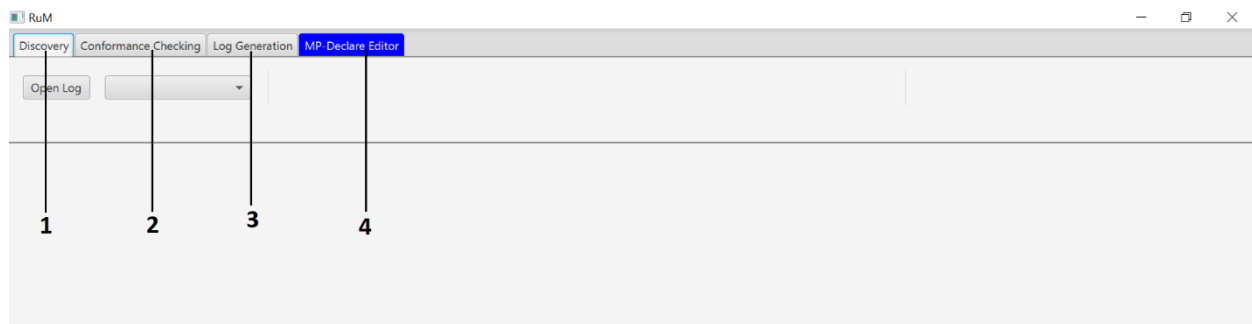# USER MANUAL

About

This document presents a manual for RuM tool. It mentions each tab and what the elements are responsible for in the corresponding sections. Before you proceed, the following materials are recommended to follow the manual more clearly. You can access these papers via Google Scholar.

1. F. M. Maggi, C. Di Ciccio, C. Di Francescomarino & T. Kala. **"Parallel algorithms for the automated discovery of declarative process models"**. Information Systems pp. 136-152, 2018.

2. A. Burattin, F. M. Maggi and A. Sperduti, **"Conformance checking based on multiperspective declarative process models,"** Expert Systems with Applications, pp. 194-211, 2016.

1. Main Screen

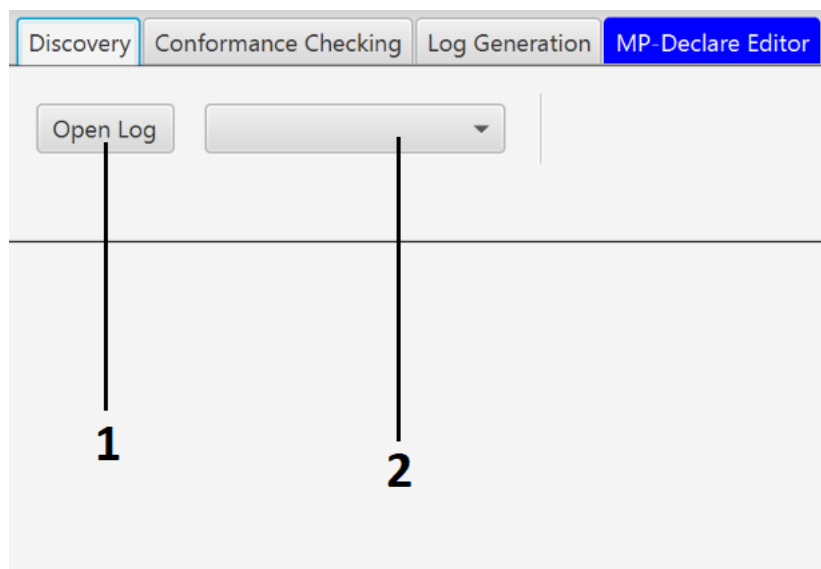After starting the tool, Figure 1 is observed.

