Robust Control Toolbox™ Release Notes

How to Contact MathWorks

www.mathworks.com	Web
comp.soft-sys.matlab	Newsgroup
www.mathworks.com/contact_TS.html	Technical Support

suggest@mathworks.com	Product enhancement suggestions
bugs@mathworks.com	Bug reports
doc@mathworks.com	Documentation error reports
service@mathworks.com	Order status, license renewals, passcodes
info@mathworks.com	Sales, pricing, and general information

508-647-7000 (Phone)

508-647-7001 (Fax)

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

For contact information about worldwide offices, see the MathWorks Web site.

Robust Control Toolbox™ Release Notes

Trademarks

Patents

Contents

Summary by Version

This table provides quick access to what's new in each version. For clarification, see "Using Release Notes" on page 2.

Version (Release)	New Features and Changes	Version Compatibility Considerations	Fixed Bugs and Known Problems
Latest Version V4.0 (R2011b)	Yes Details	Yes Summary	Bug Reports Includes fixes
V3.6 (R2011a)	Yes Details	No	Bug Reports Includes fixes
V3.5 (R2010b)	Yes Details	No	NoBug Reports Includes fixes
V3.4.1 (R2010a)	No	No	Bug Reports Includes fixes
V3.4 (R2009b)	Yes Details	Yes Summary	Bug Reports Includes fixes
V3.3.3 (R2009a)	No	No	Bug Reports Includes fixes
V3.3.2(R2008b)	No	No	Bug Reports Includes fixes
V3.3.1 (R2008a)	Yes Details	No	No bug fixes
V3.3 (R2007b)	No	No	No bug fixes
V3.2 (R2007a)	Yes Details	No	No bug fixes
V3.1.1 (R2006b)	Yes Details	No	No bug fixes
V3.1 (R2006a)	No	No	No bug fixes

Version (Release)	New Features and Changes	Version Compatibility Considerations	Fixed Bugs and Known Problems
V3.0.2 (R14SP3)	No	No	No bug fixes
V3.0.1 (R14SP2)	Yes Details	No	No bug fixes

Using Release Notes

Use release notes when upgrading to a newer version to learn about:

- New features

- Changes

- Potential impact on your existing files and practices

Review the release notes for other MathWorks® products required for this product (for example, MATLAB® or Simulink®). Determine if enhancements, bugs, or compatibility considerations in other products impact you.

If you are upgrading from a software version other than the most recent one, review the current release notes and all interim versions. For example, when you upgrade from V1.0 to V1.2, review the release notes for V1.1 and V1.2.

What Is in the Release Notes

New Features and Changes

- New functionality

- Changes to existing functionality

Version Compatibility Considerations

When a new feature or change introduces a reported incompatibility between versions, the **Compatibility Considerations** subsection explains the impact.

Compatibility issues reported after the product release appear under Bug Reports at the MathWorks Web site. Bug fixes can sometimes result in incompatibilities, so review the fixed bugs in Bug Reports for any compatibility impact.

Fixed Bugs and Known Problems

MathWorks offers a user-searchable Bug Reports database so you can view Bug Reports. The development team updates this database at release time and as more information becomes available. Bug Reports include provisions for any known workarounds or file replacements. Information is available for bugs existing in or fixed in Release 14SP2 or later. Information is not available for all bugs in earlier releases.

Access Bug Reports using your MathWorks Account.

Documentation on the MathWorks Web Site

Related documentation is available on `mathworks.com` for the latest release and for previous releases:

- Latest product documentation
- Archived documentation

3

Version 4.0 (R2011b) Robust Control Toolbox Software

This table summarizes what's new in Version 4.0 (R2011b):

New Features and Changes	Version Compatibility Considerations	Fixed Bugs and Known Problems
Yes Details below	Yes Summary	Bug Reports Includes fixes

New features introduced in this version:

- "looptune Tunes Fixed-Structure Control Systems" on page 4

- "Control System Tuning for Simulink Models with looptune or hinfstruct Using slTunable Interface" on page 5

- "wcgainplot for Visualizing Worst-Case Gains" on page 5

- "Functionality Being Removed or Changed" on page 6

looptune Tunes Fixed-Structure Control Systems

Use `looptune` to tune fixed-structure control systems to meet your requirements. To use `looptune`, specify design requirements such as loop bandwidth, stability margin, setpoint tracking, or target loop shape. `looptune` automatically tunes the parameters of your controller to meet the specified requirements.

