parameters

Main

Planet (P) to sun (S) teeth ratio (NP/NS)

Ratio g_{PS} of the planet gear wheel radius to the sun gear wheel radius. This gear ratio must be strictly greater than 1. The default value is 2.

Assembly orientation

Relative orientation of sun and planet gears, controlling their corotation direction. Left or right orientation imply, respectively, that the gears corotate in the same or opposite direction.

The default is `Left — Sun and planet gears rotate in same direction`.

Meshing Losses

Parameters for meshing and friction losses vary with the block variant chosen—one with a thermal port for thermal modeling and one without it.

Without Thermal Port

Friction model

Select how to implement friction losses from nonideal meshing of gear teeth. The default is `No meshing losses`.

- `No meshing losses — Suitable for HIL simulation` — Gear meshing is ideal.

- `Constant efficiency` — Transfer of torque between gear wheel pairs is reduced by a constant efficiency η satisfying $0 < \eta \leq 1$. If you select this option, the panel changes from its default.

Constant Efficiency

Ordinary efficiency

Torque transfer efficiency η_{SP} for sun-planet gear wheel pair meshing.

Sun-carrier power threshold

Power threshold above which the full efficiency factor is in effect. Below this value, a hyperbolic tangent function smooths the efficiency factor, lowering the efficiency losses to zero when no power is transmitted. The default value is `0.001` W.

With Thermal Port

Temperature

Array of temperatures used to construct a 1-D temperature-efficiency lookup table. The array values must increase left to right. The temperature array must be the same size as the **Efficiency** array. The default array is `[280 300 320]` K.

Efficiency

Array of component efficiencies used to construct a 1-D temperature-efficiency lookup table. The array values are the efficiencies at the temperatures in the **Temperature** array. The two arrays must be the same size. The default array is `[0.95 0.9 0.85]`.

Sun-carrier power threshold

Power threshold above which the full efficiency factor is in effect. A hyperbolic tangent function smooths the efficiency factor between zero at rest and the value provided by the temperature-efficiency lookup table at the power threshold. The default value is `0.001` W.

Viscous Losses

Sun-carrier viscous friction coefficient

Viscous friction coefficient μ_S for the sun-carrier gear motion. The default is `0`.

From the drop-down list, choose units. The default is newton-meters/(radians/second) (`N*m/(rad/s)`).

Thermal Port

Thermal mass

Thermal energy required to change the component temperature by a single degree. The greater the thermal mass, the more resistant the component is to temperature change. The default value is 50 J/K.

Initial temperature

Component temperature at the start of simulation. The initial temperature alters the component efficiency according to an efficiency vector that you specify, affecting the starting meshing or friction losses. The default value is 300 K.

Real-Time Simulation

Hardware-in-the-Loop Simulation

For optimal simulation performance, use the **Meshing Losses** > **Friction model** parameter default setting, `No meshing losses - Suitable for HIL simulation`.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also

Simscape Blocks
Differential | Planet-Planet | Planetary Gear | Ring-Planet | Sun-Planet | Sun-Planet Worm Gear

Topics
"Basic Motion, Torque, and Force Modeling"
"Limited Slip Differential with Clutches"

"Custom Planetary Gear"
"Model Thermal Losses in Driveline Components"
"Prepare Simscape Driveline Models for Real-Time Simulation Using Simscape Checks"

Sun-Planet Worm Gear

Planetary gear set of carrier, worm planet, and sun wheels with adjustable gear ratio, worm thread type, and friction losses

Library

Simscape / Driveline / Gears / Planetary Subcomponents

Description

The Sun-Planet Worm Gear block represents a two-degree-of-freedom planetary gear built from carrier, sun, and planet gears. By type, the sun and planet gears are crossed helical spur gears arranged as a worm-gear transmission, in which the planet gear is a worm. Such transmissions are used in the Torsen type 1 differential. When transmitting power, the sun gear can be independently rotated by the worm (planet) gear, or by the carrier, or both.
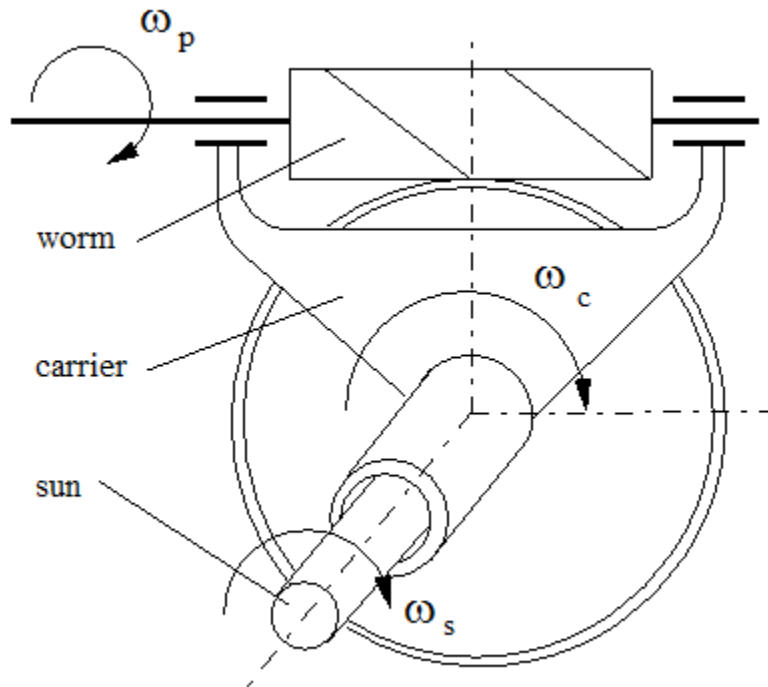
You specify a fixed gear ratio, which is determined as the ratio of the worm angular velocity to the sun gear angular velocity. You control the direction by setting the worm thread type, left-hand or right-hand. Rotation of the right-hand worm in positive direction causes the sun gear to rotate in positive direction too. The positive directions of the sun gear and the carrier are the same.

Thermal Modeling

You can model the effects of heat flow and temperature change through an optional thermal conserving port. By default, the thermal port is hidden. To expose the thermal port, right-click the block in your model and, from the context menu, select **Simscape > Block choices**. Select a variant that includes a thermal port. Specify the associated thermal parameters for the component.

Sun-Planet Worm Gear Model

Model Variables

R_{WG}	Gear, or transmission, ratio determined as the ratio of the worm angular velocity to the gear angular velocity. The ratio is positive for the right-hand worm and negative for the left-hand worm.
ω_S	Angular velocity of the sun gear
ω_P	Planet (that is, worm) angular velocity
ω_C	Carrier angular velocity
ω_{SC}	Angular velocity of the sun with respect to the carrier
α	Normal pressure angle
λ	Worm lead angle
L	Worm lead
d	Worm pitch diameter
τ_S	Torque applied to the sun shaft
τ_P	Torque applied to the planet shaft
τ_C	Torque applied to the carrier shaft
τ_{loss}	Torque loss due to meshing friction. The loss depends on the device efficiency and the power flow direction. To avoid abrupt change of the friction torque at $\omega_S = 0$, the friction torque is introduced via the hyperbolic function.
τ_{instfr}	Instantaneous value of the friction torque added to the model to simulate friction losses
τ_{fr}	Steady-state value of the friction torque
k	Friction coefficient
η_{WG}	Efficiency for worm-gear power transfer
η_{GW}	Efficiency for gear-worm power transfer
p_{th}	Power threshold
μ_{SC}	Sun-carrier viscous friction coefficient

μ_{WC}	Worm-carrier viscous friction coefficient

Ideal Gear Constraints and Gear Ratio

Sun-planet worm gear imposes one kinematic constraint on the three connected axes:

$$\omega_S = \omega_P/R_{WG} + \omega_C .$$

The gear has two independent degrees of freedom. The gear pair is (1,2) = (S,P).

The torque transfer is:

$$R_{WG}\tau_P + \tau_S - \tau_{loss} = 0 ,$$

$$\tau_C = - \tau_S,$$

with $\tau_{loss} = 0$ in the ideal case.

Nonideal Gear Constraints

For general considerations on nonideal gear modeling, see "Model Gears with Losses".

In a nonideal gear, the angular velocity and geometric constraints are unchanged. But the transferred torque and power are reduced by:

- Coulomb friction between thread surfaces on W and G, characterized by friction coefficient k or constant efficiencies [η_{WG}, η_{GW}]
- Viscous coupling of driveshafts with bearings, parametrized by viscous friction coefficients μ_{SC} and μ_{WC}

Because the transmission incorporates a worm gear, the efficiencies are different for the direct and reverse power transfer. The following table shows the value of the efficiency for all combinations of the power transfer.

Driving shaft	Driven shaft		
	Planet	Sun	Carrier
Planet	n/a	η_{WG}	η_{WG}
Sun	η_{GW}	n/a	No loss

Carrier	η_{GW}	No loss	n/a

Geometric Surface Contact Friction

In the contact friction case, η_{WG} and η_{GW} are determined by:

- The worm-gear threading geometry, specified by lead angle λ and normal pressure angle α.
- The surface contact friction coefficient k.

$$\eta_{WG} = (\cos\alpha - k\cdot\tan\lambda)/(\cos\alpha + k/\tan\lambda) \quad ,$$

$$\eta_{GW} = (\cos\alpha - k/\tan\lambda)/(\cos\alpha + k\cdot\tan\alpha) \quad .$$

Constant Efficiencies

In the constant efficiency case, you specify η_{WG} and η_{GW}, independently of geometric details.

Self-Locking and Negative Efficiency

If you set efficiency for the reverse power flow to a negative value, the train exhibits *self-locking*. Power cannot be transmitted from sun gear to worm and from carrier to worm unless some torque is applied to the worm to release the train. In this case, the absolute value of the efficiency specifies the ratio at which the train is released. The smaller the train lead angle, the smaller the reverse efficiency.

Meshing Efficiency

The efficiencies η of meshing between worm and gear are fully active only if the transmitted power is greater than the power threshold.

If the power is less than the threshold, the actual efficiency is automatically regularized to unity at zero velocity.

Viscous Friction Force

The viscous friction coefficients of the worm-carrier and sun-carrier bearings control the viscous friction torque experienced by the carrier from lubricated, nonideal gear threads. For details, see "Nonideal Gear Constraints" on page 1-469.

Limitations

- Gear inertia is assumed negligible.
- Gears are treated as rigid components.
- Coulomb friction slows down simulation. See "Adjust Model Fidelity".

Ports

Port	Description
C	Rotational conserving port representing the gear carrier
W	Rotational conserving port representing the worm gear
S	Rotational conserving port representing the sun gear
H	Thermal conserving port for thermal modeling

Parameters

Main

Gear ratio

Gear or transmission ratio R_{WG} determined as the ratio of the worm angular velocity to the gear angular velocity. The default is 25.

Worm thread type

Choose the directional sense of gear rotation corresponding to positive worm rotation. The default is `Right-hand`. If you select `Left-hand`, rotation of the worm in the generally-assigned positive direction results in the gear rotation in negative direction.

Meshing Losses

Parameters for meshing and friction losses vary with the block variant chosen—one with a thermal port for thermal modeling and one without it.

Without Thermal Port

Friction model

Select how to implement friction losses from nonideal meshing of gear threads. The default is `No meshing losses`.

- `No meshing losses — Suitable for HIL simulation` — Gear meshing is ideal.
- `Constant efficiency` — Transfer of torque and force between worm and gear is reduced by friction. If you select this option, the panel expands.

Constant Efficiency

Friction parametrization

Select how to characterize the friction between gear threads. The default is `Friction coefficient and geometrical parameters`.

- `Friction coefficient and geometrical parameters` — Friction is determined by contact friction between surfaces.

    Normal pressure angle

    The thread pressure angle α in the normal plane. Must be greater than zero and less than 90 degrees. The default is `17.5`.

    From the drop-down list, choose units. The default is degrees (`deg`).

    Lead angle

    Thread helix angle $\lambda = \arctan[L/(\pi d)]$. L is the worm lead, d is the worm pitch diameter. Must be greater than zero. The default is `20`.

    From the drop-down list, choose units. The default is degrees (`deg`).

    Friction coefficient

    Dimensionless coefficient of normal friction in the thread. Must be greater than zero. The default is `0.08`.

- `Efficiencies` — Friction is determined by constant efficiencies $0 < \eta < 1$. If you select this option, the panel changes from its default.

Efficiencies

Worm-gear efficiency

Efficiency η_{WG} of the power transfer from worm to gear.

Gear-worm efficiency

Efficiency η_{GW} of the power transfer from gear to worm.

Power threshold

Power threshold above which the full efficiency factor is in effect. Below this value, a hyperbolic tangent function smooths the efficiency factor, lowering the efficiency losses to zero when no power is transmitted. The default value is `0.001` W.

With Thermal Port

Temperature

Array of temperatures used to construct a 1-D temperature-efficiency lookup table. The array values must increase left to right. The temperature array must be the same size as the **Efficiency** array. The default array is `[280 300 320]` K.

Worm-gear efficiency

Array of component efficiencies with the worm as the driver—that is, with power flowing from the worm to the gear. The array values are the efficiencies at the temperatures in the **Temperature** array. The two arrays must be the same size. The default array is `[0.75 0.7 0.65]`.

Gear-worm efficiency

Array of component efficiencies with the gear as the driver—that is, with power flowing from the gear to the worm. The array values are the efficiencies at the temperatures in the **Temperature** array. The two arrays must be the same size. The default array is `[0.5 0.45 0.4]`.

Power threshold

Absolute value of the angular velocity above which the full efficiency factor is in effect. A hyperbolic tangent function smooths the efficiency factor between zero at rest and the value provided by the temperature-efficiency lookup table at the power threshold. The default value is `0.001` W.

Viscous Losses

Worm-carrier and sun-carrier viscous friction coefficients

Vector of viscous friction coefficients $[\mu_{WC}\ \mu_{SC}]$, for the worm-carrier and sun-carrier shafts, respectively. The default is `[0 0]`.

From the drop-down list, choose units. The default is newton-meters/(radians/second) (`N*m/(rad/s)`).

Thermal Port

Thermal mass

Thermal energy required to change the component temperature by a single degree. The greater the thermal mass, the more resistant the component is to temperature change. The default value is 50 J/K.

Initial temperature

Component temperature at the start of simulation. The initial temperature alters the component efficiency according to an efficiency vector that you specify, affecting the starting meshing or friction losses. The default value is 300 K.

Real-Time Simulation

Hardware-in-the-Loop Simulation

For optimal simulation performance, use the **Meshing Losses > Friction model** parameter default setting, `No meshing losses - Suitable for HIL simulation`.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also

Simscape Blocks
Leadscrew | Sun-Planet | Sun-Planet Bevel | Worm Gear
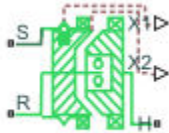
Topics
"Basic Motion, Torque, and Force Modeling"
"Torsen Differential"
"Model Thermal Losses in Driveline Components"
"Prepare Simscape Driveline Models for Real-Time Simulation Using Simscape Checks"

Synchronizer

Cone clutch, dog clutch, and translational detent assembled to provide smooth gear engagement

Library

Simscape / Driveline / Clutches

Description

The block represents a synchronizer that contains a dog clutch, a cone clutch, and a translational detent. The shift linkage first translates to engage the cone clutch. Frictional torque causes the shift linkage and cone clutch shaft to rotate at equal speed. When the force acting on the shift linkage exceeds the detent force, the dog clutch can engage.
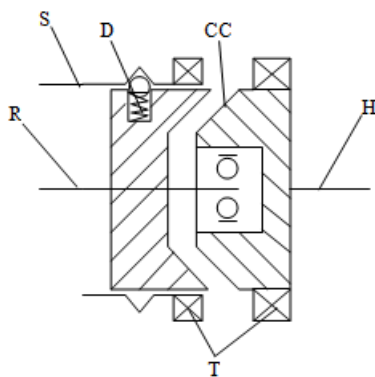
The schematic illustrates a synchronizer in the disengaged state. In this state, the ring (R) and hub (H) shafts can spin independently at different speeds. To synchronize ring and hub shaft speeds, the shift linkage (S) translates toward the hub shaft to engage the cone clutch. The friction surfaces of the cone clutch produce a frictional torque that equalizes the rotational speeds of the ring and hub shafts. The dog clutch teeth (T) can engage when the translational force acting on the shift linkage exceeds the peak detent force. The peak detent force should allow sufficient time and normal force to equalize ring and hub shaft speeds so that the dog clutch can engage.
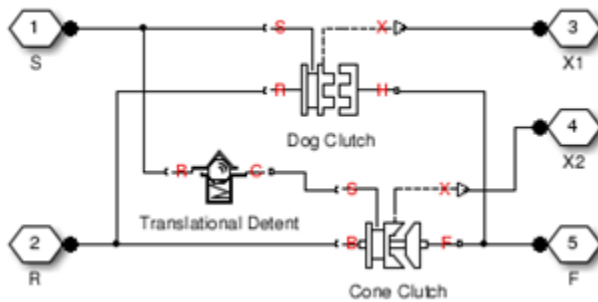
The model implements Dog Clutch, Translational Detent and a modified version of Cone Clutch blocks. Refer to each block reference page for more information on the corresponding block function. You can use a similar approach to model customized versions of the synchronizer. One example is the Transmission (Detailed) subsystem in the model sdl_vehicle_manual_transmission.

Connections R and H are mechanical rotational conserving ports that represent the ring (R) and hub (H), respectively. Connection S is a mechanical translational conserving port that represents the ring shifter handle.

Connections X1 and X2 are physical signal outputs that report the shift linkage positions of the dog clutch and cone clutch, respectively. The shift linkage positions are zero when the clutch is fully disengaged. When the dog clutch is fully engaged, the dog clutch shift linkage position has a magnitude equal to the sum of the dog clutch ring-hub gap and the tooth height. When cone clutch is fully engaged, the cone clutch shift linkage position has a magnitude equal to the cone clutch ring-hub gap.

Thermal Modeling

You can model the effects of heat flow and temperature change through an optional thermal conserving port. By default, the thermal port is hidden. To expose the thermal port, right-click the block in your model and, from the context menu, select **Simscape > Block choices**. Select a variant that includes a thermal port. Specify the associated thermal parameters for the component.

Assumptions and Limitations

- The model does not account for inertia effects. You can add a Simscape Inertia block at each port to add inertia to the synchronizer model.

Ports

S

    Conserving translational port that represents the shift linkage

R

    Conserving rotational port that represents the ring shaft

H

    Conserving rotational port that represents the hub shaft

X1

    Physical signal output port that measures the magnitude of the dog clutch translation

X2

    Physical signal output port that measures the magnitude of the cone clutch translation

T

    Thermal conserving port. The thermal port is optional and is hidden by default. To expose the port, select a variant that includes a thermal port.

Parameters

Dog Clutch

The methods that are available for parameterizing the torque transmission depend on the variant of the block that you use. If you use the default variant, which has no thermal port, you parameterize the block using one of the options for the **Torque Transmission Model** parameter. If you select a variant that has a thermal port, you parameterize the block based on temperature.

Clutch teeth mean radius

Mean radius of the clutch teeth. The parameter must be greater than zero. The default value is 50 mm.

Maximum engagement speed

The relative angular speed between the ring and hub shafts above which the dog clutch cannot engage. The value is specific to the specific gearbox or transmission and must be minimized to avoid high dynamic impact during engagement. The default value is `inf rad/s`.

Additional Parameters for the Default Block Variant

Torque Transmission Model

Model used to represent torque transmission. The default setting is `Friction clutch approximation - Suitable for HIL and linearization`.

`Friction clutch approximation - Suitable for HIL and linearization`

Maximum transmitted torque

Maximum expected torque across the clutch. The default value is `1000 N*m`.

`Dynamic with Backlash`

Number of teeth

Number of teeth in the clutch. The parameter must be greater than or equal to one. The default value is `6`.

Rotational backlash

Angular play of the ring with respect to the hub. Greater backlash values correspond to wider tooth gaps that makes engagement easier. The parameter must be greater than zero. The default value is `10 deg`.

Torsional stiffness

Stiffness of the tooth contact. The model assumes deformation due to tooth interaction is elastic. The angular deformation at the contact surface is proportional to the applied torque and the reciprocal of the contact stiffness. The parameter must be greater than zero. The default value is `1e+7 N*m/rad`.

Torsional damping

Damping at the tooth contact surface. The value of the parameter is inversely proportional to the number of oscillations following impact. The parameter must be greater than zero. The default value is `100 N*m/(rad/s)`.

Tooth-tooth friction coefficient

Friction coefficient for the translational relative motion between the hub and ring teeth. The value of the parameter depends on the tooth material, lubrication, and quality of contacting surfaces. The parameter must be greater than zero. The default value is `0.05`.

Additional Parameters for the Thermal Block Variant

Temperature vector

Specify the input values for the temperature as a vector. The minimum number of values depends on the interpolation method that you select. For linear interpolation, provide at least two values per dimension. For smooth interpolation, provide at least three values per dimension. The values in the vector must increase from left to right. The default value is `[280.0, 300.0, 320.0] K`.

Maximum transmitted torque vector

Largest torque that the clutch can transmit, corresponding to a nonslip engaged configuration, specified as a vector. If the torque transmitted between the ring and the hub exceeds this value, the two components begin to slip with respect to each other. This torque determines the static friction limit in the friction clutch approximation. The vector has the same number of elements as the temperature vector. The default value is `[1000.0, 1050.0, 1000.0] N*m`.

Interpolation method

Select one of these interpolation methods to approximate the output value when the input value is between two consecutive grid points:

- `Linear` — Select this, default, option to get the best performance.
- `Smooth` — Select this option to produce a continuous curve with continuous first-order derivatives.

For more information on interpolation algorithms, see the PS Lookup Table (1D) block reference page.

Extrapolation method

Select one of these extrapolation methods to determine the output value when the input value is outside the range specified in the argument list:

- `Linear` — Select this, default, option to produce a curve with continuous first-order derivatives in the extrapolation region and at the boundary with the interpolation region.
- `Nearest` — Select this option to produce an extrapolation that does not go above the highest point in the data or below the lowest point in the data.

For more information on extrapolation algorithms, see the PS Lookup Table (1D) block reference page.

Cone Clutch

Common parameters for the cone clutch are:

Contact surface maximum diameter

Outer diameter of the friction material linen that lines the cone clutch disks. The parameter must be greater than zero. The default value is `150 mm`.

Contact surface minimum diameter

Inner diameter of the friction material linen that lines the cone clutch disks. The parameter must be greater than zero but smaller than the value of **Contact surface maximum diameter**.

Cone half angle

Half angle of the clutch cone. The parameter must be greater than zero and less than or equal to 90 degrees. The default value is `12 deg`.

Velocity tolerance

Relative velocity below which the two surfaces can lock. The surfaces lock if the torque across the B and F rotational ports is less than the product of the effective

radius, the static friction coefficient, and the applied normal force. The default value is `0.001 rad/s`.

Threshold force

The normal force applied to the physical signal port **N** is applied to the contact only if the amount of force exceeds the value of the **Threshold force** parameter. Forces below the **Threshold force** are not applied, and there is consequently no transmitted frictional torque. The default value is `1 N`.

Friction model

Select a parameterization method to model the kinetic friction coefficient. The options and default values for this parameter depend on the variant that you select for the block. The options are:

- `Fixed kinetic friction coefficient` — Provide a fixed value for the kinetic friction coefficient. This option:

  - Is visible only if you use the default variant of the block
  - Is the default method for parameterizing the default variant of the block
  - Affects the visibility of other parameters

  Fixed kinetic friction coefficient
  Kinetic friction coefficient

  The kinetic, or Coulomb, friction coefficient. The coefficient must be greater than zero. The default value is `0.3`.

  Static friction coefficient

  The static, or peak, value of the friction coefficient. The static friction coefficient must be greater than the kinetic friction coefficient. The default value is `0.35`.

- `Table lookup kinetic friction coefficient` — Define the kinetic friction coefficient by one-dimensional table lookup based on the relative angular velocity between disks. This option:

  - Is visible only if you use the default variant of the block
  - Affects the visibility of other parameters

  Table lookup kinetic friction coefficient
  Kinetic friction coefficient relative velocity vector

  Specify input values for the relative velocity as a vector. The values in the vector must increase from left to right. The minimum number of values

depends on the interpolation method that you select. For linear interpolation, provide at least two values per dimension. For smooth interpolation, provide at least three values per dimension. The default value is `[0.0, 100.0, 1000.0] rad/s`.

Kinetic friction coefficient vector

Specify the output values for the kinetic friction coefficient as a vector. All values must be positive. The default value is `[0.3, 0.22, 0.19]`.

Friction coefficient interpolation method

Select one of these interpolation methods to approximate the output value when the input value is between two consecutive grid points:

- `Linear` — Select this, default, option to get the best performance.
- `Smooth` — Select this option to produce a continuous curve with continuous first-order derivatives.

For more information on interpolation algorithms, see the PS Lookup Table (1D) block reference page.

Friction coefficient extrapolation method

Select one of these extrapolation methods to determine the output value when the input value is outside the range specified in the argument list:

- `Linear` — Select this, default, option to produce a curve with continuous first-order derivatives in the extrapolation region and at the boundary with the interpolation region.
- `Nearest` — Select this option to produce an extrapolation that does not go above the highest point in the data or below the lowest point in the data.

For more information on extrapolation algorithms, see the PS Lookup Table (1D) block reference page.

Static friction coefficient

The static, or peak, value of the friction coefficient. The static friction coefficient must be greater than the kinetic friction coefficient. The default value is `0.35`.

