

Reactis V2013

Released August 14, 2013



New Simulink Support

- ▶ Initial support for R2013a
- ▶ Support for .slx Format
 - ▶ Introduced in R2012a to replace .mdl format
 - ▶ XML-based format
- ▶ Support Simulink *configuration set reference* concept

Multiple Condition Coverage (MCC)

MCC metric tracks if all combinations of condition outcomes for a decision have been exercised.

Multiple Condition Coverage (MCC)

MCC metric tracks if all combinations of condition outcomes for a decision have been exercised. For decision $A \ \&\& \ B \ \&\& \ C$, MCC targets are:

Multiple Condition Coverage (MCC)

MCC metric tracks if all combinations of condition outcomes for a decision have been exercised. For decision $A \ \&\& \ B \ \&\& \ C$, MCC targets are:

No Short-Circuiting

A	B	C	Decision
F	F	F	F
F	F	T	F
F	T	F	F
F	T	T	F
T	F	F	F
T	F	T	F
T	T	F	F
T	T	T	T

Multiple Condition Coverage (MCC)

MCC metric tracks if all combinations of condition outcomes for a decision have been exercised. For decision $A \ \&\& \ B \ \&\& \ C$, MCC targets are:

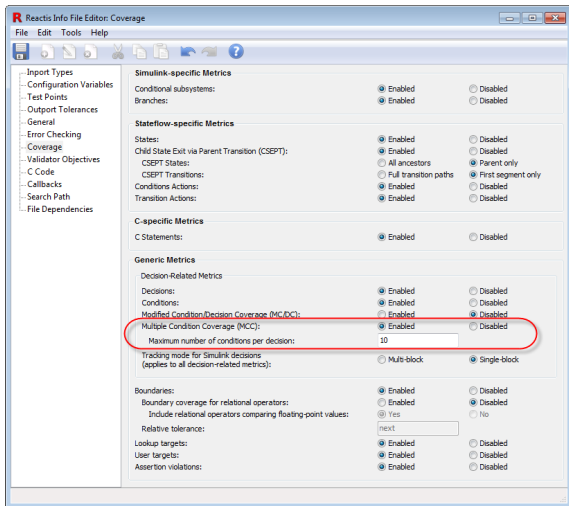
No Short-Circuiting

A	B	C	Decision
F	F	F	F
F	F	T	F
F	T	F	F
F	T	T	F
T	F	F	F
T	F	T	F
T	T	F	F
T	T	T	T

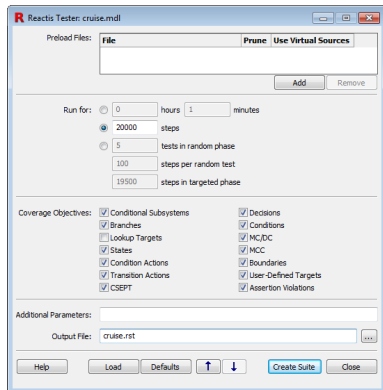
Short-Circuiting

A	B	C	Decision
F	x	x	F
T	F	x	F
T	T	F	F
T	T	T	T

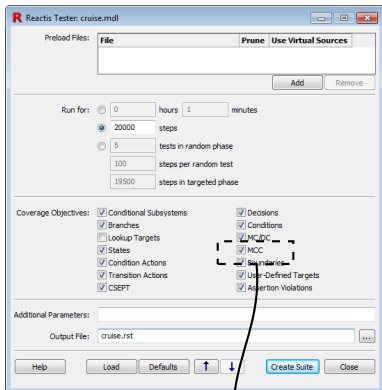
MCC Now Supported by Reactis



MCC Support in Reactis Tester

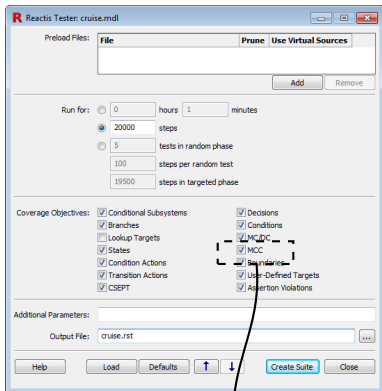


MCC Support in Reactis Tester

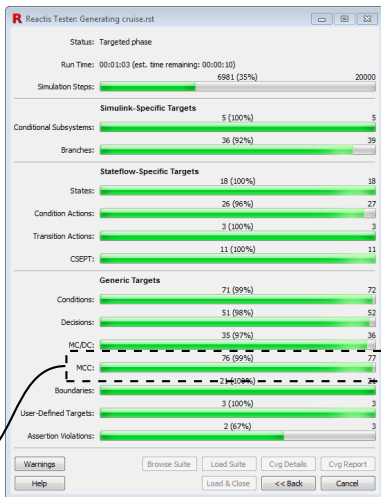


Tester attempts to cover MCC targets

MCC Support in Reactis Tester



Tester attempts to cover MCC targets



Tracking MCC in Simulator

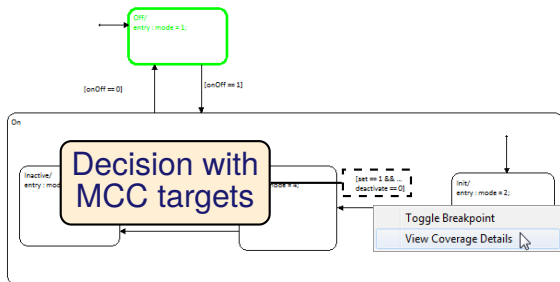
The screenshot displays the Reactis simulator window for a file named 'cruise.mdl (R2009a)'. The interface includes a menu bar (File, Edit, View, Simulate, Test Suite, Validate, Coverage, Window, Help), a toolbar with navigation and simulation controls, and a status bar at the bottom showing 'Test #29 (29/29), Step 1/1, Total 1463/1463' and 't=0.0'. On the left, a project tree lists components such as 'CruiseMain', 'CruiseMDL', 'DeactivateActivate', 'DesiredSpeed', 'Mode', 'SpdCheck', 'ThrottleCtl', 'Plant', 'LowSpeedOn', 'Speed', and 'Configuration Variables'. The