Tab 1 => For declarative process discovery

Tab 2 => For conformance checking

Tab 3 => For log generation

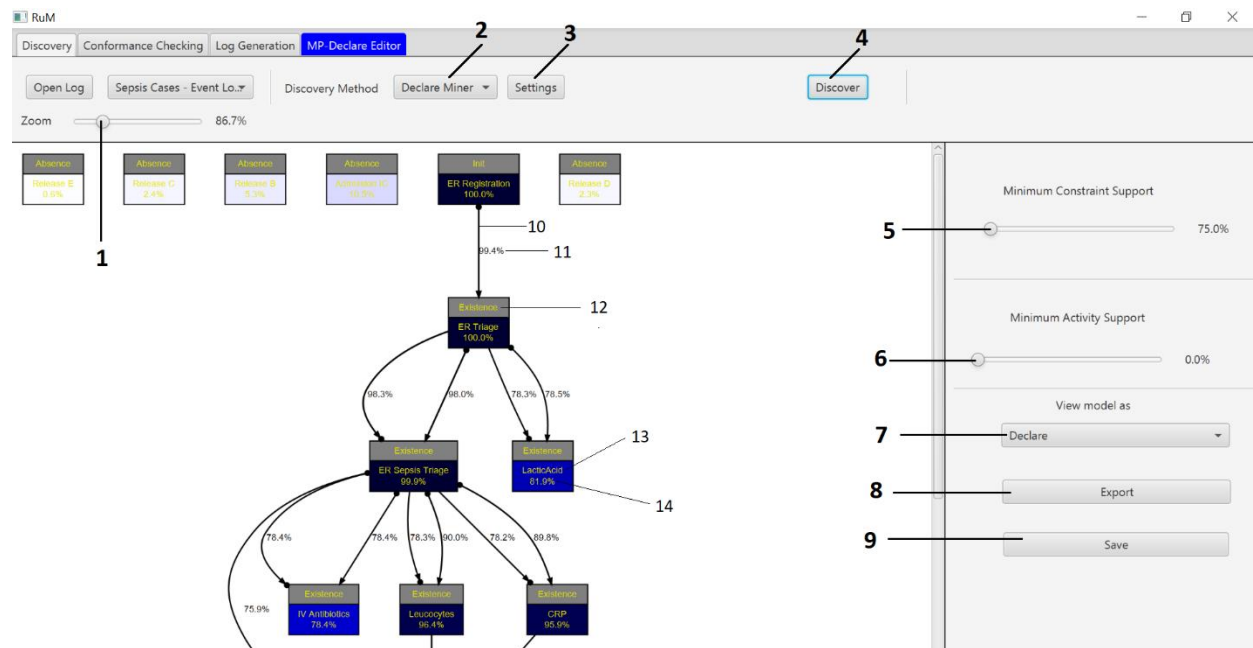Tab 4 => For editing a MP-Declare model

2. Discovery tab

This is Discovery tab with a blank screen. An event log can be opened from the file system by clicking (1). Opened logs can be chosen later from (2).

For a log only: .mxml, .mxml.gz, .xes and .xes.gz formats are supported.

2.1 Discovery output

(1) – Zoom slider is used to zoom in or out the view. It can be scrolled with mouse.

(2) – to select a discovery method: Declare Miner [1] or Minerful [2]

(3) – button to open configuration panel for the method selected in (2) (see **Section 2.2**).

(4) – to start the discovery with the selected file (see **Section 2** (2)) and the selected method in (2)

(5) – slider to set a value such that constraints having a support less than this value are not displayed

(6) – slider to set a value such that activities having a support less than this value are not displayed

(7) – to select how to see the result model: Declare, Textual or Automaton

(8) – to save in the file system in the format depending on (7)

(9) – to save the result as DECL [3] format in the tool (not in the file system).