- `Temperature-dependent kinetic friction coefficient` — Define the kinetic friction coefficient by table lookup based on the temperature. This option:

  - Is visible only if you use a thermal variant of the block

- Is the default method for parameterizing the thermal variant of the block
- Affects the visibility of other parameters

`Temperature-dependent kinetic friction coefficient`

Temperature vector

Specify the input values for the temperature as a vector. The minimum number of values depends on the interpolation method that you select. For linear interpolation, provide at least two values per dimension. For smooth interpolation, provide at least three values per dimension. The values in the vector must increase from left to right. The default value is `[280.0, 300.0, 320.0]` K.

Kinetic friction coefficient vector

Specify the output values for kinetic friction coefficient as a vector. The vector must have the same number of elements as the temperature vector. All values must be greater than zero. The default value is `[0.3, 0.28, 0.25]`.

Static friction coefficient vector

Specify the static, or peak, values of the friction coefficient as a vector. The vector must have the same number of elements as the temperature vector. Each value must be greater than the value of the corresponding element in the kinetic friction coefficient vector. The default value is `[0.4, 0.38, 0.36]`.

Friction coefficient interpolation method

Select one of these interpolation methods to approximate the output value when the input value is between two consecutive grid points:

- `Linear` — Select this, default, option to get the best performance.
- `Smooth` — Select this option to produce a continuous curve with continuous first-order derivatives.

For more information on interpolation algorithms, see the PS Lookup Table (1D) block reference page.

Friction coefficient extrapolation method

Select one of these extrapolation methods to determine the output value when the input value is outside the range specified in the argument list:

- `Linear` — Select this, default, option to produce a curve with continuous first-order derivatives in the extrapolation region and at the boundary with the interpolation region.

- **Nearest** — Select this option to produce an extrapolation that does not go above the highest point in the data or below the lowest point in the data.

  For more information on extrapolation algorithms, see the PS Lookup Table (1D) block reference page.

- **Temperature and speed-dependent kinetic friction coefficient** — Define the kinetic friction coefficient by table lookup based on the temperature and the relative angular velocity between disks. This option:

  - Is visible only if you use the default variant of the block
  - Affects the visibility of other parameters

Temperature and speed-dependent kinetic friction coefficient

Temperature vector

Specify the input values for the temperature as a vector. The minimum number of values depends on the interpolation method that you select. For linear interpolation, provide at least two values per dimension. For smooth interpolation, provide at least three values per dimension. The values in the vector must increase from left to right. The default value is `[280.0, 300.0, 320.0]` K.

Kinetic friction coefficient relative velocity vector

Specify input values for the relative velocity as a vector. The values in the vector must increase from left to right. The minimum number of values depends on the interpolation method that you select. For linear interpolation, provide at least two values per dimension. For smooth interpolation, provide at least three values per dimension. The default value is `[0.0, 100.0, 1000.0]` rad/s.

Kinetic friction coefficient matrix

Specify the output values for kinetic friction coefficient as a matrix. All the values must be greater than zero. The size of the matrix must equal the size of the matrix that is the result of the temperature vector × the kinetic friction coefficient relative velocity vector. The default value is `[0.34, 0.32, 0.3; 0.3, 0.28, 0.25; 0.25, 0.2, 0.15]`.

Static friction coefficient vector

Specify the static, or peak, values of the friction coefficient as a vector. The vector must have the same number of elements as the temperature vector. Each value must be greater than the value of the corresponding element in the kinetic friction coefficient vector. The default value is `[0.4, 0.38, 0.36]`.

Friction coefficient interpolation method

Select one of these interpolation methods to approximate the output value when the input value is between two consecutive grid points:

- `Linear` — Select this, default, option to get the best performance.
- `Smooth` — Select this option to produce a continuous curve with continuous first-order derivatives.

For more information on interpolation algorithms, see the PS Lookup Table (1D) block reference page.

Friction coefficient extrapolation method

Select one of these extrapolation methods to determine the output value when the input value is outside the range specified in the argument list:

- `Linear` — Select this, default, option to produce a curve with continuous first-order derivatives in the extrapolation region and at the boundary with the interpolation region.
- `Nearest` — Select this option to produce an extrapolation that does not go above the highest point in the data or below the lowest point in the data.

For more information on extrapolation algorithms, see the PS Lookup Table (1D) block reference page.

Detent

Peak force

Peak shear force of the detent. The default value is `500 N`.

Notch width

Width of the region where the detent exhibits shear force. The default value is `3 mm`.

Viscous friction coefficient

Viscous friction coefficient at the contact surface of the detent. The parameter must be greater than or equal to zero. The default value is `0.1 N/(m/s)`.

Friction to peak force ratio

Ratio of the kinetic friction to the peak shear force of the detent. The parameter is used to set the value of the kinetic friction. The parameter must be greater than or equal to zero. The default value is `0.01`.

Friction velocity threshold

Velocity required for peak kinetic friction at the contact surface of the detent. The parameter ensures the force is continuous when the travel direction changes, increasing the numerical stability of the simulation. The parameter must be greater than zero. The default value is `0.05 m/s`.

Shift Linkage

Linkage travel direction

Direction the shift linkage must translate along to engage the clutch. The default setting is `Positive shift linkage displacement engages clutch`

Ring-hub clearance when dog clutch disengaged

Distance between ring and dog clutch hub with the ring fully retracted. The parameter specifies the maximum travel distance if engagement requirements are not met. The parameter must be greater than the value of **Ring-hub clearance when cone clutch disengaged**. The default value is `5 mm`.

Tooth overlap to engage

Minimum value of the tooth overlap beyond which the dog clutch is considered engaged. The parameter must be greater than zero. The default value is `3 mm`.

Tooth height

Height of dog clutch teeth. The parameter must be greater than zero. The default value is `10 mm`.

Ring-hub clearance when cone clutch disengaged

Distance between the ring and cone clutch hub when the ring is fully retracted. The cone clutch can begin to engage once the slider has translated this distance. The ring can travel only this distance if the engagement requirements are not satisfied. The parameter must be greater than zero. The default value is `3 mm`.

Hard stop at back of shift linkage

Hard stop that prevents the shift linkage from traveling beyond the fully disengaged position. Choices include:

- `No hard stop when fully disengaged`
- `Hard stop when fully disengaged`

The default setting is `Hard stop when fully disengaged`.

Dog clutch ring stop stiffness

Stiffness of the hard stops on both sides of the dog clutch ring. The model assumes the ring and stops behave elastically. Contact deformation is proportional to the applied force and the reciprocal of the contact stiffness. The value of the stiffness must be assigned with reference to the parameter **Tooth overlap to engage**. Too low a stiffness could cause the deformation to exceed the required overlap and initiate a false engagement. The parameter must be greater than zero. The default value is `1e +6 N/m`.

Cone clutch ring stop stiffness

Stiffness of the hard stops on both sides of the cone clutch ring. The model assumes the ring and stops behave elastically. Contact deformation is proportional to the applied force and the reciprocal of the contact stiffness. The parameter must be greater than zero. The default value is `1e+6 N/m`.

Dog clutch ring stop damping

Translational contact damping between the dog clutch ring and the hub. The value of the damping is inversely proportional to the number of oscillations that occur after impact. The parameter must be greater than zero. The default value is `1e+3 N/(m/s)`.

Cone clutch ring stop damping

Translational contact damping between the cone clutch ring and the hub. The value of damping is inversely proportional to the number of oscillations that occur after impact. The parameter must be greater than zero. The default value is `1e+3 N/(m/s)`.

Viscous friction coefficient

Viscous friction coefficient for the relative translational motion between the hub and the ring. The value of the parameter depends on lubrication state and quality of contacting surfaces. The coefficient must be greater than or equal to zero. The default value is `100 N/(m/s)`.

Initial Conditions

Initial state

Beginning configuration of cone and dog clutches. Choices include:

- `All clutches unlocked` — Cone and dog clutches transmit zero torque between the ring and hub shafts.

- `Cone clutch locked` — Cone clutch transmits torque between the ring and hub shafts.
- `All clutches locked` — Cone and dog clutches transmit torque between the ring and hub shafts.

The default value is `All clutches unlocked`.

Dog clutch initial shift linkage position

Initial position of the shift linkage section that attaches to the dog clutch. The value of the parameter has the following restrictions:

Linkage Travel Direction	Dog Clutch State	Parameter Restriction
`Positive shift linkage displacement engages clutch`	Initially engaged	Parameter must be greater than the sum of parameters **Ring-hub clearance when dog clutch disengaged** and **Tooth overlap to engage**
	Initially disengaged	Parameter must be smaller than the sum of parameters **Ring-hub clearance when dog clutch disengaged** and **Tooth overlap to engage**
`Negative shift linkage displacement engages clutch`	Initially engaged	Negative of the parameter must be greater than the sum of parameters **Ring-hub clearance when dog clutch disengaged** and **Tooth overlap to engage**

Linkage Travel Direction	Dog Clutch State	Parameter Restriction
	Initially disengaged	Negative of the parameter must be smaller than the sum of parameters **Ring-hub clearance when dog clutch disengaged** and **Tooth overlap to engage**

The default value is 0 mm.

Cone clutch initial shift linkage position

Initial position of the shift linkage section that attaches to the cone clutch. The value of the parameter has the following restrictions:

Linkage Travel Direction	Dog Clutch State	Parameter Restriction
Positive shift linkage displacement engages clutch	Initially engaged	Parameter must be greater than the value of **Ring-hub clearance when cone clutch disengaged**
	Initially disengaged	Parameter must be smaller than the value of **Ring-hub clearance when cone clutch disengaged**
Negative shift linkage displacement engages clutch	Initially engaged	Negative of the parameter must be greater than the value of **Ring-hub clearance when dog cone disengaged**

Linkage Travel Direction	Dog Clutch State	Parameter Restriction
	Initially disengaged	Negative of the parameter must be smaller than the value of **Ring-hub clearance when dog cone disengaged**

The default value is `0 mm`.

Initial dog clutch ring-hub offset angle

Initial angle between the ring and hub portions of the dog clutch. If the clutch is disengaged, the initial angle must lie in the range $-\pi/N \leq \theta \leq +\pi/N$, where N denotes the number of teeth in the dog clutch. If the clutch is engaged, the initial angle must be in the range $-\delta/2 \leq \theta \leq +\delta/2$, where δ is the backlash angle. The default value is `0 deg`.

This parameter is only visible if you select a block variant that does not include a thermal port and set **Dog Clutch > Torque transmission model** to `Dynamic with backlash`.

Thermal Port

These thermal parameters are visible only when you select a block variant that includes a thermal port.

Thermal mass

Thermal energy required to change the component temperature by a single degree. The greater the thermal mass, the more resistant the component is to temperature change. The default value is `50 kJ/K`.

Initial temperature

Component temperature at the start of simulation. The initial temperature alters the component efficiency according to an efficiency vector that you specify, affecting the starting meshing or friction losses. The default value is `300 K`.

Linearization and Real-Time Simulation

Linearization

To optimize your model for linearization, use the **Dog Clutch** > **Torque transmission model** parameter default setting, `Friction clutch approximation - Suitable for HIL and linearization.`

Hardware-in-the-Loop Simulation

For optimal simulation performance, use the **Dog Clutch** > **Torque transmission model** parameter default setting, `Friction clutch approximation - Suitable for HIL and linearization.`

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also
Cone Clutch | Dog Clutch | Double-Sided Synchronizer | Translational Detent

Topics
"Vehicle with Manual Transmission"
"Model Thermal Losses in Driveline Components"
"Prepare Simscape Driveline Models for Real-Time Simulation Using Simscape Checks"

Tire (Friction Parameterized)

Tire with friction parameterized in terms of static and kinetic coefficients
Library: Simscape / Driveline / Tires & Vehicles

Description

The Tire (Friction Parameterized) block models a tire with friction parameterized in terms of static and kinetic coefficients. The static friction coefficient determines the applied torque at which the tire loses traction and begins to slip. The kinetic friction coefficient determines the amount of torque that the tire transmits to the pavement once it begins to slip. The tire regains traction once its relative velocity over the pavement falls below the traction velocity tolerance specified.

To increase the fidelity of the tire model, the block enables you to specify properties such as tire compliance, inertia, and rolling resistance. However, these properties increase the complexity of the tire model and can slow down simulation. Consider ignoring tire compliance and inertia if simulating the model in real time or if preparing the model for hardware-in-the-loop (HIL) simulation.

The traction state model of this block is based on the traction state model of the Fundamental Friction Clutch block. For more information on the state model, see "Clutch States" on page 1-224.

Ports

Input

N — Normal force
physical signal

Physical signal input port associated with the normal force acting on the tire. The normal force is positive if it acts downward on the tire, pressing it against the pavement.

M — Friction coefficients
physical signal | vector | $[\mu_s, \mu_k]$

Physical signal input port associated with the static (μ_s) and kinetic (μ_k) friction coefficients. Provide the friction coefficients as a two-element vector, specified in the order $[\mu_s, \mu_k]$.

Dependencies

Port **M** is exposed only if the **Main > Friction model** parameter is set to `Physical signal friction coefficients`. For more information, see Main Parameter Dependencies.

Output

S — Slip
physical signal

Physical signal output port associated with the relative slip between the tire and road.

Conserving

A — Axle
mechanical rotational

Mechanical rotational port associated with the axle that the tire sits on.

H — Hub
mechanical translational

Mechanical translational port associated with the wheel hub that transmits the thrust generated by the tire to the remainder of the vehicle.

Parameters

Main

Specify characteristics of the tire, such as the rolling radius or static friction coefficient. The table shows how the visibility of some **Main** parameters depends on the options that you choose for other parameters. To learn how to read the table, see "Parameter Dependencies" on page B-2.

Main Parameter Dependencies

Main		
Rolling radius		
Friction model — Choose `Fixed kinetic friction coefficient`, `Table lookup kinetic friction coefficient`, or `Physical signal friction coefficients`		
Fixed kinetic friction coefficient	**Table lookup kinetic friction coefficient**	**Physical signal friction coefficients** — Exposes physical signal input port **M** for providing the static (μ_s) and kinetic (μ_k) friction coefficients to the block as an array of two elements in the order [μ_s, μ_k].
Static friction coefficient		
Kinetic friction coefficient	Tire slip vector	
	Kinetic friction coefficient vector	
	Interpolation method	
	Extrapolation method	

Rolling radius — Unloaded tire-wheel radius
`0.3 m` (default) | positive scalar

Distance between the pavement and the center of the tire.

Friction model — Friction model
`Fixed kinetic friction coefficient` (default) | `Table lookup kinetic friction coefficient` | `Physical signal friction coefficients`

The block provides three friction models. The default model, `Fixed kinetic friction coefficient`, uses constant static and kinetic friction coefficients that you specify.

To specify friction using table lookup, set the **Friction model** parameter to `Table lookup kinetic friction coefficient`. While this model treats the static coefficient as a constant, it treats the kinetic coefficient as a constant or function of tire slip. Use this model to model tire dynamics under constant pavement conditions.

To model tire dynamics under variable pavement conditions, set the **Friction model** parameter to `Physical signal friction coefficients`. Selecting this model exposes physical signal inport **M**. Use the **M** port to provide the static (μ_s) and kinetic (μ_k) friction coefficients to the block as an array of two elements in the order [μ_s, μ_k].

Dependencies

Each friction model option exposes related parameters and hides unrelated parameters. Selecting `Physical signal friction coefficients` exposes physical signal input port **M**. For more information, see Main Parameter Dependencies.

`Static friction coefficient` — **Static friction coefficient**
`0.90` (default) | positive scalar

Ratio of the allowable longitudinal force to the normal force allowed before the tire begins to slip (μ_s). The parameter must be greater than either the kinetic friction coefficient or the largest value in the kinetic friction coefficient vector.

Dependencies

This parameter is visible when the **Friction model** parameter is set to `Fixed kinetic friction coefficient` or `Table lookup kinetic friction coefficient`. For more information, see Main Parameter Dependencies.

`Kinetic friction coefficient` — **Kinetic friction coefficient**
`0.70` (default) | positive scalar

Ratio of the transmitted longitudinal force to the normal force allowed during tire slip (μ_k). The ratio must be greater than zero.

Dependencies

This parameter is visible when the **Friction model** parameter is set to `Fixed kinetic friction coefficient`. For more information, see Main Parameter Dependencies.

`Tire slip vector` — **Tire slip vector**
`[0, .02, .06, .15, .6, 1]` rad/s (default) | vector

Vector of tire slip values that correspond to the kinetic friction coefficients in the **Kinetic friction coefficient vector** parameter. The vectors must be the same size. If the **Tire slip vector** parameter contains only nonnegative values, the slip versus friction function is assumed to be symmetric about the slip axis.

Dependencies

This parameter is visible when the **Friction model** parameter is set to `Table lookup kinetic friction coefficient`. For more information, see Main Parameter Dependencies.

Kinetic friction coefficient vector — Kinetic friction coefficient
`[.89, .88, .8, .75, .7, .7]` (default) | vector

Vector of kinetic friction coefficients that correspond to the tire slip values specified in the **Tire slip vector** parameter. The vectors must be the same size.

Dependencies

This parameter is visible when the **Friction model** parameter is set to `Table lookup kinetic friction coefficient`. For more information, see Main Parameter Dependencies.

Interpolation method — Interpolation method
`Linear` (default) | `Smooth`

Interpolation method for the lookup table to use for processing the tire slip-kinetic friction coefficient characteristic. To prioritize performance, select `Linear`. To produce a continuous curve with continuous first-order derivatives, select `Smooth`.

For more information on interpolation algorithms, see the PS Lookup Table (1D) block reference page.

Dependencies

This parameter is visible when the **Friction model** parameter is set to `Table lookup kinetic friction coefficient`. For more information, see Main Parameter Dependencies.

Extrapolation method — Extrapolation method
`Linear` (default) | `Nearest` | `Error`

Extrapolation method for the lookup table to use for processing the tire slip-kinetic friction coefficient characteristic. To produce:

- A curve with continuous first-order derivatives in the extrapolation region and at the boundary with the interpolation region, select `Linear`.

- An extrapolation that does not go above the highest point in the data or below the lowest point in the data, select `Nearest`.

- An error if the input signal is outside the range of the table, select `Error`. This option ensures that simulation occurs only if your data is within the table range.

For more information on extrapolation algorithms, see the PS Lookup Table (1D) block reference page.

Dependencies

This parameter is visible when the **Friction model** parameter is set to `Table lookup kinetic friction coefficient`. For more information, see Main Parameter Dependencies.

Dynamics

The table shows how the visibility of some parameters depends on the options that you choose for other parameters. To learn how to read the table, see "Parameter Dependencies" on page B-2.

Dynamics Parameter Dependencies Table

Dynamics	
Compliance — Choose `No compliance - Suitable for HIL simulation` or `Specify stiffness and damping`	
No compliance - Suitable for HIL simulation	**Specify stiffness and damping**
	Longitudinal stiffness
	Longitudinal damping
Inertia — Choose `No Inertia` or `Specify inertia and initial velocity`	
No Inertia	**Specify inertia and initial velocity**
	Tire inertia
	Initial velocity

Compliance — Dynamical compliance model
`No compliance - Suitable for HIL simulation` (default) | `Specify stiffness and damping`

Model for the dynamical compliance of the tire.

- `No compliance - Suitable for HIL simulation` — Tire is modeled with no dynamical compliance.
- `Specify stiffness and damping` — Tire is modeled as a stiff, dampened spring and deforms under load.

Dependencies

Selecting the `Specify stiffness and damping` parameterization method, exposes stiffness and damping parameters. For more information, see Dynamics Parameter Dependencies Table.

Longitudinal stiffness — Longitudinal stiffness
`1e6` N/m (default) | positive scalar

Tire longitudinal stiffness C_{Fx}.

Dependencies

Selecting `Specify stiffness and damping` for the **Compliance** parameter, exposes this parameter. For more information, see Dynamics Parameter Dependencies Table.

Longitudinal damping — Longitudinal damping
`1000` N/(m/s) (default) | positive scalar

Tire longitudinal damping b_{Fx}.

Dependencies

Selecting `Specify stiffness and damping` for the **Compliance** parameter, exposes this parameter. For more information, see Dynamics Parameter Dependencies Table.

Inertia — Inertia model
`No inertia` (default) | `Specify inertia and initial velocity`

Model for the rotational inertia of the tire.

- `No inertia` — Tire is modeled with no dynamical compliance.
- `Specify inertia and initial velocity` — Tire is modeled as a stiff, dampened spring and deforms under load.

Dependencies

Selecting the `Specify inertia and initial velocity` parameterization method, exposes inertia and velocity parameters. For more information, see Dynamics Parameter Dependencies Table.

`Tire inertia` — **Rotational inertia**
`1 kg*m^2` (default) | positive scalar

Rotational inertia I_w of the wheel-tire assembly.

Dependencies

Selecting `Specify inertia and initial velocity` for the **Inertia** parameter, exposes this parameter. For more information, see Dynamics Parameter Dependencies Table.

`Initial velocity` — **Initial rotational velocity**
`0 rad/s` (default) | scalar

Initial angular velocity, $\Omega(0)$, of the tire.

Dependencies

Selecting `Specify inertia and initial velocity` for the **Inertia** parameter, exposes this parameter. For more information, see Dynamics Parameter Dependencies Table.

Rolling Resistance

The table shows how the visibility of some **Rolling Resistance** parameters depends on the options that you choose for other parameters. To learn how to read the table, see "Parameter Dependencies" on page B-2.

Rolling Resistance Parameter Dependencies Table

Rolling Resistance			
Rolling resistance — Choose `Off` or `On`			
Off	**On**		
	Resistance model — Choose `Constant coefficient` or `Pressure and velocity dependent`		
	Constant coefficient	**Pressure and velocity dependent**	
	Constant coefficient	Tire pressure	
		Alpha	
		Beta	
		Coefficient A	
		Coefficient B	
		Coefficient C	
	Velocity threshold		

Rolling resistance — Rolling resistance option
`Off` (default) | `On`

Options for modeling rolling resistance are:

- `Off` — Neglect rolling resistance.
- `On` — Include rolling resistance.

Dependencies

Selecting `On` exposes rolling resistance parameters. For more information, see Rolling Resistance Parameter Dependencies Table.

Resistance model — Rolling resistance model
`Constant coefficient` (default) | `Pressure and velocity dependent`

Model for the rolling resistance of the tire.

- `Constant coefficient` — Neglect rolling resistance.
- `Pressure and velocity dependent` — Include rolling resistance.

1-501

Dependencies

Each **Resistance model** option exposes related parameters. For more information, see Rolling Resistance Parameter Dependencies Table.

`Constant coefficient` — **Proportionality constant**
`0.015` (default) | positive scalar

Coefficient that sets the proportionality between the normal force and the rolling resistance force. The parameter must be greater than zero.

Dependencies

Selecting `On` for the **Rolling resistance** parameter and `Constant coefficient` for the **Resistance model** parameter exposes this parameter. For more information, see Rolling Resistance Parameter Dependencies Table.

`Tire pressure` — **Tire pressure**
`250e3 Pa` (default) | positive scalar

Inflation pressure of the tire. The parameter must be greater than zero.

Dependencies

Selecting `On` for the **Rolling resistance** parameter and `Pressure and velocity dependent` for the **Resistance model** parameter exposes this parameter. For more information, see Rolling Resistance Parameter Dependencies Table.

`Alpha` — **Tire pressure equation exponent**
`-0.003` (default) | scalar

Exponent of the tire pressure in the model equation. See Rolling Resistance Parameter Dependencies Table.

Dependencies

Selecting `On` for the **Rolling resistance** parameter and `Pressure and velocity dependent` for the **Resistance model** parameter exposes this parameter. For more information, see Rolling Resistance Parameter Dependencies Table.

`Beta` — **Normal force equation exponent**
`0.97` (default) | scalar

Exponent of the normal force model equation. See Rolling Resistance Parameter Dependencies Table.

Dependencies

Selecting `On` for the **Rolling resistance** parameter and `Pressure and velocity dependent` for the **Resistance model** parameter exposes this parameter. For more information, see Rolling Resistance Parameter Dependencies Table.

Coefficient A — Velocity-independent force component *A*
`84e-4` (default)

Velocity-independent force component in the model equation. The parameter must be greater than zero. See "Pressure and Velocity Dependent Model" on page 1-388.

Dependencies

Selecting `On` for the **Rolling resistance** parameter and `Pressure and velocity dependent` for the **Resistance model** parameter exposes this parameter. For more information, see Rolling Resistance Parameter Dependencies Table.

Coefficient B — Velocity-dependent force component *B*
`6.2e-4 s/m` (default) | positive scalar

Velocity-dependent force component in the model equation. The parameter must be greater than zero. See "Pressure and Velocity Dependent Model" on page 1-388.

Dependencies

Selecting `On` for the **Rolling resistance** parameter and `Pressure and velocity dependent` for the **Resistance model** parameter exposes this parameter. For more information, see Rolling Resistance Parameter Dependencies Table.

Coefficient C — Velocity-dependent force component *C*
`1.6e-4 s^2/m^2` (default) | positive scalar

Force component that depends on the square of the velocity term in the model equation. The parameter must be greater than zero. See "Pressure and Velocity Dependent Model" on page 1-388.

Dependencies

Selecting On for the **Rolling resistance** parameter and Pressure and velocity dependent for the **Resistance model** parameter exposes this parameter. For more information, see Rolling Resistance Parameter Dependencies Table.

Velocity threshold — Wheel hub velocity threshold for mathematical slip model
0.001 m/s (default) | positive scalar

Velocity at which the full rolling resistance force is transmitted to the rolling hub. The parameter ensures that the force remains continuous during velocity direction changes, which increases the numerical stability of the simulation. The parameter must be greater than zero.

Dependencies

Selecting On for the **Rolling resistance** parameter exposes this parameter. For more information, see Rolling Resistance Parameter Dependencies.

Advanced

Use the **Advanced** tab to specify parameters that the state machine uses to determine the model of the tire.

