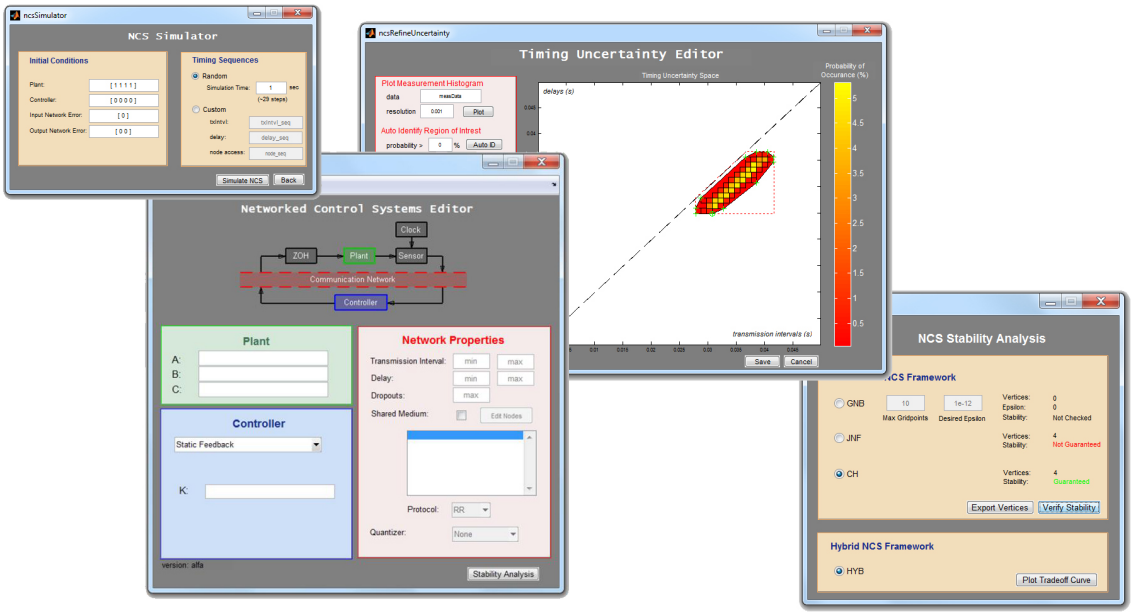
Networked Control Systems Toolbox

Eindhoven University of Technology

December 10, 2013



1 Introduction

This toolbox was created to facilitate quick and easy verification of theoretical properties of linear time-invariant (LTI) control systems which communicate via a network. The ability to simulate the models used for stability analysis is also included in this toolbox.

For robust stability analysis, this toolbox is based on both a discrete-time framework and a hybrid framework. The theoretical developments which use the discrete-time framework are [1, 4, 3, 5, 7, 8]. For theory related to the hybrid framework developments, please refer to [6].

For an overview and more information on how to use the toolbox, please refer to [2]. We strongly encourage the user to be familiar with the aforementioned literature before using the toolbox.

2 Required MATLAB (2007a or later) Toolboxes

- MPT toolbox [includes Yalmip and SeDuMi] (latest version) (http://control.ee.ethz.ch/~mpt/)
- Linear Systems *Toolkit* (http://hdd.ece.nus.edu.sg/~bmchen/linsyskit/index.html)
- MATLAB Symbolic Toolbox
- MATLAB Robust Control Toolbox

- MATLAB Optimization Toolbox

3 Installation

Unzip all the contents of ncsToolbox.zip into a folder on your computer. Add this folder and all subfolders to the MATLAB path using 'File > Set Path...'

Once the folders are added to the path try to run the four demos included in the toolbox:

- example_ncs1
- example_ncs2
- example_ncs3
- example_ncs4

If the demos complete without an error, then the toolbox has been installed successfully. If the demos produce an error during execution, make sure that the required toolboxes are installed correctly.

4 NCS object

This toolbox defines a class object of type `ncs`. An `ncs` object can be created with the NCS Editor GUI, which can be viewed by typing '`ncsEditor`' in the workspace. Information regarding the input of the NCS editor can be found in [2]. Once an `ncs` object has been created, the list of properties can be viewed (and edited) and functions associated with it can be executed. Notably, an `ncs` object has the following class functions:

- `isNcsStable` - determines if robust stability can be guaranteed
- `getPolyOvrAprx` - extracts the polytopic overapproximation using one of the included overapproximation techniques (JNF [3],CH [5], or GNB [4])
- `findMatiMadBoundry` - produces the data vectors need to plot the (MATI,MAD) robustness region, according to [6]

To view the full list of properties and functions associated with an `ncs` object, type '`help ncs`' in the workspace.