The requirements objects `TuningGoal.MaxGain`, `TuningGoal.Tracking`, and `TuningGoal.LoopShape` let you express your design requirements directly. You do not have to first convert them to weighting functions or mathematical constraints on an optimization problem.

You can use `loopview` to validate the performance the performance of the tuned control structure against your specified design requirements.

For more information, see "Tuning Fixed Control Architectures" and the `looptune` and `loopview` reference pages.

Control System Tuning for Simulink Models with looptune or hinfstruct Using slTunable Interface

If you have Simulink® Control Design™ software, you can use tuning commands, such as `slTunable.looptune` and `hinfstruct`, to tune control systems modeled in Simulink. The `slTunable` object provides an interface between your Simulink model and these commands.

Use `slTunable` to specify information about your control structure and parametrization. `slTunable` also automates tasks such as linearizing the Simulink model, parametrizing the tunable blocks of your system, and applying tuned parameter values to the model. After you create and configure an `slTunable` object for your control architecture, you can tune the control system using `slTunable.looptune` or `hinfstruct`.

For more information, see "Tuning Fixed Control Architectures" and the following demos:

- Tuning of a Digital Motion Control System
- Decoupling Controller for a Distillation Column
- Tuning of a Two-Loop Autopilot
- Tuning of Cascaded PID Loops
- Loop Shaping Design with HINFSTRUCT
- Fixed-Structure Autopilot for a Passenger Jet

wcgainplot for Visualizing Worst-Case Gains

`wcgainplot` plots the nominal, sampled, and worst-case gains of uncertain systems as a function of frequency. Use `wcgainplot` for visual analysis of uncertain systems.

For more information, see the `wcgainplot` reference page.

Functionality Being Removed or Changed

Functionality	What Happens When You Use This Functionality?	Use This Instead	Compatibility Considerations
umat object can no longer contain ultidyn or udyn uncertainty.	• Presence of ultidyn or udyn uncertain elements forces model type to uss or ufrd rather than umat. • Mixing ureal or ucomplex models with udyn or ultidyn objects produces uss instead of umat.	Expect a model type of uss or ufrd instead of umat when working with udyn or ultidyn uncertain elements.	Update code to work with uss or ufrd instead of umat when udyn or ultidyn elements are present.
uss(sys_frd), where sys_frd is a frd model object no longer converts sys_frd to ufrd.	Errors.	ufrd(sys_frd).	Replace uss(sys_frd) with ufrd(sys_frd).
ufrd(udat,freq,...) no longer constructs an uncertain frd model from the umat object udat.	Converts udat to a ufrd object with frequencies freq.	Use frd(udat,freq,...) to construct an uncertain frd model from the umat object udat.	Replace ufrd(udat,freq,...) with frd(udat,freq,...).
frd(sys_uss,w) where sys_uss is a uss model.	Warns; returns frd model containing data based on nominal response of sys_uss.	ufrd(sys_uss,w) to obtain a ufrd model.	Replace frd(sys_uss,w) with ufrd(sys_uss,w).
Nominal value of ultidyn object.	Nominal value is ss model object.	None.	Update code to work with ss model objects when working nominal value of ultidyn.

Functionality	What Happens When You Use This Functionality?	Use This Instead	Compatibility Considerations
usubs.	Applied to array of uncertain models, default substitution is '-once'.	Use '-batch' to perform batch substitution on uncertain model arrays.	Replace usubs(...) with usubs(...,'-batch').
	usubs(M,{a1;a2;...}, {v1;v2;...}) returns error.	usubs(M,a1,v1,a2, v2,...).	Replace usubs(M,{a1;a2;...}, {v1;v2;...}) with usubs(M,a1,v1,a2, v2,...).
usample(sys,'a',na, 'b',nb) where uncertain element b does not exist in sys.	Returns na-by-nb array with constant values across nb dimension, instead of na-by-1 array.	None.	Update code to reflect correct dimensionality.
wcgopt.	Still runs.	wcgainOptions or wcmarginOptions.	Replace wcgopt with wcgainOptions or wcmarginOptions.
robuststab and robustperf.	For ufrd models, BadUncertainValues field of Info output returns Nf-by-1 struct array, where Nf is the number of frequency points.	None.	Update code to work with Nf-by-1 struct array for BadUncertainValues instead of Nf-by-1 cell array.
	For nominally unstable models, performance margin is zero (instead of a negative value).	None.	Update code to reflect correct performance margin .