Traction velocity tolerance — Traction velocity tolerance
0.01 m/s (default) | positive scalar

Magnitude of the relative velocity between the tire and ground at which the tire regains traction. Setting this value too low prevents the tire from entering a state where it has traction. Setting it too high can cause the tire velocity to change suddenly when the tire gains traction, and can result in an unstable simulation. The parameter must be greater than zero.

Engagement threshold force — Engagement threshold force
10 N (default) | positive scalar

Normal force values below the **Engagement threshold force** are not applied to the tire. Setting this value too low can cause the tire to gain and lose traction rapidly. Setting this value too high can give unrealistically low static and dynamic friction forces. The parameter must be greater than zero.

`Initial traction state` — Initial traction state
`Tire is initially slipping` (default) | `Tire is initially in traction`

Option to have the tire in traction or slipping at the start of simulation.

Extended Capabilities

Real-Time and Hardware-in-the-Loop Simulation

For optimal simulation performance, set the **Dynamics** > **Compliance** parameter to `No compliance - Suitable for HIL simulation`.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using Simulink® Coder™.

See Also
Fundamental Friction Clutch | Tire (Magic Formula) | Tire (Simple) | Tire-Road Interaction (Magic Formula)

Topics
"Complete Vehicle Model"
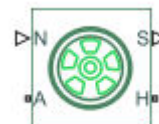"Prepare Simscape Driveline Models for Real-Time Simulation Using Simscape Checks"
"Clutch States" on page 1-224

Introduced in R2018b

Tire (Magic Formula)

Tire with longitudinal behavior given by Magic Formula coefficients
Library: Simscape / Driveline / Tires & Vehicles

Description

The Tire (Magic Formula) block models a tire with longitudinal behavior given by the Magic Formula [1], an empirical equation based on four fitting coefficients. The block can model tire dynamics under constant or variable pavement conditions.

The longitudinal direction of the tire is the same as its direction of motion as it rolls on pavement. This block is a structural component based on the Tire-Road Interaction (Magic Formula) block.

To increase the fidelity of the tire model, you can specify properties such as tire compliance, inertia, and rolling resistance. However, these properties increase the complexity of the tire model and can slow down simulation. Consider ignoring tire compliance and inertia if simulating the model in real time or if preparing the model for hardware-in-the-loop (HIL) simulation.

Tire Model

The Tire (Magic Formula) block models the tire as a rigid wheel-tire combination in contact with the road and subject to slip. When torque is applied to the wheel axle, the tire pushes on the ground (while subject to contact friction) and transfers the resulting reaction as a force back on the wheel. This action pushes the wheel forward or backward. If you include the optional tire compliance, the tire also flexibly deforms under load.

The figure shows the forces acting on the tire. The table defines the tire model variables.

Tire Model Variables

Symbol	Description and Unit
r_w	Wheel radius
V_x	Wheel hub longitudinal velocity
u	Tire longitudinal deformation
Ω	Wheel angular velocity
Ω'	Contact point angular velocity. If there is no tire longitudinal deformation, that is, if $u = 0$, $\Omega' = \Omega$.
$r_w\Omega'$	Tire tread longitudinal velocity
$V_{sx} = r_w\Omega - V_x$	Wheel slip velocity
$V'_{sx} = r_w\Omega' - V_x$	Contact slip velocity. If there is no tire longitudinal deformation, that is, if $u = 0$, $V'_{sx} = V_{sx}$.
$k = \dfrac{V_{sx}}{\|V_x\|}$	Wheel slip
$k' = \dfrac{V'_{sx}}{\|V_x\|}$	Contact slip. If there is no tire longitudinal deformation, that is, if $u = 0$, $k' = k$.
V_{th}	Wheel hub threshold velocity
F_z	Vertical load on tire
F_x	Longitudinal force exerted on the tire at the contact point
$C_{F_x} = \left(\dfrac{\partial F_x}{\partial u}\right)_0$	Tire longitudinal stiffness under deformation
$b_{F_x} = \left(\dfrac{\partial F_x}{\partial \dot{u}}\right)_0$	Tire longitudinal damping under deformation
I_w	Wheel-tire inertia, such that the effective mass is equal to $\dfrac{I_w}{r_w^2}$
τ_{drive}	Torque applied by the axle to the wheel

Tire Kinematics and Response

Roll and Slip

A nonslipping tire would roll and translate as $V_x = r_w \Omega$. However, as tires do slip, they respond by developing a longitudinal force, F_x.

The wheel slip velocity is $V_{sx} = r_w \Omega - V_x$. The *wheel slip* is $k = \dfrac{V_{sx}}{|V_x|}$. For a locked, sliding wheel, $k = -1$. For perfect rolling, $k = 0$.

Slip at Low Speed

For low speeds, as defined by $|V_x| \le |V_{th}|$, the wheel slip becomes:

$$k = \frac{2V_{sx}}{\left(V_{th} + \dfrac{V_x^2}{V_{th}}\right)}$$

This modification allows for a nonsingular, nonzero slip at zero wheel velocity. For example, for perfect slipping, that is, in the case of a nontranslating spinning tire, $V_x = 0$ while $k = \dfrac{2r_w \Omega}{V_{th}}$ is finite.

Deformation

If the tire is modeled with compliance, it is also flexible. In this case, because the tire deforms, the tire-road contact point turns at a slightly different angular velocity, Ω', from the wheel, Ω, and requires, instead of the wheel slip, the *contact point* or *contact patch slip* κ'. The block models the deforming tire as a translational spring-damper of stiffness, C_{F_x}, and damping, b_{F_x}.

If you model a tire without compliance, that is, if $u = 0$, then there is no tire longitudinal deformation at any time in the simulation and:

- $k' = k$
- $V'_{sx} = V_{sx}$
- $\Omega' = \Omega$

Tire and Wheel Dynamics
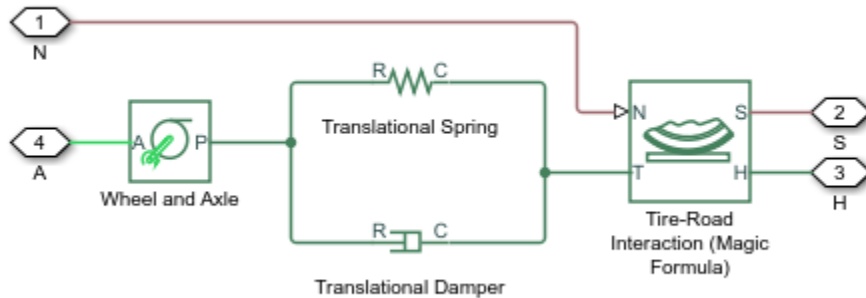
The full tire model is equivalent to this Simscape/Simscape Driveline component diagram. It simulates both transient and steady-state behavior and correctly represents starting from, and coming to, a stop. The Translational Spring and Translational Damper are equivalent to the tire stiffness C_{F_x} and damping b_{F_x}. The Tire-Road Interaction (Magic Formula) block models the longitudinal force F_x on the tire as a function of F_z and k' using the Magic Formula, with k' as the independent slip variable.

The Wheel and Axle radius is the wheel radius r_w. The Mass value is the effective mass, $\dfrac{I_w}{r_w^2}$. The tire characteristic function $f(k', F_z)$ determines the longitudinal force F_x. Together with the driveshaft torque applied to the wheel axis, F_x determines the wheel angular motion and longitudinal motion.

Without tire compliance, the Translational Spring and Translational Damper are omitted, and contact variables revert to wheel variables. In this case, the tire effectively has infinite stiffness, and port **P** of Wheel and Axle connects directly to port **T** of Tire-Road Interaction (Magic Formula).

Without tire inertia, the Mass is omitted.

Assumptions and Limitations

- The Tire (Magic Formula) block assumes longitudinal motion only and includes no camber, turning, or lateral motion.

- Tire compliance implies a time lag in the tire response to the forces on it. Time lag simulation increases model fidelity but reduces simulation performance. See "Adjust Model Fidelity".

Ports

Input

N — Normal force
physical signal

Physical signal input port associated with the normal force acting on the tire. The normal force is positive if it acts downward on the tire, pressing it against the pavement.

M — Magic formula coefficients
physical signal | vector | [B, C, D, E]

Physical signal input port associated with the Magic Formula coefficients. Provide the Magic Formula coefficients as a four-element vector, specified in the order [B, C, D, E].

Dependencies

Port **M** is exposed only if the **Main > Parameterize by** parameter is set to `Physical signal Magic Formula coefficients`. For more information, see Main Parameter Dependencies.

Output

S — Slip
physical signal

Physical signal output port associated with the relative slip between the tire and road.

Conserving

A — Axle
mechanical rotational

Mechanical rotational port associated with the axle that the tire sits on.

H — Hub
mechanical translational

Mechanical translational port associated with the wheel hub that transmits the thrust generated by the tire to the remainder of the vehicle.

Parameters

Main

The table shows how the visibility of some **Main** parameters depends on the options that you choose for other parameters. To learn how to read the table, see "Parameter Dependencies" on page B-2.

Main Parameter Dependencies

Main			
Parameterize by — Choose `Peak longitudinal force and corresponding slip`, `Constant Magic Formula coefficients`, `Load-dependent Magic Formula coefficients`, or `Physical signal Magic Formula coefficients`			
Peak longitudinal force and corresponding slip	**Constant Magic Formula coefficients**	**Load-dependent Magic Formula coefficients**	**Physical signal Magic Formula coefficients** — Exposes physical signal input port **M** for providing the Magic Formula coefficients to the block as an array of elements in this order [*B*, *C*, *D*, *E*].
Rated vertical load	Magic Formula B coefficient	Magic Formula C-coefficient parameter, p_Cx1	
Peak longitudinal force at rated load	Magic Formula C coefficient	Magic Formula D-coefficient parameters, [p_Dx1 p_Dx2]	
Slip at peak force at rated load (percent)	Magic Formula D coefficient	Magic Formula E-coefficient parameters, [p_Ex1 p_Ex2 p_Ex3 p_Ex4]	
	Magic Formula E coefficient	Magic Formula BCD-coefficient parameters, [p_Kx1 p_Kx2 p_Kx3]	
		Magic Formula H-coefficient parameters, [p_Hx1 p_Hx2]	
		Magic Formula V-coefficient parameters, [p_Vx1 p_Vx2]	

Parameterize by — Parameterization method
`Peak longitudinal force and corresponding slip` (default) | `Constant Magic Formula coefficients` | `Load-dependent Magic Formula coefficients` | `Physical signal Magic Formula coefficients`

Magic Formula tire-road interaction model.

To model tire dynamics under constant pavement conditions, select one of these models:

- `Peak longitudinal force and corresponding slip` — Parameterize the Magic Formula with physical characteristics of the tire.
- `Constant Magic Formula coefficients` — Specify the parameters that define the constant *B*, *C*, *D*, and *E* coefficients as scalars, with these default values.

Coefficient	Default Value
B	10
C	1.9
D	1
E	0.97

- `Load-dependent Magic Formula coefficients` — Specify the parameters that define the load-dependent *C*, *D*, *E*, *K*, *H*, and *V* coefficients as vectors, one for each coefficient, with these default values.

Coefficient	Parameters	Default Values
C	**p_Cx1**	1.685
D	**[p_Dx1 p_Dx2]**	[1.21 –0.037]
E	**[p_Ex1 p_Ex2 p_Ex3 p_Ex4]**	[0.344 0.095 –0.02 0]
K	**[p_Kx1 p_Kx2 p_Kx3]**	[21.51 –0.163 0.245]
H	**[p_Hx1 p_Hx2]**	[–0.002 0.002]
V	**[p_Vx1 p_Vx2]**	[0 0]

To model tire dynamics under variable pavement conditions, select `Physical signal Magic Formula coefficients`. Selecting this model exposes a physical signal port **M**. Use the port to input the Magic Formula coefficients as a four-element vector, specified in the order [*B*, *C*, *D*,*E*].

Dependencies

Each parameterization method option exposes related parameters and hides unrelated parameters. Selecting `Physical signal Magic Formula coefficients` exposes a physical signal input port. For more information, see Main Parameter Dependencies.

Rated vertical load — Rated load force
`3000 N` (default) | positive scalar

Rated vertical load force F_{z0}.

Dependencies

This parameter is exposed only when you select the `Peak longitudinal force and corresponding slip` parameterization method. For more information, see Main Parameter Dependencies.

Peak longitudinal force at rated load — Maximum longitudinal force at rated load
`3500 N` (default) | positive scalar

Maximum longitudinal force F_{x0} that the tire exerts on the wheel when the vertical load equals its rated value F_{z0}.

Dependencies

This parameter is exposed only when you select the `Peak longitudinal force and corresponding slip` parameterization method. For more information, see Main Parameter Dependencies.

Slip at peak force at rated load (percent) — Percent slip at maximum longitudinal force and rated load
`10` (default) | positive scalar

Contact slip κ'_0, expressed as a percentage (%), when the longitudinal force equals its maximum value F_{x0} and the vertical load equals its rated value F_{z0}.

Dependencies

This parameter is exposed only when you select the `Peak longitudinal force and corresponding slip` parameterization method. For more information, see Main Parameter Dependencies.

Magic Formula B coefficient — Constant Magic Formula *B* coefficient
`10` (default) | positive scalar

Load-independent Magic Formula *B* coefficient.

Dependencies

This parameter is exposed only when you select the `Constant Magic Formula coefficients` parameterization method. For more information, see Main Parameter Dependencies.

`Magic Formula C coefficient` — Constant Magic Formula *C* coefficient
`1.9` (default) | positive scalar

Load-independent Magic Formula *C* coefficient.

Dependencies

This parameter is exposed only when you select the `Constant Magic Formula coefficients` parameterization method. For more information, see Main Parameter Dependencies.

`Magic Formula D coefficient` — Constant Magic Formula *D* coefficient
`1` (default) | positive scalar

Load-independent Magic Formula *D* coefficient.

Dependencies

This parameter is exposed only when you select the `Constant Magic Formula coefficients` parameterization method. For more information, see Main Parameter Dependencies.

`Magic Formula E coefficient` — Constant Magic Formula *E* coefficient
`0.97` (default) | positive scalar

Load-independent Magic Formula *E* coefficient.

Dependencies

This parameter is exposed only when you select the `Constant Magic Formula coefficients` parameterization method. For more information, see Main Parameter Dependencies.

`Tire nominal vertical load` — Normal force
`4000` N (default) | positive scalar

Nominal normal force F_{z0} on tire.

Dependencies

This parameter is exposed only when you select the `Load-dependent Magic Formula coefficients` parameterization method. For more information, see Main Parameter Dependencies.

Magic Formula C-coefficient parameter, p_Cx1 — Variable Magic Formula *C* coefficient

`1.685` (default) | scalar

Load-dependent Magic Formula *C* coefficient.

Dependencies

This parameter is exposed only when you select the `Load-dependent Magic Formula coefficients` parameterization method. For more information, see Main Parameter Dependencies.

Magic Formula D-coefficient parameters, [p_Dx1 p_Dx2] — Variable Magic Formula *D* coefficient

`[1.21, -.037]` (default) | vector

Load-dependent Magic Formula *D* coefficient.

Dependencies

This parameter is exposed only when you select the `Load-dependent Magic Formula coefficients` parameterization method. For more information, see Main Parameter Dependencies.

Magic Formula E-coefficient parameters, [p_Ex1 p_Ex2 p_Ex3 p_Ex4] — Variable Magic Formula *E* coefficient

`[.344, .095, -.02, 0]` (default) | vector

Load-dependent Magic Formula *E* coefficient.

Dependencies

This parameter is exposed only when you select the `Load-dependent Magic Formula coefficients` parameterization method. For more information, see Main Parameter Dependencies.

**Magic Formula BCD-coefficient parameters, [p_Kx1 p_Kx2 p_Kx3] —
Variable Magic Formula *K* coefficient**
[21.51, -.163, .245] (default) | vector

Load-dependent Magic Formula *K* coefficient.

Dependencies

This parameter is exposed only when you select the Load-dependent Magic Formula
coefficients parameterization method. For more information, see Main Parameter
Dependencies.

**Magic Formula H-coefficient parameters, [p_Hx1 p_Hx2] — Variable Magic
Formula *H* coefficient**
[-.002, .002] (default) | vector

Load-dependent Magic Formula *H* coefficient.

Dependencies

This parameter is exposed only when you select the Load-dependent Magic Formula
coefficients parameterization method. For more information, see Main Parameter
Dependencies.

**Magic Formula V-coefficient parameters, [p_Vx1 p_Vx2] — Variable Magic
Formula *V* coefficient**
[0, 0] (default) | vector

Load-dependent Magic Formula *V* coefficient.

Dependencies

This parameter is exposed only when you select the Load-dependent Magic Formula
coefficients parameterization method. For more information, see Main Parameter
Dependencies.

Geometry

Rolling radius — Unloaded tire-wheel radius
0.3 m (default) | positive scalar

Unloaded tire-wheel radius, r_w.

Dynamics

The table shows how the visibility of some **Dynamics** parameters depends on the options that you choose for other parameters. To learn how to read the table, see "Parameter Dependencies" on page B-2.

Dynamics Parameter Dependencies

Dynamics		
Compliance — Choose `No compliance - Suitable for HIL simulation` or `Specify stiffness and damping`		
No compliance - Suitable for HIL simulation	**Specify stiffness and damping**	
	Longitudinal stiffness	
	Longitudinal damping	
Inertia — Choose `No Inertia` or `Specify inertia and initial velocity`		
No Inertia	**Specify inertia and initial velocity**	
	Tire inertia	
	Initial velocity	

Compliance — Dynamical compliance model
`No compliance - Suitable for HIL simulation` (default) | `Specify stiffness and damping`

Model for the dynamical compliance of the tire.

- `No compliance - Suitable for HIL simulation` — Tire is modeled with no dynamical compliance.
- `Specify stiffness and damping` — Tire is modeled as a stiff, dampened spring and deforms under load.

Dependencies

Selecting the `Specify stiffness and damping` parameterization method, exposes stiffness and damping parameters. For more information, see Dynamics Parameter Dependencies.

Longitudinal stiffness — Longitudinal stiffness
`1e6 N/m` (default) | positive scalar

Tire longitudinal stiffness C_{Fx}.

Dependencies

Selecting `Specify stiffness and damping` for the **Compliance** parameter, exposes this parameter. For more information, see Dynamics Parameter Dependencies.

Longitudinal damping — Longitudinal damping
1000 `N/(m/s)` (default) | positive scalar

Tire longitudinal damping b_{Fx}.

Dependencies

Selecting `Specify stiffness and damping` for the **Compliance** parameter, exposes this parameter. For more information, see Dynamics Parameter Dependencies.

Inertia — Inertia model
`No inertia` (default) | `Specify inertia and initial velocity`

Model for the rotational inertia of the tire.

- `No inertia` — Tire is modeled with no dynamical compliance.
- `Specify inertia and initial velocity` — Tire is modeled as a stiff, dampened spring and deforms under load.

Dependencies

Selecting the `Specify inertia and initial velocity` parameterization method, exposes inertia and velocity parameters. For more information, see Dynamics Parameter Dependencies.

Tire inertia — Rotational inertia
1 kg*m^2 (default) | positive scalar

Rotational inertia I_w of the wheel-tire assembly.

Dependencies

Selecting `Specify inertia and initial velocity` for the **Inertia** parameter, exposes this parameter. For more information, see Dynamics Parameter Dependencies.

Initial velocity — Initial rotational velocity
0 rad/s (default) | scalar

Initial angular velocity, $\Omega(0)$, of the tire.

Dependencies

Selecting `Specify inertia and initial velocity` for the **Inertia** parameter, exposes this parameter. For more information, see Dynamics Parameter Dependencies.

Rolling Resistance

The table shows how the visibility of some parameters depends on the options that you choose for other parameters. To learn how to read the table, see "Parameter Dependencies" on page B-2.

Rolling Resistance Parameter Dependencies

Rolling Resistance		
Rolling resistance — Choose `Off` or `On`		
Off	**On**	
	Resistance model — Choose `Constant coefficient` or `Pressure and velocity dependent`	
	Constant coefficient	**Pressure and velocity dependent**
	Constant coefficient	Tire pressure
		Alpha
		Beta
		Coefficient A
		Coefficient B
		Coefficient C
	Velocity threshold	

Rolling resistance — Rolling resistance option
`Off` (default) | `On`

Options for modeling rolling resistance are:

• `Off` — Neglect rolling resistance.

- `On` — Include rolling resistance.

Dependencies

Selecting `On` exposes rolling resistance parameters. For more information, see Rolling Resistance Parameter Dependencies.

Resistance model — Rolling resistance model
`Constant coefficient` (default) | `Pressure and velocity dependent`

Model for the rolling resistance of the tire.

- `Constant coefficient` — Neglect rolling resistance.
- `Pressure and velocity dependent` — Include rolling resistance.

Dependencies

Each **Resistance model** option exposes related parameters. For more information, see Rolling Resistance Parameter Dependencies.

Constant coefficient — Proportionality constant
`0.015` (default) | positive scalar

Coefficient that sets the proportionality between the normal force and the rolling resistance force. The parameter must be greater than zero.

Dependencies

Selecting `On` for the **Rolling resistance** parameter and `Constant coefficient` for the **Resistance model** parameter exposes this parameter. For more information, see Rolling Resistance Parameter Dependencies.

Tire pressure — Tire pressure
`250e3 Pa` (default) | positive scalar

Inflation pressure of the tire. The parameter must be greater than zero.

Dependencies

Selecting `On` for the **Rolling resistance** parameter and `Pressure and velocity dependent` for the **Resistance model** parameter exposes this parameter. For more information, see Rolling Resistance Parameter Dependencies.

Alpha — Tire pressure equation exponent
-0.003 (default) | scalar

Exponent of the tire pressure in the model equation. See "Pressure and Velocity Dependent Model" on page 1-388.

Dependencies

Selecting On for the **Rolling resistance** parameter and Pressure and velocity dependent for the **Resistance model** parameter exposes this parameter. For more information, see Rolling Resistance Parameter Dependencies.

Beta — Normal force equation exponent
0.97 (default) | scalar

Exponent of the normal force model equation. See "Pressure and Velocity Dependent Model" on page 1-388.

Dependencies

Selecting On for the **Rolling resistance** parameter and Pressure and velocity dependent for the **Resistance model** parameter exposes this parameter. For more information, see Rolling Resistance Parameter Dependencies.

Coefficient A — Velocity-independent force component *A*
84e-4 (default)

Velocity-independent force component in the model equation. The parameter must be greater than zero. See "Pressure and Velocity Dependent Model" on page 1-388.

Dependencies

Selecting On for the **Rolling resistance** parameter and Pressure and velocity dependent for the **Resistance model** parameter exposes this parameter. For more information, see Rolling Resistance Parameter Dependencies.

Coefficient B — Velocity-dependent force component *B*
6.2e-4 s/m (default) | positive scalar

Velocity-dependent force component in the model equation. The parameter must be greater than zero. See "Pressure and Velocity Dependent Model" on page 1-388.

Dependencies

Selecting On for the **Rolling resistance** parameter and `Pressure and velocity dependent` for the **Resistance model** parameter exposes this parameter. For more information, see Rolling Resistance Parameter Dependencies.

`Coefficient C` — Velocity-dependent force component *C*
`1.6e-4 s^2/m^2` (default) | positive scalar

Force component that depends on the square of the velocity term in the model equation. The parameter must be greater than zero. See "Pressure and Velocity Dependent Model" on page 1-388.

Dependencies

Selecting On for the **Rolling resistance** parameter and `Pressure and velocity dependent` for the **Resistance model** parameter exposes this parameter. For more information, see Rolling Resistance Parameter Dependencies.

`Velocity threshold` — Wheel hub velocity threshold for mathematical slip model
`0.001 m/s` (default) | positive scalar

Velocity at which the full rolling resistance force is transmitted to the rolling hub. The parameter ensures that the force remains continuous during velocity direction changes, which increases the numerical stability of the simulation. The parameter must be greater than zero.

Dependencies

Selecting On for the **Rolling resistance** parameter exposes this parameter. For more information, see Rolling Resistance Parameter Dependencies.

Advanced

`Velocity threshold` — Wheel hub velocity threshold for mathematical slip model
`0.1 m/s` (default) | positive scalar

Wheel hub velocity V_{th} below which the slip calculation is modified to avoid singular evolution at zero velocity. Must be positive.

Extended Capabilities

Real-Time and Hardware-in-the-Loop Simulation

For optimal simulation performance, set the **Dynamics** > **Compliance** parameter to `No compliance - Suitable for HIL simulation`.

References

[1] Pacejka, H. B. *Tire and Vehicle Dynamics.* Elsevier Science, 2005.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also
Tire (Friction Parameterized) | Tire (Simple) | Tire-Road Interaction (Magic Formula)

Topics
"Complete Vehicle Model"
"Prepare Simscape Driveline Models for Real-Time Simulation Using Simscape Checks"

Introduced in R2011a

Tire-Road Interaction (Magic Formula)

Tire-road dynamics given by magic formula coefficients
Library: Simscape / Driveline / Tires & Vehicles / Tire
                    Subcomponents
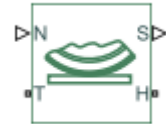
Description

The Tire-Road Interaction (Magic Formula) block models the interaction between the tire tread and road pavement. The longitudinal force arising from this interaction is given by the magic formula, an empirical equation based on four fitting coefficients. Tire properties such as compliance and inertia are ignored.

Tire-Road Interaction Model

The Tire-Road Interaction (Magic Formula) block models the longitudinal forces at the tire-road contact patch using the Magic Formula of Pacejka [1].

The figure displays the forces on the tire. The table defines the tire model variables.