main workspace shows a state machine diagram with a central state labeled 'Decision with MCC targets'. This state has an 'On' label and is connected to other states: an 'init/entry: mode = 2;' state, a state with '[set = 1 S.S. ... deactivate = 0]', and a state with 'entry: mode = 1;'. Transitions are labeled with conditions like '[onOff == 0]' and '[onOff == 1]'. A green box highlights the 'entry: mode = 1;' state.

Tracking MCC in Simulator

The screenshot displays the Reactis simulator window titled "Reactis: cruise.mdl (R2009a) - cruise.rsi - cruise.rst". The interface includes a menu bar (File, Edit, View, Simulate, Test Suite, Validate, Coverage, Window, Help) and a toolbar with various simulation controls. A progress bar at the top right shows "1463" and "All (1463 steps)".

On the left, a project tree lists the following components:

- cruise
 - SL CruiseMain
 - SL CruiseMDL
 - DeactivateActivate
 - SL DesiredSpeed
 - SF Mode
 - SpdCheck
 - SL ThrottleCtl
 - SL Plant
 - LowSpeedOn
 - Speed
 - Configuration Variables



Test #29 (29/29), Step 1/1, Total 1463/1463

t=0.0

Tracking MCC in Simulator

The screenshot displays the Reactis simulator interface. The main window shows a state machine diagram with a highlighted state: `On` with `entry : mode = 1;`. A transition labeled `[onOff == 1]` leads to another state: `init/entry : mode = 2;`. A transition labeled `[set == 1 && ... deactivate == 0]` leads to a state labeled `Decision with MCC targets`. A `Toggle Breakpoint` button is visible, with a `View Coverage Details` button below it. A legend on the left indicates `speed` (yellow arrow) and `Configuration Variables` (green circle). The status bar at the bottom shows `Test #29 (29/29), Step 1/1, Total 1463/1463` and `t=0.0`.

The **Coverage Details** window is open, showing the following table:

Decision		Condition	Condition		MC/DC		MC/DC	
True	False		True	False	True	False	True	False
1/30	1/4	set==1.0 deactivate==0.0	1/20	1/4	TT: 1/30	Fx: 1/4	TT: 1/30	TF: 1/20

Tracking MCC in Simulator

The image displays a simulator interface with two 'Coverage Details' windows and a state transition diagram. The diagram includes a state labeled 'Decision with MCC targets' and a 'Toggle Breakpoint / View Coverage Details' button.

