Obfuscate Model Tool

December 2020

McMaster Centre for Software Certification (McSCert)

Contents

1 Introduction

The Obfuscate Model Tool removes, renames, and/or hides various details of a Simulink model in order to hide confidential information. This can be useful for eliminating proprietary details when sending models to third-parties, or even by removing details from models in order to create images suitable for publication.

2 How to Use the Tool

This section describes what must be done to setup the Obfuscate Model Tool, as well as how to use the tool.

2.1 Prerequisites and Installation

1. Use MATLAB/Simulink 2011b or newer.
2. To install the tool,
   (a) from a `.zip` file — unzip the contents into your desired location. Ensure the unzipped folder and subfolders are present in your MATLAB search path, or add them if they are not present.
   (b) from a `.mltbx` file — simply open MATLAB and double-click on the file. Your MATLAB search path should be automatically configured.
   (c) from the files only — add the folders and subfolders to your MATLAB search path.
   - *Note:* If running the command "`which ObfuscateModelGUI`" indicates that the script is not found, then the tool needs to be added to the MATLAB search path. For information on adding files to the MATLAB search path, please see the MathWorks documentation.
3. Ensure the Simulink-Utility folder is on your MATLAB search path. This is a dependency for the tool to work correctly.
4. Ensure your model is open and unlocked.

2.2 Getting Started

The tool can launched by double-clicking the `ObfuscateModelGUI.mlapp` file, or by running the command `ObfuscateModelGUI` from the Command Window. This will open the Graphical User Interface (GUI) shown in Figure 1.

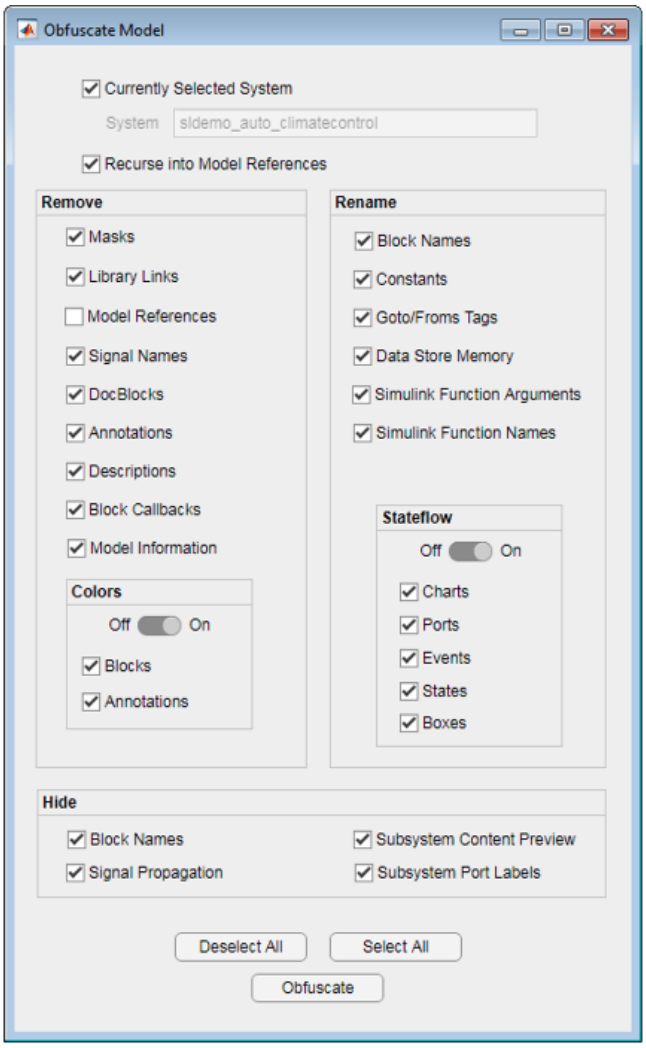

Figure 1: The tool GUI.

2.3 Functionality

This section describes the tool functionality when being used from the GUI (Figure 1). Each section describes one of the groups of options in the GUI.

2.3.1 Remove

The options in the *Remove* group will discard elements of the model.

- Masks – Block masks are commonly used to customize the block appearance of custom blocks. This option removes the masks of all blocks.

- Library Links – Library links can be used in a model to reference blocks that reside in other libraries. This option removes (or "breaks") all library links so that the blocks are stored directly in the model instead of the library. This means that the model is no longer dependant on external libraries.
- Model References – The use of a model block introduces a reference to another model. This option resets all model reference blocks so that they no longer point to other models. *Note: This may impact the model functionality.*
- Signal Names – This option turns off signal propagation.
- DocBlocks – A DocBlock stores documentation about the model. This options removes all DocBlocks.
- Annotations – This option deletes all text, area, or image annotations.
- Descriptions – This options removes the *description* information of lines, blocks, and annotations.
- Block Callbacks – Blocks can have custom callbacks. This option removes all callbacks.
- Model Information – A Simulink model stores information about itself, such as its creator's name and version number. This option resets this data.
- Colors – These options remove the colours of blocks and annotations so that they revert to their default color.

2.3.2 Rename

The options in the *Rename* group will rename elements of the model to use generic names.