(10) – A binary Declare template

(11) – Constraint support

(12) - A unary Declare template

(13) – Activity's name

(14) – Activity support

2.2 Discovery configuration for Declare Miner
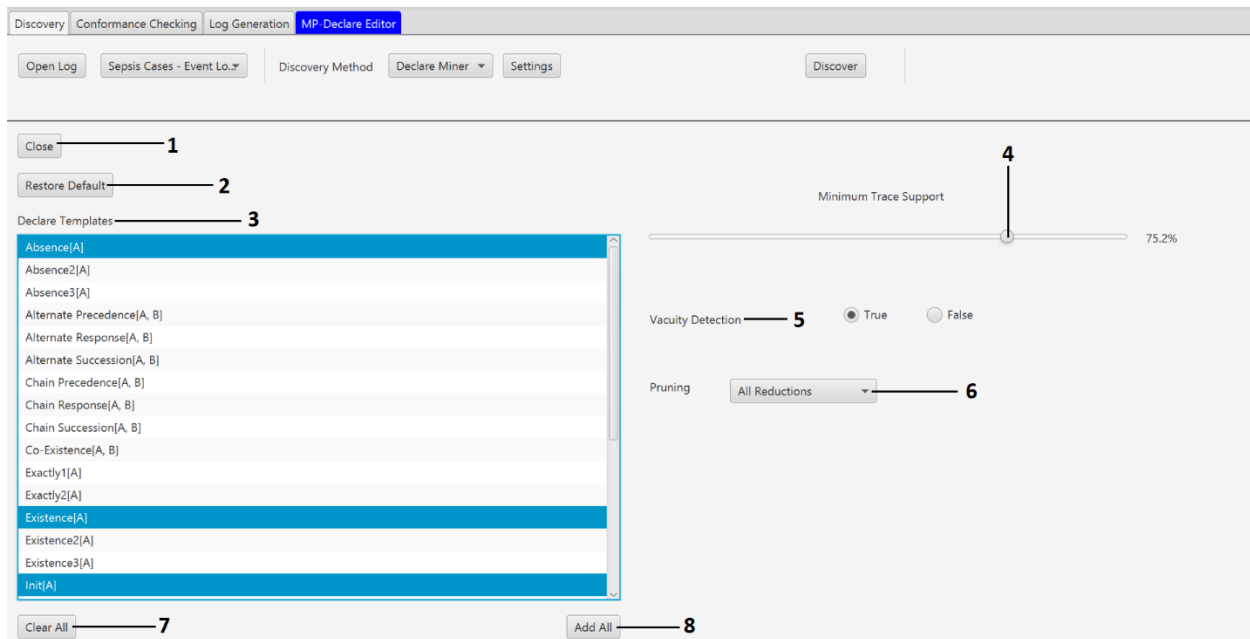
(1) – to close the configuration

(2) – to restore the default settings for the configuration

(3) – A list of Declare templates to observe in the output, multiple selection is possible (ctrl for Windows, Cmd for Mac) the button used to multi-select in the file system.

(4) – set minimum constraint support for the output

(5) – enable vacuity detection

(6) – to select a pruning option for the output

(7) – to clear the selections in (3)

(8) – select all templates in (3)

