5G Toolbox™ Release Notes

MATLAB®

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

5G Toolbox™ Release Notes

Contents

R2025a

R2024b

R2024a

R2023b

R2023a

R2022b

R2022a

R2021b

R2021a

R2020b

R2020a

R2019b

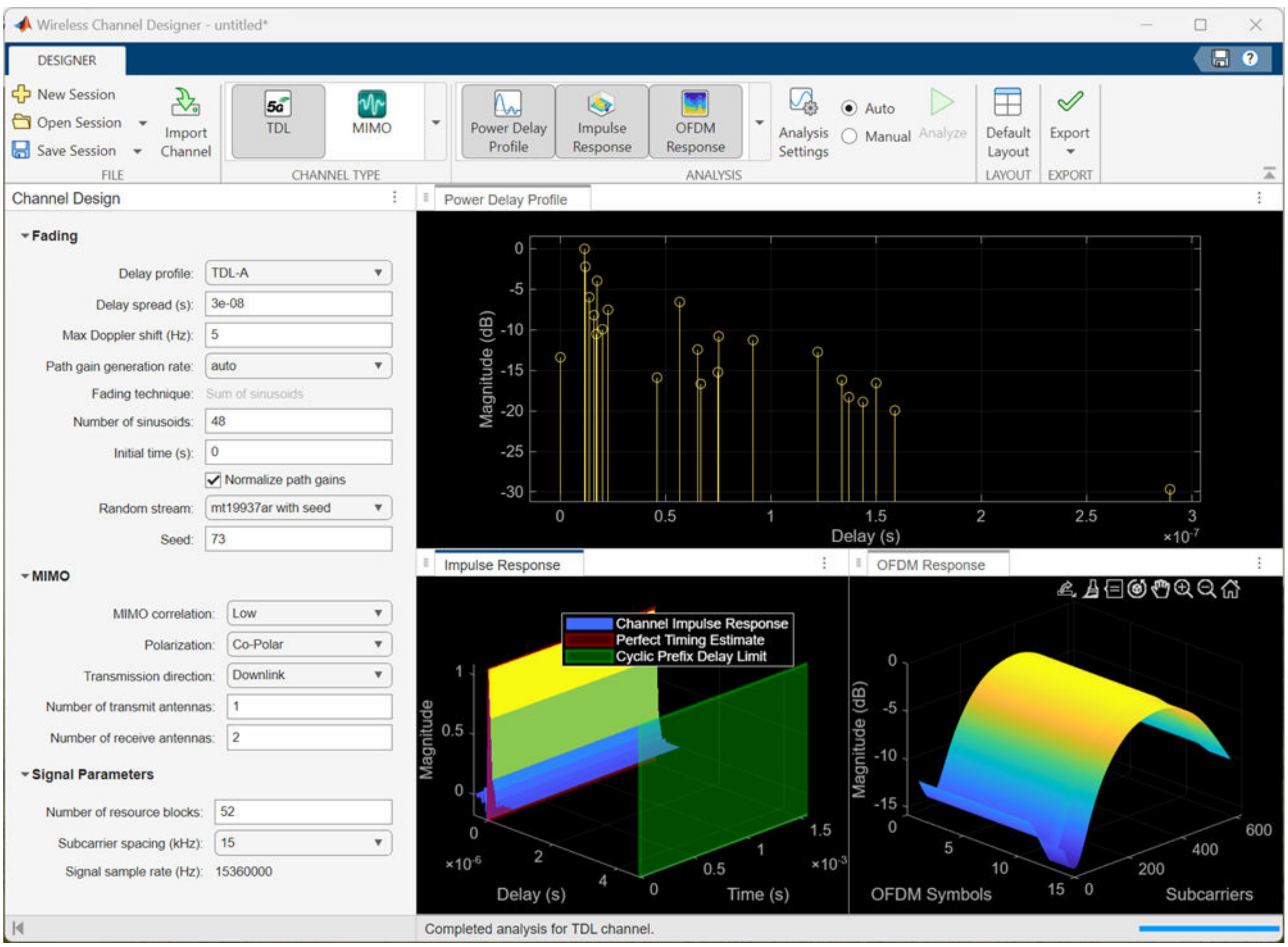
R2019a

R2025a

★Wireless Channel Designer App: Design and analyze 5G TDL channel model

Use the Wireless Channel Designer app to design and analyze a 5G tapped delay line (TDL) channel model. With this app, you can:

- Configure TDL channel model parameters. Each TDL channel model parameter in the app corresponds to a property of the `nrTDLChannel` object.
- Visualize the power delay profile, impulse response, and OFDM response of the configured channel.
- Export the channel configuration and the path gains to workspace variables or to a `.mat` file.
- Export the channel configuration to an executable MATLAB® script. Use the script to programmatically generate the path gains and to pass a waveform through the channel.



★5G Waveform Generator App: Configure custom TDD pattern in uplink FRC and downlink FRC waveforms

The 5G Waveform Generator app enables you to:

- Configure a custom time division duplex (TDD) pattern in uplink FRC and downlink FRC waveforms, as defined by the *TDD-UL-DL-Pattern* information element (IE) from TS 38.331.
- Configure transmissions in the special slots.

5G Waveform Generator App: Configure custom payloads and RV sequences for SIB1 transmissions

The 5G Waveform Generator app adds support for an increased number of system information block 1 (SIB1) configurations. You can now:

- Configure CORESET multiplexing pattern 1, including up to two PDCCH and PDSCH transmissions per slot.
- Configure custom payloads for SIB1 transmissions.
- Configure redundancy version (RV) sequences as a vector for SIB1 transmissions.

5G Waveform Analyzer App: Measure channel power and EVM; get CRC channel decoding results

The 5G Waveform Analyzer app enables you to:

- Measure the channel power for each instance of the physical downlink shared channel (PDSCH), physical uplink shared channel (PUSCH), and physical downlink control channel (PDCCH).
- Measure the error vector magnitude (EVM) of the PDSCH, PUSCH, PDCCH, and the associated reference signals for each channel instance.
- Get CRC decoding results for each instance of the downlink shared channel (DL-SCH), uplink shared channel (UL-SCH), and downlink control information (DCI).

5G Waveform Analyzer App: Analyze waveforms with nonzero initial subframe number

The 5G Waveform Analyzer app enables you to analyze downlink and uplink waveforms with nonzero initial subframe number.

5G Waveform Analyzer App: Analyze uplink waveforms that support interlaced PUSCH transmission and intracell guard bands, two codewords, and separate power scaling

The 5G Waveform Analyzer app enables you to:

- Analyze Release 16 uplink waveforms with interlaced PUSCH transmission and intracell guard bands for operation in unlicensed spectrum (NR-U).
- Analyze waveforms with two-codeword PUSCH transmissions.
- Analyze waveforms with separate power scaling for each PUSCH transmission in a period.

5G Waveform Analyzer App: Import waveform from workspace or waveform configuration from workspace or .mat file

The 5G Waveform Analyzer app enables you:

- Import a 5G waveform from the MATLAB workspace. For example, if you export a 5G waveform to the workspace by using the 5G Waveform Generator app, you can then import this waveform in the analyzer.
- Specify demodulation parameters by importing an `nrDLCarrierConfig` or `nrULCarrierConfig` waveform configuration object from the MATLAB workspace or a `.mat` file.

5G Waveform Analyzer App: Open the app from the MATLAB command prompt

Open the app from the MATLAB command prompt by calling the `nrWaveformAnalyzer` function.

5G Waveform Generator and Analyzer Apps: Configure Release 18 enhanced DM-RS multiplexing for MU-MIMO in PDSCH and PUSCH

The 5G Waveform Generator and 5G Waveform Analyzer apps enable you to configure Release 18 enhanced demodulation reference signal (DM-RS) multiplexing for multi-user multiple-input multiple-output (MU-MIMO) transmissions in the PDSCH or PUSCH instance of a 5G NR waveform.

5G Waveform Generator and Analyzer Apps: Configure CCE offset in PDCCH

The 5G Waveform Generator and 5G Waveform Analyzer apps enable you to configure control channel element (CCE) offset in the PDCCH instance of a 5G NR downlink waveform.

Programmatic Waveform Generation: Specify antenna mapping and precoding matrix in channels and signals

The object properties listed in this table enable you to specify the antenna mapping and a precoding matrix in channels and signals for programmatic waveform generation. The `info.WaveformResources` output structure of the `nrWaveformGenerator` function returns information about the CSI-RS and each configured MIMO precoding matrix.

Object	Property
nrWavegenCSIRSConfig nrWavegenPDCCHConfig nrWavegenPDSCHConfig nrWavegenPUCCH0Config nrWavegenPUCCH1Config nrWavegenPUCCH2Config nrWavegenPUCCH3Config nrWavegenPUCCH4Config nrWavegenSRSConfig nrWavegenPUSCHConfig	AntennaMapping, PrecodingMatrix
nrWavegenSSBurstConfig	AntennaMapping, SSBlockPrecodingMatrix

Programmatic Waveform Generation: Get CSI-RS configuration information

- The `nrWavegenCSIRSConfig` object enables you to obtain the number of configured channel state information reference signal (CSI-RS) ports through the `NumCSIRSPorts` read-only object property.
- The `info.WaveformResources` output structure of the `nrWaveformGenerator` function now returns CSI-RS resources information through the `CSIRS` field.

DL-SCH and UL-SCH Encoder and Decoder: Enable support for CBG-based transmission

- The `nrDLSCH`, `nrDLSCHDecoder` and `nrULSCH`, `nrULSCHDecoder` System objects add support for transmission based on code block groups (CBGs). To enable support for CBG-based transmission in any of these objects, set the `CBGTransmission` object property to `true`, then call the object with additional CBG transmission-related input and output arguments.
- The “NR PDSCH Throughput” and “NR PUSCH Throughput” examples now enable you to specify CBG-based transmission.

DL-SCH and UL-SCH Encoder and Decoder: Get transport block sizes and decoding information, and control flushing of soft buffer

- Use the `TransportBlockSizesPerProcess` read-only property of the `nrDLSCH`, `nrDLSCHDecoder`, `nrULSCH`, or `nrULSCHDecoder` objects to get the transport block sizes per each HARQ process.

- Call the `info` object function with an `nrDLSCHDecoder` or `nrULSCHDecoder` object to get decoding status information associated with the last call of the decoder.
- Disable the automatic flushing of the soft buffer associated with a transport block on a transport block CRC pass by setting the `AutoFlushSoftBuffer` property of the `nrDLSCHDecoder` or `nrULSCHDecoder` object to `false`. In previous releases, the objects automatically flush these soft buffers.

Get TBS directly from PDSCH or PUSCH

- The `nrTBS` function enables you to get the transport block size (TBS) directly from a PDSCH or PUSCH configuration object by using object-specific syntaxes.
- Use the `TransportBlockSize` property of the `nrWavegenPDSCHConfig` or `nrWavegenPUSCHConfig` objects to get the transport block size associated with each codeword in the PDSCH or PUSCH transmission, respectively.

Practical Channel Estimation: Turn off interpolation and enable subband channel estimation

The `nrChannelEstimate` function enables you to:

- Turn off interpolation by setting the `Interpolation` name-value input argument to `'off'`. When you turn off interpolation, the function returns the channel estimates only for the reference symbol locations that are specified by the `refInd` or `refGrid` inputs for each receive antenna across all ports. The function returns zero for all other grid locations.
- Estimate channel subbands independently by setting the `PRGBundleSize` name-value input argument to a nonempty value. In this case, the function returns the `nVar` and `info` outputs in terms of the number of the transmitted PRGs.

Enable phase correction in CSI-RS and SSB measurements

Use the `EnablePhaseCorrection` name-value argument in the `nrCSIRSMeasurements` and `nrSSBMeasurements` functions to enable phase correction in CSI-RS and SSB measurements, respectively. When you enable phase correction, the functions handle timing offsets due to synchronization errors and frequency selective channels in the received waveform.

Configure CSI reporting using nrCSIReportConfig

Use the `nrCSIReportConfig` object to configure CSI reporting parameters. The object sets the parameters to measure the downlink channel using CSI-RS, as defined in the *CSI-ReportConfig* IE of TR 38.331. The default object configures the report for the wideband type I single panel codebook mode. This object replaces the example-specific implementations in previous releases.

Measure EVM using nrEVM

Use the `nrEVM` function to measure the EVM of PDSCH, PUSCH, and PDCCH. This function replaces the example-specific implementations in previous releases.

CDL Channel Model: Some object properties become tunable

Starting in R2025a, these properties of the `nrCDLChannel` System object™ are tunable, so you can change their values at any time. In previous releases, these properties are nontunable.

- `PathDelays`
- `AveragePathGains`
- `AnglesAoD`
- `AnglesAoA`
- `AnglesZoD`
- `AnglesZoA`
- `MaximumDopplerShift`
- `UTDirectionOfTravel`
- `ReceiveArrayOrientation`
- `TransmitArrayOrientation`
- `InitialPhases`

Generate training data for NR SSB temporal-domain beam prediction

The “Generate Training Data for NR SSB Temporal-Domain Beam Prediction” example shows how to simulate 2-D trajectories within an urban macrocell (UMa) scenario and then calculate the optimal beam pair along the trajectory.

Model hybrid beamforming with CDL channel

The “Model Hybrid Beamforming with CDL Channel” example shows how to apply digital and analog beamforming to a 5G NR PDSCH link by using a CDL channel.

GPU Array Support: Run LDPC, CRC, DL-SCH, and UL-SCH encoding and decoding on GPU

Run the functions listed in this table on a graphics processing unit (GPU) using Parallel Computing Toolbox™. When you call any of these functions with at least one `gpuArray` object as a data input argument, the function executes on the GPU. To retrieve the array from the GPU, for example, when using a function that does not support `gpuArray` objects, use the `gather` function. For more information, see “Run MATLAB Functions on a GPU” (Parallel Computing Toolbox).

Function	GPU Array Support
`nrChannelEstimate`	The function accepts and returns GPU arrays but does not run on a GPU.
`nrCRCEncode`	Full support
`nrCRCDecode`	Full support
`nrCodeBlockDesegmentLDPC`	Full support
`nrCodeBlockSegmentLDPC`	Full support

Function	GPU Array Support
`nrLDPCEncode`	Full support
`nrRateMatchLDPC`	Full support
`nrRateRecoverLDPC`	Full support
`nrDLSCH`	To return the `codedBits` output on the GPU, specify the transport block as a `gpuArray` object in the `setTransportBlock` function call.
`nrDLSCHDecoder`	Full support
`nrULSCH`	To return the `codedBits` output on the GPU, specify the transport block as a `gpuArray` object in the `setTransportBlock` function call.
`nrULSCHDecoder`	Full support

Accelerate 5G simulation using GPU

When your simulation runs on a very large data set, using a GPU can accelerate the overall simulation time by performing the data-intensive part of the computation on the GPU using GPU arrays (requires Parallel Computing Toolbox).

- The “Accelerate 5G Simulation Using GPU” example shows how to use GPU arrays in a simplified PDSCH link-level simulation to speed up execution time. This simplified simulation consists of PDSCH symbol encoding, OFDM modulation, transmission through a CDL channel, OFDM demodulation, and PDSCH symbol decoding.
- The “NR PDSCH Throughput” example now enables you to use GPU arrays in a full PDSCH link-level simulation.

