Reach/Coreach Tool

March 2021

McMaster Centre for Software Certification (McSCert)

Contents

1 Introduction

The Reach/Coreach Tool is used to trace and highlight data flow and control flow dependencies in Simulink models. Performing a *reachability analysis*, or "*reach*", on a portion of the model will show the effects that it has on other parts of the model. A *coreachability analysis*, or "*coreach*", does the opposite, by showing what a portion of the model is dependent on.

The tool has many uses:

- The reachability and coreachability analyses effectively generate a dependency diagram of data and control flow in the model. This assists with model comprehension, and is also useful in identifying areas of the model where seemingly independent data flows have hidden dependencies.
- The tool supports impact analysis, because it identifies the parts of a model affected by a potential change of a given block or signal. Impact analysis can be of great value in indicating what effect a change in requirements or design can have on a system's design.
- The tool can be used to find unreachable parts of a model. When reachability analysis is performed on all of a model's inputs, unhighlighted blocks/signals represent unreachable parts of the model.
- When the coreachability analysis is performed on all of a model's outputs, the unhighlighted blocks/signals are unnecessary in the sense that they have no (data or control) effect on the outputs of the model.
- The tool can isolate a model into smaller pieces for further analysis, by *slicing* out the highlighted traces.

1.1 More Information

For more information on the tool and how it can be used in a model-based development with Simulink, please refer to the following papers:

Vera Pantelic, Steven Postma, Mark Lawford, Monika Jaskolka, Bennett Mackenzie, Alexandre Korobkine, Marc Bender, Jeff Ong, Gordon Marks, Alan Wassyng, "Software engineering practices and Simulink: bridging the gap," *International Journal on Software Tools for Technology Transfer (STTT)*, 2017, 95–117.

Vera Pantelic, Steven Postma, Mark Lawford, Alexandre Korobkine, Bennett Mackenzie, Jeff Ong, and Marc Bender, "A Toolset for Simulink: Improving Software Engineering Practices in Development with Simulink,"*Proceedings of 3rd International Conference on Model-Driven Engineering and Software Development (MODELSWARD 2015)*, SCITEPRESS, 2015, 50–61.

2 How to Use the Tool

This section describes what must be done to setup the 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, use one of the following approaches:

   (a) **Download the `.zip` from GitHub**

   i. Unzip the contents into your desired location.

   ii. Add the unzipped folder and subfolders to your MATLAB search path.

   iii. Download the Simulink-Utility in the same manner. Add the folder and subfolders to your MATLAB search path also. This is a dependency for the tool to work correctly.

   (b) **Use the Git command line**

   i. Use the following command to download the tool and any necessary submodules.

   ```
   git clone --recursive https://github.com/McSCert/Reach-Coreach
   ```

   ii. Add the folder and subfolders to your MATLAB search path.

   (c) **If you already have the files**

   i. Add the tool folder and subfolders to your MATLAB search path.

3. Run sl_refresh_customizations to refresh the Context Menu.

4. Ensure your model is open and unlocked.

Toubleshooting: If running the command "`which ReachCoreach`" 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.

2.2 Getting Started

Most features of the tool can be used via the Simulink Context Menu, which can be viewed by right-clicking in a model. The following options can be available in the Reach/Coreach menu, depending on what is selected in the model. These options are shown in Figure 1.

- *Reach From Selected* – Available at all times.
- *Coreach From Selected* – Available at all times.