After the settings, a discovery can be started using *Discover* button (see **Section 2.1** (4)).
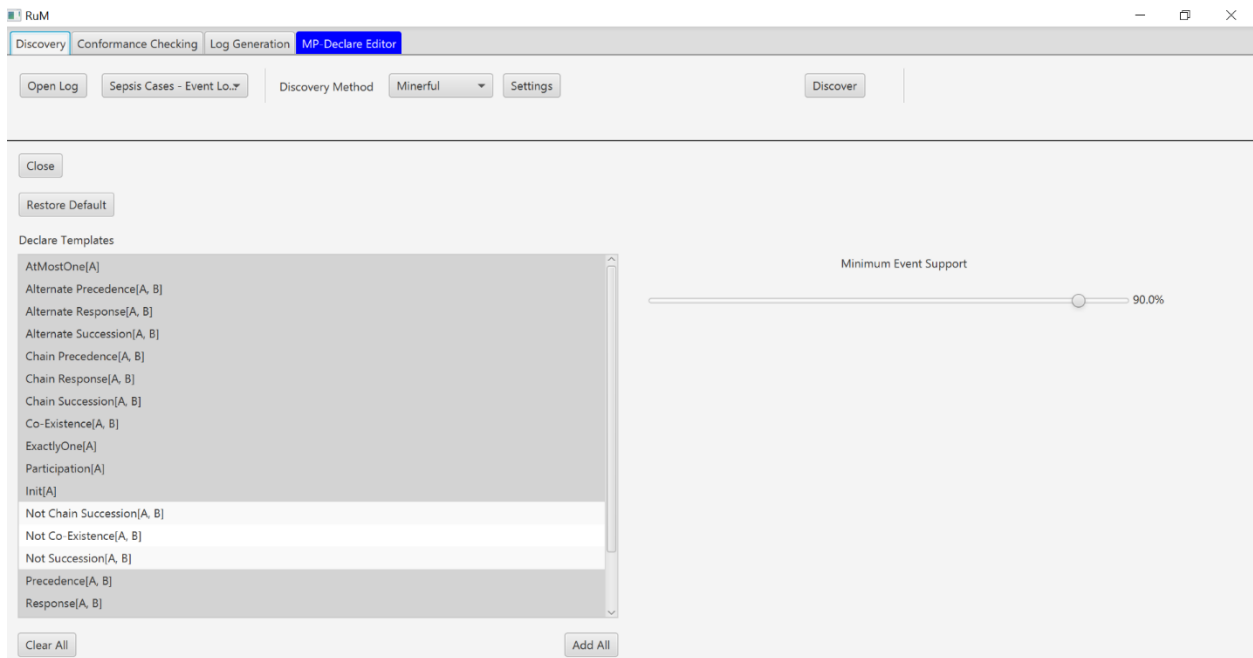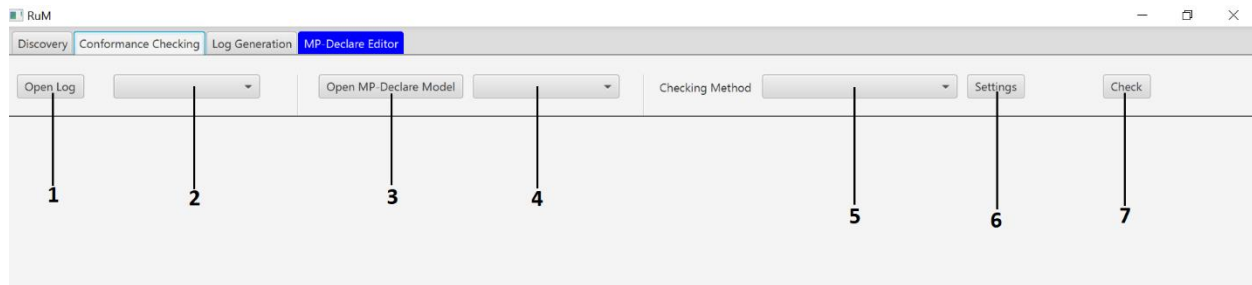
2.3 Discovery configuration for Minerful

Figure 5 – Configuration screen for Minerful

Figure 5 displays the configuration screen for Minerful. The slider *Minimum Event Support* is used to set a minimum constraint support for the output. The rest is same as in **2.2 Discovery configuration for Declare Miner.**

3. Conformance Checking tab

 (1) – to open an event log as described in Tab 1: Discovery (see **Section 2**).

(2) – opened log can be chosen again from here

(3) – to open a MP-Declare model in DECL format

(4) – opened model with (3) can be chosen again from here

(5) – to select a checking method: Declare Analyzer [4], DataAware Declare Replayer [5] and Declare Replayer [6]

(6) – to open the setting panel for the method selected in (5)

(7) – to start conformance checking with log in (2) and model in (4) using the settings

3.1 Result view for Declare Analyzer

 (1) – to switch Detailed results

(2) – A table showing the activations, fulfillments and violations for each constraint in the input model.