AI for 5G: Apply PyTorch coexecution workflows

These examples demonstrate workflows for MATLAB coexecution with PyTorch®:

- “Train PyTorch Channel Prediction Models” trains PyTorch channel prediction neural networks using data generated in MATLAB.
- “Offline Training and Testing of PyTorch Model for CSI Feedback Compression” trains an autoencoder-based PyTorch neural network offline and tests for CSI compression.
- “Online Training and Testing of PyTorch Model for CSI Feedback Compression” trains an autoencoder-based PyTorch neural network online and test for CSI compression.
- “Apply Transfer Learning on PyTorch Model to Identify 5G and LTE Signals” demonstrates coexecution with Python® to identify 5G NR and LTE signals by using the transfer learning technique on a pretrained PyTorch semantic segmentation network for spectrum sensing.

AI for 5G: Prepare data for CSI feedback

The “Prepare Data for CSI Processing” example generates channel estimates and prepare a data set to train AI-based systems, such as an autoencoder for CSI feedback compression.

Use a 5G waveform for integrated sensing and communication

The "Integrated Sensing and Communication Using 5G Waveform" (Phased Array System Toolbox) example demonstrates integrated sensing and communication (ISAC) capabilities of a 5G waveform.

System-Level Simulation: Specify SRS transmission periodicity of UE node

Use the `SRSPeriodicityUE` object property of the `nrGNB` object to specify the sounding reference signal (SRS) transmission periodicity of the UE nodes.

★System-Level Simulation: Use SRS-based downlink MU-MIMO transmission

The `configureScheduler` object function adds support for SRS-based downlink MU-MIMO transmission. Use the `MUMIMOConfigDL` name-value argument of the function to configure SRS-based downlink MU-MIMO parameters.

System-Level Simulation: Specify maximum number of layers for CSI-RS-based downlink MU-MIMO transmission

The `MUMIMOConfigDL` name-value argument of the `configureScheduler` object function now enables you to specify the maximum number of layers for a CSI-RS-based downlink MU-MIMO transmission by using the `MaxNumLayers` field.

6G Exploration Library: Configure custom modulation schemes in PDSCH

Configure custom modulation schemes in PDSCH by specifying the `Modulation` property of the `pre6GPDSCHConfig` object as `'Custom'` and then by specifying the `Constellation` property as a complex vector of constellation points.

The "Configure Custom Modulation Schemes for 6G" example shows how to use custom modulation schemes to explore constellation design.

6G Exploration Library: Use AI-native fully convolutional receiver

The "AI-Native Fully Convolutional Receiver" example shows how to replace conventional channel estimation, equalization, and symbol demodulation operations with an AI-native air interface (AI-AI).

6G Exploration Library: Simulate an AI-AI using PyTorch coexecution

The "Verify Performance of 6G AI-Native Receiver Using MATLAB and PyTorch Coexecution" example shows how to integrate a pretrained PyTorch network with MATLAB-based data generation to simulate an AI-AI.

6G Exploration Library: Analyze SINR in 6G FR3 network using digital twin

The "Analyze SINR in 6G FR3 Network Using Digital Twin" example shows how to use digital twin techniques to analyze the signal-to-interference-plus-noise ratio (SINR) performance in a 6G FR3 network.

6G Exploration Library: Explore a large-bandwidth 6G candidate waveform using USRP X410

The "Explore 400 MHz 6G Waveform Using USRP X410" example shows how to generate, transmit, capture, and analyze a large-bandwidth 6G candidate waveform.

6G Exploration Library: Use the library in MATLAB Online

You can now use 6G Exploration Library for 5G Toolbox in MATLAB Online™.

Functionality being removed or changed

HST channel model limits the number of antennas to discrete set of values
Behavior change

Starting in R2025a, the `nrHSTChannel` object limits the number of antennas to a discrete set of values:

- The number of transmit antennas specified by the `NumTransmitAntennas` property and the N_T parameter of the `signalIn` input is now limited to 1, 2, 4, and 8.
- The number of receive antennas specified by the `NumReceiveAntennas` property is now limited to 1, 2, and 4.

In previous releases, the number of antennas can be any positive integer.

HST channel model uses static channel propagation matrices
Behavior change

Starting in R2025a, the `nrHSTChannel` object uses static channel propagation matrices, as defined in TS 38.101-4 Annex B.1. This change affects the channel path gains and filtered signal outputs when you call the object. In previous releases, the channel uses static channel propagation matrices only when you set the `Velocity` and `MaximumDopplerShift` properties to `0`.

Connecting more than 16 UEs to a gNB node requires manual adjustment of the SRS transmission periodicity
Behavior change

Starting in R2025a, when you connect more than 16 UE nodes to a gNB node using the `connectUE` object function, you must adjust the SRS transmission periodicity of the `nrGNB` object manually. Set the `SRSPeriodicityUE` object property to the next multiple of L, where L is the interval in slots for reserving one symbol for the SRS resource across the entire bandwidth. In previous releases, the `nrGNB` object automatically increases the SRS transmission periodicity for over 16 UE nodes.

Uplink and downlink power normalization changes in nrUE and nrGNB
Behavior change

Starting in R2025a, the `nrUE` and `nrGNB` objects do not perform uplink and downlink power normalization using the square root of the number of transmitting antennas because the uplink and downlink precoders are already normalized. In previous releases, the `nrUE` and `nrGNB` objects apply power normalization twice, first using the square root of the number of transmitting antennas and then through the normalized precoders.

Initial MCS index changes for PDSCH and PUSCH

Behavior change

Starting in R2025a, the `nrGNB` object sets the initial modulation and coding scheme (MCS) index to 0 for the PDSCH and instructs the UE node to use 0 for the PUSCH. This initial MCS applies when measurement reports are unavailable during the initial slots of the simulation. Setting MCS to 0 can lower the block error rate (BLER) associated with UE nodes that experience poor channel conditions. In previous releases, the object sets the MCS index to 11, which can increase the BLER and necessitate up to three retransmissions of the initial packets.

nrPUSCHCodebook returns different precoding matrix for specific input argument combinations

Behavior change

The `nrPUSCHCodebook` function returns a different precoding matrix for specific input argument combinations, in line with the V18.4.0 update of TS 38.211 for Tables 6.3.1.5-12, 6.3.1.5-19, 6.3.1.5-20, 6.3.1.5.23, 6.3.1.5.24, and 6.3.1.5.26. This behavior change applies when you call the function with `nPorts` = `8`, `transformPrecode` = `false`, and any of these additional input argument combinations:

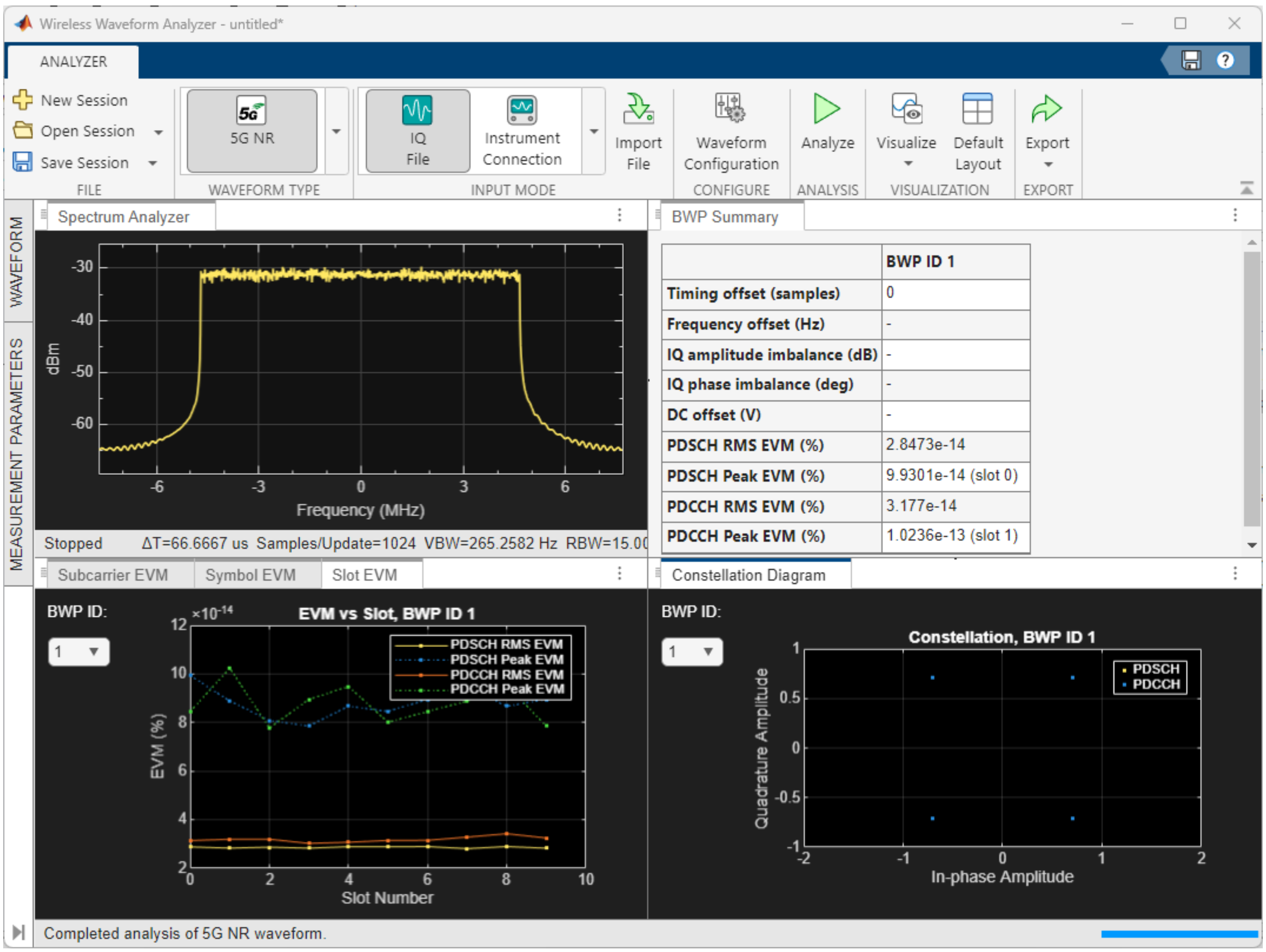
- `nLayers` = `4`, `tpmi` = `14`, and `codebookType` = `'codebook1_ng1n4n1'`
- `nLayers` = `3`, `tpmi` = `17`, and `codebookType` = `'codebook1_ng1n2n2'`
- `nLayers` = `4`, `tpmi` = `17`, and `codebookType` = `'codebook1_ng1n2n2'`
- `nLayers` = `7`, `tpmi` = `5`, and `codebookType` = `'codebook1_ng1n2n2'`
- `nLayers` = `8`, `tpmi` = `5`, and `codebookType` = `'codebook1_ng1n2n2'`
- `nLayers` = `2`, `tpmi` = `6`, and `codebookType` = `'codebook2'`
- `nLayers` = `3`, `tpmi` is specified as `14`, `22`, `30`, `38`, `46`, `54`, `62`, `70`, `78`, `86`, `94`, `102`, `110`, `118`, `126`, `134`, or an integer in the range [232, 247], and `codebookType` = `'codebook2'`
- `nLayers` = `4`, `tpmi` is specified as `10`, `18`, `26`, `34`, `42`, `50`, `52`, `53`, `54`, `55`, `56`, `57`, `58`, `59`, or `66`, and `codebookType` = `'codebook2'`
- `nLayers` = `5`, `tpmi` is specified as `24`, `25`, `26`, or `27`, and `codebookType` = `'codebook2'`

R2024b

★5G Waveform Analyzer App: Analyze and visualize imported or captured 5G waveforms and export analysis results

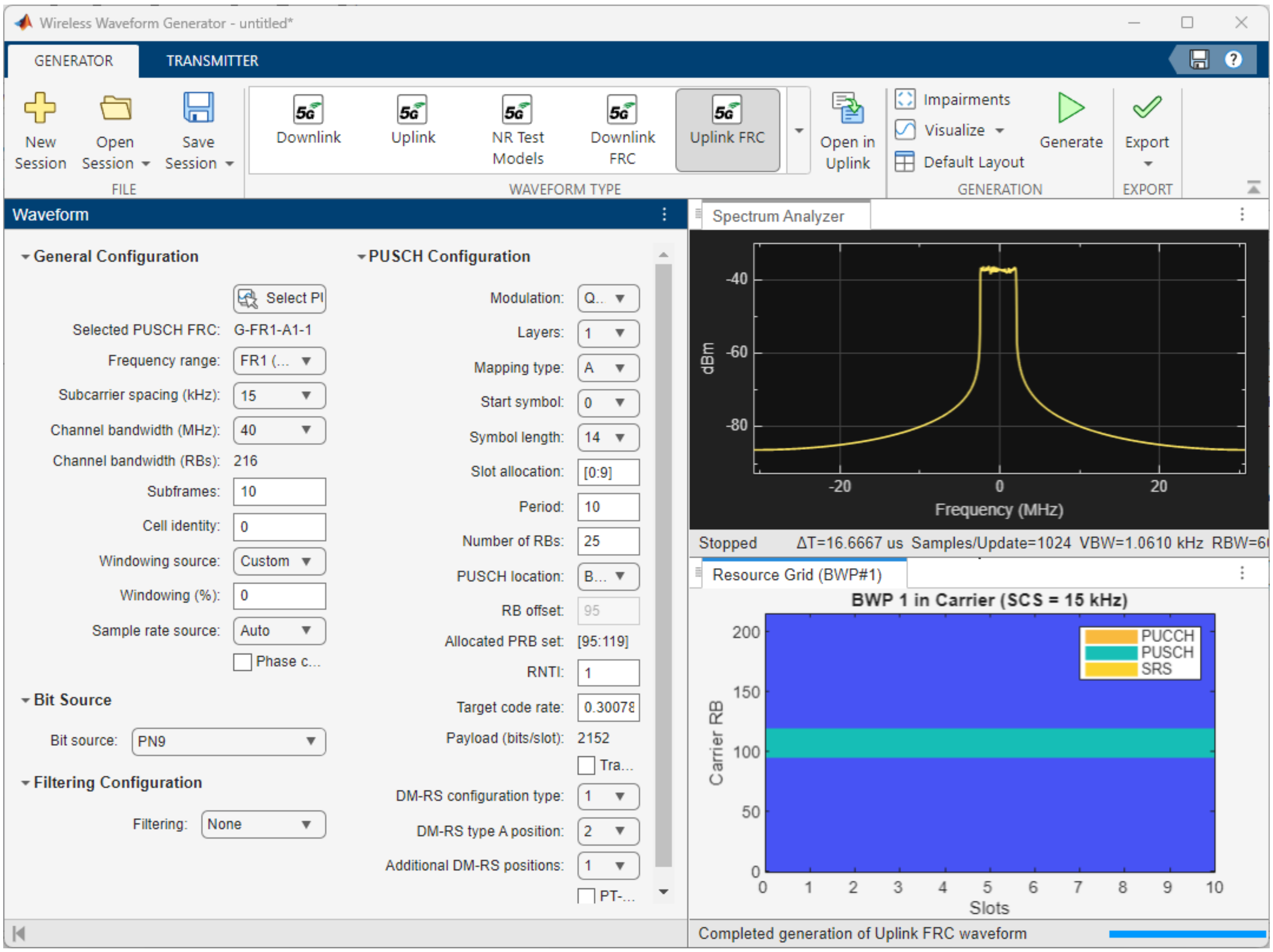
Use the 5G Waveform Analyzer app to analyze and visualize imported or captured 5G waveforms and export analysis results. With the app, you can:

- Import waveforms from a `.bb`, `.mat`, or `.txt` file.
- Capture waveforms using a connected spectrum analyzer instrument that supports the virtual instrument software architecture (VISA) interface (requires Instrument Control Toolbox™).
- Specify the waveform configuration and analysis parameters.
- Choose from a range of measurement visualization options such as a spectrum analyzer, constellation diagram, or various error vector magnitude (EVM) plots.
- Export waveform analysis results to your workspace or to a `.mat` file.