Tire-Road Contact Variables

Symbol	Description and Unit
Ω	Wheel angular velocity
r_w	Wheel radius
V_x	Wheel hub longitudinal velocity
$r_w\Omega$	Tire tread longitudinal velocity
$V_{sx} = r_w\Omega - V_x$	Wheel slip velocity, defined as the difference between the longitudinal velocities of the wheel hub and the tire tread
$k = \dfrac{V_{sx}}{\lvert V_x \rvert}$	Wheel slip
F_z	Vertical load on tire
F_{z0}	Nominal vertical load on tire
$F_x = f(\kappa, F_z)$	Longitudinal force exerted on the tire at the contact point. Also a characteristic function f of the tire.

Tire Response

Forces and Characteristic Function

A tire model provides a steady-state tire characteristic function, $F_x = f(\kappa, F_z)$, the longitudinal force F_x on the tire, based on:

- Vertical load F_z
- Wheel slip κ

Magic Formula with Constant Coefficients

The Magic Formula is a specific form for the tire characteristic function, characterized by four dimensionless coefficients, B, C, D, and E, or stiffness, shape, peak, and curvature:

$$F_X = f(\kappa, F_z) = F_z \cdot D \cdot \sin(C \cdot \arctan\{B\kappa - E[B\kappa - \arctan(B\kappa)]\})$$

The slope of f at $k = 0$ is $BCD \cdot F_z$.

Magic Formula with Load-Dependent Coefficients

A more general Magic Formula uses dimensionless coefficients that are functions of the tire load. A more complex set of parameters p_i, entered in the dialog box, specifies these functions:

$$F_{x0} = D_x \sin(C_x \arctan\{B_x\kappa_x - E_x[B_x\kappa_x - \arctan(B_x\kappa_x)]\}) + S_{Vx}$$

Where:

$$df_z = \frac{(F_z - F_{z0})}{F_z}$$

$$\kappa_x = \kappa + S_{Hx}$$

$$C_x = \text{p\_Cx1}$$

$$D_x = \mu_x \cdot F_z$$

$$\mu_x = \text{p\_Dx1} + \text{p\_Dx2} \cdot df_z$$

$$E_x = (\text{p\_Ex1} + \text{p\_Ex2} \cdot df_z + \text{p\_Ex3} \cdot df_z^2)[1 - \text{p\_Ex4} \cdot \text{sgn}(\kappa_x)]$$

$$K_{xK} = F_z \cdot (\mathrm{p\_Kx1} \ + \ \mathrm{p\_Kx2} \cdot df_z) \cdot e^{(\mathrm{p\_Kx3} \cdot df_z)}$$

$$B_x = \frac{K_{xK}}{C_x D_x + \varepsilon_x}$$

$$S_{Hx} = \ \mathrm{p\_Hx1} \ + \ \mathrm{p\_Hx2} \cdot df_z$$

$$S_{Vx} = F_z \cdot (\mathrm{p\_Vx1} \ + \ \mathrm{p\_Vx2} \cdot df_z)$$

S_{Hx} and S_{Vx} represent offsets to the slip and longitudinal force in the force-slip function, or horizontal and vertical offsets if the function is plotted as a curve. μ_x is the longitudinal load-dependent friction coefficient. ε_x is a small number inserted to prevent division by zero as F_z approaches zero.

Peak Longitudinal Force and Corresponding Slip

The block uses a representative set of Magic Formula coefficients. The block scales the coefficients to yield the peak longitudinal force F_{x0} at the corresponding slip κ_0 that you specify, for rated vertical load F_{z0}.

Magic Formula Coefficients for Typical Road Conditions

Numerical values are based on empirical tire data. These values are typical sets of constant Magic Formula coefficients for common road conditions.

Surface	B	C	D	E
Dry tarmac	10	1.9	1	0.97
Wet tarmac	12	2.3	0.82	1
Snow	5	2	0.3	1
Ice	4	2	0.1	1

Assumptions and Limitations

- The Tire-Road Interaction (Magic Formula) block assumes longitudinal motion only and includes no camber, turning, or lateral motion.

Ports

Input

N — Normal force
physical signal

Physical signal input port associated with the normal force acting on the tire. The normal force is positive if it acts downward on the tire, pressing it against the pavement.

M — Magic formula coefficients
physical signal | vector | $[B, C, D, E]$

Physical signal input port associated with the Magic Formula coefficients. Provide the Magic Formula coefficients as a four-element vector, specified in the order $[B, C, D, E]$.

Dependencies

Port **M** is exposed only if the **Parameterize by** parameter is set to `Physical signal Magic Formula coefficients`. For more information, see Parameter Dependencies.

Output

S — Slip
physical signal

Physical signal output port associated with the relative slip between the tire and road.

Conserving

A — Axle
mechanical rotational

Mechanical rotational port associated with the axle that the tire sits on.

H — Hub
mechanical translational

Mechanical translational port associated with the wheel hub that transmits the thrust generated by the tire to the remainder of the vehicle.

Parameters

The table shows how the visibility of some **Main** parameters depends on the options that you choose for other parameters. To learn how to read the table, see "Parameter Dependencies" on page B-2.

Parameter Dependencies

Parameters			
Parameterize by — Choose `Peak longitudinal force and corresponding slip`, `Constant Magic Formula coefficients`,`Load-dependent Magic Formula coefficients`, or `Physical signal Magic Formula coefficients`			
Peak longitudinal force and corresponding slip	**Constant Magic Formula coefficients**	**Load-dependent Magic Formula coefficients**	**Physical signal Magic Formula coefficients** — Exposes physical signal input port **M** for providing the Magic Formula coefficients to the block as an array of elements in this order [*B*, *C*, *D*, *E*].
Rated vertical load	Magic Formula B coefficient	Magic Formula C-coefficient parameter, p_Cx1	
Peak longitudinal force at rated load	Magic Formula C coefficient	Magic Formula D-coefficient parameters, [p_Dx1 p_Dx2]	
Slip at peak force at rated load (percent)	Magic Formula D coefficient	Magic Formula E-coefficient parameters, [p_Ex1 p_Ex2 p_Ex3 p_Ex4]	
	Magic Formula E coefficient	Magic Formula BCD-coefficient parameters, [p_Kx1 p_Kx2 p_Kx3]	
		Magic Formula H-coefficient parameters, [p_Hx1 p_Hx2]	
		Magic Formula V-coefficient parameters, [p_Vx1 p_Vx2]	
Velocity threshold			

Parameterize by — Parameterization method
Peak longitudinal force and corresponding slip (default) | Constant Magic
Formula coefficients | Load-dependent Magic Formula coefficients |
Physical signal Magic Formula coefficients

Magic Formula tire-road interaction model.

To model tire dynamics under constant pavement conditions, select one of these models:

- Peak longitudinal force and corresponding slip — Parametrize the Magic
  Formula with physical characteristics of the tire.

- Constant Magic Formula coefficients — Specify the parameters that define
  the constant B, C, D, and E coefficients as scalars, with these default values.

Coefficient	Default Value
B	10
C	1.9
D	1
E	0.97

- Load-dependent Magic Formula coefficients — Specify the parameters that
  define the load-dependent C, D, E, K, H, and V coefficients as vectors, one for each
  coefficient, with these default values.

Coefficient	Parameters	Default Values
C	**p_Cx1**	1.685
D	**[p_Dx1 p_Dx2]**	[1.21 –0.037]
E	**[p_Ex1 p_Ex2 p_Ex3 p_Ex4]**	[0.344 0.095 –0.02 0]
K	**[p_Kx1 p_Kx2 p_Kx3]**	[21.51 –0.163 0.245]
H	**[p_Hx1 p_Hx2]**	[–0.002 0.002]
V	**[p_Vx1 p_Vx2]**	[0 0]

To model tire dynamics under variable pavement conditions, select Physical signal
Magic Formula coefficients. Selecting this model exposes a physical signal port **M**.
Use the port to input the Magic Formula coefficients as a four-element vector, specified in
the order [B, C, D,E].

Dependencies

Each parameterization method option exposes related parameters and hides unrelated parameters. Selecting `Physical signal Magic Formula coefficients` exposes a physical signal input port. For more information, see Parameter Dependencies.

Rated vertical load — Rated load force
`3000 N` (default) | positive scalar

Rated vertical load force F_{z0}.

Dependencies

This parameter is exposed only when you select the `Peak longitudinal force and corresponding slip` parameterization method. For more information, see Parameter Dependencies.

Peak longitudinal force at rated load — Maximum longitudinal force at rated load
`3500 N` (default) | positive scalar

Maximum longitudinal force F_{x0} that the tire exerts on the wheel when the vertical load equals its rated value F_{z0}.

Dependencies

This parameter is exposed only when you select the `Peak longitudinal force and corresponding slip` parameterization method. For more information, see Parameter Dependencies.

Slip at peak force at rated load (percent) — Percent slip at maximum longitudinal force and rated load
`10` (default) | positive scalar

Contact slip κ'_0, expressed as a percentage (%), when the longitudinal force equals its maximum value F_{x0} and the vertical load equals its rated value F_{z0}.

Dependencies

This parameter is exposed only when you select the `Peak longitudinal force and corresponding slip` parameterization method. For more information, see Parameter Dependencies.

Magic Formula B coefficient — Constant Magic Formula *B* coefficient
10 (default) | positive scalar

Load-independent Magic Formula *B* coefficient.

Dependencies

This parameter is exposed only when you select the Constant Magic Formula coefficients parameterization method. For more information, see Parameter Dependencies.

Magic Formula C coefficient — Constant Magic Formula *C* coefficient
1.9 (default) | positive scalar

Load-independent Magic Formula *C* coefficient.

Dependencies

This parameter is exposed only when you select the Constant Magic Formula coefficients parameterization method. For more information, see Parameter Dependencies.

Magic Formula D coefficient — Constant Magic Formula *D* coefficient
1 (default) | positive scalar

Load-independent Magic Formula *D* coefficient.

Dependencies

This parameter is exposed only when you select the Constant Magic Formula coefficients parameterization method. For more information, see Parameter Dependencies.

Magic Formula E coefficient — Constant Magic Formula *E* coefficient
0.97 (default) | positive scalar

Load-independent Magic Formula *E* coefficient.

Dependencies

This parameter is exposed only when you select the Constant Magic Formula coefficients parameterization method. For more information, see Parameter Dependencies.

Tire nominal vertical load — Normal force
4000 N (default) | positive scalar

Nominal normal force F_{z0} on tire.

Dependencies

This parameter is exposed only when you select the `Load-dependent Magic Formula coefficients` parameterization method. For more information, see Parameter Dependencies.

Magic Formula C-coefficient parameter, p_Cx1 — Variable Magic Formula *C* coefficient
1.685 (default) | scalar

Load-dependent Magic Formula *C* coefficient.

Dependencies

This parameter is exposed only when you select the `Load-dependent Magic Formula coefficients` parameterization method. For more information, see Parameter Dependencies.

Magic Formula D-coefficient parameters, [p_Dx1 p_Dx2] — Variable Magic Formula *D* coefficient
[1.21, -.037] (default) | vector

Load-dependent Magic Formula *D* coefficient.

Dependencies

This parameter is exposed only when you select the `Load-dependent Magic Formula coefficients` parameterization method. For more information, see Parameter Dependencies.

Magic Formula E-coefficient parameters, [p_Ex1 p_Ex2 p_Ex3 p_Ex4] — Variable Magic Formula *E* coefficient
[.344, .095, -.02, 0] (default) | vector

Load-dependent Magic Formula *E* coefficient.

Dependencies

This parameter is exposed only when you select the `Load-dependent Magic Formula coefficients` parameterization method. For more information, see Parameter Dependencies.

`Magic Formula BCD-coefficient parameters, [p_Kx1 p_Kx2 p_Kx3]` — Variable Magic Formula *K* coefficient
[21.51, -.163, .245] (default) | vector

Load-dependent Magic Formula *K* coefficient.

Dependencies

This parameter is exposed only when you select the `Load-dependent Magic Formula coefficients` parameterization method. For more information, see Parameter Dependencies.

`Magic Formula H-coefficient parameters, [p_Hx1 p_Hx2]` — Variable Magic Formula *H* coefficient
[-.002, .002] (default) | vector

Load-dependent Magic Formula *H* coefficient.

Dependencies

This parameter is exposed only when you select the `Load-dependent Magic Formula coefficients` parameterization method. For more information, see Parameter Dependencies.

`Magic Formula V-coefficient parameters, [p_Vx1 p_Vx2]` — Variable Magic Formula *V* coefficient
[0, 0] (default) | vector

Load-dependent Magic Formula *V* coefficient.

Dependencies

This parameter is exposed only when you select the `Load-dependent Magic Formula coefficients` parameterization method. For more information, see Parameter Dependencies.

Velocity threshold — Wheel hub velocity threshold for mathematical slip model
`0.1` m/s (default) | positive scalar

Wheel hub velocity V_{th} below which the slip calculation is modified to avoid singular evolution at zero velocity. Must be positive.

References

[1] Pacejka, H. B. *Tire and Vehicle Dynamics.* Elsevier Science, 2005.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using Simulink® Coder™.

See Also
Tire (Friction Parameterized) | Tire (Magic Formula) | Tire (Simple)

Topics
"Complete Vehicle Model"
"Prepare Simscape Driveline Models for Real-Time Simulation Using Simscape Checks"

Introduced in R2018b

Tire (Simple)

No-slip tire model with minimal parameters
Library: Simscape / Driveline / Tires & Vehicles

Description

The Tire (Simple) block represents a simple, no-slip model of a tire that you parameterize by tire radius.

You can optionally add inertia, stiffness, or damping. These dynamics require additional computation that tends to make models less suitable for real-time simulation.

Assumptions and Limitations

- The tire does not slip.
- Torque-transfer capability is infinite.

Ports

Conserving

A — Axle
mechanical rotational

Mechanical rotational port associated with the axle that the tire sits on.

H — Hub
mechanical translational

Mechanical translational port associated with the wheel hub that transmits the thrust generated by the tire to the remainder of the vehicle.

Parameters

The table shows how the visibility of some Tire (Simple) parameters depends on the option that you choose for other parameters. To learn how to read the table, see "Parameter Dependencies" on page B-2.

Parameter Dependencies

Parameter	
Rolling Radius	
Compliance — Choose `No compliance - Suitable for HIL simulation` or `Specify stiffness and damping`	
No compliance - Suitable for HIL simulation	**Specify stiffness and damping**
	Longitudinal stiffness
	Longitudinal damping
Inertia — Choose `No Inertia` or `Specify inertia and initial velocity`	
No Inertia	**Specify inertia and initial velocity**
	Tire inertia
	Initial velocity

Rolling radius — Radius
`0.3 m` (default) | positive scalar

Rolling radius of the tire. The rolling radius must be greater than zero.

Compliance — Compliance model
`No compliance - Suitable for HIL simulation` (default) | `Specify stiffness and damping`

Include or neglect longitudinal stiffness and damping.

Dependencies

Selecting `Specify stiffness and damping` exposes other parameters. For more information, see Parameter Dependencies.

Longitudinal stiffness — Longitudinal stiffness
1e6 N/m (default) | positive scalar

Longitudinal stiffness of the tire. The parameter must be greater than zero.

Dependencies

This parameter appears only if the **Compliance** parameter is set to Specify stiffness and damping. For more information, see Parameter Dependencies.

Longitudinal damping — Longitudinal damping
1000 N/(m/s) (default) | positive scalar

Longitudinal damping of the tire. The parameter must be greater than zero.

Dependencies

This parameter appears only if the **Compliance** parameter is set to Specify stiffness and damping. For more information, see Parameter Dependencies.

Inertia — Inertia model
No inertia (default) | Specify inertia and initial velocity

Include or neglect tire inertia.

Dependencies

Selecting Specify inertia and initial velocity exposes other parameters. For more information, see Parameter Dependencies.

Tire inertia — Tire inertia
1 kg*/m^2 (default) | positive scalar

Rotational inertia of the tire.

Dependencies

This parameter appears only if the **Inertia** parameter is set to Specify inertia and initial velocity. For more information, see Parameter Dependencies.

Initial velocity — Initial velocity
0 rad/s (default)

Initial rotational velocity of the tire.

Dependencies

This parameter appears only if the **Inertia** parameter is set to `Specify inertia and initial velocity`. For more information, see Parameter Dependencies.

Extended Capabilities

Real-Time and Hardware-in-the-Loop Simulation

For optimal simulation performance, set the **Parameters** > **Compliance** parameter to `No compliance - Suitable for HIL simulation`.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using Simulink® Coder™.

See Also
Tire (Friction Parameterized) | Tire (Magic Formula) | Tire-Road Interaction (Magic Formula)

Topics
"Complete Vehicle Model"
"Prepare Simscape Driveline Models for Real-Time Simulation Using Simscape Checks"

Introduced in R2012a

Torque Converter

Viscous fluid coupling between rotating driveline shafts

Library

Simscape / Driveline / Couplings & Drives

Description

A torque converter couples two driveline axes, transferring torque and angular motion by the hydrodynamic action of a viscous fluid. Unlike a friction clutch, a torque converter cannot lock the axes together. The Torque Converter block acts between the two ports I and T. The block acts as a lookup function of the relative angular velocity of the two connected driveline axes. This function is defined at discrete angular velocities. For model details, see "Torque Converter Model" on page 1-546.
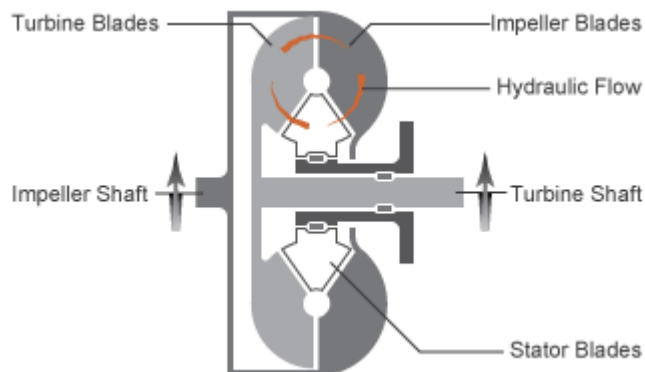
Ports and Conventions

The impeller or pump port I and turbine port T are rotational conserving ports.

The Torque Converter block follows these conventions:

- The impeller port I is the port that connects to the engine, and the turbine port T is the port that connects to the load. In normal operation, power thus flows from the impeller to the turbine.
- Forward power flow implies power flowing from I to T. Reverse power flow implies power flowing from T to I.
- The power input is through the shaft with the larger speed. The power output is through the shaft with the smaller speed.

Parameters

Torque Characteristics

Speed ratio vector

Vector of values of the independent variable, the dimensionless speed ratio R_ω. You must order these values in ascending order.

Torque ratio vector

Vector of values of the block function's first dependent variable, the dimensionless torque ratio R_τ. Each torque ratio value corresponds to a speed ratio value.

Capacity factor parameterization

Definition of the capacity factor, either K (ratio of impeller speed ω_I to square root of impeller torque τ_I) or K^* (ratio of τ_I to ω^2_I). The default is K.

Capacity factor reference speed

Choice of speed in the capacity factor definition, depending on speed ratio R_ω. Select either:

- Impeller speed ω_I for all values of R_ω.
- Impeller speed ω_I for $R_\omega < 1$, and turbine speed ω_T for $R_\omega > 1$.

Capacity factor vector

Vector of values of the block function's second dependent variable, the torque conversion capacity factor K. Each capacity factor value corresponds to a speed ratio value.

From the drop-down list, choose units.

- If you choose the default capacity factor definition K, the default units are radians/second/√(newton-meters) (`rad/s/(N*m)^0.5`).

- If you choose the alternative capacity factor definition K^*, the default units are newton-meters/(radians/second)^2 (`N*m/(rad/s)^2`).

Interpolation method

Interpolates torque ratio and capacity factor functions between discrete relative velocity values within the range of definition. The default is `Linear`.

Extrapolation method

Extrapolates torque ratio and capacity factor functions outside the range of definition. The default is `Linear`.

Dynamics

Model transmission lag

Select how to model transmission lag from input to output driveshaft. The default is `Specify time constant and initial torque`.

- `No lag — Suitable for HIL simulation` — Torque transfer is instantaneous.

- `Specify time constant and initial value` — Torque is transferred with a time lag. If you select this option, the panel changes from its default.

Time Constant and Initial Torque Ratio

Torque transmission time constant

Dynamic torque transfer time constant t_c. The default is `0.02`.

From the drop-down list, choose units. The default is seconds (`s`).

Initial turbine-to-impeller torque ratio

Ratio of turbine torque τ_T to impeller torque τ_I at the start of the simulation.

Torque Converter Model

The torque converter is a mechanism for transferring torque between impeller and turbine shafts. Because the coupling of I and T occurs by viscous action, the torque transfer depends on the difference $\omega = \omega_T - \omega_I \neq 0$, or the speed ratio $R_\omega \neq 1$. In normal operation, the two axes have different speeds, and the output axis speed never exactly reaches the input axis speed. The torque transfer is largest when $R_\omega \to 0$ or ∞, and shrinks as $R_\omega \to 1$. Because R_ω can never reach exactly one, a torque converter always transfers some torque.

Speed Ratio, Torque Ratio, and Capacity Factor

You specify the torque ratio and the capacity factor of the torque converter as discrete functions of the speed ratio with tabular vector entries. The three vectors of the variable values must have the same length.

- The speed ratio R_ω is the turbine angular speed divided by the impeller angular speed:

$$R_\omega = \omega_T/\omega_I \quad .$$

- The torque ratio R_τ is the output (turbine) torque divided by the input (impeller) torque:

$$R_\tau = \tau_T/\tau_I \quad .$$

- The capacity factor K or K^* is defined in two ways for $R_\omega < 1$, with either the default or the alternative definition:

  - Default, the input speed divided by the square root of the input torque:

  $$K = \omega_I / \sqrt{\tau_I} \quad .$$

  - Alternative, the input torque divided by the square of the input speed:

  $$K^* = \tau_I / \omega^2_I \quad .$$

- The capacity factor reference speed for $R_\omega > 1$ is ω_I by default. That is, the input speed ω_I is used in the ratio that defines either K or K^*.

  For $R_\omega > 1$, the alternative choice for reference speed is to replace ω_I by the output speed ω_T in this defining ratio.

The two dependent variables, R_τ and K, are functions of the independent variable R_ω. They specify the characteristics of the torque converter:

$$R_\tau = R_\tau(R_\omega) \quad , \quad K = K(R_\omega) \quad .$$

Continuation Near Speed Ratio of Unity

If you do not specify capacity factor data for a speed ratio of 1, the block uses a capacity factor value of $10 \cdot K_{Max}$, where K_{Max} is the maximum value in the specified capacity factor vector. The corresponding torque ratio is assumed to be 0. For all other speed ratio values not explicitly specified in the lookup table data, the block uses the interpolation or extrapolation method selected in the block dialog box.

Instantaneous Torque Transfer

When there is no time lag, the input impeller (I) and output turbine (T) torques are:

$$\tau_I = \text{sgn}(1 - \omega_T/\omega_I) \cdot [\omega_I / K]^2 \quad , \quad \tau_T = \tau_I \cdot R_\tau \quad ,$$

in normal operation (forward power flow).

Lagged Torque Transfer

You can optionally include the effect of torque transmission time lag, caused by internal fluid flow and compressibility. Instead of τ_T and τ_I being instantaneously constrained to one another, a first-order time lag introduces a delayed response in the impeller torque:

$$t_c \cdot (d\tau_I/dt) + \tau_I = \tau_I(\text{steady state}) \quad .$$

The preceding instantaneous function of the capacity factor K determines the steady-state value of τ_I.

Limitations

The impeller shaft must always rotate in a positive direction. Simulation is not valid for $\omega_I < 0$.

If you drive the Torque Converter from a torque source, such as the Generic Engine, you must include an inertia in the source, to represent the engine, shaft inertia, or other source components. To ensure that the impeller starts by rotating in a positive direction, set the initial speed for this inertia to a positive value.

Torque Transmission Time Lag

Torque converters lag in their response to changing input torque. By default, Torque Converter includes no time lag in its response. You can include a response lag by specifying a time constant. Time lag simulation increases model fidelity but reduces simulation performance. See "Adjust Model Fidelity".

Real-Time Simulation

Hardware-in-the-Loop Simulation

For optimal simulation performance, for the **Dynamics** > **Model transmission lag** parameter, select `No lag - Suitable for HIL simulation`.

Reference

Society of Automotive Engineers, *Hydrodynamic Drive Test Code (Surface Vehicle Recommended Practice),* SAE J643, May 2000.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also

Simscape Blocks
Disk Friction Clutch | Fundamental Friction Clutch | Generic Engine

Topics
"Complete Vehicle Model"

"Vehicle with Four-Speed Transmission"
"Vehicle with Four-Wheel Drive"
"Prepare Simscape Driveline Models for Real-Time Simulation Using Simscape Checks"

Torque Noise Source

Model zero-mean normally (Gaussian) distributed torque
Library: Simscape / Driveline / Sources

Description

The Torque Noise Source block generates zero-mean normally (Gaussian) distributed torque using the Random Number source in the Simscape Foundation library. The torque produced by the block depends on two factors:

- The noise power spectral density input to the block
- A random number seed that you control using the **Repeatability** and, as applicable, **Seed** parameters

The defining equation for the torque that the block generates is

$$T = \sqrt{PSD/2}\frac{N(0, 1)}{\sqrt{h}},$$

where:

- T is the output torque.
- PSD is the single-sided noise power spectral density.
- N is a Gaussian random number with zero mean and standard deviation of one.
- h is the sampling interval.

Ports

Input

P — Single-sided noise power spectral density
physical signal

The power spectral density (*PSD*) of noise is the average noise power per unit of bandwidth, in $(N*m)^2/Hz$.

Conserving

B — Base interface
mechanical rotational

Mechanical rotational conserving port associated with the base interface.

F — Follower interface
mechanical rotational

Mechanical rotational conserving port associated with the follower interface.