3.2 Detailed view for Declare Analyzer

(1) – to select a trace in the input log

(2) – to select a constraint in the input model

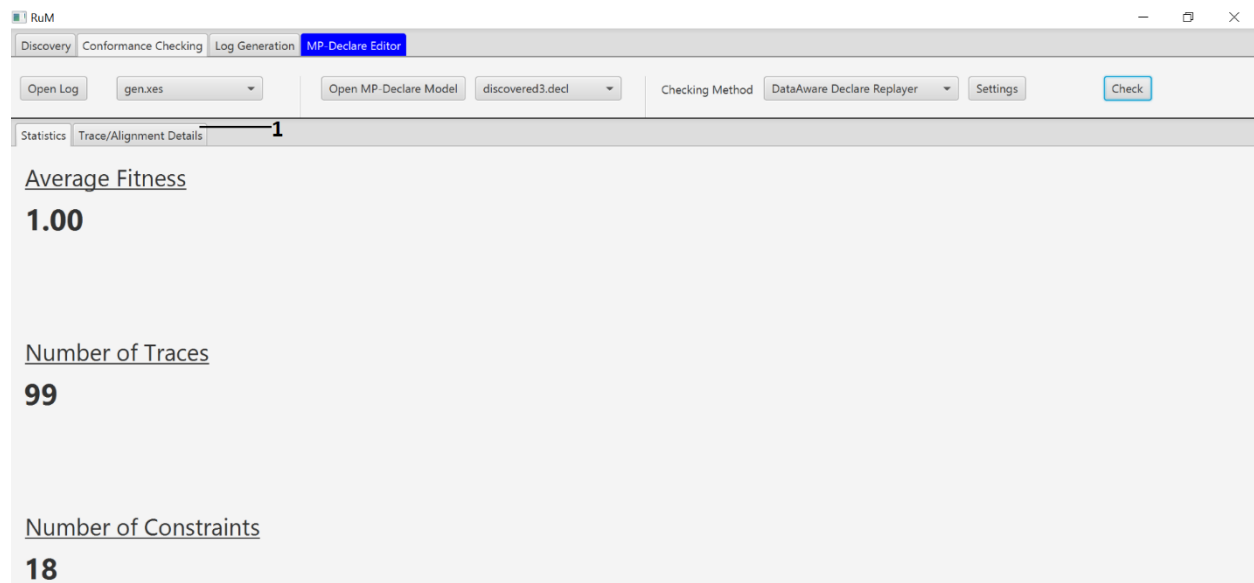(3) – a legend showing what the colors refer to

(4) – sorting option for the traces: Activation, violation, fulfillment numbers and Alphabetical

(5) – a result view from the trace selected in (1) and constraint selected in (2)

(6) – An event in the trace

(7) – Details of an event, it appears when an event is moused over.