★5G Waveform Generator App: Easily configure basic PUSCH waveforms

The 5G Waveform Generator app adds support for easy PUSCH waveform configuration for a single numerology and a single sequence PUSCH transmission. This capability is available in the 5G uplink FRC waveform type.



5G Waveform Generator App: Configure Release 15, 16, 17, and 18 PUSCH FRC waveforms

The 5G Waveform Generator app enables you to configure Release 15, 16, 17, and 18 physical uplink shared channel (PUSCH) FRC waveforms, as defined in TS 38.104.

5G Waveform Generator App: Configure up to eight layers and limited buffer rate matching for PDSCH and PUSCH

The 5G Waveform Generator app enables you to configure up to eight layers, eight antenna ports, two codewords, and limited buffer rate matching for the physical downlink shared channel (PDSCH) and

PUSCH transmissions. You can specify the maximum number of layers and maximum modulation order for limited buffer rate matching.

5G Waveform Generator App: Configure separate power scaling for each PDSCH, PUSCH, and PDCCH transmission in a period

The 5G Waveform Generator app enables you to configure separate power scaling for each PDSCH, PUSCH, and physical downlink control channel (PDCCH) transmission in a period.

5G Waveform Generator App: Configure periodicity and half-frame offset of synchronization signal burst

The 5G Waveform Generator app enables you to configure the periodicity and half-frame offset of the synchronization signal burst in a 5G NR downlink waveform.

5G Waveform Generator App: Configure MIB/SIB1 transmission in downlink waveforms

The 5G Waveform Generator app enables you to configure system information block 1 (SIB1) transmissions when you select MIB for the SSB payload in 5G NR downlink waveforms.

Programmatic Waveform Generation: Configure limited buffer rate matching for PDSCH and PUSCH

Configure limited buffer rate matching for the PDSCH and PUSCH transmissions by using the `LimitedBufferRateMatching` property of the `nrWavegenPDSCHConfig` and `nrWavegenPUSCHConfig` objects, respectively. You can specify the maximum number of layers and maximum modulation order for the limited buffer rate matching by using the `MaxNumLayers` and `MCSTable` properties of the corresponding objects, respectively.

Programmatic Waveform Generation: Configure separate power scaling for each PDSCH, PUSCH, PDCCH, PUCCH, and SRS transmission in a period

Configure separate power scaling for each PDSCH, PUSCH, PDCCH, physical uplink control channel (PUCCH), and sounding reference signal (SRS) transmission in a period by setting the `Power` property of the corresponding configuration objects to a vector of length equal to the number of allocated transmissions in a period. This support is now available for the following waveform configuration objects:

- `nrWavegenPDSCHConfig`
- `nrWavegenPUSCHConfig`
- `nrWavegenPDCCHConfig`
- `nrWavegenPUCCH0Config`
- `nrWavegenPUCCH1Config`
- `nrWavegenPUCCH2Config`

- `nrWavegenPUCCH3Config`
- `nrWavegenPUCCH4Config`
- `nrWavegenSRSConfig`

Programmatic Waveform Generation: Configure periodicity and half-frame offset of synchronization signal burst

Configure the periodicity and half-frame offset of the synchronization signal burst in a 5G NR downlink waveform by using the `Period` property of the `nrWavegenSSBurstConfig` object. Specify `Period` as a vector of the form [*period offset*].

Generate ACS test waveform for 5G NR receiver testing

The ACS Test Waveform Generation for 5G NR Receiver Testing example provides a framework that enables you to generate a waveform for the adjacent channel selectivity (ACS) receiver test, as specified in TS 38.141-1, Section 7.4.1. The example shows how to use the generated waveform to test an arbitrary receiver model that measures throughput and EVM.

Generate NR V2X sidelink waveforms

The NR Sidelink Vector Waveform Generation example shows how to configure and generate a 5G NR V2X sidelink baseband waveform containing a sequence of physical sidelink control channel (PSCCH) and physical sidelink shared channel (PSSCH) transmissions, including sidelink control information (SCI) and SL-SCH transport channel coding.

Configure CDL channel model interactively

The Interactive CDL Channel Model Configuration example provides an app that enables you to configure a CDL channel model and visualize the channel characteristics and OFDM channel response. Using the app, you can save or load CDL channel configurations to and from the workspace as an `nrCDLChannel` object.

CDL and TDL Channel Models: Obtain OFDM channel response and timing offset directly from channel

The `nrCDLChannel` and `nrTDLChannel` System objects enable you to obtain the OFDM channel response and timing offset directly from the channel. To enable this functionality, set the `ChannelResponseOutput` object property to `'ofdm-response'`. For more information on how to use this functionality, see the NR PDSCH Throughput and NR PUSCH Throughput examples.

TDL Channel Model: Configure sample rate for path gain generation

Configure the sample rate for path gain generation by using the `PathGainSampleRate` property of the `nrTDLChannel` object. To enable the channel to automatically reduce the number of path gains samples for the channel filtering process, set `PathGainSampleRate` to `'auto'`. For more information on how to use this functionality, see the NR PDSCH Throughput and NR PUSCH Throughput examples.

Perform beam selection using TR 38.901 channel model and FR2 SSB

The NR SSB Beam Sweeping and Neural Network for Beam Selection examples now use a TR 38.901 channel model, baseline system-level simulation assumptions for AI/ML from TR 38.843 Table 6.3.1-1, and FR2 synchronization signal bursts (SSBs).

Deep learning data synthesis updates for 5G channel estimation

The Deep Learning Data Synthesis for 5G Channel Estimation example now uses a convolutional neural network (CNN) with two filters in each convolutional layer and shows how to measure the performance of the CNN over different channel realizations.

Measure EVM for FRC and NR-TM with 1024-QAM modulation and increased subcarrier spacings

The following examples now enable you to measure the EVM for FR2-2 fixed reference channel (FRC) waveforms or NR test models (NR-TMs) configured with 1024-QAM modulation and subcarrier spacings of 480 kHz and 960 kHz:

- EVM Measurement of 5G NR PUSCH Waveforms
- EVM Measurement of 5G NR Downlink Waveforms with RF Impairments
- 5G NR Downlink ACLR Measurement

Measure power per RE for 5G NR waveforms

The following examples now enable you to measure the power per resource element (RE) for various channels in a 5G NR waveform:

- EVM Measurement of 5G NR PUSCH Waveforms
- EVM Measurement of 5G NR Downlink Waveforms with RF Impairments

Obtain per configuration EVM measurements for PDSCH, PDCCH, DM-RS, and PT-RS

The EVM Measurement of 5G NR Downlink Waveforms with RF Impairments example now enables you to obtain EVM measurements for each PDSCH, PDCCH, DM-RS, and PT-RS configurations.

★System-Level Simulation: Implement custom PUSCH and PDSCH resource scheduling strategies

The `nrScheduler` base class enables you to implement a custom PUSCH or PDSCH resource scheduling strategy in a subclass. To add this custom scheduler into the gNB node, specify the `Scheduler` name-value argument of the `configureScheduler` function as an object of the subclass derived from the `nrScheduler` base class. For more information on how to use this functionality, see the Use Custom Scheduler in 5G System-Level Simulation example.

System-Level Simulation: Configure reference signal for downlink transmission at gNB node

Configure the reference signal for the downlink transmission at a gNB node by specifying the `CSIMeasurementSignalDL` name-value argument of the `configureScheduler` function as `'CSI-RS'` (default) or `'SRS'`. For more information on how to use this functionality, see the NR Cell Performance Evaluation with MIMO example.

System-Level Simulation: Customize RV sequence and number of retransmissions

Customize the redundancy version (RV) sequence and the number of retransmissions by specifying the `RVSequence` name-value argument of the `configureScheduler` function as an array of unique integer values from 0 to 3.

System-Level Simulation: Evaluate performance of cell-free mMIMO network

The Evaluate Performance of Cell-Free mMIMO Networks example shows how to evaluate the system performance of a cell-free (CF) massive multiple-input multiple-output (mMIMO) network.

6G Exploration Library: Configure 4096-QAM modulation in PDSCH

Configure 4096-QAM modulation in PDSCH by specifying the `Modulation` property of the `pre6GPDSCHConfig` object as `'4096QAM'`.

6G Exploration Library: Generate and analyze DFT-s-OFDM waveform

The Measure Impact of Sub-THz Hardware Impairments on 6G Waveforms example now enables you to generate and analyze a discrete Fourier transform-spread-OFDM (DFT-s-OFDM) pre-6G waveform.

GPU Array Support: Run nrCDLChannel, nrPDSCH, nrPDSCHDecode, nrPUSCH, nrPUSCHDecode, nrPUSCHScramble, nrPerfectChannelEstimate, nrPerfectTimingEstimate, and nrExtractResources on GPU

Run the functions listed in this table on a graphics processing unit (GPU) using Parallel Computing Toolbox. When you call any of these functions with at least one `gpuArray` (Parallel Computing Toolbox) object as a data input argument, the function executes on the GPU. To retrieve the array from the GPU, for example, when using a function that does not support `gpuArray` objects, use the `gather` (Parallel Computing Toolbox) function. For more information, see Run MATLAB Functions on a GPU (Parallel Computing Toolbox).

Function	GPU Array Support
`nrCDLChannel`	Only when you set the `ChannelFiltering` property to `true`.
`nrPDSCH`	Full support

Function	GPU Array Support
`nrPDSCHDecode`	Full support
`nrPUSCH`	Full support
`nrPUSCHDecode`	Full support
`nrPUSCHScramble`	Full support
`nrPerfectChannelEstimate`	Full support
`nrPerfectTimingEstimate`	Full support
`nrExtractResources`	Full support

Functionality being removed or changed

Downlink waveform generation implements limited buffer rate matching for PDSCH

Behavior change

The 5G Waveform Generator app and the `nrWaveformGenerator` function now implement limited buffer rate matching for the PDSCH transmission. Starting in R2024b, the PDSCH is generated with limited buffer rate matching. In previous releases, the PDSCH is generated without limited buffer rate matching.

For backward compatibility, the app and the programmatic waveform generation add support for disabling limited buffer rate matching through these options:

- In the app, deselect **Limited buffer rate matching**.
- In the `nrWavegenPDSCHConfig` object, set the `LimitedBufferRateMatching` property to `false`.

System-level simulation dynamically selects uplink rank

Behavior change

System-level simulation dynamically selects the uplink rank based on the measured channel quality. This behavior change affects the simulation results when you configure multiple-input and multiple-output (MIMO) antenna configuration in the uplink direction. Starting in R2024b, when you set the `NumTransmitAntennas` property of the `nrUE` object and the `NumReceiveAntennas` property of the `nrGNB` object to a value greater than one, the simulation dynamically selects the uplink rank. In previous releases, irrespective of the antenna configuration in the uplink direction, the simulation sets the uplink rank to one.

R2024a

★Introducing 6G Exploration Library for 5G Toolbox

6G Exploration Library for 5G Toolbox enables you to explore, model, simulate, and test candidate 6G waveforms and technologies. Examples show you how to:

- Generate waveforms with parameters extended beyond the limits of the 5G NR specifications.
- Simulate 6G-candidate links, including transmitter operations, channel models, RF impairments, and reference receiver algorithms. Compute metrics including throughput and EVM.
- Evaluate link performance in 7–20 GHz, mmWave, and sub-Terahertz ranges.
- Model reconfigurable intelligent surfaces and experiment with links with and without the presence of blockages.
- Apply AI techniques to solve 6G wireless communications problems.
- Accelerate your simulation by using multicore computers and clusters.

For more information, see 6G Exploration Library. To use this functionality, download and install the 6G Exploration Library for 5G Toolbox add-on.

★Simulate 5G-Advanced Release 18 eMIMO for PDSCH, PUSCH, and SRS transmissions

PUSCH Updates: Specify eight antenna ports, up to eight layers, and two codewords in PUSCH transmissions

- The `nrPUSCHConfig` and `nrWavegenPUSCHConfig` objects add support for eight antenna ports, up to eight layers, and two codewords in physical uplink shared channel (PUSCH) transmissions, as defined in Release 18 of TS 38.211 Section 6.3.1. Use the object properties and the corresponding values listed in this table to enable this support.

Property	Value Summary for Eight Ports and Up to Eight Layers
`NumLayers`	5, 6, 7, and 8
`Modulation`	One-element or two-element cell array
`NumAntennaPorts`	8
`TPMI`	Integer between 0 and 304, depending on the `NumLayers`, `NumAntennaPorts`, `CodebookType` property values
`CodebookType`	Precoding matrices for codebook transmission with eight antenna ports
`NumCodewords` (read-only)	The number of configured codewords corresponding to the specified number of layers
`TargetCodeRate` (`nrWavegenPUSCHConfig` only)	One-element or two-element array
`RVSequence` (`nrWavegenPUSCHConfig` only)	One-element or two-element cell array

  This support is available in the `nrULCarrierConfig` object and the `nrPUSCH`, `nrPUSCHIndices`, `nrPUSCHDecode`, `nrPUSCHCodebook`, `nrPUSCHDMRS`, `nrPUSCHPTRS`, and `nrWaveformGenerator` functions.

- The nrPUSCHDecode function enables you to configure the format of output arguments. When you add UniformCellOutput=true to the list of input arguments, the function returns the outputs as cell arrays irrespective of the number of layers.
- The nrPUSCHScramble and nrPUSCHDescramble functions now enable you to specify two codewords for scrambling.
- The nrPUSCHPRBS function now enables you to specify the codeword number as an optional input argument. The function uses this input to calculate the scrambler initialization, as described in Release 18 of TS 38.211 Section 6.3.1.1.

UL-SCH and DL-SCH Updates: Encode and decode UL-SCH on two transport blocks and configure output format of UL-SCH and DL-SCH encoding and decoding

- The nrULSCH and nrULSCHDecoder System objects now support the uplink shared channel (UL-SCH) encoder and decoder processing chain on two transport blocks, as defined in Release 18 of TS 38.212 Section 6.2.
- The nrULSCHInfo, nrULSCHMultiplex, and nrULSCHDemultiplex functions now support two-codeword configurations.
- The getTransportBlock, setTransportBlock, and resetSoftBuffer object functions now enable you to specify the HARQ process number and transport block number as name-value arguments. This support is available in both the UL-SCH and downlink shared channel (DL-SCH) processing chain.
- The nrULSCH and nrDLSCH encoder objects and the nrULSCHDecoder and nrDLSCHDecoder decoder objects now enable you to configure the output format of the encoded codewords and decoded transport blocks, respectively, through the UniformCellOutput object property.