Parameters

Sample time — Time step period and offset
`1e-3` s (default) | `[`*step*`,` *offset*`]` s

The values of the time *step* period and the initial time *offset*. If you specify a scalar value for *step*, the block assumes an *offset* value of `0`. To specify a nonzero value for the initial time offset, specify the parameter values using the vector `[`*step*`,` *offset*`]`. The *offset* value must be less than the *step* value and greater than or equal to zero.

Repeatability — Random number seed control
`Not repeatable` (default) | `Repeatable` | `Specify seed`

The random number seed is the number that initializes the random number generator. The seed is `0` or a positive integer. To control the random number seed, set this parameter to:

- `Not repeatable` — The seed changes every time you simulate your model. The block resets the random seed using the MATLAB random number generator command:

  `seed = randi(2^32-1);`

- `Repeatable` — The seed is the same random number at the start of every simulation. The block sets the value using the same MATLAB random number generator command used by the `Not repeatable` parameter.

  When you add a Torque Noise Source block to your model from the Sources library, the block generates and stores a random value for the repeated seed. When you make a copy of the Torque Noise Source block from an existing block in a model, the copy generates a new random value for the repeated seed.

- `Specify seed` — The seed is a number that you specify using the **Seed** parameter. The **Seed** parameter is only available when you choose `Specify seed` for the **Repeatability** parameter.

Seed — Random number generation seed value
0 (default)

The seed must be 0 or a positive integer. This parameter is only available when you select `Specify seed` for the **Repeatability** parameter.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also

Simscape Blocks
Force Noise Source | Random Number | Rotational Velocity Noise Source | Sinusoidal Torque Source | Translational Velocity Noise Source

Functions
randi

Introduced in R2016b

Torsional Spring-Damper

Rotational spring and damper coupling, with Coulomb friction, locking, and hard stops

Library

Simscape / Driveline / Couplings & Drives

Description

The Torsional Spring-Damper block represents a dynamic element that imposes a combination of internally generated torques between the two connected driveshaft axes, the rod and the case. The complete torque includes these components:

• Linear damped spring
• Coulomb friction (including locking static friction)
• Hard-stop compliance

The second and third components are optional.

The Torsional Spring-Damper block uses the models of these blocks:

Block	Contribution	Library
Loaded-Contact Rotational Friction	Coulomb friction	**Simscape** / **Driveline** / **Brakes & Detents** / **Rotational**
Rotational Damper, Rotational Spring	Damping	**Simscape** / **Foundation Library** / **Mechanical** / **Rotational Elements**
Rotational Spring	Spring	
Rotational Hard Stop	Hard stop	

Assumptions and Limitations

- Including hard-stop and Coulomb friction enhances model fidelity, but reduces simulation speed. For more information, see "Driveline Simulation Performance".

Ports

Conserving

C

Mechanical rotational port associated with the slider that travels between stops installed on the case.

R

Mechanical rotational port associated with the rod.

Parameters

Spring-Damper

Restoring spring stiffness

Torsional spring stiffness *k* acting between connected driveshafts. The default is `1000 N*m/rad`. The value must be greater than zero.

Viscous friction coefficient

Torsional damping *μ* acting between the connected driveshafts. The default is `10 N*m/(rad/s)`. The value must be greater than or equal to zero.

Coulomb friction torque

Constant kinetic friction torque τ_K acting between connected driveshafts. The default is `0 N*m`. The value must be greater than or equal to zero.

Ratio of static to kinetic friction

Constant ratio *R* of static Coulomb friction torque τ_S to kinetic Coulomb friction torque τ_K acting between connected driveshafts. The default is `1.1`. The value must be greater than or equal to one.

Velocity tolerance

Minimum relative angular speed ω_{Tol} below which the two connected driveshafts can lock and rotate together. The default is `0.001 rad/s`. The value must be greater than zero

Hard Stops

Hard stop

Include or exclude hard-stop torque by selecting one of these options:

- `No hard stops — Suitable for HIL simulation` — To enhance simulation speed by excluding the hard-stop torque contribution, select this default option.
- `Compliant hard stops` — To enhance model fidelity by including the hard-stop torque contribution, select this option. Selecting this option enables other parameters.

Upper bound

Upper hard-stop angular displacement δ_+ from the zero-torque reference angle $\phi = 0$. The default is `10 deg`. The value must be greater than δ_-.

Selecting `Compliant hard stops` for the **Hard stop** parameter enables this parameter.

Lower bound

Lower hard-stop angular displacement δ_- from the zero spring force reference angle $\phi = 0$. The default is `-10 deg`. The value must be less than δ_+.

Selecting `Compliant hard stops` for the **Hard stop** parameter enables this parameter.

Contact stiffness

Hard-stop stiffness k_{HS} applied if the relative angle ϕ moves into the hard-stop region. The default is `1e6`. `N*m/rad`. The value must be greater than or equal to zero.

Selecting `Compliant hard stops` for the **Hard stop** parameter enables this parameter.

Contact damping

Hard-stop damping μ_{HS} applied if the relative angle ϕ moves into the hard-stop region.The default is `10 N*m/(rad/s)`. The value must be greater than zero.

Selecting `Compliant hard stops` for the **Hard stop** parameter enables this parameter.

Hard stop model

Select the hard-stop model:

- `Stiffness and damping applied smoothly through transition region, damped rebound` — Specify a transition region, in which the torque is scaled from zero. At the end of the transition region, the full stiffness and damping are applied. This model has damping applied on the rebound, but it is limited to the value of the stiffness torque. In this sense, damping can reduce or eliminate the torque provided by the stiffness, but never exceed it. All equations are smooth and produce no zero crossings.

- `Full stiffness and damping applied at bounds, undamped rebound` — This model has full stiffness and damping applied with impact at upper and lower bounds, with no damping on the rebound. Equations produce no zero crossings when velocity changes sign, but there is a position-based zero crossing at the bounds. Having no damping on rebound helps to push the slider past this position quickly. This model has nonlinear equations.

- `Full stiffness and damping applied at bounds, damped rebound` — This model has full stiffness and damping applied with impact at upper and lower bounds, with damping applied on the rebound as well. Equations are switched linear, but produce position-based zero crossings. Use this hard stop model if `simscape.findNonlinearBlocks` indicates that this is the block that prevents the whole network from being switched linear.

Selecting `Compliant hard stops` for the **Hard stop** parameter enables this parameter.

Transition region

Region where the torque is ramped up from zero to the full value. At the end of the transition region, the full stiffness and damping are applied. The default value is `0.001 rad`.

Selecting `Stiffness and damping applied smoothly through transition region, damped rebound` for the **Hard stop model** enables this parameter.

Initial Conditions

Initial deformation

Initial deformation of the torsional spring relative to the zero-torque reference angle $\phi = 0$. The default is `0 deg`.

Real-Time Simulation

Hardware-in-the-Loop Simulation

For optimal simulation performance, use the **Hard Stops** > **Hard stop** parameter default setting, `No hard stops - Suitable for HIL simulation`.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also

Simscape Blocks
Loaded-Contact Rotational Friction | Rotational Damper | Rotational Hard Stop | Rotational Spring

Topics
"Gear Coupling Control Using Clutches"
"Prepare Simscape Driveline Models for Real-Time Simulation Using Simscape Checks"

Translational Damper

Faultable linear translational damper
Library: Simscape / Driveline / Couplings & Drives / Springs &
                      Dampers
                      Simscape / Foundation Library / Mechanical /
                      Translational Elements

Description

The Translational Damper block models a faultable linear translational viscous damper. From the start of the simulation, the block uses the nominal damping coefficient. If you enable faults for the block, the damping coefficient changes in response to one or both of these triggers:

- Simulation time — A fault occurs at a specified time.

- Simulation behavior — If the magnitude of the translational acceleration exceeds a specified maximum value, the block increments the number of shocks by one. A fault occurs if the number of shocks exceeds a specified maximum number of shocks.

If a fault trigger event occurs, the block uses the faulted damping coefficient for the remainder of the simulation. You can program the block to issue a fault report as a warning or error message.

The defining equations are

$$F = Dv$$

and

$$v = v_R - v_C,$$

where

- F is the force transmitted through the damper.
- D is the damping (viscous friction) coefficient.
- v is the relative velocity.

- v_R and v_C are the absolute velocities of terminals R and C, respectively.

The positive block direction is from port R to port C. Therefore, the torque is positive if it acts in the direction from R to C.

Ports

Conserving

R — Rod interface
mechanical translational

Mechanical translational conserving port associated with the rod interface.

C — Case interface
mechanical translational

Mechanical translational conserving port associated with the case interface.

Parameters

Main

Damping coefficient — Damping coefficient for the no-fault condition
100 N/(m/s) (default)

The oscillation-reducing force due to viscous friction that the block applies if no trigger events occur. Specify a value that is greater than or equal to zero.

Faults

Enable faults — Fault mode capability
Off (default) | On

To enable the block to use a different damping coefficient based on the simulation time or when the magnitude of the translational acceleration exceeds the specified number of shocks, select On.

Dependencies

Selecting On makes these parameters visible:

- **Faulted damping coefficient**
- **Enable temporal fault trigger**
- **Enable behavioral fault trigger**
- **Reporting when fault occurs**

Faulted damping coefficient — Damping coefficient for the fault condition
1e6 N/(m/s) (default)

The oscillation-reducing force due to viscous friction that the block applies if a trigger event occurs. Specify a value that is greater than or equal to zero.

Dependencies

This parameter is only visible if you select On for the **Enable faults** parameter.

Enable temporal fault trigger — Fault trigger capability
Off (default) | On

To use the faulted damping coefficient starting at a specified time in the simulation, select On.

Dependencies

This parameter is only visible if you select On for the **Enable faults** parameter.

Selecting On makes the **Simulation time for fault event** parameter visible.

Simulation time for fault event — Temporal fault trigger
5 s (default)

Specify the time during the simulation that you want the block to start using the faulted damping coefficient.

Dependencies

This parameter is only visible if you select On for the **Enable faults** parameter and On for the **Enable temporal fault trigger** parameter.

Enable behavioral fault trigger — Fault trigger capability
Off (default) | On

To use the faulted damping coefficient when the number of sustained shocks exceeds a specified maximum number of shocks, select `On`.

Dependencies

This parameter is only visible if you select `On` for the **Enable faults** parameter.

Selecting `On` makes these parameters visible:

- **Maximum permissible acceleration**
- **Maximum number of shocks**

`Maximum permissible acceleration` — Behavioral fault trigger
`10 m/s^2` (default)

The maximum value that the magnitude of the translational acceleration can reach without causing the block to increment the number of sustained shocks.

Dependencies

This parameter is only visible if you select `On` for the **Enable faults** parameter and `On` for the **Enable behavioral fault trigger** parameter.

`Maximum number of shocks` — Behavioral fault trigger
`1` (default)

The maximum number of shocks that can be sustained without triggering the block to use the faulted damping coefficient.

Dependencies

This parameter is only visible if you select `On` for the **Enable faults** parameter and `On` for the **Enable behavioral fault trigger** parameter.

`Reporting when fault occurs` — Fault reporting
`None` (default) | `Warning` | `Error`

Select your reporting preference for the fault condition.

Dependencies

This parameter is only visible if you select `On` for the **Enable faults** parameter.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also
Nonlinear Rotational Damper | Nonlinear Translational Damper | Rotational Damper | Variable Rotational Damper | Variable Translational Damper

Introduced in R2016b

Translational Detent

Double-sided, spring-loaded translational detent

Library

Simscape / Driveline / Brakes & Detents / Translational

Description

The Translational Detent block represents a double-sided, spring-loaded translational detent with a ball and conical notch.

The detent slider slides horizontally over the detent case. These two parts develop a horizontal shear force between them. The inside of the slider is the conical notch. The detent case contains a vertical spring. The detent ball lies between the spring and the conical notch. With the block, you can model ball and angled notch detents with geometric and spring characteristics, peak force and notch width, or a table lookup. The model also includes viscous damping and kinetic friction between the slider and case. You can set the friction to zero.

Limitations

- The model does not account for inertia. Add mass terms externally to the R and C ports as required.

- If you use the peak force-notch width or the table lookup parameterization, the kinetic friction is independent of the detent normal force.

Ports

R and C are translational conserving ports associated with the slider and case, respectively.

Parameters

Detent Characteristics

Parameterization

Select how to specify the detent characteristics. The default is By peak force and notch width.

• By peak force and notch width — Specify detent characteristics by the peak shear force and the notch width.

Peak Force and Notch Width

Peak force

The peak shear force developed inside the detent. The default is 10.

From the drop-down list, choose units. The default is newtons (N).

Notch width

The width of the region where the shear force develops. The default is 30.

From the drop-down list, choose units. The default is millimeters (mm).

• By table lookup — Define the detent characteristics by one-dimensional table lookup based on the relative displacement between the slider and case. If you select this option, the panel changes from its default option.

By Table Lookup

Relative displacement vector

Specify the vector of input relative displacements that correspond to the values in **Force vector**. These two vectors must be of the same size. The displacements must be specified in strictly increasing order. The default values are [-15 -12.3 -7.6 -1.1 1.1 7.6 12.3 15].

From the drop-down list, choose units. The default is millimeters (mm).

Force vector

Specify the vector of input values for the shear force that correspond to the values in **Relative displacement vector**. The default values are [0 0 -8.365 -5.87 5.87 8.365 0 0].

From the drop-down list, choose units. The default is newtons (N).

Interpolation method

Select one of these interpolation methods to approximate the output value when the input value is between two consecutive grid points:

- Linear — Select this, default, option to get the best performance.
- Smooth — Select this option to produce a continuous curve with continuous first-order derivatives.

For more information on interpolation algorithms, see the PS Lookup Table (1D) block reference page.

- By geometry — Define the detent characteristics by ball-notch geometry and dynamics. If you select this option, the panel changes from its default option.

Geometric and Spring Parameters

Ball diameter

Specify the diameter of the ball. The default is 12.

From the drop-down list, choose units. The default is millimeters (mm).

Notch angle

Specify the full angle of the notch. Must be greater than 0 and less than or equal to 179 degrees. The default is 120.

From the drop-down list, choose units. The default is degrees (deg).

Notch depth

Specify the depth of the notch in the slider. Must be greater than zero. The default is 4.5.

From the drop-down list, choose units. The default is millimeters (mm).

Preload force

Specify the force to which the detent spring is preloaded. Must greater than or equal to 0. The default is 15.

From the drop-down list, choose units. The default is newtons (`N`).

Spring stiffness

Specify the stiffness of the detent spring. Must be greater than 0. The default is `3000`.

From the drop-down list, choose units. The default is newtons/meter (`N/m`).

Shear reversal region

Specify the width of the region where the ball moves from one notch face to the other. Must be greater than 0. The default is `0.1`.

From the drop-down list, choose units. The default is millimeters (`mm`).

Initial relative position

The relative displacement of the slider and case when simulation starts. The default is `0`.

From the drop-down list, choose units. The default is millimeters (`mm`).

Friction

Viscous friction coefficient

Specify the viscous friction coefficient for the ball-notch contact. Must be greater than or equal to 0. The default is `0.1`.

From the drop-down list, choose units. The default is newtons/(meters/second) (`N/(m/s)`).

Friction to peak force ratio

Specify the kinetic friction coefficient for the ball-notch contact. Must be greater than or equal to 0. The default is `0.01`.

The kinetic friction is this ratio multiplied by the peak shear force.

Friction velocity threshold

Specify the relative velocity required for peak kinetic friction in the detent. Must be greater than 0. The default is `0.05`.

From the drop-down list, choose units. The default is meters/second (`m/s`).

Translational Detent Model

The geometry of the ball-notch detent is shown in the following figure. As the conical slider slides horizontally over the case, the relative displacement $x = x_R - x_C$ causes a horizontal shear force F to develop.
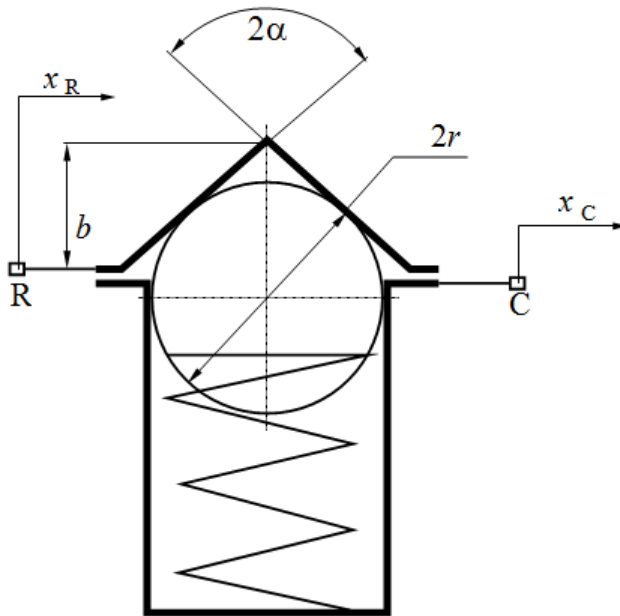

b	Notch depth
r	Ball radius
α	Notch half angle

Depending on your choice of parameterization, the shear force model is defined by geometric and spring parameters, by the peak force and notch width, or by a table lookup specifying relative displacement versus force.

Geometric and Spring Parameters

The geometric and spring parameterization uses four regions to define the shear force. The following figure displays these regions.
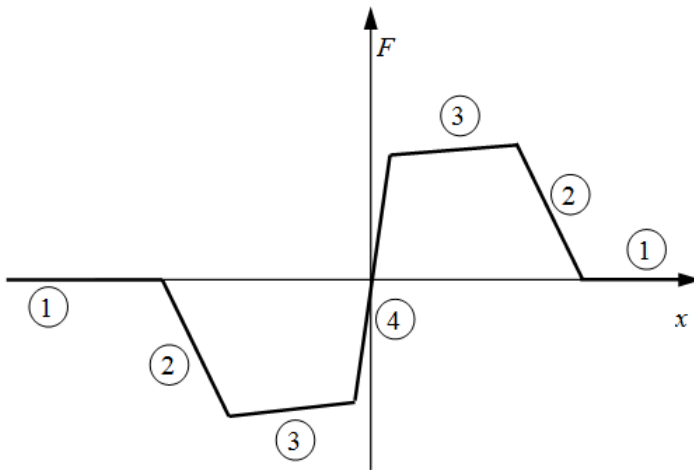
Because the ball is spherical and the notch is symmetric, the horizontal force versus displacement function is symmetric about the origin. (See "Peak Force and Notch Width" on page 1-569.)

Region	Ball Position and Contact Angle	Shear Force
1	Ball is outside conical notch. Contact angle is vertical.	Viscous damping and kinetic friction only. No spring contribution.
2	Ball is entering notch, in contact with notch corner. Contact angle rotates from vertical.	Spring, viscous damping, and kinetic friction contributions. Peaks at transition from region 2 to region 3.
3	Ball slides along face of notch. Contact angle is constant.	Force decreases as spring extends, but does not go to zero (spring preload force).
4	Ball moves from one face to the other. Contact angle switches direction.	Force switches direction. Shear reversal region width controls how smooth this switch is.

Peak Force and Notch Width

If you choose the peak force and notch width parameterization, the block ensures that the force-relative displacement curve provides a continuous force and force derivative over

the detent region. The peak forces are halfway between the detent center and detent edge, as shown in the following figure.
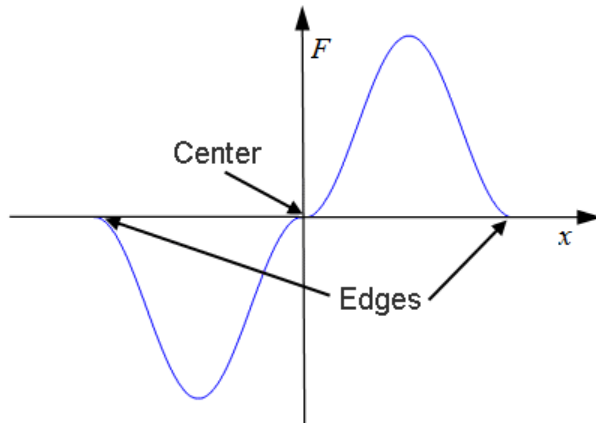
By Table Lookup

With the table lookup parameterization, you can create an arbitrary function relating shear force to relative displacement. If you create such a function, consider the following best practices.

Energy Conservation

If you want to ensure that the detent conserves energy, the total integral of the force-relative displacement curve (area under the curve) must be zero.

Simulation Performance

To stabilize simulation of the detent, avoid discontinuities in the force-relative displacement function. The most important requirement is to have a shear reversal region of nonzero width, analogous to region 4 in the geometric parameterization.

See Also

Loaded-Contact Translational Friction | Rotational Detent

Topics

"Clutches, Clutch-Like Elements, and Coulomb Friction"
"Stepping Mechanism with Detents"

Translational Power Sensor

Mechanical sensor used to measure average or instantaneous translational power
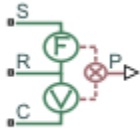
Library

Sensors

Description

The block represents a sensor that measures the translational power change due to one or more driveline components connected between ports R and C. The power output can be an instantaneous or average value. Two averaging options are available: period and vibration averaging.

The instantaneous power is the product of the current velocity and force. Use the instantaneous power to analyze peak power transmission and identify heavily loaded components.

The time-averaged power is the integral of the instantaneous power over one sample period, divided by that period. For simulation times smaller than the sample period, the average power is zero. The power is constant for each sample period.

The vibration-averaged power is the integral of the power over one vibration period, divided by that period. The vibration period is the time between two consecutive crossings from negative to positive velocity values. Use this averaging method only for oscillations with velocity passing through zero. If the vibration has a variable frequency, the averaging period can change. For simulation times smaller than the vibration period, the average power is zero. The power output is constant for each vibration period.

Parameters

Measurement Type

Select either instantaneous or average power measurement. The default type is `Instantaneous Power`.

- `Instantaneous power` — Output the power measured for the current time step.
- `Average power over a specified period`— Output the power averaged over a fixed time period.

  Averaging period

  Enter the time period used to compute the average power. Typical values include the cycle length of a reciprocating device or the actuation force.

  The default value is `1`. The default unit is `s`.

- `Average power over a vibration`— Output the power averaged over one vibration period.

  Maximum vibration frequency

  Enter the maximum vibration frequency over which to average the power. Very high values can reduce simulation speed. Very low values can result in lower power measurements.

  The default value is `1000`. The default unit is `Hz`.

Ports

Port	Description
P	Physical signal output port for translational power
S	Mechanical translational conserving port associated with driving source
R	Mechanical translational conserving port associated with loaded driveline components
C	Mechanical translational conserving port associated with case holding loaded driveline components

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also
Rotational Power Sensor

Introduced in R2013a

Translational Velocity Noise Source

Produce zero-mean normally (Gaussian) distributed translational velocity
Library: Simscape / Driveline / Sources

Description

The Translational Velocity Noise Source block generates zero-mean normally (Gaussian) distributed translational velocity using the Random Number source in the Simscape Foundation library. The translational velocity produced by the block depends on two factors:

- The noise power spectral density input to the block
- A random number seed that you control using the **Repeatability** and, as applicable, **Seed** parameters

The defining equation for the translational velocity that the block generates is

$$v = \sqrt{PSD/2}\frac{N(0, 1)}{\sqrt{h}},$$

where:

- v is the output translational velocity.
- PSD is the single-sided noise power spectral density.
- N is a Gaussian random number with zero mean and standard deviation of one.
- h is the sampling interval.

Ports

Input

P — Single-sided noise power spectral density
physical signal

The power spectral density (*PSD*) of noise is the average noise power per unit of bandwidth, in $(m/s)^2/Hz$.

Conserving

B — Base interface
mechanical translational

Mechanical translational conserving port associated with the base interface.

F — Follower interface
mechanical translational

Mechanical translational conserving port associated with the follower interface.

Parameters

Sample time — Time step period and offset
`1e-3` s (default) | `[`*step*`,` *offset*`]` s

The values of the time *step* period and the initial time *offset*. If you specify a scalar value for *step*, the block assumes an *offset* value of `0`.To specify a nonzero value for the initial time offset, specify the parameter values using the vector `[`*step*`,` *offset*`]`. The *offset* value must be less than the *step* value and greater than or equal to zero.

Repeatability — Random number seed control
`Not repeatable` (default) | `Repeatable` | `Specify seed`

The random number seed is the number that initializes the random number generator. The seed is `0` or a positive integer. To control the random number seed, set this parameter to:

- `Not repeatable` — The seed changes every time you simulate your model. The block resets the random seed using the MATLAB random number generator command:

  `seed = randi(2^32-1);`

- `Repeatable` — The seed is the same random number at the start of every simulation. The block sets the value using the same MATLAB random number generator command used by the `Not repeatable` parameter.

  When you add a Translational Velocity Noise Source block to your model from the Sources library, the block generates and stores a random value for the repeated seed. When you make a copy of the Translational Velocity Noise Source block from an existing block in a model, the copy generates a new random value for the repeated seed.

- `Specify seed` — The seed is a number that you specify using the **Seed** parameter. The **Seed** parameter is only available when you choose `Specify seed` for the **Repeatability** parameter.

Seed — Random number generation seed value
0 (default)

The seed must be 0 or a positive integer. This parameter is only available when you select `Specify seed` for the **Repeatability** parameter.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also

Simscape Blocks
Force Noise Source | Random Number | Rotational Velocity Noise Source | Sinusoidal Translational Velocity Source | Torque Noise Source

Functions
randi

Introduced in R2016b

Unbalanced Load

Load with angle-dependent rotational inertia

Library

Simscape / Driveline / Inertias & Loads

Description

This block models the effects of a load whose rotational inertia varies instantaneously with the rotation angle. Examples include the wobbling of a spinning axle and shaking of an off-center rotating machine. You can specify the load inertia in terms of its rotation path or as an angle-inertia lookup table.

