

5G Toolbox™ Release Notes



MATLAB®

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

5G Toolbox™ Release Notes

© COPYRIGHT 2018–2019 by The MathWorks, Inc.

The software described in this document is furnished under a license agreement. The software may be used or copied only under the terms of the license agreement. No part of this manual may be photocopied or reproduced in any form without prior written consent from The MathWorks, Inc.

FEDERAL ACQUISITION: This provision applies to all acquisitions of the Program and Documentation by, for, or through the federal government of the United States. By accepting delivery of the Program or Documentation, the government hereby agrees that this software or documentation qualifies as commercial computer software or commercial computer software documentation as such terms are used or defined in FAR 12.212, DFARS Part 227.72, and DFARS 252.227-7014. Accordingly, the terms and conditions of this Agreement and only those rights specified in this Agreement, shall pertain to and govern the use, modification, reproduction, release, performance, display, and disclosure of the Program and Documentation by the federal government (or other entity acquiring for or through the federal government) and shall supersede any conflicting contractual terms or conditions. If this License fails to meet the government's needs or is inconsistent in any respect with federal procurement law, the government agrees to return the Program and Documentation, unused, to The MathWorks, Inc.

Trademarks

MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See www.mathworks.com/trademarks for a list of additional trademarks. Other product or brand names may be trademarks or registered trademarks of their respective holders.

Patents

MathWorks products are protected by one or more U.S. patents. Please see www.mathworks.com/patents for more information.

R2019b

Receiver Design and Synchronization: Perform practical timing and channel estimation	1-2
LDPC decoder enhancements and support for layered belief propagation and min-sum approximation	1-2
Support for CSI-RS	1-2
5G NR-TM waveform generation	1-3
NR downlink ACLR measurement	1-3
System-Level Simulation: Schedule 5G NR PUSCH resources	1-3
Enhanced 5G NR waveform generation, including CSI-RS, SRS, and PT-RS	1-3

R2019a

Support for uplink physical channels PUSCH and PUCCH ...	2-2
Support for uplink shared channel (UL-SCH) and uplink control information (UCI)	2-2
UL-SCH	2-2
UCI	2-3
Parity-check support for polar coding	2-3

Addition of new downlink shared channel (DL-SCH) System objects	2-3
5G NR uplink waveform generation	2-3
5G NR PUSCH link-level reference simulation	2-4

R2018b

Introducing 5G Toolbox	3-2
Standard-compliant models for 3GPP 5G NR Release 15	3-2
Link-level simulation with reference examples, including 5G NR PDSCH throughput simulation	3-2
OFDM waveform generation with NR subcarrier spacings and frame numerologies	3-2
TR 38.901 propagation channel models, including tapped delay line (TDL) and clustered delay line (CDL)	3-3
Downlink transport and physical channels (shared, control, broadcast); synchronization and demodulation reference signals	3-3
Signal processing functions, including channel coding (LDPC and polar codes), channel estimation, synchronization, and equalization	3-3
C and C++ code generation support	3-3

R2019b

Version: 1.2

New Features

Bug Fixes

Receiver Design and Synchronization: Perform practical timing and channel estimation

When processing received signals, use these functions to perform practical timing or channel estimation.

Functions	Description
<code>nrTimingEstimate</code>	Practical timing estimation
<code>nrChannelEstimate</code>	Practical channel estimation

The “NR PDSCH Throughput” and “NR PUSCH Throughput” examples now include practical timing and channel estimation in addition to perfect timing and channel estimation.

LDPC decoder enhancements and support for layered belief propagation and min-sum approximation

Low-density parity-check (LDPC) decoding now uses multicore processing. Additional LDPC decoding algorithms, such as layered belief propagation, normalized min-sum approximation, and offset min-sum approximation, are also available. These enhancements apply to these decoders: `nrLDPCDecode` function, `nrDLSCHDecoder` System object™, and `nrULSCHDecoder` System object.

Support for CSI-RS

Use these functions and objects to model standard-compliant channel state information reference signals (CSI-RS).

Functions	Description
<code>nrCSIRS</code>	Generate CSI-RS symbols
<code>nrCSIRSIndices</code>	Generate CSI-RS resource element indices

Objects	Description
<code>nrCSIRSConfig</code>	CSI-RS configuration parameters
<code>nrCarrierConfig</code>	Carrier configuration parameters

5G NR-TM waveform generation

Generate standard-compliant 5G NR test models (NR-TMs) for frequency range 1 (FR1) and FR2. For NR-TM waveform generation, you can specify the NR-TM name, the channel bandwidth, the subcarrier spacing, and the duplexing mode. For an example, see “5G NR-TM and FRC Waveform Generation”.

NR downlink ACLR measurement

Measure the adjacent channel leakage ratio (ACLR) for 5G NR test models in FR1 and FR2. For an example, see “5G NR Downlink ACLR Measurement”.

System-Level Simulation: Schedule 5G NR PUSCH resources

Schedule physical uplink shared channel (PUSCH) resources in FDD mode and evaluate scheduling performance. You can choose from various MAC scheduling strategies. For more information, see “NR PUSCH FDD Scheduling”.

Enhanced 5G NR waveform generation, including CSI-RS, SRS, and PT-RS

Downlink waveform generation now includes CSI-RS and physical downlink shared channel (PDSCH) phase tracking reference signal (PT-RS). For more information, see “5G NR Downlink Carrier Waveform Generation”.

Uplink waveform generation now includes sounding reference signal (SRS) and PUSCH PT-RS. For more information, see “5G NR Uplink Carrier Waveform Generation”.