Left Coverage Details Window:

Decision		Condition	Condition	Condition	MC/DC	MC/DC
True	False		True	False	True	False
1/30	1/4	set==1.0	1/20	1/4	TT: 1/30	Fx: 1/4
		deactivate==0.0	1/30	1/20	TT: 1/30	TF: 1/20

Right Coverage Details Window:

Click on column headers to enable filters Clear Filter

set==1.0	deactivate==0.0	Decision	Covered
False	x	False	1/4
True	False	False	1/20
True	True	True	1/30

State Transition Diagram:

- State: **Decision with MCC targets**
- Transitions: `mode == 1;`, `[set == 1 && ... deactivate == 0]`, `init/entry : mode = 2;`
- Buttons: **Toggle Breakpoint**, **View Coverage Details**

Bottom Status Bar: Test #29 (29/29), Step 1/1, Total 1463/1463, t=0.0

Tracking MCC in Simulator

Row for each MCC target

Decision with MCC targets

Decision MCC

Decision	Decision	Condition	Condition	Condition	MC/DC	MC/DC
True	False		True	False	True	False
1/30	1/4	set==1.0	1/20	1/4	TT: 1/30	Fx: 1/4
		deactivate==0.0	1/30	1/20	TT: 1/30	TF: 1/20

OH/
entry : mode = 1;

Click on column headers to enable filters

set==1.0	deactivate==0.0	Decision	Covered
False	x	False	1/4
True	False	False	1/20
True	True	True	1/30

Toggle Breakpoint
View Coverage Details

Test #29 (29/29), Step 1/1, Total 1463/1463 t=0.0

MCC Filtering

Decision MCC

Click on column headers to enable filters Clear Filter

set==1.0	deactivate==0.0	Decision	Covered
False	x	False	1/4
True	False	False	1/20
True	True	True	1/30

Help Close

MCC Filtering

Click column header

The image shows two side-by-side screenshots of a software window titled "Coverage Details". The window has a tab labeled "Decision" and "MCC". Below the tab, there is a text prompt "Click on column headers to enable filters" and a "Clear Filter" button. The main content is a table with columns for filters and data.

Left Screenshot: The table has four columns: "set==1.0", "deactivate==0.0", "Decision", and "Covered". The data rows are:

set==1.0	deactivate==0.0	Decision	Covered
False	x	False	1/4
True	False	False	1/20
True	True	True	1/30

Right Screenshot: The table has four columns: "set==1.0", "deactivate==0.0", "T: Decision", and "Covered". The data rows are:

set==1.0	deactivate==0.0	T: Decision	Covered
True	True	True	1/30

A yellow callout box with blue text says "Only display rows with *True* in column". An arrow points from this box to the "T: Decision" column header in the right screenshot. Another arrow points from the "Decision" column header in the left screenshot to the "T: Decision" column header in the right screenshot. The right screenshot also has a "Close" button at the bottom right.

MCC Filtering

Click column header

R Coverage Details

Decision MCC

Click on column headers to enable filters

Clear Filter

set==1.0	deactivate==0.0	Decision	Covered
False	x	False	1/4
True	False	False	1/20
True	True	True	1/30

Help

Close

Only display rows with *True* in column

Click column header again

R Coverage Details

Decision MCC

Click on column headers to enable filters

Clear Filter

set==1.0	deactivate==0.0	F: Decision	Covered
False	x	False	1/4
True	False	False	1/20

Help

Only display rows with *False* in column

MCC Considerations

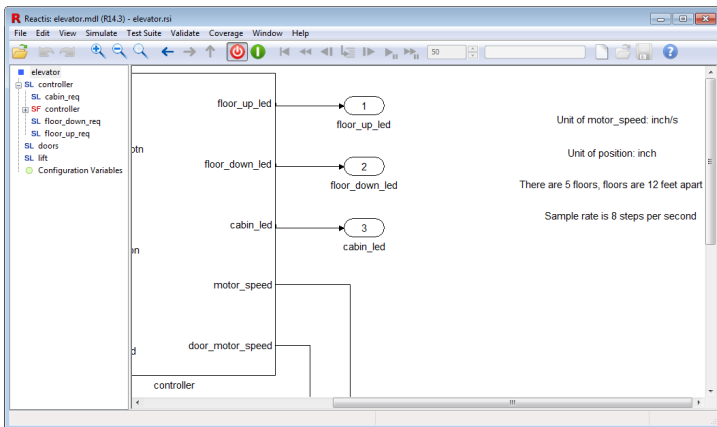
- ▶ Decision with n conditions has 2^n MCC targets.
- ▶ When short-circuiting is enabled, many fewer MCC targets. Number is between MC/DC ($n+1$) and 2^n .
- ▶ Even if 100% MCC is not goal, MCC coverage details can be helpful in obtaining MC/DC.
- ▶ In model-specific settings (*Edit* → *Coverage...*), can set upper-bound on number of conditions in a decision; if exceeded MCC will not be tracked.