The block provides two rotation path parameterizations. One assumes circular motion of the unbalanced load. The other assumes elliptical motion. In each case, the distance between the unbalanced load and the main rotation axis varies with the rotation angle. The varying distance causes the moment of inertia, given by the parallel-axis theorem, to vary during rotation:

$$I(\theta) = I_0 + Mr^2(\theta),$$

where:

- I is the moment of inertia about the main rotation axis.

- I_0 is the moment of inertia about the intrinsic rotation axis of the unbalanced load.

- M is the total mass of the unbalanced load.

- r is the distance between the unbalanced load and the main rotation axis.

1-579

The angle dependence of the moment of inertia leads to an angle dependence of the inertial torque acting on the unbalanced load:

$$T_I(\theta) = I(\theta)\dot{\omega},$$

where:

- T_I is the inertial torque.
- ω is the angular velocity of the load.

If the load moves in a circular path, its distance to the main rotation axis is given by

$$r(\theta) = \sqrt{(r_C cos(\theta) + r_A)^2 + (r_C sin(\theta))^2},$$

where:

- r_A is the distance between the main rotation axis and the offset rotation axis.
- r_C is the radius of the circular path.
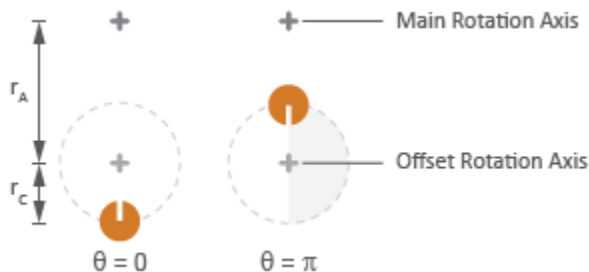
Circular Rotation Path

If the load moves in an elliptical path, its distance to the main rotation axis is given by

$$r(\theta) = \sqrt{(r_{\parallel} cos(\theta) + r_A)^2 + (r_{\perp} sin(\theta))^2},$$

where:

- $r_{\parallel}$ is the radius of the elliptical path in line with the distance from the main rotation axis.

- $r_\perp$ is radius of the elliptical path perpendicular to the distance from the main rotation axis.
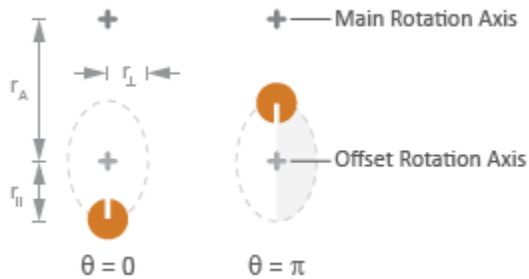
Elliptical Rotation Path

Ports

The block has one conserving rotational port. This port identifies the location of the unbalanced load in a driveline model.

Parameters

Inertia

Parameterization

Choice of parameterization for modeling the moment of inertia of the unbalanced load. Options include `Circular rotation`, `Elliptical rotation`, and `By table lookup`. The default setting is `Circular rotation`.

Select `Circular rotation` or `Elliptical rotation` to parameterize the load inertia in terms of the load's rotation path. Select `By table lookup` to provide the load inertia directly across its rotation path.

Moment of inertia about own axis

Moment of inertia about the intrinsic rotation axis of the unbalanced load. This moment of inertia is assumed constant throughout simulation. This parameter

1-581

appears only when **Parameterization** is set to `Circular rotation`. The default value is `1` kg*m2.

Mass

Mass of the unbalanced load. This parameter appears only when **Parameterization** is set to `Circular rotation`. The default value is `1` kg.

Distance from axis to center of rotation

Distance between the main rotation axis and the center of the rotation path. This parameter appears only when **Parameterization** is set to `Circular rotation`. The default value is `0.01` m.

Radius of rotation

Radius of the circular rotation path traversed by the unbalanced load. This parameter appears only when **Parameterization** is set to `Circular rotation`. The default value is `0.01` m.

Radius of rotation inline with distance from axis

Length of the semimajor axis of the elliptical rotation path traversed by the unbalanced load. The semimajor axis is in line with the line segment joining the main rotation axis and the elliptical rotation center. This parameter appears only when **Parameterization** is set to `Elliptical rotation`. The default value is `0.01` m.

Radius of rotation perpendicular to distance from axis

Length of the semiminor axis of the elliptical rotation path traversed by the unbalanced load. The semiminor axis is perpendicular to the line segment joining the main rotation axis and the elliptical rotation center. This parameter appears only when **Parameterization** is set to `Elliptical rotation`.The default value is `0.005` m.

Angle vector

Array of angular positions at which to specify the load inertia. The block calculates the load inertia between the angles specified using linear interpolation. This parameter appears only when **Parameterization** is set to `By table lookup`. The default vector is `[0,pi,3*pi/2]` rad.

Inertia vector

Array of load inertias about the intrinsic rotation axis of the load. Each inertia corresponds to an angle specified in the **Angle vector** parameter. This parameter appears only when **Parameterization** is set to `By table lookup`. The default vector is `[1,10,1]` kg*m^2.

Initial Conditions

Initialization

    Choice of parameter through which to specify the initial state of the unbalanced load. Options include `Specify angular momentum` and `Specify angular velocity`.

Initial angular momentum

    Angular momentum of the unbalanced load at the start of simulation. This parameter appears only when the **Initialization** parameter is set to `Specify angular momentum`. The default value is `0` m^2*kg/s.

Initial angular velocity

    Angular velocity of the unbalanced load at the start of simulation. This parameter appears only when the **Initialization** parameter is set to `Specify angular velocity`. The default value is `0` rad/s.

Initial angle

    Angular position of the unbalanced load at the start of simulation. The initial angle determines the initial inertia of the unbalanced load. The default value is `0` rad.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also
Variable Inertia | Variable Mass

Unidirectional Clutch

Clutch that transmits power in a single direction

Library

Simscape / Driveline / Clutches

Description

The Unidirectional Clutch block models an ideal one-way clutch based on a coupled ring-hub. It transmits torque when the relative velocity of hub and ring becomes positive or negative, depending on how you set the transmission direction. For model details, see "Unidirectional Clutch Model" on page 1-585.

Ports

H and R are rotational conserving ports representing, respectively, the hub and ring driveshaft axes. The clutch motion is measured as the *slip* $\omega = \omega_H - \omega_R$, the angular velocity of hub relative to ring.

Parameters

Transmission direction

From the drop-down list, select the relative rotation direction (sign of) for which the clutch locks and transfers torque. The default is `From hub to ring for positive rotation`.

Initial state

Clutch state at the start of simulation. The clutch can be in one of two states, locked and unlocked. A locked clutch constrains the base and follower shafts to spin at the

same velocity, that is, as a single unit. An unlocked clutch allows the two shafts to spin at different velocities, resulting in slip between the clutch plates. The default setting is `Unlocked`.

Unidirectional Clutch Model

The Unidirectional Clutch, if locked, transmits torque from hub to ring. If it is unlocked, it transmits no torque. The clutch does not have a state of being unlocked but still engaged (transmitting torque).

Clutch Geometry, Variable, and State Summary

The figure displays the geometry of the unidirectional clutch. The table summarizes the unidirectional clutch variables.
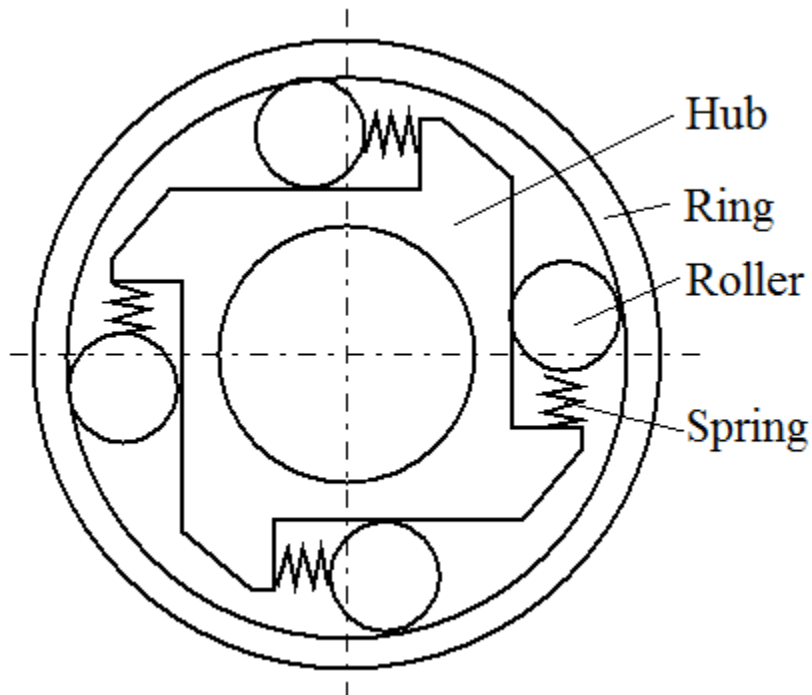
Clutch Variable

Symbol	Definition	Significance
ω	Relative angular velocity (slip)	$\omega_H - \omega_R$

The Unidirectional Clutch block implements an ideal model of a one-way clutch and does not include any internal dynamics, such as the rollers and springs.

Locking Condition

The transmission direction determines the condition for locking.

- If you choose transmission for the positive direction, the clutch locks if $\omega = \omega_H - \omega_R \geq 0$ and becomes zero from positive values. The slip ω then remains zero, and the clutch transfers torque.

  If the driveline dynamics then causes $\omega < 0$ at a later time, the clutch abruptly unlocks and stops transferring torque.

- If you choose transmission for the negative direction, the clutch locks if $\omega = \omega_H - \omega_R \leq 0$ and becomes zero from negative values. The slip ω then remains zero, and the clutch transfers torque.

  If the driveline dynamics then causes $\omega > 0$ at a later time, the clutch abruptly unlocks and stops transferring torque.

For example, locking and torque transfer in the positive direction as follows.

- When the hub rotates faster than the ring ($\omega > 0$), the small gap prearranged between the hub and the ring is cleared. The spring-loaded rollers wedge between the hub and the ring, causing the two to contact, lock, and rotate together; ω resets to zero. (The rotation convention in the figure is clockwise.)

- If the ring rotates faster than the hub ($\omega < 0$), the rollers are pushed out of contact and the clutch unlocks.

For a clutch schematic, see "Clutch Geometry, Variable, and State Summary" on page 1-585.

Power Dissipated by the Clutch

The unidirectional clutch is ideal and does not dissipate power. Either the clutch is unlocked and applies no kinetic friction, or it is locked (no slip) and applies only static friction.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.
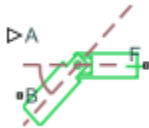
See Also
Dog Clutch | Fundamental Friction Clutch

Topics
"Ratchet Leadscrew Mechanism"
"Gear Coupling Control Using Clutches"

Universal Joint

Rotational coupling between two driveline shafts

Library

Couplings & Drives

Description

This block represents a rotational coupling between two driveline shafts. The coupling transfers torque between the shafts so they spin as a unit under an applied load. Two rotational degrees of freedom, internal to the coupling, allow the shafts to connect at an angle. This intersection angle varies according to the physical signal input from port A. Compliance, modeled as a parallel spring-damper set, allows the coupling to deform under load.

You can use the Universal Joint block as a connection between two rotational driveline components—for example, between the driving and driven shafts in an automobile drive train.
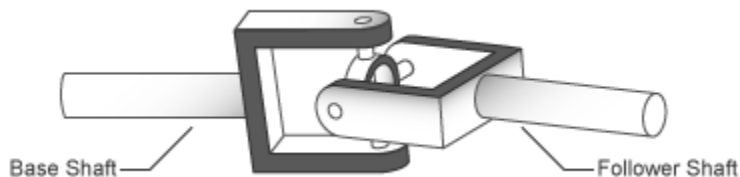
The ratio of the shaft angular velocities depends on two parameters: the intersection angle between the two shafts and the rotation angle of the base shaft. A physical signal

input provides the intersection angle while a dialog box parameter provides the initial base shaft angle. These two angles fix the ratio of the two shaft angular velocities according to the nonlinear equation:

$$\omega_F = \frac{\cos(A)}{1 - \sin^2(A) \cdot \cos^2(\theta_B)} \omega_B,$$

where:

- ω_F is the angular velocity of the follower shaft about its length axis.

- ω_B is the angular velocity of the base shaft about its length axis.

- θ_B is the rotation angle of the base shaft about its length axis.

- A is the intersection angle between base and follower shafts about the base shaft pin.

The two schematics in the figure illustrate the equation parameters. In each schematic, the left shaft represents the base shaft, while the right shaft represents the follower shaft. The right schematic shows the coupling seen in the left schematic after the shafts spin 90° about their length axes (dashed line segments).
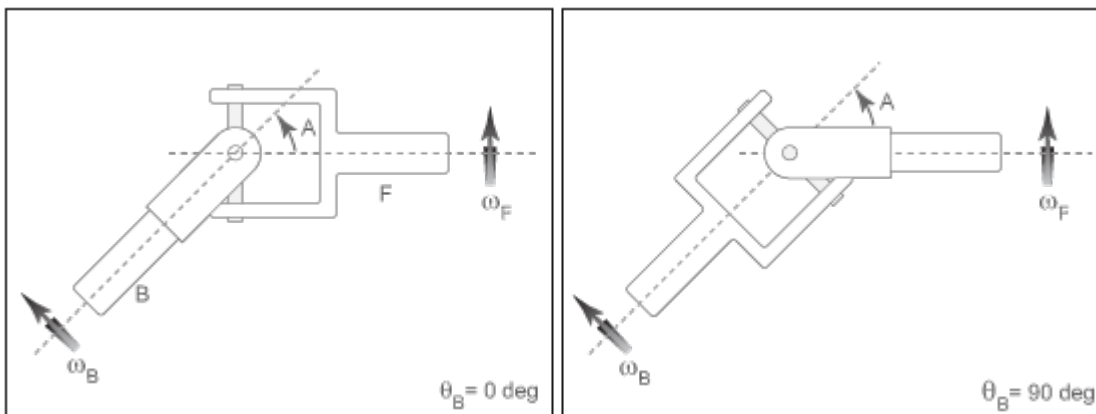
In the figure, the intersection angle is the angle between the two shafts about the pin of the base shaft. The absolute value of this angle must fall in the range `0≤A<Maximum intersection angle`. The base shaft angle is the angle of the base shaft about its length axis. The base shaft angle is also the time-integral of the base shaft angular velocity, ω_B.

Parameters

Maximum joint angle

Maximum intersection angle the joint allows. This angle measures the rotation between base and follower shafts about the base shaft pin. The value of this angle must fall in the range `0 ≤ A < pi/2`. From the drop-down list, select a physical unit. The default value is `pi/4` rad.

Joint stiffness

Linear spring stiffness of the joint. The spring stiffness accounts for elastic energy storage in the joint due to material compliance. In the drop-down list, select a physical unit. The default value is `1e6` N*m/rad.

Joint damping

Linear damping coefficient of the joint. The damping coefficient accounts for energy dissipation in the joint due to material compliance. In the drop-down list, select a physical unit. The default value is `1e3` N*m/(rad/s).

Initial base shaft angle

Rotation angle of the base shaft about its length axis at time zero. From the drop-down list, select a physical unit. The default value is `0` rad.

Initial torque from base to follower shaft

Torque that the base shaft transfers to the follower shaft at time zero. This torque determines the initial state of material compliance at the joint. Set this value to greater than zero to preload the shafts with torque. Changing this value alters the initial transient response due to material compliance. Select a physical unit. The default value is `0` N*m.

Ports

Port	Description
B	Conserving rotational port associated with the base shaft
F	Conserving rotational port associated with the follower shaft
A	Physical signal input port for the intersection angle

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also
Belt Drive | Belt Pulley | Chain Drive | Flexible Shaft | Rope Drum

Variable Inertia

Time-varying inertia specified as physical signal input

Library

Inertias & Loads

Description

This block models the effects of a time-varying inertia. Components with variable inertia include satellites unfolding arrays of solar panels. A physical signal input port provides the instantaneous inertia at each time step throughout simulation. To minimize solver errors, avoid discontinuities and abrupt changes in the input signal.

As the component inertia changes, so does the inertial torque acting on that component:

$$T_I = I\dot{\omega},$$

where:

- T_I is the inertial torque.
- I is the variable inertia.
- ω is the angular velocity of the inertia.

The physical signal input saturates at the minimum inertia specified in the block dialog box. Whenever the input falls below the minimum inertia, the block performs its calculations on the minimum inertia instead. The inertial torque becomes:

$$T_I = I_{Min}\dot{\omega},$$

where:

- I_{Min} is the minimum inertia allowed.

The angular momentum of the variable inertia is conserved. In the absence of external torques, a decrease in the moment of inertia leads to a proportional increase in angular velocity. The angular momentum remains constant until a net external torque causes it to change.

Variables

Use the **Variables** tab to set the priority and initial target values for the block variables before simulating. For more information, see "Set Priority and Initial Target for Block Variables" (Simscape).

Unlike block parameters, variables do not have conditional visibility. The **Variables** tab lists all the existing block variables. If a variable is not used in the set of equations corresponding to the selected block configuration, the values specified for this variable are ignored.

Ports

The block has one conserving rotational port and one physical signal input port. The conserving rotational port identifies the driveline component whose inertia varies with time. The physical signal input port provides the means to specify the inertia as a function of time.

Parameters

Minimum inertia

Lowest value that the variable inertia can take on during simulation. The physical signal input saturates at this value. If the input falls below the minimum inertia, it is set equal to the minimum inertia value. The default value is `0.1 kg*m^2`.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also
Unbalanced Load | Variable Mass

Variable Mass

Time-varying mass specified as physical signal input

Library

Inertias & Loads

Description

This block models the effects of a time-varying mass. Components with variable mass include fuel tanks with progressively lower fuel contents. A physical signal input port provides the instantaneous mass at each time step throughout simulation. To minimize solver errors, avoid discontinuities and abrupt changes in the input signal.

As the component mass changes, so does the inertial force acting on that component:

$$F_I = m\dot{v},$$

where:

- F_I is the inertial force.
- m is the variable mass.
- v is the velocity of the mass.

The physical signal input saturates at the minimum mass specified in the block dialog box. Whenever the input falls below the minimum mass, the block performs its calculations on the minimum mass instead. The inertial force becomes:

$$F_I = m_{Min}\dot{v},$$

where:

- m_{Min} is the minimum mass allowed.

The linear momentum of the variable mass is conserved. In the absence of external forces, a decrease in mass leads to a proportional increase in linear velocity. The linear momentum remains constant until a net external force causes it to change.

Variables

Use the **Variables** tab to set the priority and initial target values for the block variables before simulating. For more information, see "Set Priority and Initial Target for Block Variables" (Simscape).

Unlike block parameters, variables do not have conditional visibility. The **Variables** tab lists all the existing block variables. If a variable is not used in the set of equations corresponding to the selected block configuration, the values specified for this variable are ignored.

Ports

The block has one conserving translational port and one physical signal input port. The conserving translational port identifies the driveline component whose mass varies with time. The physical signal input port provides the means to specify the mass as a function of time.

Parameters

Minimum mass

Lowest value that the variable mass can take on during simulation. The physical signal input saturates at this value. If the input falls below the minimum mass, it is set equal to the minimum mass value. The default value is `0.1 kg`.

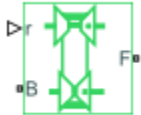# Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also
Unbalanced Load | Variable Inertia

Variable Ratio Transmission

Dynamic gearbox with variable and controllable gear ratio, transmission compliance, and friction losses

Library

Simscape / Driveline / Couplings & Drives

Description

The Variable Ratio Transmission block represents a gearbox that dynamically transfers motion and torque between the two connected driveshaft axes, base and follower.

Ignoring the dynamics of transmission compliance, the driveshafts are constrained to corotate with a variable gear ratio that you control. You can choose whether the follower axis rotates in the same or opposite direction as the base axis. If they rotate in the same direction, ω_F and ω_B have the same sign. If they rotate in opposite directions, ω_F and ω_B have opposite signs.

Transmission compliance introduces internal time delay between the axis motions. Therefore, unlike a gear, a variable ratio transmission does not act as a kinematic constraint. You can also control torque loss caused by transmission and viscous losses. For model details, see "Variable Ratio Transmission Model" on page 1-599.

Ports

B and F are rotational conserving ports representing, respectively, the base and follower driveshafts.

You specify the unitless variable gear ratio $g_{FB}(t)$ as a function of time at the physical signal input at port r. If the signal value becomes zero or negative, the simulation stops with an error.

Variable Ratio Transmission Model

Ideal Motion and Torque Transfer

Variable Ratio Transmission is a dynamical mechanism for transferring motion and torque between base and follower.

If the relative compliance ϕ between the axes is absent, the block is equivalent to a gear with a variable ratio $g_{FB}(t)$. Such a gear imposes a time-dependent kinematic constraint on the motions of the two driveshafts:

$$\omega_B = \pm g_{FB}(t)\cdot\omega_F \quad , \quad \tau_F = \pm g_{FB}(t)\cdot\tau_B \quad .$$

However, Variable Ratio Transmission does include compliance between the axes. Dynamical motion and torque transfer replace the kinematic constraint, with a nonzero ϕ that dynamically responds through base compliance parameters k_p and k_v:

$$d\phi/dt = \pm g_{FB}(t)\cdot\omega_F - \omega_B \quad ,$$

$$\tau_B = -k_p\phi - k_v d\phi/dt \quad ,$$

$$\pm g_{FB}(t)\cdot\tau_B + \tau_F - \tau_{loss} = 0 \quad .$$

$\tau_{loss} = 0$ in the ideal case.

Estimating Compliance Parameters

- You can estimate the base angular compliance k_p from the transmission time constant t_c and inertia J: $k_p = J(2\pi/t_c)^2$.
- You can estimate the base angular velocity compliance k_v from the transmission time constant t_c, inertia J, and damping coefficient C: $k_v = (2Ct_c)/2\pi = 2C\sqrt{(J/k_p)}$.

Nonideal Torque Transfer and Losses

With nonideal torque transfer, $\tau_{\text{loss}} \neq 0$. Losses in the Variable Ratio Transmission are modeled similarly to how losses are modeled in nonideal gears. For general information on nonideal gear modeling, see "Model Gears with Losses".

In a nonideal gearbox, the angular velocity and compliance dynamics are unchanged. The transferred torque and power are reduced by:

- Coulomb friction (for example, between belt and wheel, or internal belt losses due to stretching) characterized by an efficiency η.
- Viscous coupling of driveshafts with bearings, parametrized by viscous friction coefficients μ.

$$\tau_{\text{loss}} = \tau_{\text{Coul}} \cdot \tanh(4\omega_{\text{out}}/\omega_{\text{th}}) + \mu_B \omega_B + \mu_F \omega_F \quad, \quad \tau_{\text{Coul}} = |\tau_F| \cdot (1 - \eta) \quad .$$

When the angular velocity changes sign, the hyperbolic tangent regularizes the sign change in the Coulomb friction torque.

Power Flow	Power Loss Condition	Output Driveshaft ω_{out}
Forward	$\omega_B \tau_B > \omega_F \tau_F$	Follower, ω_F
Reverse	$\omega_B \tau_B < \omega_F \tau_F$	Base, ω_B

Meshing Efficiency

The friction loss represented by efficiency η is fully applied only if the absolute value of the follower angular velocity ω_F is greater than a velocity threshold ω_{th}.

If this absolute velocity is less than ω_{th}, the actual efficiency is automatically regularized to one at zero velocity.

Parameters

Main

Output shaft rotates

From the drop-down list, choose how the output driveshaft rotates relative to the input driveshaft. The default is In same direction as input shaft.

Compliance

Transmission stiffness at base (B)

Reciprocal of transmission angular compliance k_p, angular displacement per unit torque, measured at the base. The default is `30000`.

From the drop-down list, choose units. The default is newton-meters/radian (`N*m/rad`).

Transmission damping at base (B)

Reciprocal of transmission angular compliance damping k_v, angular speed per unit torque, measured at the base. The default is `0.05`.

From the drop-down list, choose units. The default is newton-meters/(radian/second) (`N*m/(rad/s)`).

Initial input torque at base (B)

Torque applied at the base driveshaft at the start of simulation ($t = 0$). The default is `0`.

From the drop-down list, choose units. The default is newton-meters (`N*m`).

Transmission Losses

Losses model

Select how to implement friction losses from nonideal torque transfer. The default is `No losses`.

- `No losses — Suitable for HIL simulation` — Torque transfer is ideal.
- `Constant efficiency` — Transfer of torque across gearbox is reduced by a constant efficiency η satisfying $0 < \eta \leq 1$. If you select this option, the panel changes from its default.

   Constant Efficiency

   Efficiency

   Effective torque transfer efficiency η between the base and follower driveshafts.

   Follower angular velocity threshold

   Absolute angular velocity threshold ω_{th} above which full efficiency loss is applied, for follower velocity ω_F.

From the drop-down list, choose units. The default is radians/second (`rad/s`).

Viscous Losses

Viscous friction coefficients at base (B) and follower (F)

Vector of viscous damping coefficients [μ_B μ_F] applied at the base and follower driveshafts, respectively. The default is [0 0].

From the drop-down list, choose units. The default is newton-meters/(radian/second) (`N*m/(rad/s)`).