PDSCH and PUSCH DM-RS Updates: Use enhanced DM-RS multiplexing for MU-MIMO downlink and uplink transmissions

- The nrPDSCHDMRSConfig and nrPUSCHDMRSConfig objects add support for enhanced demodulation reference signal (DM-RS) multiplexing for multi-user multiple-input multiple-output (MU-MIMO) transmissions through the DMRSEnhancedR18 object property.

SRS Updates: Specify eight antenna ports and enable cyclic shift and comb offset hopping in SRS transmissions

- The nrSRSConfig and nrWavegenSRSConfig objects now enable you to specify the use of eight-port time division multiplexing (TDM) for sounding reference signal (SRS) transmission by using the EnableEightPortTDM object property. This support is available also in the nrSRS and nrSRSIndices functions.
- Use the nrSRSConfig and nrWavegenSRSConfig object properties listed in this table to configure cyclic shift hopping and comb offset hopping in SRS transmissions. This support is available also in the nrSRS and nrSRSIndices functions.

Property	Description
CyclicShiftHopping	Enable cyclic shift hopping
CyclicShiftHoppingID	Cyclic shift hopping identity
CyclicShiftHoppingSubset	Cyclic shift hopping subset
HoppingFinerGranularity	Enable cyclic shift hopping finer granularity
CombOffsetHopping	Enable comb offset hopping

Property	Description
CombOffsetHoppingID	Comb offset hopping identity
CombOffsetHoppingSubset	Comb offset hopping subset
HoppingWithRepetition	Enable comb offset hopping with repetition

5G Waveform Generator App: Configure interlaced PUSCH and PUCCH transmissions for operation in NR-U

The 5G Waveform Generator app now enables you to:

- Configure Release 16 interlacing for operation in unlicensed spectrum (NR-U). You can configure interlaced PUSCH transmissions and interlaced physical uplink control channel (PUCCH) transmissions.
- Configure intracell guard bands for operation with shared spectrum channel access for frequency range 1 (FR1) in NR-U.

5G Waveform Generator App: Export IQ data without exporting metadata

Save IQ data to a `.txt` or `.mat` file without saving metadata when exporting 5G NR waveforms from the 5G Waveform Generator app.

TDL Channel Model Updates: Configure additional Release 17 channel profiles and NTN channel profiles

The `nrTDLChannel` System object now enables you to:

- Configure additional simplified tapped delay line (TDL) delay profiles from Release 17 of TS 38.101-4 Annexes B.2.1.1 and B.2.1.2 by setting the `DelayProfile` object property to `'TDLD30'`, `'TDLA10'`, or `'TDLD10'`.
- Configure nonterrestrial network (NTN) channel profiles from TR 38.811 by setting `DelayProfile` to `'NTN-TDL-A'`, `'NTN-TDL-B'`, `'NTN-TDL-C'`, or `'NTN-TDL-D'`.
- Configure the NTN simplified delay profiles from TS 38.101-5 by setting `DelayProfile` to `'NTN-TDLA100'` or `'NTN-TDLC5'`.
- Configure the Doppler shift due to satellite movement for all channel taps by using the `SatelliteDopplerShift` object property. To enable this property, set `DelayProfile` to `'NTN-TDL-A'`, `'NTN-TDL-B'`, `'NTN-TDL-C'`, `'NTN-TDL-D'`, `'NTN-TDLA100'`, or `'NTN-TDLC5'`.

Access CQI and MCS lookup tables

- Use the `nrCQITables` object to retrieve the channel quality indicator (CQI) lookup tables, as defined in TS 38.214 Section 5.2.2.1.
- Use the `nrPDSCHMCSTables` and `nrPUSCHMCSTables` objects to retrieve the PDSCH and PUSCH modulation and coding scheme (MCS) lookup tables, as defined in TS 38.214 Section 5.1.3.1 and Section 6.1.4.1, respectively.

Use AI for UE positioning

The AI for Positioning Accuracy Enhancement example shows how to use artificial intelligence (AI) for positioning NR user equipment (UE) and compares the AI-based performance with the traditional time difference of arrival (TDoA) technique commonly employed in NR cellular networks.

Capture synchronization signal burst with software-defined radio

The 5G NR Synchronization Signal Capture Using Software-Defined Radio example shows how to use a software-defined radio to capture a synchronization signal burst. The example also shows how to analyze the capture and identify the strongest synchronization signal block.

O-RAN Fronthaul Testing: Analyze CU-Plane messages, apply modulation compression for downlink data, and generate CU-Plane messages for uplink data

- The Analyze CU-Plane Messages for O-RAN Fronthaul Testing example shows how to analyze a packet capture (PCAP) file that contains control plane and user plane (CU-Plane) messages sent on the open radio access network (O-RAN) fronthaul.
- The Generate CU-Plane Messages for O-RAN Fronthaul Test example now enables you to:
  - Apply modulation compression for downlink data, as defined in TS O-RAN.WG4.CUS Annex A.5.
  - Generate CU-Plane messages for uplink data.

Accelerate link-level simulations using parallel processing

The Accelerate Link-Level Simulations with Parallel Processing example shows how to accelerate link-level simulations by using a cluster of workers from a parallel pool.

Compute EVM for DM-RS and PT-RS in 5G NR downlink and uplink waveforms

- The EVM Measurement of 5G NR Downlink Waveforms with RF Impairments example now supports EVM computation for PDSCH DM-RS, PDSCH phase tracking reference signal (PT-RS), and physical downlink control channel (PDCCH) DM-RS.
- The EVM Measurement of 5G NR PUSCH Waveforms example now supports EVM computation for PUSCH DM-RS and PUSCH PT-RS.

System-Level Simulation: Enable RLC-AM radio bearer and link adaptation

- The `RLCEntityType` property of the `nrRLCBearerConfig` object now enables you to specify a radio link control (RLC) acknowledged mode (AM) radio bearer between a gNB and UE node. Setting this property to `'AM'` enables you to configure the AM radio bearer through the `PollPDU`, `PollByte`, `MaxRetxThreshold`, `PollRetransmitTimer`, and `StatusProhibitTimer` properties.
- The `configureScheduler` function now enables you to configure link adaptation for the downlink and uplink directions at a gNB node by using the `LinkAdaptationConfigDL` and

LinkAdaptationConfigUL name-value arguments, respectively. When you enable link adaptation, the system dynamically adjusts the modulation and coding scheme (MCS) based on the data transmission success or failure. This feature requires the Communications Toolbox Wireless Network Simulation Library add-on.

System-Level Simulation: Evaluate 3GPP reference scenarios

The Evaluate 3GPP Indoor Reference Scenario example shows how to model, simulate, and evaluate the system-level performance of a 3GPP enhanced mobile broadband (eMBB) indoor hotspot (InH) scenario, as described in 3GPP TR 38.913.

System-Level Simulation: Use TR 38.901 channel model to simulate custom scenarios

The NR Cell Performance Evaluation with MIMO, NR Intercell Interference Modeling, and NR FDD Scheduling Performance Evaluation examples now use the 3GPP TR 38.901 channel model to simulate custom scenarios.

GPU Array Support: Run nrOFDMModulate, nrOFDMDemodulate, nrSymbolModulate, nrSymbolDemodulate, nrLDPCDecode, and nrEqualizeMMSE functions on GPU

Run the functions listed in this table on a graphics processing unit (GPU) using Parallel Computing Toolbox. When you call any of these functions with at least one `gpuArray` (Parallel Computing Toolbox) object as a data input argument, the function executes on the GPU. To retrieve the array from the GPU, for example, when using a function that does not support `gpuArray` objects, use the `gather` (Parallel Computing Toolbox) function. For more information, see Run MATLAB Functions on a GPU (Parallel Computing Toolbox).

Function	GPU Array Support
`nrOFDMModulate`	Full support
`nrOFDMDemodulate`	Full support
`nrSymbolModulate`	Full support
`nrSymbolDemodulate`	Full support
`nrLDPCDecode`	Full support
`nrEqualizeMMSE`	Full support

Functionality being removed or changed

Value range of DMRSPortSet property has changed

Behavior change

The value range of the `DMRSPortSet` property of the `nrPDSCHDMRSConfig` and `nrPUSCHDMRSConfig` objects has changed to match the PDSCH DM-RS and PUSCH DM-RS antenna port values from TS 38.211 Tables 7.4.1.1.2-5 and 6.4.1.1.3-5, respectively. In previous releases, the valid antenna port numbers range from 0 to 11 and the nominal value depends on the DM-RS configuration type and DM-RS symbol duration. Starting in R2024a, the valid antenna port numbers range from 0 to 23 and the nominal value also depends on whether enhanced DM-RS multiplexing is enabled. In the case of PDSCH DM-RS, the range from 0 to 23 in the `DMRSPortSet` property corresponds to the range from 1000 to 1023, respectively, in Table 7.4.1.1.2-5.

R2023b

5G Waveform Generator app updates

You can now use the 5G Waveform Generator app to:

- Set the physical downlink control channel (PDCCH) data source to an arbitrary sequence of bits in addition to the predefined pseudorandom noise (PN) sequences.
- Specify the initial subframe number in 5G NR downlink and uplink waveforms.
- Transmit generated waveforms using NI VST hardware (requires Instrument Control Toolbox).

Open 5G waveform configuration parameters in app

Use the `openInGenerator` object function of the `nrDLCarrierConfig` and `nrULCarrierConfig` objects to open your 5G waveform configuration parameters in the 5G Waveform Generator app. You can then visualize, edit, and generate your waveform in the app.

Specify initial subframe number in 5G NR downlink and uplink waveforms

Use the `InitialNSubframe` property of the `nrDLCarrierConfig` and `nrULCarrierConfig` objects to specify the initial subframe number in 5G NR downlink and uplink waveforms, respectively.

Release 16 PUSCH and PUCCH interlacing for operation in unlicensed spectrum (NR-U)

5G NR end-to-end simulations and uplink waveform generation now support Release 16 interlacing.

- Use the `Interlacing`, `RBSetIndex`, and `InterlaceIndex` properties of the `nrPUSCHConfig` and `nrWavegenPUSCHConfig` objects to configure interlaced physical uplink shared channel (PUSCH) transmissions. This support is available in the `nrPUSCH`, `nrPUSCHIndices`, `nrPUSCHDecode`, `nrPUSCHDMRS`, `nrPUSCHDMRSIndices`, `nrPUSCHPTRS`, and `nrPUSCHPTRSIndices` functions.
- The `nrPUSCHIndices` function now also returns the physical resource blocks (PRBs) allocated for PUSCH within the bandwidth part (BWP).
- Use the object properties in this table to configure interlaced physical uplink control channel (PUCCH) transmissions. This support is available in the `nrPUCCH`, `nrPUCCHIndices`, `nrPUCCHDecode`, `nrPUCCHDMRS`, and `nrPUCCHDMRSIndices` functions.

Object	Interlacing Properties
`nrPUCCH0Config`	`Interlacing`, `RBSetIndex`, `InterlaceIndex`
`nrPUCCH1Config`	`Interlacing`, `RBSetIndex`, `InterlaceIndex`
`nrPUCCH2Config`	`Interlacing`, `RBSetIndex`, `InterlaceIndex`, `OCCI`, `SpreadingFactor`
`nrPUCCH3Config`	`Interlacing`, `RBSetIndex`, `InterlaceIndex`, `OCCI`, `SpreadingFactor`
`nrWavegenPUCCH0Config`	`Interlacing`, `RBSetIndex`, `InterlaceIndex`
`nrWavegenPUCCH1Config`	`Interlacing`, `RBSetIndex`, `InterlaceIndex`

Object	Interlacing Properties
nrWavegenPUCCH2Config	Interlacing, RBSetIndex, InterlaceIndex, OCCI, SpreadingFactor
nrWavegenPUCCH3Config	Interlacing, RBSetIndex, InterlaceIndex, OCCI, SpreadingFactor

- The nrPUCCHIndices function now also returns the PRBs allocated for PUCCH within the BWP.
- The nrPUCCHHoppingInfo function now enables you to specify the resource block indices in the interlace.
- Use the IntraCellGuardBands property of the nrULCarrierConfig and nrCarrierConfig objects to specify intracell guard bands for uplink transmissions.
- The nrIntraCellGuardBandsConfig object enables you to configure intracell guard bands for uplink transmissions.

Release 16 updates for low PAPR sequences and DM-RS scrambling identities for PUCCH formats 3 and 4

Use the DMRSUplinkTransformPrecodingR16 and NID0 properties of the nrPUCCH3Config, nrPUCCH4Config, nrWavegenPUCCH3Config, and nrWavegenPUCCH4Config objects to specify type 2 low peak-to-average power ratio (PAPR) PUCCH demodulation reference signal (DM-RS) sequences and DM-RS scrambling identities, as defined in Release 16 of TS 38.211 Section 6.4.1.3.3.1.

CDL channel model updates

The nrCDLChannel System object now enables you to:

- Customize a CDL channel model that is initialized with a predefined delay profile.
- Use the MaximumDopplerShift and UTDirectionOfTravel object properties to configure V2X fading channels for the transmitter and receiver. To control clutter mobility, use the MovingScattererProportion and MaximumScattererSpeed object properties.

The channelInfo output argument of the info object function now returns these two additional fields when the channel input is an nrCDLChannel object.

- ClusterAngleSpreads — Cluster-wise root mean square (RMS) angle spreads
- XPR — Cross-polarization power ratio

Enhanced support for subband PDSCH precoding

The functions in this table provide support for physical downlink shared channel (PDSCH) precoding resource block group (PRG) bundling, as defined in TS 38.214 Section 5.1.2.3. These functions replace the example-specific implementations in previous releases.

Function	Description
nrPDSCHPrecode	Precoding for PDSCH PRG bundling
nrPRGInfo	PRG-related information

Reference simulation to measure NR V2X Sidelink PSCCH and PSSCH throughput

Use the NR V2X Sidelink PSCCH and PSSCH Throughput reference simulation to measure the link-level throughput and BLER of the NR V2X physical sidelink control channel (PSCCH) and physical sidelink shared channel (PSSCH), including sidelink control information (SCI) and SL-SCH transport channel coding.

UAV tracking using 5G PRS

The Vehicle Tracking for Urban Air Mobility Using 5G Position Reference Signal example shows how to track unmanned aerial vehicles (UAV) in urban air mobility scenarios using a 5G-specific positioning reference signal (PRS) and a tracking algorithm. The example then compares the positional estimate of the UAVs with the raw estimates obtained using the multilateration technique.

Measure the PDSCH throughput of 5G NR link in NTN channel

The NR NTN PDSCH Throughput example shows how to measure the PDSCH throughput of a 5G NR link in a nonterrestrial network (NTN) channel and perform the receiver-end Doppler compensation in the throughput measurement. Additionally, this example models a power amplifier with or without memory and static or time-varying propagation delays.