- Block Names – Each block in a model has a *name* that is typically displayed underneath the block. This option renames all block names to a generic name based on the block type. For example, an Inport block will be renamed to Inport1.
- Constants – Each constant block has a *value* that is a number or a variable name. This option renames any values that are variable names to be generic (e.g., Constant1). Numerical constants will remain. *Note: You must manually change the definition of the data to match the new name. This tool does not currently rename data in the base workspace or data dictionaries.*
- Goto/From Tags – A Goto block has a *goto tag* that matches it to its From blocks. This option renames tags to generic names (e.g., GotoFrom1) and renames any matching From blocks as well.

- Data Store Memory – A Data Store Memory block has a *data store name*. This option renames all Data Store Memory blocks to be generic (e.g., DataStore1) as well as all associated Data Store Read and Data Store Write blocks.
- Simulink Function Arguments – A Simulink Function can have inputs and outputs using ArgIn and ArgOut blocks. This option renames the *argument name* of ArgIn and ArgOut blocks to be generic (e.g. u1 for an input, and y1 for an output).
- Simulink Function Names – The trigger within a Simulink Function specifies the function's name. This option renames it to a generic name (e.g., f1), and updates any corresponding Function Caller blocks to match.
- Stateflow – These options rename the various Stateflow elements to generic names.

2.3.3 Hide

The options in the *Hide* group will hide elements of the model from view.

- Block Names – This option hides the name of a block from view.
- Signal Propagation – This option turns off signal propagation.
- Subsystem Content Preview – This option turns off the content preview that is displayed in blocks such as subsystems.
- Subsystem Port Labels – This option hides the port labels shown on blocks such as subsystems.

3 Example

Use the command `sldemo_auto_climatecontrol` in the Simulink Command Window to open the example model, shown in Figure 2. To run the tool, run the command `ObfuscateModelGUI` from the Command Window, and then press the `Obfuscate` button. The resulting model is given in Figure 3. We can see that the colors, annotations, masks, names, port labels, and many other elements have been removed from the model.

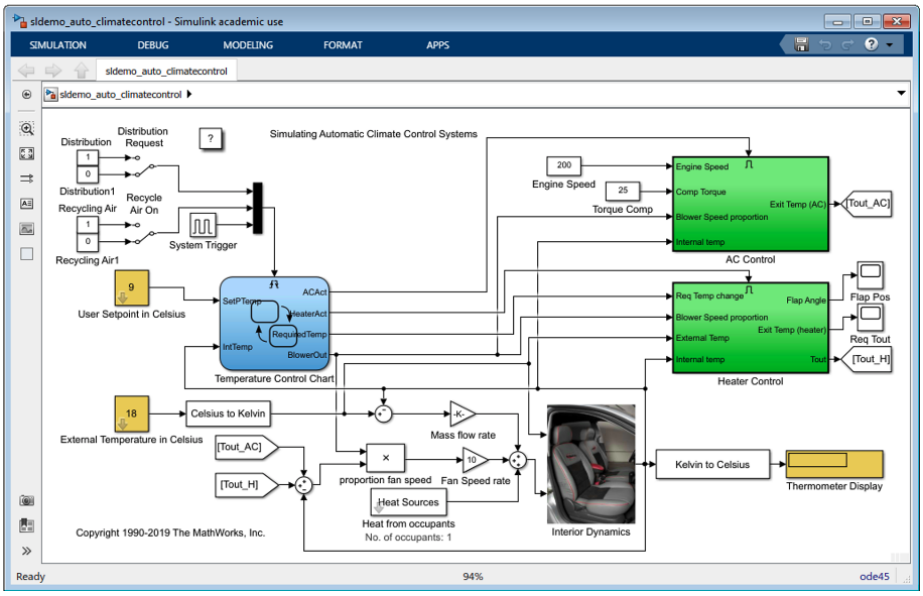

Figure 2: Original demo model.

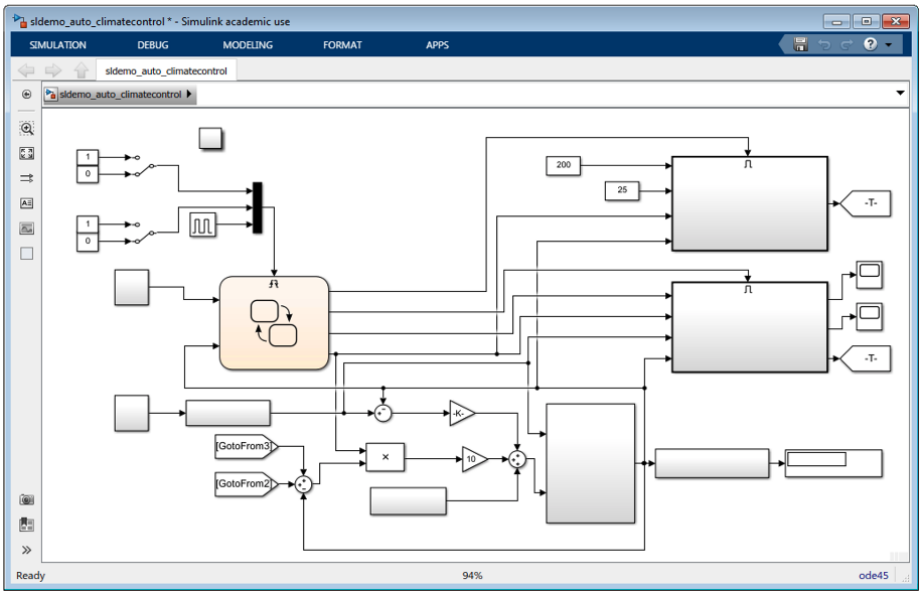

Figure 3: Resulting model after obfuscation.