Real-Time Simulation

Hardware-in-the-Loop Simulation

For optimal simulation performance, use the **Transmission Losses** > **Losses model** parameter default setting, `No losses - Suitable for HIL simulation`.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also

Simscape Blocks
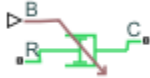Simple Gear | Torque Converter

Topics
"Gear Coupling Control Using Clutches"
"Variable Ratio Gear"
"Prepare Simscape Driveline Models for Real-Time Simulation Using Simscape Checks"

Variable Rotational Damper

Rotational damper with variable damping coefficient

Library

Couplings & Drives/Springs & Dampers

Description

The block represents a rotational viscous damper with a variable damping coefficient. A physical signal input port provides the magnitude of the damping coefficient. The magnitude of the damping torque is equal to the product of the physical signal input and the relative angular velocity between the two rotational conserving ports, R and C. A minimum damping coefficient prevents non-physical negative damping values.

The damping torque satisfies the following expression:

$$T = \begin{cases} B \cdot \omega & B \geq B_{\min} \\ B_{\min} \cdot \omega & B < B_{\min} \end{cases},$$

The parameters are:

- T — Torque transmitted through damper
- B — Viscous damping coefficient
- ω — Relative angular velocity measured between the two rotational conserving ports according to $\omega = \omega_R - \omega_C$
- ω_R — Angular velocity of rotational conserving port R
- ω_C — Angular velocity of rotational conserving port C

The block applies equal and opposite damping torques on the rotational conserving ports. The sign of the damping torque acting on port R is equal to the sign of the relative

angular velocity. A positive relative angular velocity corresponds to a positive damping torque acting on port R, and a negative damping torque of equal magnitude acting on port C.

Assumptions and Limitations

- The block represents only viscous damping
- The value of the damping coefficient must be equal to or greater than zero.

Ports

B

    Physical signal input port representing the variable damping coefficient.

C

    Rotational conserving port.

R

    Rotational conserving port.

Parameters

Minimum damping coefficient

    Minimum value allowed for the damping coefficient. The physical signal input saturates below this level. The parameter must be greater than or equal to zero. The default value is `0 N*m/(rad/s)`.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also

Nonlinear Rotational Damper | Nonlinear Translational Damper | Rotational Damper | Translational Damper | Variable Translational Damper

Variable Rotational Spring

Rotational spring with variable spring stiffness

Library

Couplings & Drives/Springs & Dampers

Description

The block represents a torsional spring with variable spring stiffness. A physical signal input port provides the value of the variable spring stiffness. The magnitude of the spring torque is equal to the product of the physical signal input and the relative angular displacement between the two rotational conserving ports. A minimum spring rate value prevents non-physical values.

The rotational spring torque satisfies the following expression:

$$T = \begin{cases} K \cdot \theta & K \geq K_{\min} \\ K_{\min} \cdot \theta & K < K_{\min} \end{cases},$$

The parameters are:

- T — Torque transmitted through the spring
- K — Spring rate (stiffness)
- K_{min} — Minimum allowable spring rate (stiffness)
- θ — Relative angular displacement measured between the two rotational conserving ports according to $\theta = \theta_{init} + \theta_R - \theta_C$
- θ_{init} — Initial angular deformation between the two rotational conserving ports
- θ_R — Absolute angular displacement of rotational conserving port R

- θ_C — Absolute angular displacement of rotational conserving port C

The block applies equal and opposite spring torques on the two rotational conserving ports. The sign of the spring torque acting on port R is equal to the sign of the relative angular velocity. A positive relative angular velocity corresponds to a positive damping torque acting on port R, and a negative damping torque of equal magnitude acting on port C.

Variables

Use the **Variables** tab to set the priority and initial target values for the block variables before simulating. For more information, see "Set Priority and Initial Target for Block Variables" (Simscape).

Unlike block parameters, variables do not have conditional visibility. The **Variables** tab lists all the existing block variables. If a variable is not used in the set of equations corresponding to the selected block configuration, the values specified for this variable are ignored.

Assumptions and Limitations

The value of the spring rate must be greater than zero.

Ports

K

Physical signal input port representing the variable spring stiffness.

C

Rotational conserving port.

R

Rotational conserving port.

Parameters

Minimum spring rate

> Minimum value allowed for the spring rate (stiffness). The physical signal input saturates below the specified value. The parameter must be greater than zero. The default value is `0.01`. The default unit is `(N*m/rad)`.
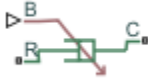
Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also
Nonlinear Rotational Spring | Nonlinear Translational Spring | Variable Translational Spring

Variable Translational Damper

Translational viscous damper with variable damping coefficient

Library

Couplings & Drives/Springs & Dampers

Description

The block represents a translational viscous damper with variable damping coefficient. A physical signal input port provides the magnitude of the translational damping coefficient. The magnitude of the damping force is equal to the product of the physical signal input and the relative linear velocity between the two translational conserving ports. A minimum damping coefficient prevents non-physical negative damping values.

The translational damping force satisfies the following expression:

$$F = \begin{cases} B \cdot v, & B \geq B_{\min} \\ B_{\min} \cdot v, & B < B_{\min} \end{cases},$$

The parameters are:

- F — Force transmitted through the translational damper between the two translational conserving ports
- B — Viscous damping coefficient
- B_{min} — Minimum allowed damping coefficient
- v — Relative linear velocity measured between the two translational conserving ports
- v_R — Linear velocity of port R
- v_C — Linear velocity of port C

The block applies equal and opposite damping forces on the two translational conserving ports. The sign of the damping force acting on port R is equal to the sign of the relative linear velocity. A positive linear velocity corresponds to a positive damping force acting on port R, and a negative damping force of equal magnitude acting on port C.

Assumptions and Limitations

- The block represents strictly viscous damping.
- The value of the damping coefficient must be greater than or equal to zero.

Ports

B

Physical signal input port representing the variable translational damping coefficient.

C

Translational conserving port.

R

Translational conserving port.

Parameters

Minimum damping coefficient

Minimum value allowed for the variable damping coefficient. The physical signal input saturates below the specified value. The parameter must be greater than or equal to zero. The default value is `0 N/(m/s)`.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also

Nonlinear Rotational Damper | Nonlinear Translational Damper | Rotational Damper | Translational Damper | Variable Rotational Damper

Variable Translational Spring

Translational spring with variable spring stiffness

Library

Couplings & Drives/Springs & Dampers

Description

The block represents a translational spring with variable spring stiffness. A physical signal input port provides the variable spring stiffness. The magnitude of the spring force is equal to the product of the physical signal input and the relative linear displacement between the two translational conserving ports. A minimum spring rate prevents non-physical values.

The translational spring force satisfies the following expression:

$$F = \begin{cases} K \cdot x, & K \geq K_{\min} \\ K_{\min} \cdot x & K < K_{\min} \end{cases},$$

The parameters are:

- F — Force transmitted through the spring between the two translational conserving ports
- K — Spring rate (stiffness)
- K_{min} — Minimum allowed spring rate (stiffness)
- x — Relative displacement between the two translational conserving ports according to $x = x_{init} + x_R - x_C$
- x_{init} — initial spring deformation

- x_R — Absolute displacement of translational conserving port R
- x_C — Absolute displacement of translational conserving port C

The block applies equal and opposite spring forces on the two translational conserving ports. The sign of the spring force acting on port R is equal to the sign of the relative linear displacement between the two ports. A positive linear velocity corresponds to a positive spring force acting on port R, and a negative spring force of equal magnitude acting on port C.

Variables

Use the **Variables** tab to set the priority and initial target values for the block variables before simulating. For more information, see "Set Priority and Initial Target for Block Variables" (Simscape).

Unlike block parameters, variables do not have conditional visibility. The **Variables** tab lists all the existing block variables. If a variable is not used in the set of equations corresponding to the selected block configuration, the values specified for this variable are ignored.

Assumptions and Limitations

The value of the variable spring stiffness must be greater than zero.

Ports

K

Physical signal input port representing the variable spring stiffness.

C

Translational conserving port.

R

Translational conserving port.

Parameters

Minimum spring rate

Minimum value allowed for the spring rate. The physical signal input saturates below the specified value. The parameter must be greater than zero. The default value is 1 N/m.

Extended Capabilities

C/C++ Code Generation
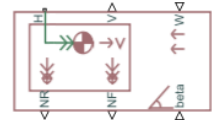
Generate C and C++ code using MATLAB® Coder™.

See Also

Nonlinear Rotational Spring | Nonlinear Translational Spring | Variable Rotational Spring

Vehicle Body

Two-axle vehicle with longitudinal dynamics and motion and adjustable mass, geometry, and drag properties
Library:　　　　Simscape / Driveline / Tires & Vehicles

Description

The Vehicle Body block represents a two-axle vehicle body in longitudinal motion. The vehicle can have the same or a different number of wheels on each axle. For example, two wheels on the front axle and one wheel on the rear axle. The vehicle wheels are assumed identical in size. The vehicle can also have a center of gravity (CG) that is at or below the plane of travel.

The block accounts for body mass, aerodynamic drag, road incline, and weight distribution between axles due to acceleration and road profile. Optionally include pitch and suspension dynamics. The vehicle does not move vertically relative to the ground.

The block has an option to include an externally-defined mass and an externally-defined inertia. The mass, inertia, and center of gravity of the vehicle body can vary over the course of simulation in response to system changes.

Model

The vehicle axles are parallel and form a plane. The longitudinal, x, direction lies in this plane and perpendicular to the axles. If the vehicle is traveling on an incline slope, β, the normal, z, direction is not parallel to gravity but is always perpendicular to the axle-longitudinal plane.

This figure and table define the vehicle motion model variables.
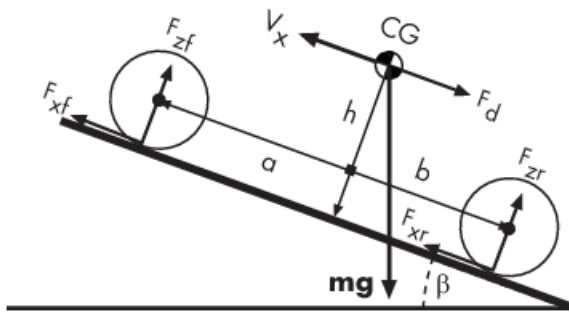
Vehicle Dynamics and Motion

Vehicle Model Variables

Symbol	Description
g	Gravitational acceleration
β	Incline angle
m	Mass of the vehicle
h	Height of vehicle center of gravity (CG) above the ground
a, b	Distance of front and rear axles, respectively, from the normal projection point of vehicle CG onto the common axle plane
V_x	Velocity of the vehicle. When $V_x > 0$, the vehicle moves forward. When $V_x < 0$, the vehicle moves backward.
V_w	Wind speed. When $V_w > 0$, the wind is headwind. When $V_w < 0$, the wind is tailwind.
n	Number of wheels on each axle
F_{xf}, F_{xr}	Longitudinal forces on each wheel at the front and rear ground contact points, respectively
F_{zf}, F_{zr}	Normal load forces on each wheel at the front and rear ground contact points, respectively
A	Effective frontal vehicle cross-sectional area
C_d	Aerodynamic drag coefficient
ρ	Mass density of air
F_d	Aerodynamic drag force

Equations

Vehicle Dynamics

The vehicle motion is a result of the net effect of all the forces and torques acting on it. The longitudinal tire forces push the vehicle forward or backward. The weight mg of the vehicle acts through its center of gravity (CG). Depending on the incline angle, the weight pulls the vehicle to the ground and pulls it either backward or forward. Whether the vehicle travels forward or backward, aerodynamic drag slows it down. For simplicity, the drag is assumed to act through the CG.

$$m\dot{V}_x = F_x - F_d - mg \cdot \sin\beta$$

$$F_x = n(F_{xf} + F_{xr})$$

$$F_d = \frac{1}{2}C_d\rho A(V_x + V_w)^2 \cdot \text{sgn}(V_x + V_w)$$

Zero normal acceleration and zero pitch torque determine the normal force on each front and rear wheel.

$$F_{zf} = \frac{-h(F_d + mg\sin\beta + m\dot{V}_x) + b \cdot mg\cos\beta}{n(a + b)}$$

$$F_{zr} = \frac{+h(F_d + mg\sin\beta + m\dot{V}_x) + a \cdot mg\cos\beta}{n(a + b)}$$

The wheel normal forces satisfy $F_{zf} + F_{zr} = mg\dfrac{\cos\beta}{n}$.

If you include an externally-defined mass or inertia, the equations are shifted by the weighted value of the input.

Pitch dynamics

The rate of the pitch that occurs during vehicle acceleration depends on three torque components and the inertia of the vehicle:

$$\alpha = \frac{(f \cdot h) + (F_{zf}a) - (F_{zr}b)}{J}$$

Where:

- α is the pitch rate.
- f is the longitudinal force.
- J is the inertia.

If you choose a linear model for suspension stiffness and damping, the block uses small angle approximation for pitch calculations. If you choose a table lookup model, the block uses the vectors that you specify calculate pitch dynamics. For hard stop equations, see Translational Hard Stop.

Variables

Use the **Variables** tab to set the priority and initial target values for the block variables before simulating. For more information, see "Set Priority and Initial Target for Block Variables" (Simscape).

Unlike block parameters, variables do not have conditional visibility. The **Variables** tab lists all the existing block variables. If a variable is not used in the set of equations corresponding to the selected block configuration, the values specified for this variable are ignored.

Limitations and Assumptions

The Vehicle Body block lets you model only longitudinal dynamics, parallel to the ground and oriented along the direction of motion. The vehicle is assumed to be in pitch and normal equilibrium. The block does not model pitch or vertical movement. As such, the equations assume that the wheels never lose contact. This constraint can result in negative normal forces.

Ports

Input

W — Headwind speed
scalar

Physical signal input port for headwind speed.

beta — Road incline angle
scalar

Physical signal input port for road incline angle.

CG — Center of gravity, in m
two-element vector

Physical signal input port for the center of gravity, in m, of the externally-defined mass relative to the CG of the vehicle body.

Dependencies

This port is visible only when the **Externally-defined additional mass** parameter in the **Main** settings is set to On.

M — Mass, in kg
scalar

Physical signal input port for the mass, in kg, of the externally-defined mass.

Dependencies

This port is visible only when the **Externally-defined additional mass** parameter in the **Main** settings is set to On.

J — External moment of inertia, in kg*m^2
scalar

Physical signal input port for the moment of inertia, in kg*m^2, of the externally-defined mass.

Dependencies

This port is visible only when the **Externally-defined additional mass** parameter in the **Main** settings and the **Pitch dynamics** parameter in the **Pitch** settings are both set to On.

Output

V — Longitudinal velocity
scalar

Physical signal output port for vehicle longitudinal velocity.

NF — Front axle normal force
scalar

Physical signal output port for normal force on the front axle. Wheel forces are considered positive if acting downwards.

NR — Rear axle normal force
scalar

Physical signal output port for normal force on the rear axle. Wheel forces are considered positive if acting downwards.

Conserving

H — Horizontal motion
mechanical translational

Conserving port associated with the horizontal motion of the vehicle body. Connect tire traction motion to this port.

Parameters

Main

Mass — Vehicle mass
1200 kg (default)

Mass of the vehicle.

Number of wheels per axle — Wheel count
2 (default) | scalar number or a two-element array

Wheel counts on the front and rear axles. If the input is a scalar number, the wheel counts of the front and rear axles are assumed the same. For example, if the input is 2, then the front and rear axles are each assumed to have two wheels.

If the input is a two-element array, the first number is the front-axle wheel count and the second number the rear-axle wheel count. For example, if the input is the array [2,1], then the front axle is assumed to have two wheels and the rear axle one wheel.

1-621

`Horizontal distance from CG to front axle` — CG-to-front axle distance
`1.4 m` (default)

Horizontal distance, *a*, from the center of gravity to the front wheel axle of the vehicle.

`Horizontal distance from CG to rear axle` — CG-to-rear axle distance
`1.6 m` (default)

Horizontal distance, *b*, from the center of gravity to the rear wheel axle of the vehicle.

`CG height above ground` — CG-to-ground distance
`0.5 m` (default)

Distance, *h*, between the center of gravity of the vehicle and the ground.

`Gravitational acceleration` — Gravitational acceleration
`9.81 m/s^2` (default)

Acceleration due to gravitational force acting at the center of gravity of the vehicle.

`Externally-defined additional mass` — Variable mass option
`Off` (default) | `On`

Option to include as time- or event-variant mass that affects the mass and CG of the vehicle body.

Dependencies

The **CG**, **M**, and **J** ports are visible only when this parameter is set to `On`. Exposure of the **J** port also requires that you select `On` for the **Pitch dynamics** parameter in the **Pitch** settings.

`Negative normal force warning` — Warning
`Off` (default) | `On`

To see a warning when the block calculates negative values for normal force during simulation, select `On`. The block can generate negative forces because the equations assume that the wheels never lose contact.

Drag

`Frontal area` — Effective cross-sectional area
`3 m^2` (default)

Effective cross-sectional area, *A*, presented by the vehicle in longitudinal motion. The block uses this value to calculate the aerodynamic drag force on the vehicle.

`Drag coefficient` — Aerodynamic drag coefficient
`0.4` (default)

Aerodynamic drag coefficient, C_d. The block uses this value to calculate the aerodynamic drag force on the vehicle.

`Air density` — Ambient air density
`1.18` kg/m^3 (default)

Density of the air that surrounds the vehicle.

Pitch

The table shows how the visibility of some parameters depends on the option that you choose for other parameters. To learn how to read the table, see "Parameter Dependencies" on page B-2.

Pitch Parameter Dependencies

Pitch	
Pitch dynamics — Choose Off or On	
Off	**On**
	Pitch moment of inertia
	Suspension model — Choose Linear or By table lookup

Linear	**By table lookup**
Front suspension stiffness	Front suspension stiffness force vector
Front suspension damping	Front suspension deformation vector
Rear suspension stiffness	Front suspension damping force vector
Rear suspension damping	Front suspension velocity vector
	Rear suspension stiffness force vector
	Rear suspension deformation vector
	Rear suspension damping force vector
	Rear suspension velocity vector
	Interpolation method
	Extrapolation method

Hard stop — Choose Off or On	
Off	**On**
	Front upper bound
	Front lower bound
	Front contact stiffness
	Front contact damping
	Rear upper bound
	Rear lower bound
	Rear contact stiffness
	Rear contact damping

Initial pitch
Initial pitch rate

Pitch dynamics — Pitch dynamics modelling
Off (default) | On

Option to model pitch dynamics.

Dependencies

Selecting On for this parameter enables parameters for:

- Suspension model
- Hard-stop model
- Initial pitch conditions

The **J** port is visible only when this parameter is set to On. Exposure of the **J** port also requires that you select On for the **Externally-defined additional mass** parameter in the **Main** settings.

For more information, see the Pitch Parameter Dependencies table.

Pitch moment of inertia — Pitch moment of inertia
4000 kg/m^2 (default) | positive scalar

Moment of inertia for pitch calculations.

Dependencies

This parameter is enabled when you set **Pitch dynamics** to On.

For more information, see the Pitch Parameter Dependencies table.

Suspension model — Suspension parameterization method
Linear (default) | Table lookup

Parameterization method for suspension modeling.

You parameterize the suspension using stiffness and damping data. To specify the data using scalar values, select Linear. To specify the data using vector values, select By table lookup.

Dependencies

This parameter is enabled when you set **Pitch dynamics** to On.

For more information, see the Pitch Parameter Dependencies table.

1-625

Front suspension stiffness — Front suspension stiffness
1e4 N/m (default) | positive scalar

Front suspension stiffness per axle.

Dependencies

This parameter is enabled when you set both of these options:

- **Pitch dynamics** to On
- **Suspension model** to Linear

For more information, see the Pitch Parameter Dependencies table.

Front suspension damping — Front suspension damping
1e4N/(m/s) (default) | positive scalar

Front suspension damping per axle.

Dependencies

This parameter is enabled when you set both of these options:

- **Pitch dynamics** to On
- **Suspension model** to Linear

For more information, see the Pitch Parameter Dependencies table.

Rear suspension stiffness — Rear suspension stiffness
1e4 N/m (default) | positive scalar

Rear suspension stiffness per axle.

Dependencies

This parameter is enabled when you set both of these options:

- **Pitch dynamics** to On
- **Suspension model** to Linear

For more information, see the Pitch Parameter Dependencies table.

Rear suspension damping — Rear suspension damping
1e4N/(m/s) (default) | positive scalar

Rear suspension damping per axle.

Dependencies

This parameter is enabled when you set both of these options:

- **Pitch dynamics** to On
- **Suspension model** to Linear

For more information, see the Pitch Parameter Dependencies table.

Front suspension stiffness force vector — Front suspension stiffness force
[-2000, -1000, 0, 1000, 2000] N (default) | increasing vector

Stiffness force of the front suspension. Specify the output values for the lookup table as a vector. The number of elements in the output vector must be the same as the number of elements in the input vector. The input vector parameter is the **Front suspension deformation vector** parameter.

Dependencies

This parameter is enabled when you set both of these options:

- **Pitch dynamics** to On
- **Suspension model** to By table lookup

For more information, see the Pitch Parameter Dependencies table.

Front suspension deformation vector — Front suspension deformation
[-.4, -.2, 0, .2, .4] m (default) | increasing vector

Deformation of the front suspension specified in terms of displacement. Specify the input values for the lookup table as a vector. The values of the elements in the vector must increase from left to right. The minimum number of elements in the vector depends on the interpolation method that you select. For linear interpolation, provide at least two elements. For smooth interpolation, provide at least three elements. The output vector parameter is the **Front suspension stiffness force vector** parameter.

Dependencies

This parameter is enabled when you set both of these options:

- **Pitch dynamics** to On

- **Suspension model** to By table lookup

For more information, see the Pitch Parameter Dependencies table.

Front suspension damping force vector — Front suspension damping force
[-200, -100, 0, 100, 200] N (default) | increasing vector

Damping force of the front suspension. Specify the output values for the lookup table as a vector. The number of elements in the output vector must be the same as the number of elements in the input vector. The input vector parameter is the **Front suspension velocity vector** parameter.

Dependencies

This parameter is enabled when you set both of these options:

- **Pitch dynamics** to On
- **Suspension model** to By table lookup

For more information, see the Pitch Parameter Dependencies table.

Front suspension velocity vector — Front suspension velocity
[-4, -2, 0, 2, 4] m/s (default) | increasing vector

Velocity of the front suspension. Specify the input values for the lookup table as a vector. The values of the elements in the vector must increase from left to right. The minimum number of elements in the vector depends on the interpolation method that you select. For linear interpolation, provide at least two elements. For smooth interpolation, provide at least three elements. The output vector parameter is the **Front suspension damping force vector** parameter.

Dependencies

This parameter is enabled when you set both of these options:

- **Pitch dynamics** to On
- **Suspension model** to By table lookup

For more information, see the Pitch Parameter Dependencies table.

Rear suspension stiffness force vector — Rear suspension stiffness force
[-2000, -1000, 0, 1000, 2000] N (default) | increasing vector

Stiffness force of the rear suspension. Specify the output values for the lookup table as a vector. The number of elements in the output vector must be the same as the number of elements in the input vector. The input vector parameter is the **Rear suspension deformation vector** parameter.

Dependencies

This parameter is enabled when you set both of these options:

- **Pitch dynamics** to On
- **Suspension model** to By table lookup

For more information, see the Pitch Parameter Dependencies table.

Rear suspension deformation vector — Rear suspension deformation
[-.4, -.2, 0, .2, .4] m (default) | increasing vector

Deformation of the rear suspension specified in terms of displacement. Specify the input values for the lookup table as a vector. The values of the elements in the vector must increase from left to right. The minimum number of elements in the vector depends on the interpolation method that you select. For linear interpolation, provide at least two elements. For smooth interpolation, provide at least three elements. The output vector parameter is the **Rear suspension stiffness force vector** parameter.

Dependencies

This parameter is enabled when you set both of these options:

- **Pitch dynamics** to On
- **Suspension model** to By table lookup

For more information, see the Pitch Parameter Dependencies table.

Rear suspension damping force vector — Rear suspension damping force
[-200, -100, 0, 100, 200] N (default) | increasing vector

Damping force of the rear suspension. Specify the output values for the lookup table as a vector. The number of elements in the output vector must be the same as the number of elements in the input vector. The input vector parameter is the **Rear suspension velocity vector** parameter.

Dependencies

This parameter is enabled when you set both of these options:

1-629

- **Pitch dynamics** to `On`
- **Suspension model** to `By table lookup`

For more information, see the Pitch Parameter Dependencies table.

Rear suspension velocity vector — Rear suspension velocity
[-4, -2, 0, 2, 4] m/s (default) | increasing vector

Velocity of the rear suspension. Specify the input values for the lookup table as a vector. The values of the elements in the vector must increase from left to right. The minimum number of elements in the vector depends on the interpolation method that you select. For linear interpolation, provide at least two elements. For smooth interpolation, provide at least three elements. The output vector parameter is the **Rear suspension damping force vector** parameter.

Dependencies

This parameter is enabled when you set both of these options:

- **Pitch dynamics** to `On`
- **Suspension model** to `By table lookup`

For more information, see the Pitch Parameter Dependencies table.

Interpolation method — Interpolation method
linear (default) | smooth

Interpolation methods for approximating the output value when the input value is between two consecutive grid points are:

- `Linear` — Select this default option to get the best performance. Provide at least two values per dimension.
- `Smooth` — Select this option to produce a continuous curve with continuous first-order derivatives. Provide at least three values per dimension.

For more information on interpolation algorithms, see the PS Lookup Table (1D) block reference page.

Dependencies

This parameter is enabled when you set both of these options:

- **Pitch dynamics** to On
- **Suspension model** to By table lookup

For more information, see the Pitch Parameter Dependencies table.

Extrapolation method — Extrapolation method
linear (default) | smooth

Extrapolation method for determining the output value when the input value is outside the range specified in the argument list are:

- Linear — Produces a curve with continuous first-order derivatives in the extrapolation region and at the boundary with the interpolation region.
- Nearest — Produces an extrapolation that does not go above the highest point in the data or below the lowest point in the data.

For more information on extrapolation algorithms, see the PS Lookup Table (1D) block reference page.