Accelerate end-to-end simulation with frequency-domain channel modeling

The Accelerate End-to-End Simulation with Frequency-Domain Channel Modeling example shows how to accelerate a NR PDSCH or PUSCH end-to-end simulation by performing frequency-domain channel modeling using a CDL channel model.

O-RAN fronthaul CU-plane message generation updates

The Generate CU-Plane Messages for O-RAN Fronthaul Test example now provides support for:

- C-plane section type 3 messages for mixed numerology
- Creation of package summary information
- Ethernet MTU parameterization and application-level packet fragmentation
- C-plane timing advance (*Tcp_adv_dl*) parameterization
- Replicating data across multiple extended antenna-carrier identifiers (eAxC IDs)
- Saving CU-plane messages into packet capture next generation (PCAPNG) files

Exclude DC subcarrier from EVM computation

These examples now support direct current (DC) subcarrier puncturing and DC offset correction in the error vector magnitude (EVM) computation:

- EVM Measurement of 5G NR Downlink Waveforms with RF Impairments
- EVM Measurement of 5G NR PUSCH Waveforms

- 5G NR Waveform Acquisition and Analysis
- 5G NR Waveform Capture and Analysis Using Software-Defined Radio.

System-level simulation updates

The Communications Toolbox Wireless Network Simulation Library add-on now provides support for:

- **Full PHY processing in 5G nodes** — The `nrGNB` and `nrUE` objects now support full physical layer (PHY) processing in 5G nodes. You can use full PHY processing when modelling single-cell scenarios. For an example of full PHY processing in 5G nodes and how to switch between abstract and full PHY processing, see NR Cell Performance Evaluation with Physical Layer Integration.
- **Open-loop uplink transmit power control** — The `nrGNB` object now enables open-loop transmit power control in UE nodes by default, as defined in 3GPP TS 38.213 Section 7.1. Use the `configureULPowerControl` object function to configure the parameters related to uplink transmit power control.
- **PF and BestCQI scheduling strategies** — Use the `configureScheduler` object function of the `nrGNB` object to specify proportional fair (PF) or best channel quality indicator (BestCQI) scheduling strategies. For an example of using these scheduling strategies, see NR FDD Scheduling Performance Evaluation.
- **Random waypoint mobility in UE nodes** — Use the `addMobility` object function of the `nrUE` object to configure random waypoint mobility in UE nodes.
- **Listing 5G nodes added to a wireless network simulator** — Use the `Nodes` property of the `wirelessNetworkSimulator` object to obtain the list of 5G nodes (`nrGNB` and `nrUE` objects) that are added to the simulator.
- **Modeling inter-arrival time** — The Generate and Visualize FTP Application Traffic Pattern example now models the packet inter-arrival time, as defined in 3GPP TR 36.889.

Functionality being removed or changed

C/C++ code generation updates

Behavior change

For C/C++ code generation, in calls to `nrPolarDecode`, `nrBCHDecode`, `nrDCIDecode`, and `nrUCIDecode`, the limitation that the input argument `L` must be a compile-time constant has been removed, however, `L` must be less than or equal to 64.

R2023a

5G Waveform Generator app updates

You can now use the 5G Waveform Generator app to:

- Configure sounding reference signals (SRS) for partial frequency sounding, as defined in Release 17 of TS 38.211 Section 6.4.1.4. You can now specify the extended ranges for the number of symbols, symbol repetition, and transmission comb number and offset.
- Configure the synchronization signal (SS) burst for frequency range 2-2 (FR2-2). The SS burst now supports Case F and Case G, corresponding to 480 kHz and 960 kHz subcarrier spacing, respectively, as defined in TS 38.213 Section 4.1.
- Configure NR uplink and downlink waveforms with subcarrier spacing and channel bandwidth for FR2-2, as defined in Release 17 of TS 38.211. You can now specify subcarrier spacing values 480 kHz or 960 kHz and channel bandwidth values 800, 1600, or 2000.
- Configure the resource block offset for the physical uplink shared channel (PUSCH) allocation in NR uplink FRC waveforms.
- Configure 1024QAM modulation for the physical downlink shared channel (PDSCH) allocation in NR downlink and NR downlink FRC waveforms.
- Configure NR-FR1-TM2b and NR-FR1-TM3.1b test models that use 1024QAM modulation.
- Configure 35 MHz and 45 MHz channel bandwidths for frequency range 1 (FR1).

You can now also use the app in MATLAB Online.

Release 17 updates for FR2-2

Specify subcarrier spacing and channel bandwidth for FR2-2 through these updated features:

- The `BlockPattern` property of the `nrWavegenSSBurstConfig` object now supports `'Case F'` and `'Case G'`, corresponding to 480 kHz and 960 kHz subcarrier spacings, respectively, as defined in TS 38.213 Section 4.1.
- The `SubcarrierSpacing` property of the `nrPRACHConfig`, `nrCarrierConfig`, `nrSCSCarrierConfig`, and `nrWavegenBWPConfig` objects now supports 480 kHz and 960 kHz subcarrier spacings, as defined in Release 17 of TS 38.211.
- The `nrOFDMDemodulate`, `nrOFDMInfo`, `nrOFDMModulate`, `nrPerfectChannelEstimate`, and `nrTimingEstimate` functions now support 480 kHz and 960 kHz subcarrier spacings.
- The `nrDLCarrierConfig` and `nrULCarrierConfig` objects now support 800 MHz, 1600 MHz, and 2000 MHz for FR2-2 and 480 kHz and 960 kHz subcarrier spacings.

Release 17 updates for partial frequency sounding SRS transmissions

The `nrSRSConfig` and `nrWavegenSRSConfig` objects and the `nrSRS` and `nrSRSIndices` functions now support SRS for partial frequency sounding, as defined in Release 17 of TS 38.211 Section 6.4.1.4.

- Use the `FrequencyScalingFactor`, `StartRBIndex`, and `EnableStartRBHopping` object properties of the `nrSRSConfig` and `nrWavegenSRSConfig` objects to configure SRS transmissions for partial frequency sounding.
- Use the `StartRBHoppingTable` read-only property of the `nrSRSConfig` object to display the frequency hopping offset indices for partial frequency sounding, as defined in TS 38.211 Table 6.4.1.4.3-3.

- For the number of symbols, symbol repetition, transmission comb number, and transmission comb offset, you can now specify the extended ranges that are defined in Release 17.

Release 17 updates for PUCCH

- You can now set the `PRBSet` property of the `nrPUCCH0Config`, `nrPUCCH1Config`, `nrPUCCH4Config`, `nrWavegenPUCCH0Config`, `nrWavegenPUCCH1Config`, and `nrWavegenPUCCH4Config` objects to a vector, as defined in Release 17 of TS 38.213 Section 9.2.1 and TS 38.211 Sections 6.3.2.3, 6.3.2.4, and 6.3.2.6.
- You can now specify the number of PUCCH resource blocks when calling the `nrPUCCH0`, `nrPUCCH1`, or `nrPUCCH4` functions.

Release 17 updates for PRACH

The `ActivePRACHSlot` property of the `nrPRACHConfig` object now supports the values 3, 7, and 15 for 480 kHz and 960 kHz subcarrier spacings, as defined in Release 17 of TS 38.211 Section 5.3.2.

Release 17 updates for DL-SCH and UL-SCH

The `nrDLSCH`, `nrDLSCHDecoder`, `nrULSCH`, and `nrULSCHDecoder` System objects now support up to 32 HARQ processes.

Release 17 updates for downlink 1024QAM for NR

You can now specify downlink 1024QAM modulation for NR through these functions and objects:

- `nrDLSCH` and `nrDLSCHDecoder` System objects
- `nrPDSCHConfig` and `nrWavegenPDSCHConfig` objects
- `nrPDSCH`, `nrPDSCHDecode`, `nrPDSCHIndices`, `nrRateMatchLDPC`, `nrRateRecoverLDPC`, `nrTBS`, `nrSymbolModulate`, and `nrSymbolDemodulate` functions

Release 17 updates for FR1

The `ChannelBandwidth` property of the `nrDLCarrierConfig` and `nrULCarrierConfig` objects now supports 35 MHz and 45 MHz for FR1.

Support for VRB in PDSCH

Use the `PRBSetType` property of the `nrPDSCHConfig` and `nrWavegenPDSCHConfig` objects to specify the resource allocation type. This property specifies whether the `PRBSet` property defines physical resource block (PRB) or virtual resource block (VRB) allocation. In previous releases, the `PRBSet` property specifies only PRB allocation.

Enhancements to nrRateRecoverPolar, nrUCIDecode,and nrDCIEncode

- The `nrRateRecoverPolar` and `nrUCIDecode` functions now perform soft combining of repeated received softbits when the number of softbits exceeds the length of the circular buffer.

- The nrDCIEncode function now also returns the RNTI-masked CRC bits of the DCI encoding process.

Enhanced CSI computation

- All featured examples that compute channel state information (CSI) parameters now support enhanced type II codebooks.
- The NR PDSCH Throughput Using Channel State Information Feedback example now supports artificial intelligence-based CSI encoding that enables you to experiment with CSI techniques beyond the 5G standard.

5G NR waveform capture and analysis using software-defined radio

The 5G NR Waveform Capture and Analysis Using Software-Defined Radio example shows how to generate and transmit a standard-compliant 5G NR waveform by using the 5G Waveform Generator app and a software-defined radio (SDR). The example then captures the transmitted over-the-air waveform with an SDR and performs error vector magnitude (EVM) analysis of the waveform.

TDD reciprocity-based PDSCH MU-MIMO using SRS with JSDM

The TDD Reciprocity-Based PDSCH MU-MIMO Using SRS example now provides the option to perform joint spatial division multiplexing (JSDM) beamforming using Phased Array System Toolbox™ functions.

System-level simulation updates

Network simulation functionality moved to Communications Toolbox Wireless Network Simulation Library

These objects have moved to the Communications Toolbox Wireless Network Simulation Library add-on. To use these features, you must download and install the add-on from File Exchange.

- networkTrafficOnOff
- networkTrafficVoIP
- networkTrafficVideoConference
- networkTrafficFTP

Enhanced support for wireless network simulation, gNB, and UE

The objects listed in this table replace the example-specific implementation in previous releases.

Object	Description	Compatibility Consideration
wirelessNetworkSimulator	Simulate a multinode 5G network of gNB nodes and UEs.	**Note** To use these features, you must download and install the Communications Toolbox Wireless Network Simulation Library add-on from File Exchange.
nrGNB	Create a 5G NR base station (gNB) with medium access control (MAC), radio link control (RLC), and an abstracted physical layer (PHY).	

Object	Description	Compatibility Consideration
nrUE	Create 5G NR user equipment (UE) with MAC, RLC, and an abstracted PHY.	
nrRLCBearerConfig	Configure RLC bearer between a gNB and UE.	

Evaluate performance of UMi, UMa, and RMa scenarios

The NR Interference Modeling with Toroidal Wrap-Around example now shows how to evaluate the performance of the urban micro (UMi), urban macro (UMa), and rural macro (RMa) scenarios that are defined in TR 38.901.

Evaluate NR cell performance with downlink MU-MIMO

The NR Cell Performance with Downlink MU-MIMO example shows how to evaluate the system performance of a codebook-based downlink multi-user (MU) multiple-input multiple-output (MIMO).

C/C++ code generation updates

- For C/C++ code generation, in calls to `nrWaveformGenerator(cfg)`:
  - The limitations imposed on the `cfg.WindowingPercent` and `cfg.CarrierFrequency` object properties have been removed.
  - The limitation imposed on the `cfg.SampleRate` object property has been updated. When you set the `SampleRate` property of the configuration object to a value that equals SCS times the FFT size, the FFT size can now be any positive integer, not just a power of 2.
- For C/C++ code generation, the limitations imposed on the name-value arguments listed in this table have been removed. You do not need to specify these name-value arguments as compile-time constants for the specified functions. Some limitations, however, still apply when you specify the `'SampleRate'` and `'Nfft'` name-value arguments in the same call. For more details, see the corresponding notes.

Function	Name-Value Argument
nrChannelEstimate	'CyclicPrefix', 'CDMLengths', 'AveraginWindow'
nrCSIRSIndices	'IndexStyle', 'IndexBase'
nrExtractResources	'IndexStyle', 'IndexBase'
nrLDPCDecode	'OutputFormat', 'Algorithm', 'ScalingFactor', 'Offset', 'Termination'
nrOFDMModulate	'CyclicPrefix', 'Windowing', 'CarrierFrequency', 'Nfft' **Note** If you specify the 'SampleRate' name-value argument, the 'Nfft' name-value argument must still be compile-time constant for code generation.

Function	Name-Value Argument
nrOFDMInfo	'CyclicPrefix', 'Windowing', 'CarrierFrequency', 'Nfft', 'SampleRate'
nrOFDMDemodulate	'CyclicPrefix', 'Windowing', 'CarrierFrequency', 'Nfft', 'CyclicPrefixFraction' **Note** If you specify the 'SampleRate' name-value argument, the 'Nfft' name-value argument must still be compile-time constant for code generation.
nrPBCHDMRSIndices	'IndexStyle', 'IndexBase'
nrPBCHIndices	'IndexStyle', 'IndexBase'
nrPDCCHResources	'IndexStyle', 'IndexBase', 'IndexOrientation'
nrPDCCHSpace	'IndexStyle', 'IndexBase', 'IndexOrientation'
nrPDSCHDMRSIndices	'IndexStyle', 'IndexBase', 'IndexOrientation'
nrPDSCHIndices	'IndexStyle', 'IndexBase', 'IndexOrientation'
nrPDSCHPTRSIndices	'IndexStyle', 'IndexBase', 'IndexOrientation'
nrPerfectChannelEstimate	'CyclicPrefix', 'CyclicPrefixFraction', 'Nfft' **Note** If you specify the 'SampleRate' name-value argument, the 'Nfft' name-value argument must still be compile-time constant for code generation.
nrPRACHDetect	'DetectionThreshold', 'PreambleIndex'
nrPRACHIndices	'IndexStyle', 'IndexBase'
nrPRSIndices	'IndexStyle', 'IndexBase'
nrPSSIndices	'IndexStyle', 'IndexBase'
nrPUCCHDMRSIndices	'IndexStyle', 'IndexBase', 'IndexOrientation'
nrPUCCHIndices	'IndexStyle', 'IndexBase', 'IndexOrientation'
nrPUSCHDMRSIndices	'IndexStyle', 'IndexBase', 'IndexOrientation'
nrPUSCHIndices	'IndexStyle', 'IndexBase', 'IndexOrientation'
nrPUSCHPTRSIndices	'IndexStyle', 'IndexBase', 'IndexOrientation'
nrSSSIndices	'IndexStyle', 'IndexBase'

Function	Name-Value Argument
nrSRSIndices	'IndexStyle', 'IndexBase'
nrTimingEstimate	'CyclicPrefix', 'CarrierFrequency, 'Nfft' **Note** If you specify the 'SampleRate' name-value argument, the 'Nfft' name-value argument must still be compile-time constant for code generation.