Functionality	What Happens When You Use This Functionality?	Use This Instead	Compatibility Considerations
robopt.	Still runs.	robuststabOptions or robustperfOptions.	Replace robopt with robuststabOptions or robustperfOptions.
actual2normalized.	First output argument is normalized uncertain block value. The second output argument is normalized distance between block value and nominal value.	[NV,ndist] = actual2normalized(BLK,AV).	Use second output argument ndist for normalized distance.
reshape(unc_sys,S).	S does not include the I/O size of the models in the array unc_sys. For example, if unc_sys is a 6-by-1 array of 2-output, 4-input models, reshape(unc_sys,[2 3]) converts unc_sys to a 2-by-3 array.	None.	Remove I/O size dimensions from reshape on uncertain model arrays.
diag(uss_sys) where uss_sys is a uss model.	Errors.	None.	Remove diag(uss_sys).

Version 3.6 (R2011a) Robust Control Toolbox Software

This table summarizes what's new in Version 3.6 (R2011a):

New Features and Changes	Version Compatibility Considerations	Fixed Bugs and Known Problems
Yes Details below	No	Bug Reports Includes fixes

New features introduced in this version:

Enhanced Workflow for H-Infinity Synthesis of Fixed-Structure Control Systems

New Generalized LTI models in Control System Toolbox™ allow you to model control systems with tunable parameters. Using these models simplifies controller tuning with `hinfstruct`. You can model a closed-loop transfer function, including tunable parameters, as a generalized state-space (`genss`) model and directly tune the parameters to minimize the closed-loop gain. The `hinfstruct` command can tune any fixed-structure SISO or MIMO control system using H_∞ synthesis techniques.

Additionally, new `realp` and `genmat` objects let you create parametric expressions. You can use such expressions to create custom tunable components. For example, you can define a low-pass filter parametrized by its cutoff frequency, or an observer-based controller parametrized by the state-feedback and observer gains.

For more information about creating tunable Generalized LTI models, see "Models with Tunable Coefficients" in the *Control System Toolbox User's Guide*.

For more information about H_∞ tuning with `hinfstruct`, see "Tuning Fixed Control Architectures" in the *Robust Control Toolbox™ Getting Started Guide*.

For examples of designing controllers for several different architectures using `hinfstruct`, see the following updated and new demos:

- Loop Shaping Design with HINFSTRUCT (updated)
- Tuning of a Two-Loop Autopilot (updated)
- Decoupling Controller for a Distillation Column (updated)
- Multi-Loop PID Control of a Robot Arm (updated)
- Fixed-Structure Autopilot for a Passenger Jet (new)

Version 3.5 (R2010b) Robust Control Toolbox Software

This table summarizes what's new in Version 3.5 (R2010b):

New Features and Changes	Version Compatibility Considerations	Fixed Bugs and Known Problems
Yes Details below	No	Bug Reports Includes fixes

New features introduced in this version:

New Commands for H-Infinity Synthesis of Fixed-Structure Control Systems

New commands in this release allow you to tune fixed-structure SISO and MIMO control systems using the techniques of H_∞ synthesis.

The new `hinfstruct` command lets you use the frequency-domain methods of H_∞ synthesis to tune control systems with a broad range of architectures and controller structures. For example, you can tune:

- Fixed-order, fixed-structure controllers, such as pure gains, PID controllers, or fixed-order transfer function or state-space models

- Single feedback-loop architectures with multiple tunable elements, such as a PID controller plus a filter

- Multiple feedback-loop architectures with multiple tunable elements

Specify the tunable elements of your system using the new parametrized Control Design blocks `ltiblock.gain`, `ltiblock.pid`, `ltiblock.tf`, and `ltiblock.ss`.

For examples of designing controllers for several different architectures using `hinfstruct`, see the following new demos:

- Loop Shaping Design with HINFSTRUCT

- Tuning of a Fixed-Structure Autopilot

- Decoupling Controller for a Distillation Column

- Multi-Loop PID Control of a Robot Arm

For more information, see "Tuning Fixed Control Architectures" in the *Robust Control Toolbox Getting Started Guide*.