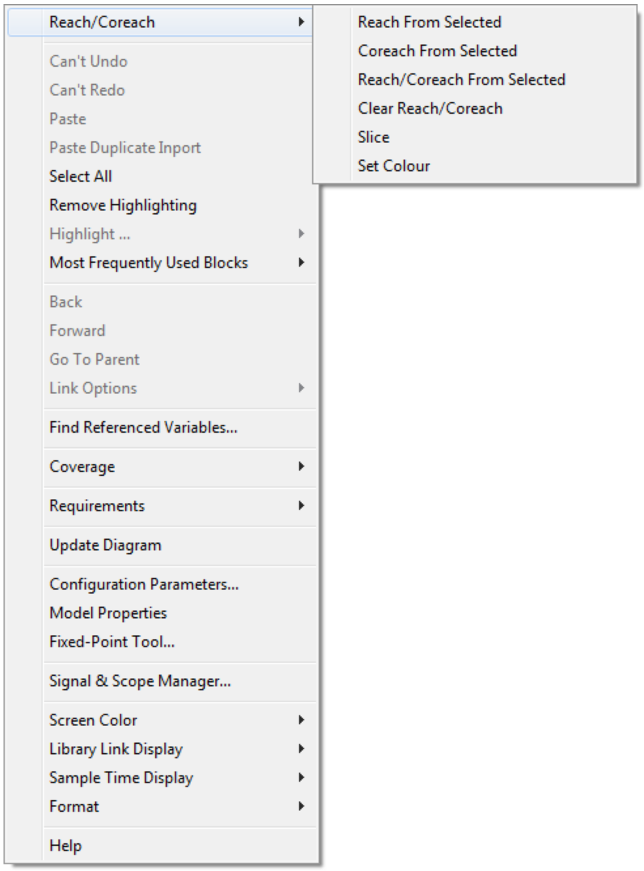


Figure 1: Simulink Context Menu with tool options visible.

- *Reach/Coreach From Selected* – Available at all times.
- *Clear Reach/Coreach* – Available when a reach and/or coreach are present in the model.
- *Slice* – Available when a reach and/or coreach are present in the model.
- *Set Colour* – Available at all times.

The only features that cannot be used via the Simulink Context Menu are the model differencing versions of the `Reach From Selected` and `Coreach From Selected` functionalities which are run via the command line, as described in Table 7 and Table 8.

2.3 Functionality

This section describes the tool functionality when being used from the Simulink Context Menu (Figure 1).

Reach From Selected

Right-clicking on one or more blocks in the model and then selecting `Reach From Selected` from the Context Menu will perform a reachability analysis, and highlight the blocks and signals which are affected by those selected blocks. The highlighting illustrates what parts of the model are dependent on the selected blocks, due to control flow or data flow.

Coreach From Selected

Right-clicking on one or more blocks in the model and then selecting `Coreach From Selected` from the Context Menu will perform a coreachability analysis, and highlight the blocks and signals which affect those selected blocks. The highlighting illustrates what parts of the model the selected blocks depend on, due to control flow or data flow.

Reach/Coreach From Selected

Right-clicking on one or more blocks in the model, and then selecting `Reach/Coreach From Selected` from the Context Menu will perform both reachability and coreachability analyses on the selected blocks. The highlighting illustrates, bidirectionally, what parts of the model either impact, or have an impact, on the selected blocks, due to control flow or data flow.

Clear Reach/Coreach

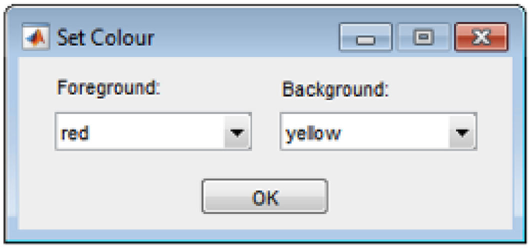
After a reach, coreach, or reach/coreach are performed on the model, this option will be enabled. Right-clicking anywhere in the model, and then selecting `Clear Reach/Coreach` from the Context Menu will clear all highlighting in the model.

Slice

After a reach, coreach, or reach/coreach are performed on the model, this option will be enabled. Right-clicking anywhere in the model, and then selecting `Slice` from the Context Menu will isolate the highlighted parts of the model by removing the non-highlighted blocks and lines.

Set Colour

Right-clicking anywhere in the model and then selecting `Set Colour` from the Context Menu will display the user interface shown in Figure 2. The foreground (text and line) and background (block) highlight colours can be customized via the drop down menus, which provide the standard Matlab colours. Clicking `OK`



Foreground: red

Background: yellow

Figure 2: Highlight colour selector interface.

will use the selected colours for all current and future highlighting in the model. The default colours are red (foreground) and yellow (background).