Dependencies

This parameter is enabled when you set both of these options:

- **Pitch dynamics** to On
- **Suspension model** to By table lookup

For more information, see the Pitch Parameter Dependencies table.

Hard stop — Hard-stop modeling option
Off (default) | On

Option to model hardstops for the front and rear suspension.

Dependencies

This parameter is enabled when you set the **Pitch dynamics** to On:

Selecting On for this parameter enables parameters for the front and rear hard stops:

- Upper bound
- Lower bound

- Contact stiffness
- Contact damping

For more information, see the Pitch Parameter Dependencies table.

Front upper bound — Hardstop front suspension upper bound
.25 (default)

Upper bound of the hard stop for the front suspension.

Dependencies

This parameter is enabled when you set both of these options:

- **Pitch dynamics** to On
- **Hard stop** to On

For more information, see the Pitch Parameter Dependencies table.

Front lower bound — Vehicle front lower bound
-.25 (default)

Lower bound of the hard stop for the front suspension.

Dependencies

This parameter is enabled when you set both of these options:

- **Pitch dynamics** to On
- **Hard stop** to On

For more information, see the Pitch Parameter Dependencies table.

Front contact stiffness — Vehicle front contact stiffness
1e6 (default)

Contact stiffness of the hard stop for the front suspension.

Dependencies

This parameter is enabled when you set both of these options:

- **Pitch dynamics** to On

- **Hard stop** to On

For more information, see the Pitch Parameter Dependencies table.

Front contact damping — Vehicle front contact damping
150 (default)

Contact damping of the hard stop for the front suspension.

Dependencies

This parameter is enabled when you set both of these options:

- **Pitch dynamics** to On
- **Hard stop** to On

For more information, see the Pitch Parameter Dependencies table.

Rear upper bound — Vehicle rear upper bound
.25 (default)

Upper bound of the hard stop for the rear suspension.

Dependencies

This parameter is enabled when you set both of these options:

- **Pitch dynamics** to On
- **Hard stop** to On

For more information, see the Pitch Parameter Dependencies table.

Rear lower bound — Vehicle rear lower bound
-.25 (default)

Lower bound of the hard stop for the rear suspension.

Dependencies

This parameter is enabled when you set both of these options:

- **Pitch dynamics** to On
- **Hard stop** to On

For more information, see the Pitch Parameter Dependencies table.

`Rear contact stiffness` — Vehicle rear contact stiffness
`1e6` (default)

Contact stiffness of the hard stop for the rear suspension.

Dependencies

This parameter is enabled when you set both of these options:

- **Pitch dynamics** to `On`
- **Hard stop** to `On`

For more information, see the Pitch Parameter Dependencies table.

`Rear contact damping` — Vehicle rear contact damping
`150` (default)

Contact damping of the hard stop for the rear suspension.

Dependencies

This parameter is enabled when you set both of these options:

- **Pitch dynamics** to `On`
- **Hard stop** to `On`

For more information, see the Pitch Parameter Dependencies table.

`Initial pitch` — Pitch at the beginning of the simulation
`0 rad` (default)

Pitch at the beginning of the simulation.

Dependencies

This parameter is enabled when you set **Pitch dynamics** to `On`.

For more information, see the Pitch Parameter Dependencies table.

`Initial pitch rate` — Pitch rate at the beginning of the simulation
`0 rad/s` (default)

Pitch rate at the beginning of the simulation.

Dependencies

This parameter is enabled when you set **Pitch dynamics** to On.

For more information, see the Pitch Parameter Dependencies table.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also
Tire (Friction Parameterized) | Tire (Magic Formula) | Tire (Simple) | Tire-Road Interaction (Magic Formula) | Translational Hard Stop

Topics
"Complete Vehicle Model"

Introduced in R2011b

Worm Gear

Worm gear with adjustable gear ratio and friction losses

Library

Simscape / Driveline / GearsSimscape / Driveline / Gears

Description

The block represents a rotational gear that constrains the two connected driveline axes, worm (W) and gear (G), to rotate together in a fixed ratio that you specify. You can choose whether the gear rotates in a positive or negative direction. Right-hand rotation is the positive direction. If the worm thread is right-hand, ω_W and ω_G have the same sign. If the worm thread is left-hand, ω_W and ω_G have opposite signs.

Thermal Modeling

You can model the effects of heat flow and temperature change through an optional thermal conserving port. By default, the thermal port is hidden. To expose the thermal port, right-click the block in your model and, from the context menu, select **Simscape** > **Block choices**. Select a variant that includes a thermal port. Specify the associated thermal parameters for the component.

Worm Gear Model

Model Variables

R_{WG}	Gear ratio
ω_W	Worm angular velocity
ω_G	Gear angular velocity
α	Normal pressure angle
λ	Worm lead angle
L	Worm lead
d	Worm pitch diameter
τ_G	Gear torque
τ_W	Torque on the worm
τ_{loss}	Torque loss due to meshing friction. The loss depends on the device efficiency and the power flow direction. To avoid abrupt change of the friction torque at $\omega_G = 0$, the friction torque is introduced via the hyperbolic function.
τ_{fr}	Steady-state value of the friction torque at $\omega_G \rightarrow \infty$.
k	Friction coefficient
η_{WG}	Torque transfer efficiency from worm to gear
η_{GW}	Torque transfer efficiency from gear to worm
p_{th}	Power threshold
$[\mu_W \ \mu_G]$	Vector of viscous friction coefficients for the worm and gear

Ideal Gear Constraint and Gear Ratio

Worm gear imposes one kinematic constraint on the two connected axes:

$$\omega_W = R_{WG}\omega_G \, .$$

The two degrees of freedom are reduced to one independent degree of freedom. The forward-transfer gear pair convention is (1,2) = (W,G).

The torque transfer is:

$$R_{WG}\tau_W \quad - \quad \tau_G \quad - \quad \tau_{loss} \quad = \quad 0 \quad ,$$

with $\tau_{loss} = 0$ in the ideal case.

Nonideal Gear Constraint

In the nonideal case, $\tau_{loss} \neq 0$. For general considerations on nonideal gear modeling, see "Model Gears with Losses".

Geometric Surface Contact Friction

In the contact friction case, η_{WG} and η_{GW} are determined by:

- The worm-gear threading geometry, specified by lead angle λ and normal pressure angle α.

- The surface contact friction coefficient k.

$$\eta_{WG} \quad = \quad (\cos\alpha \quad - \quad k{\cdot}\tan\lambda)/(\cos\alpha \quad + \quad k/\tan\lambda) \quad ,$$

$$\eta_{GW} \quad = \quad (\cos\alpha \quad - \quad k/\tan\lambda)/(\cos\alpha \quad + \quad k{\cdot}\tan\lambda) \quad .$$

Constant Efficiencies

In the constant friction case, you specify η_{WG} and η_{GW}, independently of geometric details.

Self-Locking and Negative Efficiency

η_{GW} has two distinct regimes, depending on lead angle λ, separated by the *self-locking point* at which $\eta_{GW} = 0$ and $\cos\alpha = k/\tan\lambda$.

- In the *overhauling regime*, $\eta_{GW} > 0$, and the force acting on the nut can rotate the screw.

- In the *self-locking regime*, $\eta_{GW} < 0$, and an external torque must be applied to the screw to release an otherwise locked mechanism. The more negative is η_{GW}, the larger the torque must be to release the mechanism.

η_{WG} is conventionally positive.

Meshing Efficiency

The efficiencies η of meshing between worm and gear are fully active only if the transmitted power is greater than the power threshold.

If the power is less than the threshold, the actual efficiency is automatically regularized to unity at zero velocity.

Viscous Friction Force

The viscous friction coefficient μ_W controls the viscous friction torque experienced by the worm from lubricated, nonideal gear threads and viscous bearing losses. The viscous friction torque on a worm driveline axis is $-\mu_W \omega_W$. ω_W is the angular velocity of the worm with respect to its mounting.

The viscous friction coefficient μ_G controls the viscous friction torque experienced by the gear, mainly from viscous bearing losses. The viscous friction torque on a gear driveline axis is $-\mu_G \omega_G$. ω_G is the angular velocity of the gear with respect to its mounting.

Limitations

- Gear inertia is assumed negligible.
- Gears are treated as rigid components.
- Coulomb friction slows down simulation. See "Adjust Model Fidelity".

Ports

Port	Description
W	Rotational conserving port representing the worm component
G	Rotational conserving port representing the gear component
H	Thermal conserving port for thermal modeling

Parameters

Main

Gear ratio

Gear or transmission ratio R_{WG} determined as the ratio of the worm angular velocity to the gear angular velocity. The default is 25.

Worm thread type

Choose the directional sense of gear rotation corresponding to positive worm rotation. The default is Right-hand. If you select Left-hand, rotation of the worm in the generally-assigned positive direction results in the gear rotation in negative direction.

Meshing Losses

Parameters for meshing losses vary with the block variant chosen—that with a thermal port for thermal modeling or that without a thermal port.

Without Thermal Port

Friction model

Select how to implement friction losses from nonideal meshing of gear threads. The default is No meshing losses — Suitable for HIL simulation.

- No meshing losses — Suitable for HIL simulation — Gear meshing is ideal.

- Constant efficiency — Transfer of torque between worm and gear is reduced by friction. If you select this option, the panel expands.

Constant Efficiency

Friction parametrization

Select how to characterize the friction between gear threads. The default is Friction coefficient and geometrical parameters.

- Friction coefficient and geometrical parameters — Friction is determined by contact friction between surfaces.

Normal pressure angle

> The thread pressure angle α in the normal plane. Must be greater than zero and less than 90 degrees. The default is `17.5`.

> From the drop-down list, choose units. The default is degrees (`deg`).

Lead angle

> Thread helix angle $\lambda = \arctan[L/(\pi d)]$. L is the worm lead, d is the worm pitch diameter. Must be greater than zero. The default is `20`.

> From the drop-down list, choose units. The default is degrees (`deg`).

Friction coefficient

> Dimensionless coefficient of normal friction in the thread. Must be greater than zero. The default is `0.08`.

- `Efficiencies` — Friction is determined by constant efficiencies $0 < \eta < 1$. If you select this option, the panel changes from its default.

Efficiencies

Worm-gear efficiency

> Efficiency η_{WG} of the power transfer from worm to gear.

Gear-worm efficiency

> Efficiency η_{GW} of the power transfer from gear to worm.

Power threshold

Power threshold above which full efficiency factor is in effect. A hyperbolic tangent function smooths the efficiency factor between zero at rest and the value provided by the temperature-efficiency lookup table at the power threshold. The default value is `0.001` W.

With Thermal Port

Temperature

Array of temperatures used to construct a 1-D temperature-efficiency lookup table. The array values must increase left to right. The temperature array must be the same size as the **Efficiency** array. The default array is `[280 300 320]` K.

Worm-gear efficiency

Array of component efficiencies with the worm as the driver—that is, with power flowing from the worm to the gear. The array values are the efficiencies at the

temperatures in the **Temperature** array. The two arrays must be the same size. The default array is `[0.75 0.7 0.65]`.

Gear-worm efficiency

Array of component efficiencies with the gear as the driver—that is, with power flowing from the gear to the worm. The array values are the efficiencies at the temperatures in the **Temperature** array. The two arrays must be the same size. The default array is `[0.5 0.45 0.4]`.

Power threshold

Power threshold above which the full efficiency factor is in effect. A hyperbolic tangent function smooths the efficiency factor between zero at rest and the value provided by the temperature-efficiency lookup table at the power threshold. The default value is `0.001` W.

Viscous Losses

Viscous friction coefficients at worm (W) and gear (G)

Vector of viscous friction coefficients $[\mu_W \ \mu_G]$, for the worm and gear, respectively. The default is `[0 0]`.

From the drop-down list, choose units. The default is newton-meters/(radians/second) (`N*m/(rad/s)`).

Thermal Port

Thermal mass

Thermal energy required to change the component temperature by a single degree. The greater the thermal mass, the more resistant the component is to temperature change. The default value is `50` J/K.

Initial temperature

Component temperature at the start of simulation. The initial temperature alters the component efficiency according to an efficiency vector that you specify, affecting the starting meshing or friction losses. The default value is `300` K.

Real-Time Simulation

Hardware-in-the-Loop Simulation

For optimal simulation performance, use the **Meshing Losses** > **Friction model** parameter default setting, `No meshing losses - Suitable for HIL simulation`.

Extended Capabilities

C/C++ Code Generation
Generate C and C++ code using MATLAB® Coder™.

See Also

Simscape Blocks
Leadscrew | Sun-Planet | Sun-Planet Worm Gear

Topics
"Basic Motion, Torque, and Force Modeling"
"Model Thermal Losses in Driveline Components"
"Prepare Simscape Driveline Models for Real-Time Simulation Using Simscape Checks"

Technical Conventions

Driveline Abbreviations and Conventions

Abbreviations

An important abbreviation is DoF, which means *degree of freedom* and refers to one coordinate of angular motion. All driveline rotational DoFs are measured with respect to a single absolute coordinate system at rest.

Angular Motion

Standard symbols for angular motion analysis include the following:

Symbol	Meaning (Units)
r	Gear radius
N	Number of gear teeth
g	Gear ratio
θ	Angle
ω	Angular velocity
τ	Torque

Gear Ratios

For a pair of coupled, coplanar gear wheels, the gear ratio g_{21} of gear 2 to gear 1 is defined as the ratio of the second gear wheel radius to the first. This definition is equivalent to the ratio of the number of teeth on the second gear wheel to the number of teeth on the first.

$$g_{21} \equiv r_2/r_1 = N_2/N_1$$

The gear ratio is the ratio of torques and the reciprocal of the angular velocity ratio.

$$g_{21} \qquad = \qquad \tau_2/\tau_1 \qquad = \qquad \omega_1/\omega_2$$

For gear boxes made of more than two gear wheels, the gear ratio is defined to be the ratio of torques or the reciprocal of the ratio of angular velocities, between the output and input shafts.

If the gear is reversing, the output ω and τ have opposite signs from the input ω and τ.

Parameter Dependencies

Parameter Dependencies

A Simscape block parameter is considered *visible* when it appears in the Simulink Property Inspector, in a block dialog box, or as a block choice on the Simscape context menu. A block parameter is considered *configurable* or *enabled* when you can configure it by selecting an option, entering a value, or selecting or clearing a check box. Parameters that are visible but are not enabled are typically dimmed.

The visibility and configurability of some parameters depend on the options that you select for other parameters. Parameter dependencies are typically listed in the parameter description in the documentation for the block. For some blocks, the documentation also includes a parameter dependency table.

Parameter Dependency Tables

Parameter dependency tables show how the visibility and configurability of some parameters depend on the options that you select for other parameters.

Parameter dependency tables indicate which parameter options affect the *visibility* of other parameters. For example, the next table shows the dependencies for the **Effects and Initial Conditions** parameters for the Pipe (2P) block from the **Simscape > Foundation Library > Two-Phase Fluid > Elements** library.

The row numbers in this table are for reference only. The first column of a parameter dependency table typically includes all parameters that are visible by default.

Parameters that affect the visibility or configurability of other parameters are shown in bold text. The options for the parameters that affect other parameters are shown in italic text in the following row of the table. Parameters that do not affect other parameters are shown in plain text.

Effects and Initial Conditions Parameter Dependencies

Row	Parameters and Options					
1	**Fluid Inertia**					
2	*Off*			*On*		
3	**Initial fluid regime**			**Initial fluid regime**		
4	*Subcooled liquid*	*Two-phase mixture*	*Superheated vapor*	*Subcooled liquid*	*Two-phase mixture*	*Superheated vapor*
5	Initial fluid pressure	Initial fluid pressure	Initial fluid pressure	Initial fluid pressure	Initial fluid pressure	Initial fluid pressure
6	Initial fluid temperature	Initial fluid vapor quality	Initial fluid temperature	Initial fluid temperature	Initial fluid vapor quality	Initial fluid temperature
7				Initial mass flow rate from port A to port B	Initial mass flow rate from port A to port B	Initial mass flow rate from port A to port B
8	Phase change time constant	Phase change time constant	Phase change time constant	Phase change time constant	Phase change time constant	Phase change time constant

The figure shows the **Effects and Initial Conditions** parameters that are visible by default on the block dialog box for the Pipe (2P) block.

If you change the **Fluid Inertia** parameter from `Off` to `On`, the **Initial mass flow rate from port A to port B** parameter becomes visible. The table shows this dependency in rows 1, 2, and 7.

If you then change the **Initial fluid regime** parameter from `Subcooled liquid` to `Two-phase mixture`, the **Initial fluid temperature** parameter is hidden and **Initial fluid vapor quality** becomes visible. The table shows these dependencies in rows 3, 4, and 6.
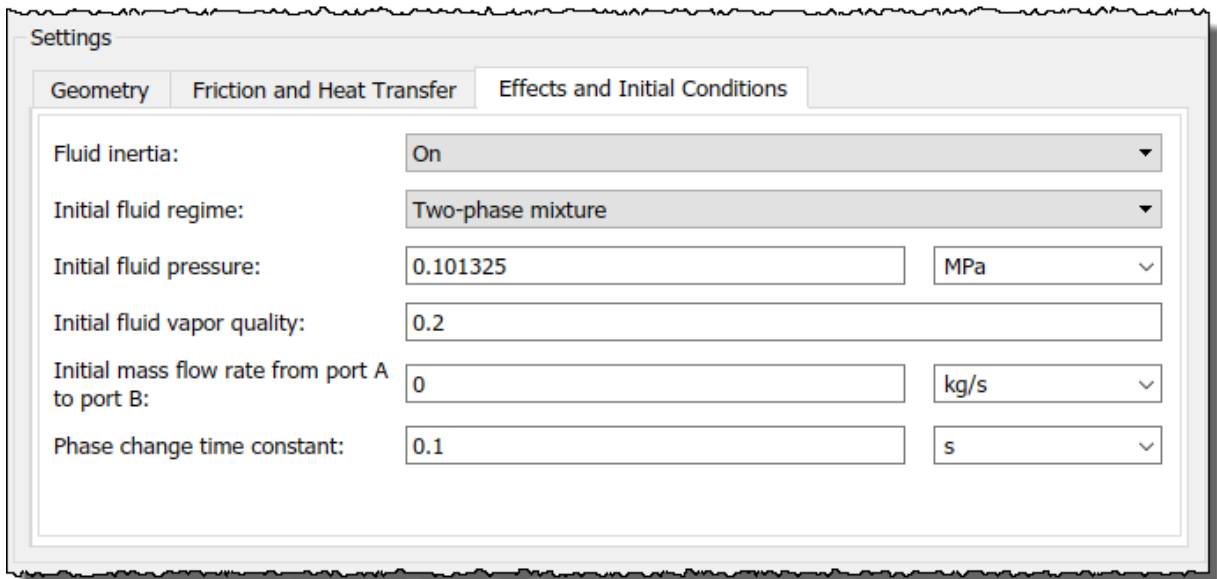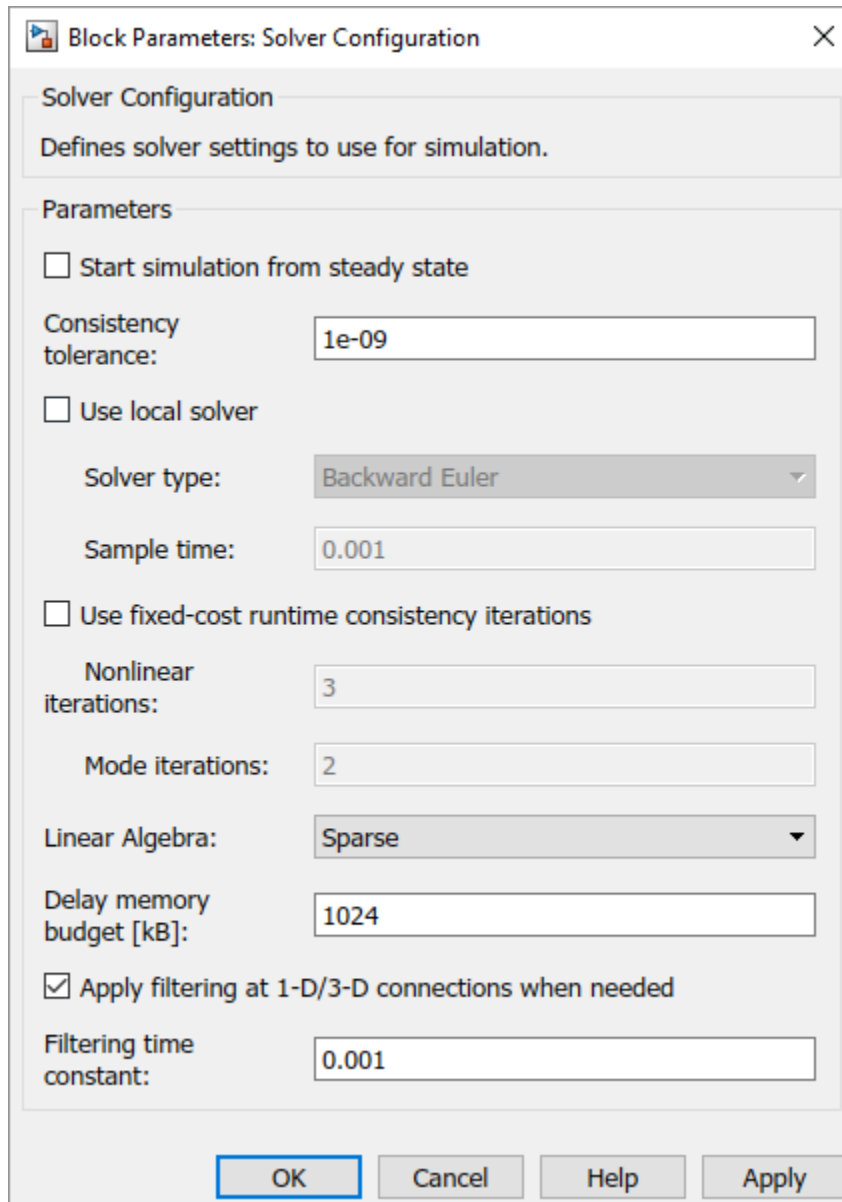
Parameter dependency tables also indicate which parameter options *enable* other parameters. For example, the next table shows the dependencies for the parameters for the Solver Configuration block from the **Simscape** > **Foundation Library** > **Utilities** library. The first column in the table contains row numbers, which are for reference only. The second column includes all parameters that are visible by default.

Parameters that affect the visibility and configurability of other parameters are shown in bold text. The options for the parameters that affect other parameters are shown as selected and cleared check boxes in the following row of the table. Parameters that are shown in plain text are enabled but do not affect other parameters. Parameters that are not enabled are not shown.

Solver Configuration Block Parameter Dependencies

Row	Parameters and Options			
1	Start simulation from steady state			
2	Consistency tolerance			
3	**Use local Solver**			
4	☐		☑	
5			Solver type	
6			Sample time	
7	**Use fixed-cost runtime consistency iterations**		**Use fixed-cost runtime consistency iterations**	
8	☐	☑	☑	☐
9		Nonlinear iterations	Nonlinear iterations	
10		Mode iterations		
11	Linear Algebra			
12	Equation formulation			
13	Delay memory budget [kB]			
14	**Apply filtering at 1-D/3-D connections when needed**			
15	☑		☐	
16	Filtering time constant			

The figure shows the parameters that are visible by default on the block dialog box for the Solver Configuration block. The dimmed parameters, for example, **Solver type**, are not enabled.

Block Parameters: Solver Configuration ✕

Solver Configuration

Defines solver settings to use for simulation.

Parameters

☐ Start simulation from steady state

Consistency
tolerance: `1e-09`

☐ Use local solver

    Solver type: Backward Euler ▾

    Sample time: 0.001

☐ Use fixed-cost runtime consistency iterations

    Nonlinear
iterations: 3

    Mode iterations: 2

Linear Algebra: Sparse ▾

Delay memory
budget [kB]:   `1024`

☑ Apply filtering at 1-D/3-D connections when needed

Filtering time
constant: `0.001`

| OK | Cancel | Help | Apply |

If you select the **Use local solver** check box, the **Use fixed-cost runtime consistency** check box becomes selected and these parameters become enabled.

- **Solver type**
- **Nonlinear iterations**
- **Sample time**

Selecting **Use local solver** does not enable the **Mode iterations** parameter. The **Mode iterations** parameter is only enabled when the **Use local solver** check box is cleared while **Use fixed-cost runtime consistency** check box is selected.

The table shows these dependencies in rows 3–10.

Bibliography

[1] Genta, G., *Motor Vehicle Dynamics: Modeling and Simulation,* Singapore, World Scientific, 1997, chapters 2, 3, and 4

[2] Jurgen, R. K., *Electronic Transmission Controls,* Troy, Michigan, Society of Automotive Engineers, 2000

[3] Juvinall, R. C., *Fundamentals of Machine Component Design,* New York, John Wiley & Sons, 1983

[4] The MathWorks Inc., https://www.mathworks.com/industries/auto/, Industries: Automotive

[5] Meriam, J. L., and L. G. Kraige, *Dynamics*, Volume 2 of *Engineering Mechanics,* New York, John Wiley & Sons, 1987

[6] Nwagboso, C. O., *Automotive Sensory Systems,* London, Chapman & Hall, 1993

[7] Pacejka, H. B. *Tire and Vehicle Dynamics,* Society of Automotive Engineers and Butterworth-Heinemann, Oxford, 2002, chapters 1,4,7, and 8

[8] Shingley, J. E., C. R. Mishke, and R. G. Budynas, *Mechanical Engineering Design,* McGraw-Hill, 2004

[9] Society of Automotive Engineers, https://www.sae.org, see "SAE Store"

[10] Wong, J. Y., *Theory of Ground Vehicles,* 3rd Ed., New York, Interscience, 2001