Version 3.4.1 (R2010a) Robust Control Toolbox Software

This table summarizes what's new in Version 3.4.1 (R2010a):

New Features and Changes	Version Compatibility Considerations	Fixed Bugs and Known Problems
No	No	Bug Reports Includes fixes

Version 3.4 (R2009b) Robust Control Toolbox Software

This table summarizes what's new in Version 3.4 (R2009b):

New Features and Changes	Version Compatibility Considerations	Fixed Bugs and Known Problems
Yes Details below	Yes Summary	Bug Reports Includes fixes

New features and changes introduced in this version are described here:

- "New Option to Improve Robust Performance by Accounting for Real Uncertain Parameters" on page 14
- "New Command to Linearize Simulink Models with Uncertainty" on page 14
- "New Interface for Simulating Effects of Uncertainty in Simulink Models" on page 15
- "New Command to Model Multiple LTI Responses as One Uncertain System" on page 15
- "New and Updated Demos" on page 15
- "Functions, Properties and Blocks Being Removed" on page 16

New Option to Improve Robust Performance by Accounting for Real Uncertain Parameters

You can now improve robust performance by accounting for real uncertain parameters when designing controllers using μ-synthesis. The user-defined options you use in the dksyn command now includes a new option MixedMU. Set this option to 'on' to account for real uncertain parameters in your system. For more information, see the dkitopt, and dksyn reference pages.

New Command to Linearize Simulink Models with Uncertainty

If you have Simulink Control Design software installed, you can take model uncertainty into account when linearizing a Simulink model. You can then

use the resulting uncertain linearized model (`uss` object) to perform linear analysis and robust control design.

If your model already contains Uncertain State Space blocks, use the new `ulinearize` command to obtain an `uss` model. If you want to account for uncertainty in your linear analysis without using Uncertain State Space blocks, you can specify individual Simulink blocks to linearize to an uncertain variable. For more information, see "Computing Uncertain State-Space Models from Simulink Models" in the *Robust Control Toolbox User's Guide*.

New Interface for Simulating Effects of Uncertainty in Simulink Models

This version of the product provides a new interface to simulate the effects of uncertainty in Simulink models. The interface includes the following:

- Uncertain State Space block to specify uncertain system in Simulink. You should replace USS System blocks in your existing models with the Uncertain State Space block. To do so, run the `slupdate` command on your models.
- `ufind` command to extract all uncertain variables from a Simulink model.
- `usample` command to generate random values of these uncertain variables.

For more information on simulating the effects of uncertainty using the new interface, see "Simulating Effects of Uncertainty" in the *Robust Control Toolbox User's Guide*.

New Command to Model Multiple LTI Responses as One Uncertain System

This version of the product includes a new `ucover` command that lets you model a family of LTI responses as one uncertain system. For more information, see the `ucover` reference page.

New and Updated Demos

The following new and updated demos illustrate use of the new features:

- "Control of Spring-Mass-Damper Using Mixed mu-Synthesis" shows use of the new `MixedMU` option and `dksyn` command for mixed-mu synthesis.

- "Linearization of Simulink Models with Uncertainty" shows how to compute uncertain state-space models using `ulinearize` and Simulink Control Design software.

- "Robustness Analysis in Simulink" uses the new interface for simulating effects of uncertainty in Simulink models.

- "Simultaneous Stabilization Using Robust Control" and "Modeling a Family of Responses as an Uncertain System" show use of the `ucover` command.

- "First-Cut Robust Design" shows use of the `usample`, `ucover` and `dksyn` commands.

To access the demos, type

```
demo('toolbox','robust control')
```

Functions, Properties and Blocks Being Removed

Function, Property or Block Name	What Happens When You Use Function or Property?	Use This Instead	Compatibility Considerations
usiminfo	Still runs	ufind	See "New Interface for Simulating Effects of Uncertainty in Simulink Models" on page 15.
usimfill	Still runs	ufind	See "New Interface for Simulating Effects of Uncertainty in Simulink Models" on page 15.
usimsamp	Still runs	usample	See "New Interface for Simulating Effects of Uncertainty in Simulink Models" on page 15.

Function, Property or Block Name	What Happens When You Use Function or Property?	Use This Instead	Compatibility Considerations
USS System block	Still runs	Uncertain State Space block	See "New Interface for Simulating Effects of Uncertainty in Simulink Models" on page 15.
ltiarray2uss	Still runs	ucover	See "New Command to Model Multiple LTI Responses as One Uncertain System" on page 15.