2.4 Errors and Warnings

Any errors or warnings during tool use will be visible in the MATLAB Command Window. Typically, errors will be shown when the model is locked or function parameters are incorrect.

3 Example

Use the command `ReachCoreachDemo` in the Simulink command window to open the example model, shown in Figure 3. This example has control and data flow pass through Data Stores, If and If Action subsystems, as well as a Bus Creator and Bus Selector. The following steps will demonstrate the various functionalities of the tool.

1. Let us perform a reachability analysis to find all possible dependencies of the Inport block In2. To do this, right-click on the In2 block and select the `Reach From Selected` option. The resulting model is shown in Figure 4. Due to the fact that In2 is used in the last two condition checks of the If block and its data flow is traced to Outport block Out1. Doing a reachability analysis on In2 also shows clearly that it impacts the Data Store Memory block A.

2. To clear the highlighting now present in the model due to the previous reach operation, right-click anywhere in the model and select the `Clear Reach/Coreach` option. This will revert the model's appearance to that of Figure 3.

3. Let us perform a coreachability analysis to find all the parts of the model which the Outport blocks Out2 and Out3 are dependent on. To do this,

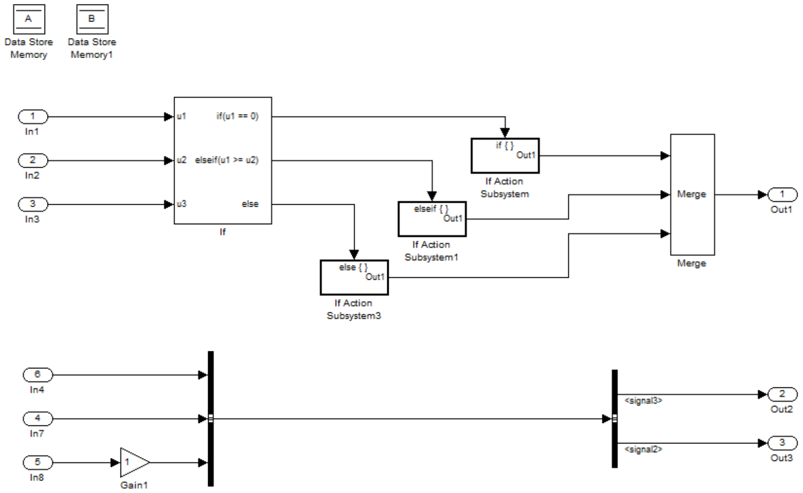


Figure 3: Reach/Coreach demo model.

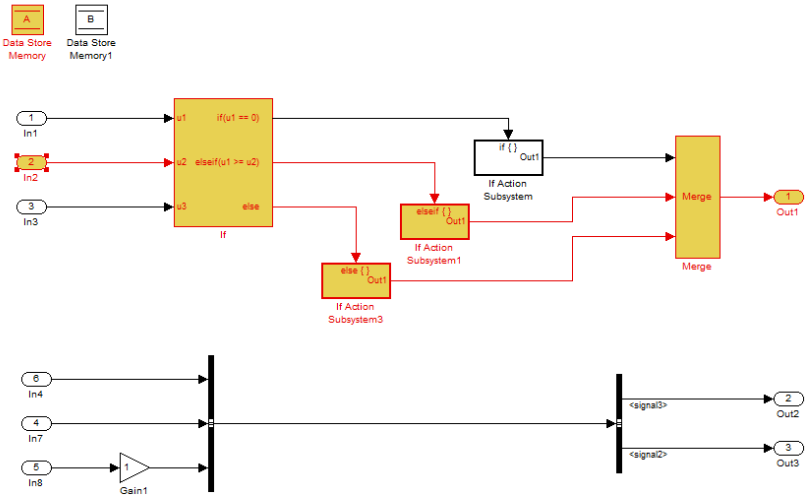


Figure 4: Resulting model after `Reach From Selected` operation.

select both blocks, right-click on them, and select the `Coreach From Selected` option. The resulting model is shown in Figure 5. Blocks Out2 and Out3 are dependent on the Inport block In7 and In8 only, and it is interesting to note that In6 has no impact in the model.