Functionality being removed or changed

nrPDSCHConfig and nrWavegenPDSCHConfig default to VRB allocation
Behavior change

The PRBSet property of the nrPDSCHConfig and nrWavegenPDSCHConfig objects now specifies VRB allocation by default. In previous releases, this property specifies only PRB allocation. Use the PRBSetType object property to specify whether the PRBSet property defines PRB or VRB allocation.

R2022b

Version: 2.5

New Features

Bug Fixes

5G Waveform Generator app updates

You can now use the 5G Waveform Generator app to:

- Initialize physical uplink shared channel (PUSCH) RA-RNTI scrambling for *msgA* on PUSCH during a two-step random access channel (RACH) procedure, as defined in Release 16 of TS 38.211 Section 6.3.1.1.
- Enable low peak-to-average-power ratio (PAPR) physical downlink shared channel (PDSCH) demodulation reference signal (DM-RS) sequences, as defined in Release 16 of TS 38.211 Section 7.4.1.1.
- Enable low PAPR PUSCH DM-RS sequences, as defined in Release 16 of TS 38.211 Section 6.4.1.1.
- Configure the frequency position of the control resource set (CORESET) by setting the RB offset.
- Configure physical downlink control channel (PDCCH) DM-RS transmission in the entire CORESET by setting the precoder granularity.
- Configure the number of subframes in an NR test model (NR-TM).
- Specify a different power level for each synchronization signal (SS) block in a burst.
- Configure an additional channel or signal in a 5G NR downlink or uplink waveform by customizing a duplicate copy of an existing channel or signal configuration.
- Open a copy of an NR-TM, downlink FRC, or uplink FRC preset definition as a 5G NR downlink or uplink waveform configuration. Use this feature to customize the standard-defined preset definitions.
- Transmit your waveforms over the air at full radio device rate. For a list of radios that support full device rates, see Supported Radio Devices (Wireless Testbench). This feature requires Wireless Testbench. For an example, see Transmit App-Generated Wireless Waveform Using Radio Transmitters.

TS 38.101 HST channel modeling

Use the `nrHSTChannel` System object to send an input signal through a high-speed train (HST) channel to obtain the channel-impaired signal. The object implements the HST single tap, HST-DPS, and HST-SFN channel profiles from TS 38.101-4 Annex B.

CDL and TDL channel model updates

The `info` function of the `nrCDLChannel` and `nrTDLChannel` System objects now also returns the maximum channel delay. This delay consists of the maximum path delay and the channel filter delay.

Support for O-RAN fronthaul CU-plane message generation

Use the `nrORANBlockCompress` and `nrORANBlockDecompress` functions to perform O-RAN fronthaul block compression and decompression. The functions implement the block floating point (BFP), block scaling, and μ-law compression and decompression methods, as defined in TS O-RAN.WG4.CUS Annex A.1, A.2, and A.3.

For an example of how to generate fronthaul control and user plane (CU-Plane) messages for O-RAN conformance tests, as defined in TS O-RAN.WG4.CONF, see Generate CU-Plane Messages for O-RAN Fronthaul Test.

CORESET enhancements

The `nrCORESETConfig` configuration object now enables you to specify the precoder granularity and RB offset through the `PrecoderGranularity` and `RBOffset` object properties, respectively.

SS burst configuration updates

The `Power` property of the `nrWavegenSSBurstConfig` object now enables you to specify a different power level for each SS block in a burst. You can use this feature to model the power level effect of beam sweeping on different blocks in the burst.

Enhanced support for RSRP, RSSI, and RSRQ measurements

- Use the `nrCSIRSMeasurements` function to measure the channel state information (CSI) reference signal received power (CSI-RSRP), the CSI received signal strength indicator (CSI-RSSI), and the CSI reference signal received quality (CSI-RSRQ), as defined in TS 38.215 Sections 5.1.2 and 5.1.4.
- Use the `nrSSBMeasurements` function to measure the synchronization signal (SS) reference signal received power (SS-RSRP), the SS received signal indicator (SS-RSSI), and the SS reference signal received quality (SS-RSRQ), as defined in TS 38.215 Sections 5.1.1 and 5.1.3.

These functions replace the example-specific implementation in previous releases.

Enhanced support for PRACH detection

- Use the `nrPRACHDetect` function to detect the physical random access channel (PRACH). This function replaces the example-specific implementation in previous releases.
- The 5G NR PRACH Detection and False Alarm Test example now enables you to run a false alarm conformance test, as defined in TS 38.141-1.

Reference simulation to measure NR PDSCH throughput using CSI feedback

Use the NR PDSCH Throughput Using Channel State Information Feedback reference simulation to measure the PDSCH throughput of a 5G NR adaptive link. The simulation adjusts the modulation and coding scheme (MCS), precoding matrix, and number of layers based on the CSI feedback that the UE reports.

EVM measurement updates

- The EVM Measurement of 5G NR Downlink Waveforms with RF Impairments example now shows how to estimate the I/Q imbalances in a 5G NR PDSCH waveform that contains PDCCH.
- The EVM Measurement of 5G NR PUSCH Waveforms example now shows how to estimate the I/Q imbalances of a 5G NR PUSCH waveform.

Type II codebooks for CSI computation

The 5G NR Downlink CSI Reporting example now supports type II codebooks for the computation of CSI parameters.

Train deep Q-network (DQN) agent for beam selection

The Train DQN Agent for Beam Selection example shows how to train a DQN reinforcement learning agent to increase beam selection accuracy in a 5G NR communications system by selecting the beam with the highest signal strength while reducing beam transition cost.

Compress CSI feedback with autoencoder neural network

The CSI Feedback with Autoencoders example shows how to use an autoencoder neural network to compress CSI feedback in a 5G NR communications system. This approach eliminates the use of a quantized codebook and can improve overall system performance.

Simulate 5G LDPC BLER using the cloud or a cluster

The 5G LDPC Block Error Rate Simulation Using the Cloud or a Cluster example shows how to use the cloud or a cluster for block error rate (BLER) simulation of low-density parity-check (LDPC) coding for the 5G NR downlink shared transport channel (DL-SCH).

System-level simulation updates

Use the `nrMACPDUDecode` function to decode the medium access control (MAC) protocol data units (PDUs).

R2022a

5G Waveform Generator app updates

The 5G Waveform Generator app now provides these capabilities.

- OFDMA channel noise generator (OCNG) for downlink FRC waveform generation, as defined in Release 16 of TS 38.101 Annex A.5.
- Resource grid visualization of conflicts across channels and signals.
- Instant resource grid and channel view plot updates during configuration of NR-TM and NR uplink and downlink FRC waveforms.
- Resource element (RE) mapping visualization. Use the RE mapping plot to visualize reference signal allocation within the resource block that contains the selected channel. You can visualize demodulation reference signal (DM-RS), phase tracking reference signal (PT-RS), channel state information reference signal (CSI-RS), and sounding reference signal (SRS) resources.
- NR waveform transmission over the air by using a software-defined radio (SDR) hardware. To transmit a waveform on one of the supported SDRs (ADALM-Pluto, USRP™, USRP embedded series, and Xilinx® Zynq-based radios), you must install the associated add-on. For more information, see Transmit Using SDR.

Enhanced support for TR 38.901 channel model

- The `nrPathLossConfig` and `nrPathLoss` features now support the indoor factory with high Tx and high Rx (InF-HH) scenario, as defined in TR 38.901 Table 7.2-4.
- The `nrCDLChannel` System object now enables you to model the fast fading channel, as defined in TR 38.901 Section 7.5, using the `XPR`, `RayCoupling`, and `InitialPhases` object properties.

Release 16 updates to PUSCH scrambling

These features now support physical uplink shared channel (PUSCH) RA-RNTI scrambling initialization for *msgA* on PUSCH during a two-step random access channel (RACH) procedure, as defined in Release 16 of TS 38.211 Section 6.3.1.1.

- `nrPUSCHConfig` and `nrWavegenPUSCHConfig` objects through the `NRAPID` object property
- `nrPUSCH` and `nrPUSCHDecode` functions through the `NRAPID` object property of the `pusch` input argument
- `nrPUSCHScramble`, `nrPUSCHDescramble`, and `nrPUSCHPRBS` functions through the optional `nrapid` input argument

Release 16 updates to PDSCH DM-RS and PUSCH DM-RS

PDSCH DM-RS

- Use the `DMRSDownlinkR16` property of the `nrPDSCHDMRSConfig` object to enable low peak-to-average-power ratio (PAPR) physical downlink shared channel (PDSCH) DM-RS sequences, as defined in Release 16 of TS 38.211 Section 7.4.1.1.
- The `NIDNSCID` property of the `nrPDSCHDMRSConfig` object now supports dynamic ID selection.
- PDSCH DM-RS symbol generation now supports type B mapping DM-RS symbol positions, as defined in Release 16 of TS 38.211 Tables 7.4.1.1.2-3 and 7.4.1.1.2-4.

PUSCH DM-RS

- Use the DMRSUplinkR16 and DMRSUplinkTransformPrecodingR16 properties of the nrPUSCHDMRSConfig object to enable low PAPR PUSCH DM-RS sequences, as defined in Release 16 of TS 38.211 Section 6.4.1.1.
- The NIDNSCID property of the nrPUSCHDMRSConfig object now supports dynamic ID selection.

Enhancements to PDCCH support

The nrPDCCHConfig and nrWavegenPDCCHConfig objects now enable you to position the physical downlink control channel (PDCCH) in the associated control resource set (CORESET) by using the CCEOffset property of the objects.

Enhanced computational performance in end-to-end simulations

Internal optimizations of the nrRateMatchLDPC, nrRateRecoverLDPC, nrEqualizeMMSE, and nrExtractResources functions now enable faster end-to-end simulations. The achievable speedup depends on the configuration parameters of the simulation. For example, running the simulation in the NR PDSCH Throughput example with 1000 subframes is on average 30% faster in comparison to previous releases.

Dynamic spectrum sharing for 5G NR and LTE coexistence

The Dynamic Spectrum Sharing for 5G NR and LTE Coexistence example shows how to generate coexisting LTE and 5G NR waveforms for dynamic spectrum sharing.

Model RF impairments in MATLAB

The EVM Measurement of 5G NR Downlink Waveforms with RF Impairments example shows how to model and analyze RF impairments in MATLAB, including phase noise, in-phase and quadrature (I/Q) imbalance, filter effects, and memoryless nonlinearity.

EVM measurement enhancements

The EVM Measurement of 5G NR Downlink Waveforms with RF Impairments example shows how to:

- Measure the error vector magnitude (EVM) of the PDCCH in addition to the PDSCH.
- Estimate and correct for carrier frequency offsets.

The EVM Measurement of 5G NR PUSCH Waveforms example now shows how to estimate and correct the carrier frequency offset of 5G NR PUSCH waveforms.

Neural network for beam selection deep learning example

The Neural Network for Beam Selection example shows how to use a neural network to reduce the overhead in the beam selection task by using only the location of the receiver rather than the knowledge of the communication channels.

System-level simulation updates

- Use the `nrMACSubPDU` function to generate a medium access control (MAC) subprotocol data unit (subPDU) for uplink or downlink direction.
- Use the `nrMACBSRDecode` function to decode the buffer status report (BSR) control element (CE) and receive logical channel group (LCG) IDs and the buffer size range.
- The NR Cell Performance Evaluation with Beam Management example shows how to beamform the downlink data based on the layer-1 reference signal received power (L1-RSRP) feedback from the user equipment (UE).

Functionality being removed or changed

SR bit output value in PUCCH format 0 decoding has changed
Behavior change

For physical uplink control channel (PUCCH) format 0, the `nrPUCCHDecode` function now returns `0` instead of `[]` for the scheduling request (SR) bit value when the detection metric is below the discontinuous transmission (DTX) detection threshold.

Detection threshold in PUCCH format 1 decoding has changed
Behavior change

For PUCCH format 1, the `nrPUCCHDecode` function now uses 0.22 instead of 0.36 as the default DTX detection threshold.

nrWavegenCSIRSConfig object enables CSI-RS by default
Behavior change

The `Enable` property of the `nrWavegenCSIRSConfig` object now defaults to `1` instead of `0`. As a result, the `nrWavegenCSIRSConfig` object now enables CSI-RS by default. To preserve the behavior of the `nrDLCarrierConfig` object from previous releases, in which CSI-RS is disabled by default, the `CSIRS` property of the `nrDLCarrierConfig` object now defaults to `{nrWavegenCSIRSConfig('Enable',0)}` instead of `{nrWavegenCSIRSConfig}`.

R2021b

5G Waveform Generator app updates

The 5G Waveform Generator app now provides these capabilities.

- Export waveforms to a Simulink® block. Use the generated block as a waveform source in a Simulink model. For more information, see the Waveform From Wireless Waveform Generator App block.
- Export NR test models (NR-TM) and NR uplink and downlink FRC waveform generation parameters to a MATLAB script. Use the exported scripts to programmatically generate waveforms from the command line.
- Physical uplink control channel (PUCCH) support.
- Sounding reference signal (SRS) for positioning support, as defined in Release 16 of TS 38.211 Section 6.4.1.4. This support includes the extended ranges for the starting position of the SRS within a slot, number of SRS symbols, symbol repetition, and comb number and offset.
- Complementary cumulative distribution function (CCDF) visualization to measure the peak-to-average power ratio (PAPR) of the generated waveform. The CCDF visualization also supports time-duplexed waveforms and signal bursts when you select burst mode in the CCDF plot.
- Control resource set (CORESET) visualization in the resource grid when configuring NR downlink, NR-TM, or NR downlink FRC waveforms.

Enhanced PUCCH support in programmatic waveform generation

The `nrULCarrierConfig` object now enables you to configure 5G NR uplink waveforms containing PUCCH by using the `PUCCH` property of the object. You can use the objects listed in this table for the PUCCH configuration. This enhanced capability replaces the example-specific implementation in previous releases.

Object	Description
`nrWavegenPUCCH0Config`	PUCCH format 0 configuration parameters for 5G waveform generation
`nrWavegenPUCCH1Config`	PUCCH format 1 configuration parameters for 5G waveform generation
`nrWavegenPUCCH2Config`	PUCCH format 2 configuration parameters for 5G waveform generation
`nrWavegenPUCCH3Config`	PUCCH format 3 configuration parameters for 5G waveform generation
`nrWavegenPUCCH4Config`	PUCCH format 4 configuration parameters for 5G waveform generation

Enhanced support for PUCCH decoding

Use the `nrPUCCHDecode` function to decode PUCCH formats 0, 1, 2, 3, and 4 and to calculate the detection metric. This function replaces the example-specific implementation in previous releases.

Release 16 updates to PRACH generation

- The `nrPRACH`, `nrPRACHIndices`, and `nrPRACHConfig` features now support physical random access channel (PRACH) preamble lengths 1151 and 571, as defined in Release 16 of TS 38.211 Section 6.3.3.
- You can now set the `LRA` property of the `nrPRACHConfig` configuration object when you set the `SubcarrierSpacing` property to `15` or `30`.
- The upper bound of the `ConfigurationIndex` property of the `nrPRACHConfig` configuration object is now 262 instead of 255 for frequency range 1 (FR1) and time division duplex (TDD) mode.

Support for TR 38.901 propagation path loss

Use the `nrPathLossConfig` object and the `nrPathLoss` function to configure and calculate the propagation path loss between a base station (BS) and user equipment (UE), as defined in TR 38.901 Section 7.4.1.