Text Search Enhancement

Text search now searches Simulink annotations

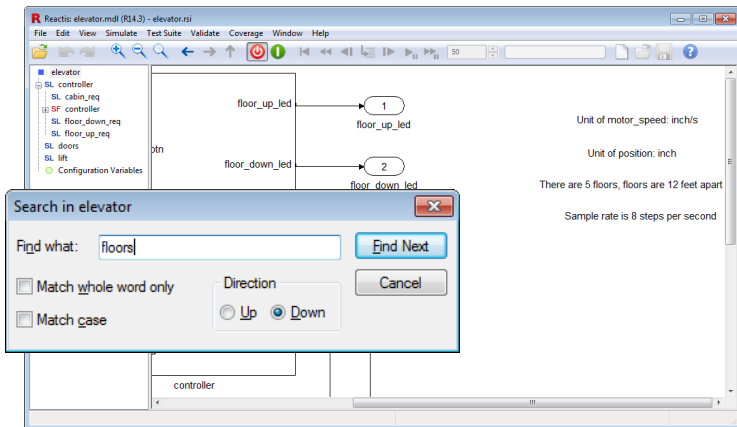
Text Search Enhancement

Text search now searches Simulink annotations



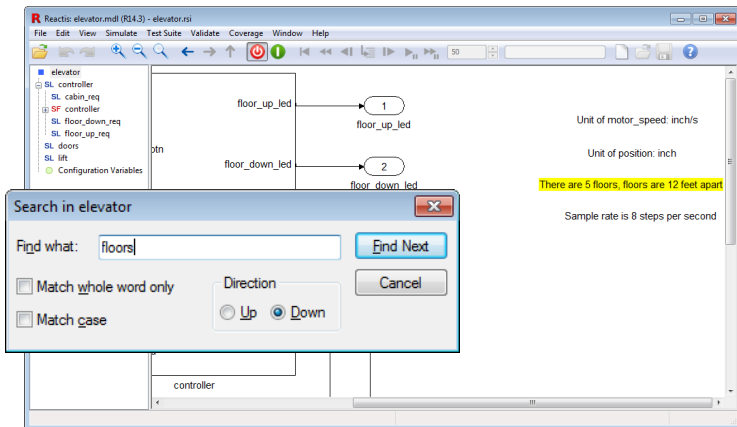
Text Search Enhancement

Text search now searches Simulink annotations



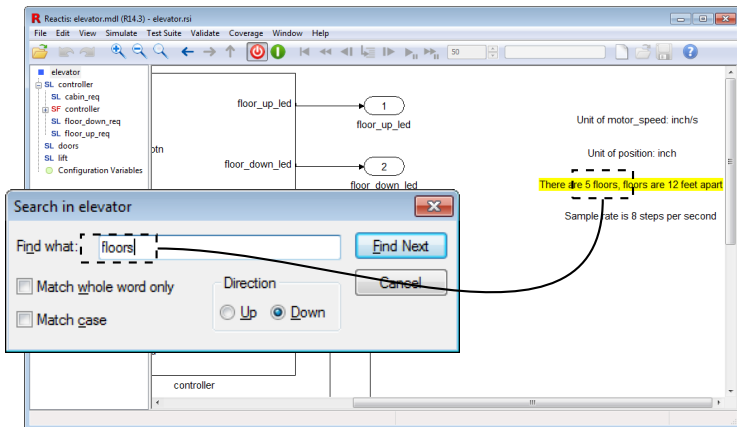
Text Search Enhancement

Text search now searches Simulink annotations



Text Search Enhancement

Text search now searches Simulink annotations



Display Model Info and Doc Block Content

Display contents of Model Info and Doc blocks

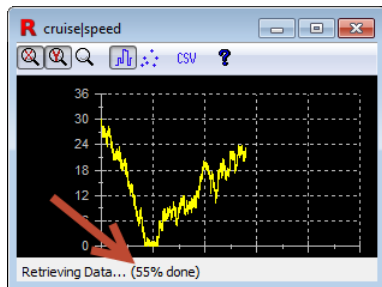
The screenshot displays the Reactis software interface for a model named 'cruise.mdl (R2009a) - cruise.rsi'. The main window shows a block diagram with the following components and connections:

- Cruise Model**: Version 1.113, Modified 10-Apr-2013 15:34:52.
- DOC**: Text block.
- Block Diagram**: Includes 'fncallgen', 'fcg', 'trigger', 'onOff', 'acceResume', 'cancel', and 'deceSet'.