4. We can isolate the coreachability analysis by performing a slice. To do so, right-click anywhere in the model and select the `Slice` option. All blocks

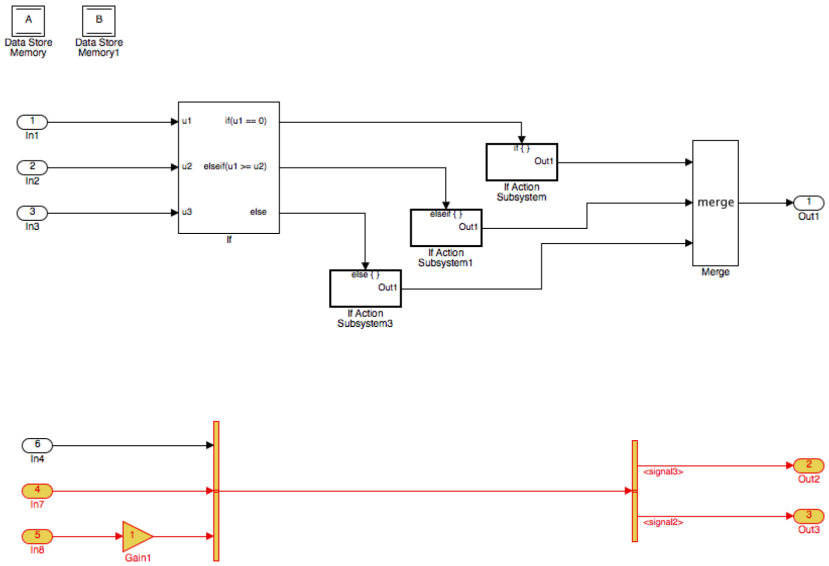


Figure 5: Resulting model after `Coreach From Selected` operation.

and signals which are not highlighted will be deleted from the model. The resulting model is shown in Figure 6.

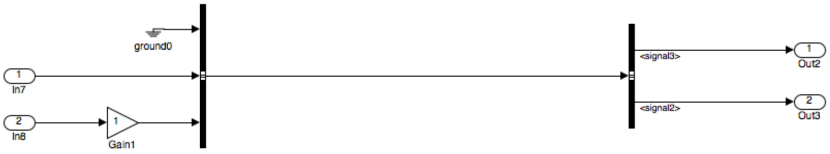


Figure 6: Resulting model after `Slice` operation.

5. Let us now perform a reach/coreach in the model. First, reopen the demo model, so as to start from new. Right-click on Data Store Memory B, and select the `Reach/Coreach From Selected` option. The resulting model is shown in Figure 7. By doing both a reachability and coreachability analysis on this block, we can see the total impact that this block has on the model. In this case, no other blocks or signals are highlighted, meaning that this block has no purpose in the model. It is not used by anything, nor does it perform any operation itself. This block can be considered “dead code”, and should be removed from the design.

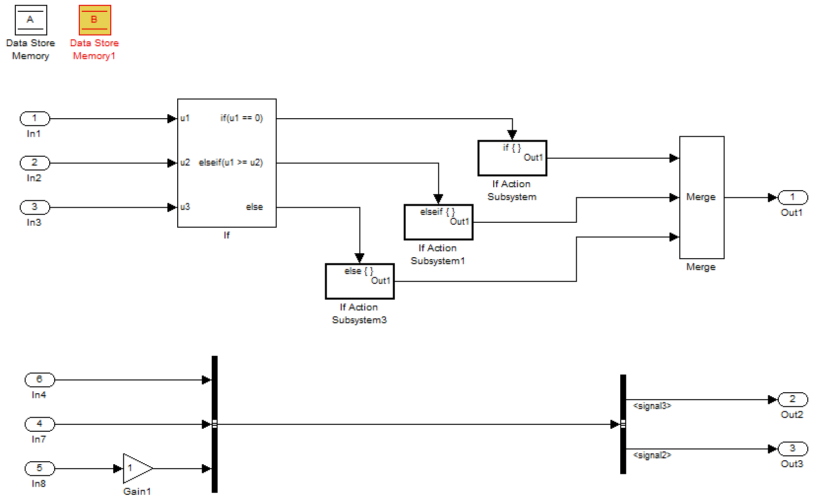


Figure 7: Resulting model after `Reach/Coreach` operation.