- The `nrPathLossConfig` object configures rural macrocell (RMa), urban macrocell (UMa), urban microcell (UMi), indoor hotspot (InH), or indoor factory (InF) scenarios.
- The `nrPathLoss` function calculates the path loss between a BS and UE in line-of-sight (LOS) or non-LOS at a carrier frequency.

The Include Path Loss in NR Link-Level Simulations example shows how to analyze the performance impact of path loss, transmit power, and receive noise in 5G NR link-level simulations.

TDL and CDL channel model updates

- The `nrTDLChannel` System object now supports simplified channel profiles. Configure the tapped delay line (TDL) delay profiles from TS 38.101-4 Annexes B.2.1.1, B.2.1.2 and TS 38.104 Annexes G.2.1.1, G.2.1.2 by specifying `'TDLA30'`, `'TDLB100'`, `'TDLC300'`, or `'TDLC60'` for the `DelayProfile` property of the object.
- You can now disable channel filtering in the `nrTDLChannel` System object. Generate channel path gains without filtering a waveform by setting the `ChannelFiltering` property of the object to `false`. The `NumTimeSamples` and `OutputDataType` properties specify the duration and output data type of the path gains when you disable channel filtering.
- The `nrCDLChannel` System object now enables you to specify the output data type of the generated path gains by using the `OutputDataType` object property when you disable channel filtering.

TDD reciprocity-based PDSCH MU-MIMO using SRS

The TDD Reciprocity-Based PDSCH MU-MIMO Using SRS example shows how to exploit channel reciprocity in a TDD scenario to calculate physical downlink shared channel (PDSCH) beamforming weights for multiuser multiple-input multiple-output (MU-MIMO).

Modeling downlink control information formats

The NR Downlink Control Information Formats example shows how to use MATLAB classes to represent downlink control information (DCI) formats and how to encode and decode DCI bit payloads.

RI computation

The 5G NR Downlink CSI Reporting example now shows how to compute the rank indicator (RI) over a TDL channel. The example also shows how to compute channel state information (CSI) parameters using type 1 multipanel codebooks.

5G NR waveform acquisition and analysis

The 5G NR Waveform Acquisition and Analysis example shows how to generate a 5G NR test model (NR-TM) waveform using the 5G Waveform Generator app and download the generated waveform to a Keysight® vector signal generator for over-the-air transmission (requires Instrument Control Toolbox). The example then captures the transmitted over-the-air signal using a Keysight signal analyzer and analyzes the signal in MATLAB.

NR RF transmitter and receiver updates

The NR RF models in the Modeling and Testing an NR RF Transmitter and Modeling and Testing an NR RF Receiver with LTE Interference examples now use test models that are configured and exported from the 5G Waveform Generator app.

UE positioning within network of gNBs

The NR Positioning Using PRS example shows how to calculate the position of the user equipment (UE) within a network of gNodeBs (gNBs) using NR positioning reference signals (PRSs).

System-level simulation updates

- Use the `nrPCAPWriter` object to write 5G NR MAC packets to PCAP or PCAPNG files by encapsulating the packets into a pseudo protocol with a link type.
- Use the `pcapReader` object to read a PCAP file and filter packets and to decode protocol packets. The object enables you to decode Ethernet and eCPRI protocol packets and to plug in custom protocol decoders.
- Use the `nrMACBSR` function to generate a regular, periodic, or padding buffer status report (BSR) control element (CE).
- The NR Cell Performance Evaluation with MIMO example now shows how to model a 5G NR cell with MIMO antenna configurations in the uplink direction by using the SRS. This example also shows how to import and use ray-trace results in the system-level simulation.
- The NR Intercell Interference Modeling example now shows how to model the effects of intracell mobility of UEs on the downlink performance of a 5G NR network with multiple cells.
- The Plug In Custom Scheduler in System-Level Simulation example shows how to create and integrate a custom scheduling strategy into system-level simulation framework and observe the network performance.

R2021a

App support for NR uplink and downlink vector waveform generation

The 5G Waveform Generator app now enables you to parameterize, visualize, and generate NR uplink and downlink vector waveforms. You can export the generated waveforms to your workspace or to a `.mat` or `.bb` file. Alternatively, you can export waveform generation parameters to a runnable MATLAB script that you can use to programmatically generate your waveform from the command line, without the app. You can also transmit the generated waveforms over the air by using signal-generator instruments. The use of the transmit feature requires the Instrument Control Toolbox product.

Enhanced support for programmatic 5G NR UL waveform generation

The `nrWaveformGenerator` function now enables you to generate 5G NR uplink waveforms. You can now use the objects listed in this table to configure uplink waveform generation. This enhanced capability replaces the example-specific implementation in previous releases.

Object	Description
`nrULCarrierConfig`	5G uplink waveform configuration parameters
`nrSCSCarrierConfig`	Subcarrier spacing (SCS) carrier configuration parameters
`nrWavegenBWPConfig`	Bandwidth part configuration parameters for 5G waveform generation
`nrWavegenPUSCHConfig`	Physical uplink shared channel (PUSCH) configuration parameters for 5G waveform generation (including uplink control information (UCI) and Release 16 CG-UCI parameters)
`nrWavegenSRSConfig`	Sounding reference signal (SRS) configuration parameters for 5G waveform generation (including Release16 SRS for positioning parameters)

For an example of how to use these features to configure and create mixed numerology uplink waveforms containing PUSCH and SRS, see 5G NR Uplink Vector Waveform Generation.

Support for PRS

Use the features listed in these tables to configure and generate positioning reference signal (PRS) symbols and resource element indices.

Function	Description
`nrPRS`	Generate PRS symbols
`nrPRSIndices`	Generate PRS resource element indices

Object	Description
`nrPRSConfig`	PRS configuration parameters

For an example of how to learn the effects of different PRS resource set configurations on the time-frequency structure of PRS resources, see NR Positioning Reference Signal.

Enhanced support for PUCCH

Use the features listed in these tables to model the physical uplink control channel (PUCCH) formats 0, 1, 2, 3, and 4 and the corresponding demodulation reference signals (DM-RS). These enhancements replace the example-specific implementation in previous releases.

Function	Description
`nrPUCCH`	Generate PUCCH modulation symbols
`nrPUCCHIndices`	Generate PUCCH resource element indices
`nrPUCCHDMRS`	Generate PUCCH DM-RS symbols
`nrPUCCHDMRSIndices`	Generate PUCCH DM-RS resource element indices

Object	Description
`nrPUCCH0Config`	PUCCH format 0 configuration parameters
`nrPUCCH1Config`	PUCCH format 1 configuration parameters
`nrPUCCH2Config`	PUCCH format 2 configuration parameters
`nrPUCCH3Config`	PUCCH format 3 configuration parameters
`nrPUCCH4Config`	PUCCH format 4 configuration parameters

For an example of how to measure the block error rate (BLER) of uplink control information (UCI) transmitted on the PUCCH in a 5G NR link, see NR PUCCH Block Error Rate.

Support for channel reciprocity in CDL and TDL channel models

Model a time division duplexing (TDD) operation while retaining channel reciprocity in the `nrCDLChannel` and `nrTDLChannel` channel models by using the `swapTransmitAndReceive` object function. Use the `TransmitAndReceiveSwapped` property of either objects to establish whether the channel model has its receive and transmit antennas swapped.

Support for custom antenna elements and arrays in CDL channel model (requires Phased Array System Toolbox)

You can now specify the `TransmitAntennaArray` and `ReceiveAntennaArray` properties of the `nrCDLChannel` channel model by using Phased Array System Toolbox antenna array objects. These antenna array objects enable you to specify different antenna array configurations, including predefined and custom antenna elements. You can design custom antenna elements by using Phased Array System Toolbox or Antenna Toolbox™ features.

To specify custom antenna elements in 5G rectangular multipanel arrays, as defined in TR 38.901 Section 7.3, use the `phased.NRRectangularPanelArray` (Phased Array System Toolbox) object.

PUSCH enhancements

- The `nrPUSCH`, `nrPUSCHDecode`, and `nrPUSCHIndices` functions now enable you to handle the phase tracking reference signal (PT-RS) within the processing of the physical uplink shared channel (PUSCH).

- The `nrPUSCHDecode` function now enables you to handle the UCI placeholder bit locations.

Release 16 updates to low-PAPR sequence generation

The `nrLowPAPRS` function now supports type 2 low peak-to-average power ratio (low-PAPR) sequence generation, as defined in Release 16 of TS 38.211 Section 5.2.3.

Release 16 updates to SRS generation

- The `nrSRS`, `nrSRSIndices`, and `nrSRSConfig` features now support SRS for positioning, as defined in Release 16 of TS 38.211 Section 6.4.1.4.
- Use the `SubcarrierOffsetTable` read-only property of the `nrSRSConfig` object to determine the SRS subcarrier offset for each OFDM symbol and to obtain valid combinations of the number of OFDM symbols and the transmission comb number.
- The range of the `SymbolStart` property of the `nrSRSConfig` object now starts from 0 instead of 6. You can transmit the SRS in any symbol of a slot.

TDD reciprocity-based PDSCH beamforming using SRS

The TDD Reciprocity-Based PDSCH Beamforming Using SRS example shows how to exploit channel reciprocity to calculate the physical downlink shared channel (PDSCH) beamforming weights in a TDD scenario. The beamforming weights calculation uses a channel estimate based on uplink SRS. Using these beamforming weights, the example uses the same channel for a downlink PDSCH transmission.

5G NR downlink carrier aggregation, demodulation, and analysis

The 5G NR Downlink Carrier Aggregation, Demodulation, and Analysis example shows how to generate, aggregate, and demodulate multiple 5G NR downlink carriers. To perform carrier aggregation (CA), calculate the frequency offsets for the intraband contiguous CA case, as described in TS 38.104 Section 5.3A. The example also supports customized intraband noncontiguous and interband CA scenarios.

EVM measurement of PUSCH FRC

Measure the standard-defined error vector magnitude (EVM) of physical uplink shared channel (PUSCH) fixed reference channel (FRC) waveforms. For more information, see the EVM Measurement of 5G NR PUSCH Waveforms example.

CDL channel model customization with ray tracing

The CDL Channel Model Customization with Ray Tracing example shows how to customize the CDL channel model parameters by using the output of a ray tracing analysis.

The example shows how to:

- Specify the location of a transmitter and a receiver in a 3-D environment
- Use ray tracing to calculate these geometric aspects of the channel: number of rays, angles, delays, and attenuations

- Configure the CDL channel model with the result of ray tracing analysis
- Specify the channel antenna arrays using Phased Array System Toolbox features
- Visualize the transmit and receive array radiation patterns based on singular value decomposition of a perfect channel estimate

Compute 5G NR CQI and PMI

Compute downlink channel state information (CSI) parameters, such as the channel quality indicator (CQI) and precoding matrix indicator (PMI) for MIMO scenarios with type-1 single panel codebooks over a tapped delay line (TDL) channel. For more information, see the 5G NR Downlink CSI Reporting example.

System-level simulation updates

Video conference application traffic pattern generation

Use the `networkTrafficVideoConference` object to generate a video conference application traffic pattern and model real-world data traffic in system-level simulations.

5G NR cell performance evaluation with configurable DM-RS parameters

You can now simulate and evaluate 5G NR cell performance by configuring DM-RS parameters. For more information, see the NR TDD Symbol Based Scheduling Performance Evaluation example.

5G NR cell performance evaluation with MIMO

Model a 5G NR cell with multiple-input multiple-output (MIMO) antenna configurations in downlink direction. You can customize the scheduling strategy to leverage the MIMO capabilities and evaluate the network performance. For more information, see NR Cell Performance Evaluation with MIMO.

5G NR cluster performance evaluation with toroidal wrap-around

The NR Interference Modeling with Toroidal Wrap-Around example shows how to evaluate the performance of a 19-cell cluster with toroidal wrap-around to remove edge effects.

Functionality being removed or changed

Subcarrier offset range in SS burst configuration has changed
Behavior change

The range of the `KSSB` property of the `nrWavegenSSBurstConfig` object for frequency range 1 (FR1) now depends on the value of the `SubcarrierSpacingCommon` property of that object.

- If `SubcarrierSpacingCommon` is `15`, specify `KSSB` as an integer from 0 to 11.
- If `SubcarrierSpacingCommon` is `30`, specify `KSSB` as an integer from 0 to 23.

Orientation field of antenna array properties in nrCDLChannel will be removed
Warns

- The `Orientation` field of the `TransmitAntennaArray` property of the `nrCDLChannel` System object will be removed in a future release. Use the `TransmitArrayOrientation` property of that object instead.

- The `Orientation` field of the `ReceiveAntennaArray` property of the `nrCDLChannel` System object will be removed in a future release. Use the `ReceiveArrayOrientation` property of that object instead.

R2020b

Enhanced support for 5G NR DL waveform generation

You can now use the features listed in these tables to generate 5G NR downlink (DL) waveforms. These features replace the example-specific implementation in previous releases.

Function	Description
`nrWaveformGenerator`	Generate 5G NR downlink waveform

Object	Description
`nrDLCarrierConfig`	5G downlink waveform configuration parameters
`nrSCSCarrierConfig`	Subcarrier spacing (SCS) carrier configuration parameters
`nrWavegenBWPConfig`	Bandwidth part configuration parameters for 5G waveform generation
`nrWavegenCSIRSConfig`	Channel state information reference signal (CSI-RS) configuration parameters for 5G waveform generation
`nrWavegenPDCCHConfig`	Physical downlink control channel (PDCCH) configuration parameters for 5G waveform generation
`nrWavegenPDSCHConfig`	Physical downlink shared channel (PDSCH) configuration parameters for 5G waveform generation
`nrWavegenSSBurstConfig`	Synchronization signal (SS) burst configuration parameters for 5G waveform generation

Additional enhancements used in 5G waveform generation

- The `nrCORESETConfig` object now provides a property to set a mnemonic description for the specified control resource set (CORESET) configuration.
- The `nrSearchSpaceConfig` object now provides properties to set a mnemonic description and ID for the specified search space set configuration.

Enhanced support for OFDM modulation and demodulation

You can now use the functions listed in this table to perform OFDM modulation and demodulation. These functions replace the example-specific implementations in previous releases.

Function	Description
`nrOFDMModulate`	Generate OFDM modulated waveform
`nrOFDMDemodulate`	Demodulate OFDM waveform
`nrOFDMInfo`	Get OFDM information
`nrResourceGrid`	Generate empty carrier slot resource grid
`nrPRACHOFDMModulate`	Generate physical random access channel (PRACH) OFDM modulated waveform
`nrPRACHOFDMInfo`	Get PRACH OFDM information

You can now use the `nrCarrierConfig` object as an input argument in the `nrChannelEstimate`, `nrPerfectChannelEstimate`, and `nrTimingEstimate` functions to specify OFDM-related parameters. In addition, `nrPerfectChannelEstimate` and `nrTimingEstimate` accept OFDM-related name-value pair arguments.

Enhanced support for UL-SCH data and control multiplexing

You can now use the functions listed in this table to perform uplink shared channel (UL-SCH) data and control multiplexing and demultiplexing. These functions replace the example-specific implementation in previous releases. For an example of how to configure and perform data and control multiplexing to form a codeword on the physical uplink shared channel (PUSCH), see NR UCI Multiplexing on PUSCH.