A 'DOC block' window is open, displaying the following text:

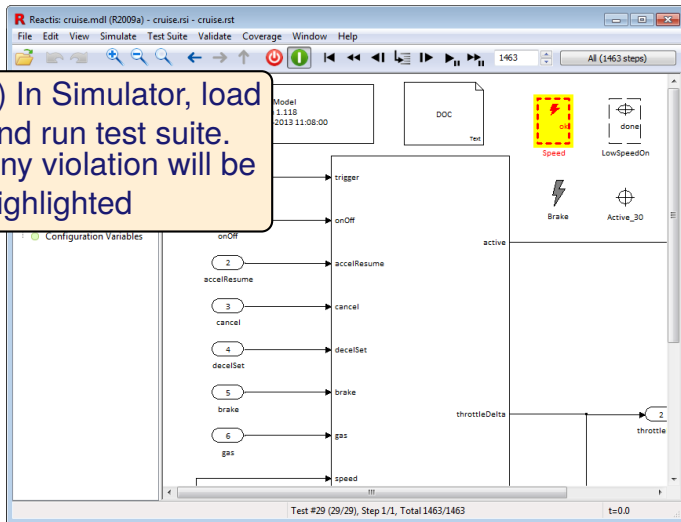
This model implements a cruise control to demonstrate the different capabilities offered by Reactis to test and validate models. Chapter 3 of the Reactis User's Guide describes how to use Reactis with this model.

Show Progress when Opening Scope



Run-to-Violation

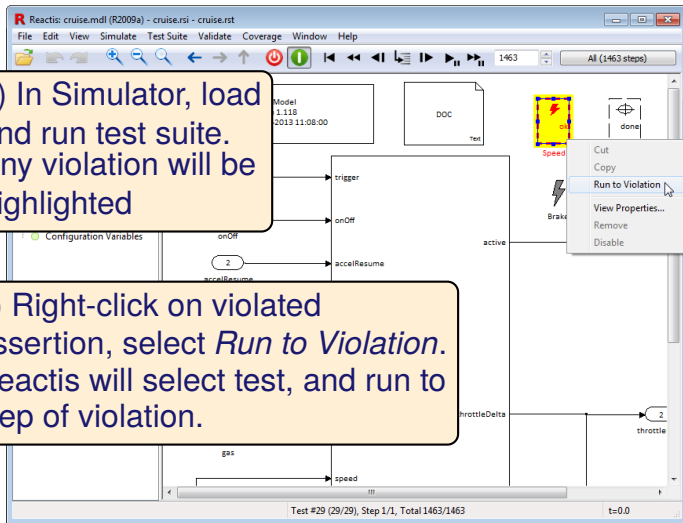
1) In Simulator, load and run test suite. Any violation will be highlighted



Run-to-Violation

1) In Simulator, load and run test suite. Any violation will be highlighted

2) Right-click on violated assertion, select *Run to Violation*. Reactis will select test, and run to step of violation.



Lookup Table Enhancements

- ▶ Enhancements to N-d lookup tables, Pre Lookup, Interpolation
 - ▶ Track coverage
 - ▶ Support more type combinations
- ▶ In coverage reporting, show breakpoint values

	<code>[-inf, c1]</code>	<code>[c1, c2]</code>	<code>[c2, c3]</code>	<code>[c3, +inf]</code>
<code>[-inf, r1]</code>	1/4	2/3	5/1	1/6
<code>[r1, r2]</code>	1/5	2/2	3/11	2/1
<code>[r2, +inf]</code>	1/1	1/3	1/2	1/7

Old

	<code>u2 < 1.0</code>	<code>u2 ≥ 1.0</code>	<code>u2 ≥ 10.0</code>	<code>u2 ≥ 100.0</code>
<code>u1 < 1.0</code>	1/4	2/3	5/1	1/6
<code>u1 ≥ 1.0</code>	1/5	2/2	3/11	2/1
<code>u1 ≥ 10.0</code>	1/1	1/3	1/2	1/7

New

- ▶ Tester gets better coverage for lookup tables

API Functions to Export Coverage Reports

- ▶ *rsTesterWithReport* produces a coverage report after a Tester run
- ▶ *rsSimRunSuiteWithReport* produces a coverage report after running a test suite in Simulator