R2019a

Version: 1.1

New Features

Bug Fixes

Support for uplink physical channels PUSCH and PUCCH

Use these functions to model the physical uplink shared channel (PUSCH) and the physical uplink control channel (PUCCH) formats 0, 1, 2, 3, and 4.

New Feature	Description
nrPUSCH	Generate PUSCH modulation symbols
nrPUSCHCodebook	Generate PUSCH precoding matrix
nrPUSCHDecode	Decode PUSCH modulation symbols
nrPUSCHDescramble	Perform PUSCH descrambling
nrPUSCHPRBS	Generate PUSCH scrambling sequence
nrPUSCHScramble	Perform PUSCH scrambling
nrPUCCH0	Generate PUCCH format 0 modulation symbols
nrPUCCH1	Generate PUCCH format 1 modulation symbols
nrPUCCH2	Generate PUCCH format 2 modulation symbols
nrPUCCH3	Generate PUCCH format 3 modulation symbols
nrPUCCH4	Generate PUCCH format 4 modulation symbols
nrPUCCHPRBS	Generate PUCCH pseudorandom scrambling sequence
nrPUCCHHoppingInfo	Get PUCCH hopping information
nrLowPAPRS	Generate low peak-to-average power ratio (low-PAPR) sequence
nrTransformPrecode	Generate transform precoded symbols
nrTransformDeprecode	Generate transform deprecoded symbols

For more information, see Uplink Physical Channels.

Support for uplink shared channel (UL-SCH) and uplink control information (UCI)

UL-SCH

These functions and System objects support UL-SCH encoding and decoding.

New Feature	Description
nrULSCH	Apply UL-SCH encoder processing chain
nrULSCHDecoder	Apply UL-SCH decoder processing chain
nrULSCHInfo	Get uplink shared channel (UL-SCH) information

For more information, see Uplink Transport Channels.

UCI

These functions support UCI encoding and decoding, including support for polar coding and channel coding of small block lengths.

New Feature	Description
nrUCIEncode	Encode uplink control information (UCI)
nrUCIDecode	Decode uplink control information (UCI)

For more information, see Uplink Control Information.

Parity-check support for polar coding

Polar coding functions `nrPolarEncode` and `nrPolarDecode` now support parity-check polar coding.

Addition of new downlink shared channel (DL-SCH) System objects

Use these System objects for DL-SCH encoding and decoding. These System objects replace the example-specific DL-SCH implementations in the previous release.

New Feature	Description
nrDLSCH	Apply DL-SCH encoder processing chain
nrDLSCHDecoder	Apply DL-SCH decoder processing chain

5G NR uplink waveform generation

You can now generate a 5G NR uplink waveform, including physical signals and channels. You can also parameterize and generate multiple bandwidth parts (BWP), and multiple

instances of the PUSCH and PUCCH channels over different BWPs. For an example, see 5G NR Uplink Carrier Waveform Generation.

5G NR PUSCH link-level reference simulation

You can measure the PUSCH throughput for various propagation conditions and parameter sets. For an example, see NR PUSCH Throughput.

R2018b

Version: 1.0

New Features

Introducing 5G Toolbox

5G Toolbox provides standard-compliant functions and reference examples for the modeling, simulation, and verification of 5G communications systems. The toolbox supports link-level simulation, golden reference verification and conformance testing, and test waveform generation.

With the toolbox you can configure, simulate, measure, and analyze end-to-end communications links. You can modify or customize the toolbox functions and use them as reference models for implementing 5G systems and devices.

The toolbox provides reference examples to help you explore baseband specifications and simulate the effects of RF designs and interference sources on system performance. You can generate waveforms and customize test benches to verify that your designs, prototypes, and implementations comply with the 3GPP 5G New Radio (NR) standard.

Standard-compliant models for 3GPP 5G NR Release 15

5G Toolbox provides standard-compliant functions and reference examples for the modeling, simulation, and verification of 5G communications systems. For more information, see Scope of 5G Toolbox.

Link-level simulation with reference examples, including 5G NR PDSCH throughput simulation

5G Toolbox provides standard-compliant functions for end-to-end physical layer transmit and receive processing. You can measure the PDSCH throughput for various propagation conditions and parameter sets. For an example, see NR PDSCH Throughput.

OFDM waveform generation with NR subcarrier spacings and frame numerologies

With 5G Toolbox, you can generate NR waveforms, including physical signals and channels. The process includes all the stages from channel coding to OFDM modulation. You can explore the effect of different subcarrier spacings and frame numerologies.

TR 38.901 propagation channel models, including tapped delay line (TDL) and clustered delay line (CDL)

5G Toolbox provides TDL and CDL channel modeling capabilities. For more details, see the `nrCDLChannel` and `nrTDLChannel` System objects.

Downlink transport and physical channels (shared, control, broadcast); synchronization and demodulation reference signals

5G Toolbox provides functions to model transport and physical channels, including shared, control, and broadcast channels. You can also model synchronization and demodulation reference signals. For more information, see [Downlink Channels](#).

Signal processing functions, including channel coding (LDPC and polar codes), channel estimation, synchronization, and equalization

5G Toolbox provides low-level functions that model subcomponents in the processing chain for transport channels, physical channels, and physical signals. For more information, see [Physical Layer Subcomponents](#). The toolbox also provides signal reception capability. Using 5G Toolbox functions, you can perform channel estimation, timing estimation, synchronization, and minimum mean-squared error (MMSE) equalization. For more information, see [Signal Reception](#).

C and C++ code generation support

All 5G Toolbox functions and System objects support ANSI®/ISO® compliant C/C++ code generation.