Function	Description
`nrULSCHMultiplex`	Perform UL-SCH data and control multiplexing
`nrULSCHDemultiplex`	Perform UL-SCH data and control demultiplexing

Additional enhancements

- The `nrULSCHInfo` function now provides the bit capacity and symbol capacity information of data and uplink control information (UCI).
- The `nrPUSCHConfig` object now contains properties to specify the parameters required to calculate the bit and symbol capacities of each UCI type when performing UCI multiplexing and demultiplexing.
- The `nrPUSCHDescramble` function now enables you to handle the UCI placeholder bit locations.

Support for CDL channel characteristics visualization

Visualize channel characteristics of the clustered delay line (CDL) channel model by using the `displayChannel` object function of the `nrCDLChannel` System object. The function displays geometric and electromagnetic characteristics of the CDL channel model at the transmitter and receiver ends. The visualization includes the position, polarization, and directivity radiation pattern of the antenna elements, cluster paths directions, and average path gains.

Enhanced support for TBS calculation

You can now use the `nrTBS` function to determine the transport block size (TBS) that is associated with each codeword in the shared channel transmission. This function replaces the example-specific implementation in previous releases.

PDCCH enhancements

If the higher-layer parameter *pdcch-DMRS-ScramblingID* is not configured, you can now also set the `DMRSScramblingID` property of the `nrPDCCHConfig` object to `[]`. The object then automatically sets the PDCCH DM-RS scrambling identity to the physical layer cell identity specified by the `NCellID` property of the carrier.

PRACH enhancements

- The `nrPRACHGrid` function now enables you to specify the output data type of the generated PRACH grid.

- The `nrPRACHConfig` object now provides read-only properties `SubframesPerPRACHSlot` and `PRACHSlotsPeriod` to determine the total number of subframes spanned by a nominal PRACH slot and the number of PRACH slots in a period.

Perform NR cell search and MIB and SIB1 recovery

Synchronize, demodulate, and decode a live gNodeB signal and decode the master information block (MIB) and the first system information block (SIB1). For more information, see NR Cell Search and MIB and SIB1 Recovery.

Perform subband PDSCH precoding

The NR PDSCH Throughput example now includes optional subband precoding of the PDSCH. When you enable subband precoding, the example selects a different beamforming matrix for each subband, which can increase performance for large bandwidths in frequency-selective channels.

Measure standard-defined EVM

Measure standard-defined error vector magnitude (EVM) of NR test model (NR-TM) or fixed reference channel (FRC) waveforms. For more information, see EVM Measurement of 5G NR PDSCH Waveforms.

Model phase noise and CPE compensation

Model the impact of phase noise in a 5G OFDM system and use the phase tracking reference signal (PT-RS) in compensating the common phase error (CPE). Measure the error vector magnitude (EVM) and bit error rate (BER) with and without CPE compensation. For more information, see NR Phase Noise Modeling and Compensation.

Employ beam sweeping and beam refinement procedures

Employ beam sweeping at both the transmitter (gNB) and receiver (UE) ends of a 5G NR system by using synchronization signal blocks (SSB) for beam management procedures during initial access. For more information, see NR SSB Beam Sweeping.

Perform beam refinement procedure at the downlink transmitter end of a 5G NR system by using CSI-RS. You can transmit multiple CSI-RS resources in different directions in a scattering environment and select the optimal transmit beam based on reference signal received power (RSRP) measurements. For more information, see NR Downlink Transmit-End Beam Refinement Using CSI-RS.

System-level simulation updates

Write MAC packets to PCAP file

Use the objects listed in this table to write generated and recovered 5G NR MAC packets to a file.

Object	Description
`pcapWriter`	PCAP file writer of protocol packets

Object	Description
`pcapngWriter`	PCAPNG file writer of protocol packets

Generate application traffic patterns

Use the objects listed in this table to generate application traffic patterns in 5G system-level simulations to accurately model real-world data traffic.

Object	Description
`networkTrafficFTP`	File transfer protocol (FTP) application traffic pattern generator
`networkTrafficOnOff`	On-Off application traffic pattern generator
`networkTrafficVoIP`	Voice over Internet protocol (VoIP) application traffic pattern generator

For an example of how to generate an FTP application traffic pattern based on the IEEE 802.11ax-Evaluation-Methodology and 3GPP TR 36.814 specification, see Generate and Visualize FTP Application Traffic Pattern.

Evaluate performance of 5G NR cell with MAC, physical layer, and channel modeling

Evaluate the performance of a 5G New Radio (NR) cell by modeling a gNB and multiple UE devices. You can simulate gNB and UE devices with application, radio link control (RLC), medium access control (MAC), and physical layers of the protocol stack along with channel model. For more information, see NR Cell Performance Evaluation with Physical Layer Integration.

Model 5G NR intercell interference

Evaluate the performance of 5G NR network with multiple cells by modeling the effect of intercell interference in downlink. For more information, see NR Intercell Interference Modeling.

Functionality being removed or changed

DM-RS reference point updates for CORESET ID 0
Behavior change

The reference point for the DM-RS sequence-to-subcarrier resource mapping for CORESET ID 0 is now the lowest physical resource block of the CORESET instead of the common resource block 0 (CRB0). This update affects the PDCCH and PDCCH DM-RS resources for CORESET ID 0 that you generate with the `nrPDCCHSpace` and `nrPDCCHResources` functions. For all other CORESET ID values, the reference point remains CRB0.

Updates to upper limit of UE identifier in PDCCH encoding
Behavior change

For the `nrPDCCH`, `nrPDCCHDecode`, and `nrPDCCHPRBS` functions, the upper limit of UE identifier `nrnti` is now 65,519 instead of 65,535.

R2020a

App support for NR-TM, NR uplink FRC, and NR downlink FRC waveform generation

Use the 5G Waveform Generator app to generate NR test models (NR-TM) and NR uplink and downlink fixed reference channel (FRC) waveforms.

Support for PRACH

Use these functions and this object to model the physical random access channel (PRACH).

Function	Description
nrPRACH	Generate PRACH symbols
nrPRACHIndices	Generate PRACH resource element indices
nrPRACHGrid	Generate PRACH resource grid

Object	Description
nrPRACHConfig	PRACH configuration parameters

For more information on PRACH time resources, preambles, configurations, and how to map PRACH symbols to the resource grid, see 5G NR PRACH Configuration.

You can configure and generate time-domain waveform for a single PRACH configuration in a single carrier. For an example, see 5G NR PRACH Waveform Generation.

You can measure the probability of correct detection of the PRACH preamble in the presence of a preamble signal. For an example, see 5G NR PRACH Detection Test.

Enhanced support for DM-RS and PT-RS

DM-RS and PT-RS for PUSCH

You can now use these functions and objects to model demodulation reference signals (DM-RS) and phase-tracking reference signals (PT-RS) for the physical uplink shared channel (PUSCH). These enhancements replace the example-specific implementations in previous releases.

Function	Description
nrPUSCHDMRS	Generate PUSCH DM-RS symbols
nrPUSCHDMRSIndices	Generate PUSCH DM-RS indices
nrPUSCHPTRS	Generate PUSCH PT-RS symbols
nrPUSCHPTRSIndices	Generate PUSCH PT-RS indices

Object	Description
nrPUSCHDMRSConfig	PUSCH DM-RS configuration parameters
nrPUSCHPTRSConfig	PUSCH PT-RS configuration parameters

For more information on how to configure and generate the PUSCH reference signals, see NR PUSCH Resource Allocation and DM-RS and PT-RS Reference Signals.

DM-RS and PT-RS for PDSCH

You can now use these functions and objects to model DM-RS and PT-RS for the physical downlink shared channel (PDSCH). These enhancements replace the example-specific implementations in previous releases.

Function	Description
`nrPDSCHDMRS`	Generate PDSCH DM-RS symbols
`nrPDSCHDMRSIndices`	Generate PDSCH DM-RS indices
`nrPDSCHPTRS`	Generate PDSCH PT-RS symbols
`nrPDSCHPTRSIndices`	Generate PDSCH PT-RS indices

Object	Description
`nrPDSCHDMRSConfig`	PDSCH DM-RS configuration parameters
`nrPDSCHPTRSConfig`	PDSCH PT-RS configuration parameters

For more information on how to configure and generate the PDSCH reference signals, see NR PDSCH Resource Allocation and DM-RS and PT-RS Reference Signals.

Enhanced support for PDSCH indices and PDSCH configuration

You can now use this function and these objects to configure and generate PDSCH resource element indices. These enhancements replace the example-specific implementation in previous releases.

Function	Description
`nrPDSCHIndices`	Generate PDSCH resource element indices

Object	Description
`nrPDSCHConfig`	PDSCH configuration parameters
`nrPDSCHReservedConfig`	PDSCH reserved physical resource block (PRB) configuration parameters

The `nrPDSCH` and `nrPDSCHDecode` functions now enable you to specify carrier and PDSCH configuration parameters by using the `nrCarrierConfig` and `nrPDSCHConfig` objects, respectively.

Enhanced support for PUSCH indices and PUSCH configuration

You can now use this function and object to configure and generate physical uplink shared channel (PUSCH) resource element indices. These enhancements replace the example-specific implementation in previous releases.

Function	Description
`nrPUSCHIndices`	Generate PUSCH resource element indices

Object	Description
`nrPUSCHConfig`	PUSCH configuration parameters

The `nrPUSCH` and `nrPUSCHDecode` functions now enable you to specify carrier and PUSCH configuration parameters by using the `nrCarrierConfig` and `nrPUSCHConfig` objects, respectively.

Enhanced support for PDCCH resources, CORESET, and search spaces

You can now use these functions to generate physical downlink control channel (PDCCH) resources and PDCCH DM-RS resources. These enhancements replace the example-specific implementation in previous releases.

Function	Description
`nrPDCCHResources`	Generate PDCCH and PDCCH DM-RS resources
`nrPDCCHSpace`	Generate PDCCH resources for all candidates and aggregation levels

You can now use these objects to configure the control resource element set (CORESET), the PDCCH, and the search space set. These enhancements replace the example-specific implementation in previous releases.

Object	Description
`nrCORESETConfig`	CORESET configuration parameters
`nrPDCCHConfig`	PDCCH configuration parameters
`nrSearchSpaceConfig`	Search space set configuration parameters

For more information on how to configure the CORESET, the PDCCH, and the search space set, see Downlink Control Processing and Procedures.

Enhanced support for SRS

You can now use these functions and this object to model the sounding reference signal (SRS). These enhancements replace the example-specific implementation in previous releases.

Function	Description
`nrSRS`	Generate uplink SRS symbols
`nrSRSIndices`	Generate uplink SRS resource element indices

Object	Description
`nrSRSConfig`	SRS configuration parameters

For more information on how to configure and generate the SRS for uplink transmission, see NR SRS Configuration.

You can use the SRS for synchronization, channel estimation, and uplink channel state information (CSI) estimation. For an example, see NR Uplink Channel State Information Estimation Using SRS.

Specify RNTI for polar and DCI decoding

The `nrPolarDecode` and `nrDCIDecode` functions now enable you to specify the radio network temporary identifier (RNTI) used for masking the CRC parity bits at the transmit end.

Uplink FRC generation

Generate standard-compliant 5G NR uplink fixed reference channels (FRCs) for frequency range 1 (FR1) and frequency range 2 (FR2). For more information, see 5G NR-TM and FRC Waveform Generation.

Characterize the impact of RF impairments in NR transmitter and receiver

Characterize the impact of radio frequency (RF) impairments, such as in-phase and quadrature (IQ) imbalance, phase noise, and power amplifier (PA) nonlinearities in the performance of a NR RF transmitter. For more information, see Modeling and Testing an NR RF Transmitter.

Characterize the impact of RF impairments in the RF reception of an NR waveform when coexisting with a long-term evolution (LTE) interference. For more information, see Modeling and Testing an NR RF Receiver with LTE Interference.

Measure EVM of NR-TM

Measure the error vector magnitude (EVM) of NR test models (NR-TM) with phase noise and memoryless nonlinearity impairments. For an example, see EVM Measurement of 5G NR-TM Waveforms.

Measure CSI-RSRP, CSI-RSSI, CSI-RSRQ and report CQI using CSI-RS

Measure the channel state information (CSI) reference signal received power (CSI-RSRP), the CSI received signal strength indicator (CSI-RSSI), and the CSI reference signal received quality (CSI-RSRQ). For an example, see 5G NR CSI-RS Measurements.

Calculate and compare CQI values using CSI-RS with practical and perfect channel estimation. For an example, see 5G NR CQI Reporting.

System-level simulation updates

Schedule 5G NR PUSCH and PDSCH resources for TDD mode with symbol based scheduling

Schedule PUSCH and PDSCH resources in time division duplexing (TDD) mode with radio link control (RLC) layer functionality in unacknowledged mode (UM). You can use slot or symbol based scheduling, choose from various MAC scheduling strategies, and evaluate the scheduling performance. For more information, see NR TDD Symbol Based Scheduling Performance Evaluation.

Schedule 5G NR PUSCH and PDSCH resources for FDD mode

Schedule PUSCH and PDSCH resources in frequency division duplexing (FDD) mode with RLC UM and MAC logical channel prioritization (LCP) procedure. You can choose from various MAC scheduling strategies and evaluate the scheduling performance. For more information, see NR FDD Scheduling Performance Evaluation.

Use RLC and LCP for 5G NR PUSCH FDD scheduling

PUSCH scheduling in frequency division duplexing (FDD) mode now includes radio link control (RLC) layer functionality in UM and MAC LCP procedure. For more information, see NR PUSCH FDD Scheduling.

Generate data for deep learning data synthesis

Train a convolutional neural network (CNN) for 5G channel estimation using Deep Learning Toolbox™ and data generated with 5G Toolbox. For an example, see Deep Learning Data Synthesis for 5G Channel Estimation.

Functionality being removed or changed

Polar decoding metric update

Behavior change

In releases R2019b and before, the function `nrPolarDecode` uses the exact form of the expression $\log(1 + e^x)$ for the internal metric evaluation of polar decoding. Because the exact form leads to numerical instability for high SNR ranges, polar decoding now uses an approximation. The function `nrPolarDecode` approximates $\log(1 + e^x)$as 0 for $x < 0$ and as x for $x \geq 0$.

Polar decoding is a common component in the broadcast channel, downlink control information, and uplink control information decoding. Therefore, the polar metric approximation affects the results of the `nrBCHDecode`, `nrDCIDecode`, and `nrUCIDecode` functions, resulting in a marginal degradation of the BLER performance in a link-level simulation.

R2019b

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
nrTimingEstimate	Practical timing estimation
nrChannelEstimate	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
nrCSIRS	Generate CSI-RS symbols
nrCSIRSIndices	Generate CSI-RS resource element indices

Objects	Description
nrCSIRSConfig	CSI-RS configuration parameters
nrCarrierConfig	Carrier configuration parameters

5G NR-TM waveform generation

Generate standard-compliant 5G NR test models (NR-TMs) for frequency range 1 (FR1) and frequency range 2 (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.

13

R2019a

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.