5 Demos

There are four demos currently included in the toolbox. To view the demos, simply type

- showdemo example_ncs1
- showdemo example_ncs2
- showdemo example_ncs3
- showdemo example_ncs4

to view the demos which show how to work with NCS objects from the MATLAB command window.

Also included in the toolbox are three demo `ncs` objects, which can be loaded into the workspace by typing '`load exampleNcs.mat`' in the workspace. These demo `ncs` objects can be imported into the NCS Editor by clicking 'File > Import' in the NCS Editor window. The simulation and analysis tools can be used on these objects.

6 List of Functions

Type 'help <function name>' in MATLAB to read a description for each function.

NCS Object Methods (Functions):

- getPlant.m
- getController.m
- getNetwork.m
- getDtModelMatrices.m
- getPolyOvrAprx.m
- setPlant.m
- setController.m
- setNetwork.m
- plotTimingUncertaintyRegion.m
- genTPoly.m
- isNcsStable.m
- findMatiMadBoundry.m
- evaluateDtNcsModel.m
- simulateNcs.m

Main Discrete-Time NCS Functions:

- dtNcs_main.m
- genDtNcsModel.m
- genPolyOvrAprx.m
- analyzePolyOvrAprx.m
- analyzeStochStability.m

Supporting Discrete-Time NCS Functions:

- sepJdnBlocks.m
- reduceAlphas.m
- getAlphaMinMax.m
- jnf.m
- jnf2ch.m
- gnb.m
- initGridPts.m
- genTimingUncertaintyRegion.m
- findNextDropPoly.m
- checkTimingBounds.m

Main Hybrid NCS Functions:

- hybNcs_main.m
- genHybNcsModel.m
- analyzeHybNcs.m

7 Known Issues

1. The toolbox uses the Linear Systems Toolkit, which may produce a case-sensitive error when attempting to run `example_dtNcs` due to the function `zzrepvaluej`. To fix this error, go to

the directory where the Linear Systems Toolkit is located and rename the file 'zzrepvalue**j**.m' to 'zzrepvalue**J**.m'.

8 Citing

If this toolbox is used for research purposes, please cite [2].

9 Community Feedback

We hope to gain feedback from the community to improve the quality and overall usefulness of both the toolbox and the theory. Any suggestions or comments are encouraged and can be emailed to Nick Bauer (`n.w.bauer@tue.nl`).

References

[1] D. Antunes, J.P. Hespanha, and C. Silvestre. Stochastic networked control systems with dynamic protocols. In *Proc. IEEE Conf. American Control Conference*, pages 1686 –1691, July 2011.

[2] N.W. Bauer, S.J.L.M. van Loon, M.C.F. Donkers, N van de Wouw, and W.P.M.H. Heemels. Networked control systems toolbox: Robust stability analysis made easy. In *Proceedings of the 3rd IFAC Workshop on Distributed Estimation and Control in Networked Systems (NECSYS)*, pages 55–60, 2012.

[3] M.B.G. Cloosterman, N. van de Wouw, W.P.M.H. Heemels, and H. Nijmeijer. Stability of networked control systems with uncertain time-varying delays. *IEEE Trans. Autom. Control*, 54(7):1575–1580, Jul. 2009.

[4] M.C.F. Donkers, W.P.M.H. Heemels, N. van de Wouw, and L. Hetel. Stability analysis of networked control systems using a switched linear systems approach. *Trans. Autom. Control*, 56(9):2101–2115, Sep. 2011.

[5] R.H. Gielen, S. Olaru, M. Lazar, W.P.M.H. Heemels, N. van de Wouw, and S.-I. Niculescu. On polytopic inclusions as a modeling framework for systems with time-varying delays. *Automatica*, 46(3):615 – 619, 2010.

[6] W.P.M.H. Heemels, A.R. Teel, N. van de Wouw, and D. Nešić. Networked control systems with communication constraints: Tradeoffs between transmission intervals, delays and performance. *IEEE Trans. Autom. Control*, 55(8):1781–1796, 2010.

[7] W.P.M.H. Heemels, N. van de Wouw, R.H. Gielen, M.C.F. Donkers, L. Hetel, S. Olaru, M. Lazar, J. Daafouz, and S. Niculescu. Comparison of overapproximation methods for stability analysis of networked control systems. In *HSCC 2010: Proc. 13th ACM Int. Conf. on Hybrid systems: Computation and control*, pages 181–190, 2010.

[8] S.J.L.M. van Loon. *Stability Analysis of Networked and Quantized Control Systems: Theory and Matlab Implementation.* Master's Thesis, Eindhoven University of Technology, Eindhoven, The Netherlands, 2012.