4 Matlab Commands

The tool can also be used via the Matlab command line, with the following classes and functions.

Table 1: ReachCoreach function description.

Function	`ReachCoreach`
Syntax	`obj = ReachCoreach(`*modelName*`)`
Description	Create a new ReachCoreach object for *model*.
Inputs	*modelName*: The Simulink model name (or top-level system name) of the model on which to perform reach/coreach operations.
Outputs	*obj*: ReachCoreach object.

Table 2: reachAll function description.

Function	`reachAll`
Syntax	`obj.reachAll(`*blocks, lines*`)`
Description	Perform a reachability analysis starting from *blocks*.
Inputs	*blocks*: A cell array of block path names from which to start the reach operation. *lines*: An array of handles representing signal lines from which to start the reach operation.
Outputs	N/A

Table 3: coreachAll function description.

Function	`coreachAll`
Syntax	`obj.coreachAll`(*blocks, lines*)
Description	Perform a coreachability analysis starting from *blocks*.
Inputs	*blocks*: A cell array of block path names from which to start the coreach operation. *lines*: An array of handles representing signal lines from which to start the coreach operation.
Outputs	N/A

Table 4: clear function description.

Function	`clear`
Syntax	`obj.clear`()
Description	Clear any reach or coreach highlighting.
Inputs	N/A
Outputs	N/A

Table 5: slice function description.

Function	`slice`
Syntax	`obj.slice`()
Description	Isolate the reach/coreach blocks by removing unhighlighted blocks.
Inputs	N/A
Outputs	N/A

Table 6: setColor function description.

Function	`setColor`
Syntax	`obj.setColor`(*foregroundColor*, *backgroundColor*)
Description	Set the color of the reach/coreach highlighting.
Inputs	*foregroundColor*: A Matlab color. *backgroundColor*: A Matlab color.
Outputs	N/A

Table 7: Reach_Diff function description.

Function	`Reach_Diff`
Syntax	[rObjs1, rObjs2] = `Reach_Diff`(*model1, model2, highlight*)
Description	Identify objects in *model1* and *model2* that were added/deleted/modified between the two versions, then perform a reachability analysis in each model starting from the identified objects.
Inputs	*model1*: A char array naming a model. *model2*: A char array naming a different version of the model given by *model1* (this version can be newer or older). *highlight*: Set as 1 to highlight impacts yellow within the models and to highlight the differences dark green; set as 0 for no highlighting. (Optional. Default is 1)
Outputs	*rObjs1*: Array of Simulink block and line handles that were reached in *model1* *rObjs2*: Array of Simulink block and line handles that were reached in *model2*.

Table 8: Coreach_Diff function description.

Function	`Coreach_Diff`
Syntax	[cObjs1, cObjs2] = `Coreach_Diff`(*model1, model2, highlight*)
Description	Identify objects in *model1* and *model2* that were added/deleted/modified between the two versions, then perform a coreachability analysis in each model starting from the identified objects.
Inputs	*model1*: A char array naming a model. *model2*: A char array naming a different version of the model given by *model1* (this version can be newer or older). *highlight*: Set as 1 to highlight impacts yellow within the models and to highlight the differences dark green; set as 0 for no highlighting. (Optional. Default is 1)
Outputs	*cObjs1*: Array of Simulink block and line handles that were coreached in *model1*. *cObjs2*: Array of Simulink block and line handles that were coreached in *model2*.

Example: The following commands perform the same operations as described previously for the `ReachCoreachDemo` example, however via the Matlab command line.

1. `r = ReachCoreach('ReachCoreachDemo')`
2. `r.reachAll({'ReachCoreachDemo/In2'})`
3. `r.clear()`
4. `r.coreachAll({'ReachCoreachDemo/Out2','ReachCoreachDemo/Out3'})`
5. `r.slice()`