Version 3.3.3 (R2009a) Robust Control Toolbox Software

This table summarizes what's new in Version 3.3.3 (R2009a):

New Features and Changes	Version Compatibility Considerations	Fixed Bugs and Known Problems
No	No	Bug Reports Includes fixes

Version 3.3.2 (R2008b) Robust Control Toolbox Software

This table summarizes what's new in Version 3.3.2 (R2008b):

New Features and Changes	Version Compatibility Considerations	Fixed Bugs and Known Problems
No	No	Bug Reports Includes fixes

Version 3.3.1 (R2008a) Robust Control Toolbox Software

This table summarizes what's new in Version 3.3.1 (R2008a):

New Features and Changes	Version Compatibility Considerations	Fixed Bugs and Known Problems
Yes Details below	No	No bug fixes

New features and changes introduced in this version are described here.

Ability to Use LOOPMARGIN with Simulink

This version of Robust Control Toolbox software lets you analyze the robustness of nonlinear Simulink models using the LOOPMARGIN command.

If you have the Simulink Control Design product installed, you can perform stability margin analysis of a Simulink model by passing the model name and a point within that model to the LOOPMARGIN command.

Version 3.3 (R2007b) Robust Control Toolbox Software

This table summarizes what's new in Version 3.3 (R2007b):

New Features and Changes	Version Compatibility Considerations	Fixed Bugs and Known Problems
No	No	No bug fixes

Version 3.2 (R2007a) Robust Control Toolbox Software

This table summarizes what's new in Version 3.2 (R2007a):

New Features and Changes	Version Compatibility Considerations	Fixed Bugs and Known Problems
Yes Details below	No	No bug fixes

New features and changes introduced in this version are described here.

New Simulink Blocks

- USS System — This Robust Control Toolbox version introduces a new Simulink block, USS System. You can use this block to import uncertain systems into Simulink models.

- Multiplot Graph — Plot multiple signals in one figure.

Version 3.1.1 (R2006b) Robust Control Toolbox Software

This table summarizes what's new in Version 3.1.1 (R2006b):

New Features and Changes	Version Compatibility Considerations	Fixed Bugs and Known Problems
Yes Details below	No	No bug fixes

New Function ltiarray2uss

This Robust Control Toolbox version introduces a new function, `ltiarray2uss`. This function constructs an uncertain state-space model from an LTI array.

Version 3.1 (R2006a) Robust Control Toolbox Software

This table summarizes what's new in Version 3.1 (R2006a):

New Features and Changes	Version Compatibility Considerations	Fixed Bugs and Known Problems
No	No	No bug fixes

Version 3.0.2 (R14SP3) Robust Control Toolbox Software

This table summarizes what's new in Version 3.0.2 (R14SP3):

New Features and Changes	Version Compatibility Considerations	Fixed Bugs and Known Problems
No	No	No bug fixes

Version 3.0.1 (R14SP2) Robust Control Toolbox Software

This table summarizes what's new in Version 3.0.1 (R14SP2):

New Features and Changes	Version Compatibility Considerations	Fixed Bugs and Known Problems
Yes Details below	No	No bug fixes

New features and changes introduced in this version are described here:

mussvunwrap Is Renamed

`mussvunwrap` has been renamed. It is now called `mussvextract`.

New Functions actual2normalized and normalized2actual

This Robust Control Toolbox version introduced two new functions:

- `actual2normalized` — Calculate normalized distance between nominal value and given value for uncertain atom.

- `normalized2actual` — Convert value for atom in normalized coordinates to corresponding actual value.

Compatibility Summary for Robust Control Toolbox Software

This table summarizes new features and changes that might cause incompatibilities when you upgrade from an earlier version, or when you use files on multiple versions. Details are provided with the description of the new feature or change.

Version (Release)	New Features and Changes with Version Compatibility Impact
Latest Version **V4.0 (R2011b)**	See the **Compatibility Considerations** subheading for new features and changes: • "Functionality Being Removed or Changed" on page 6
V3.6 (R2011a)	None
V3.5 (R2010b)	None
V3.4.1 (R2010a)	None
V3.4 (R2009b)	See "Functions, Properties and Blocks Being Removed" on page 16.
V3.3.3 (R2009a)	None
V3.3.2 (R2008b)	None
V3.3.1 (R2008a)	None
V3.3 (R2007b)	None
V3.2 (R2007a)	None
V3.1.1 (R2006b)	None

Version (Release)	New Features and Changes with Version Compatibility Impact
V3.1 (R2006a)	None
V3.0.2 (R14SP3)	None
V3.0.1 (R14SP2)	None