3.3 Result view for Declare Replayers

(1) – to switch Trace/Alignment details

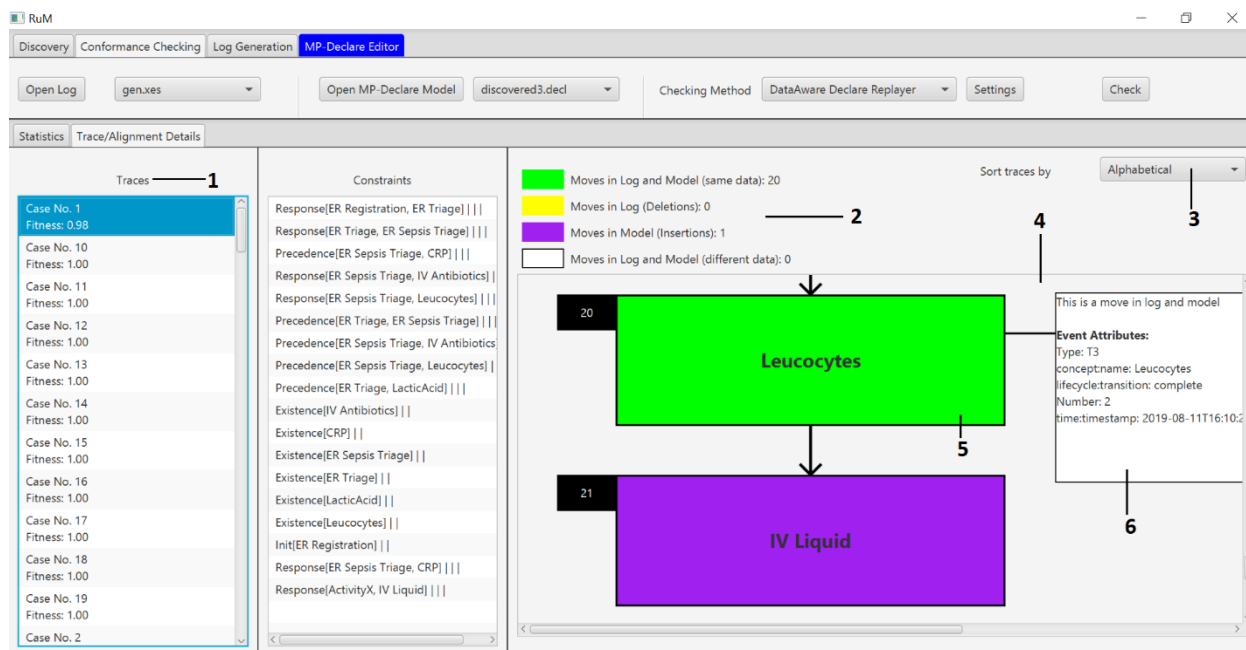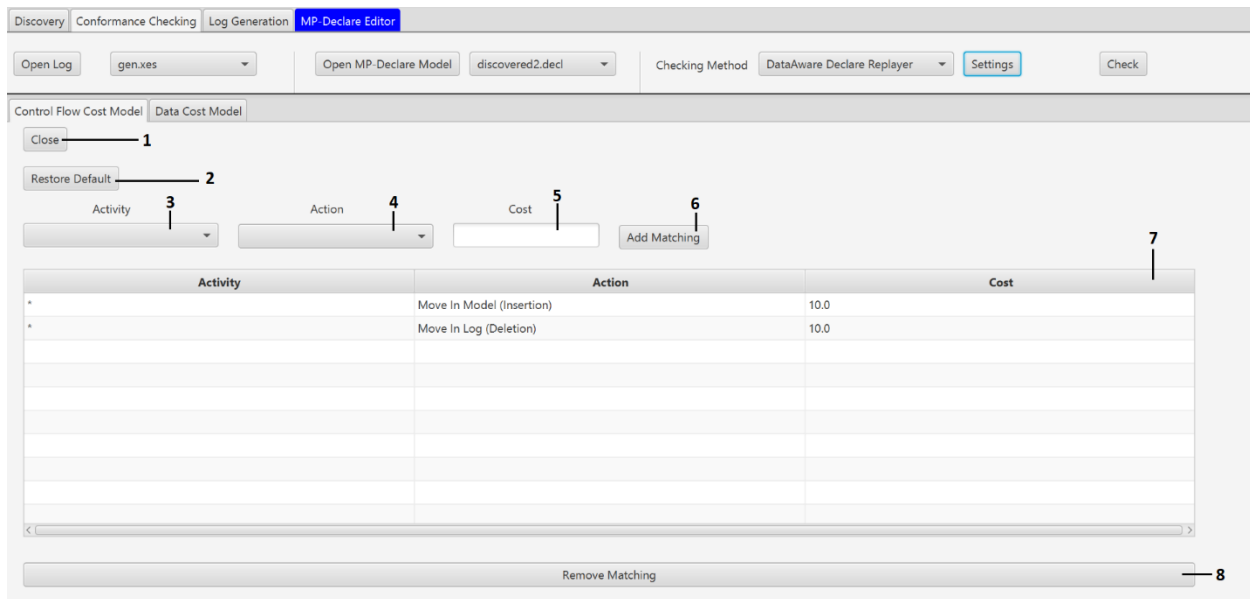## 3.4 Trace/Alignment view for Declare Replayers

Figure 10 – Trace/Alignment Details for DataAware Declare Replayer

Figure 10 displays *Trace/Alignment Details* for DataAware Declare Replayer. Here:

(1) – to select a trace in the input log

(2) – a legend about what the colors refer to

(3) – sorting options for the traces

(4) – an alignment view of selected trace in (1)

(5) – an event in the selected trace

(6) – details of an event, it appears when an event is moused over.

NOTE for Declare Replayer: The result screen for Declare Replayer is same as in Figure 9. *Statistics* tab is the same as mentioned in DataAware Declare Replayer. *Trace/Alignment Details* tab is mostly same as mentioned in DataAware Declare Replayer. The difference is that the legend in Declare Replayer does not contain *Moves in Log and Model (different data)* and *Moves in Log and Model (same data)*. It has *Moves in Log*, *Moves in Model* and *Moves in Log and Model*.

3.5 Control Flow Cost Model setting for Declare Replayers

(1) – to close the settings screen

(2) – to restore the default settings

(3) – to select an activity

(4) – to select an action for activity in (3): Move in Model (Insertion) or Move in Log (Deletion)

(5) – to define a cost value

(6) – to define a cost model

(7) – table showing the cost models

(8) – to remove a model from (7)

3.6 Data Cost Model setting for DataAware Declare Replayer
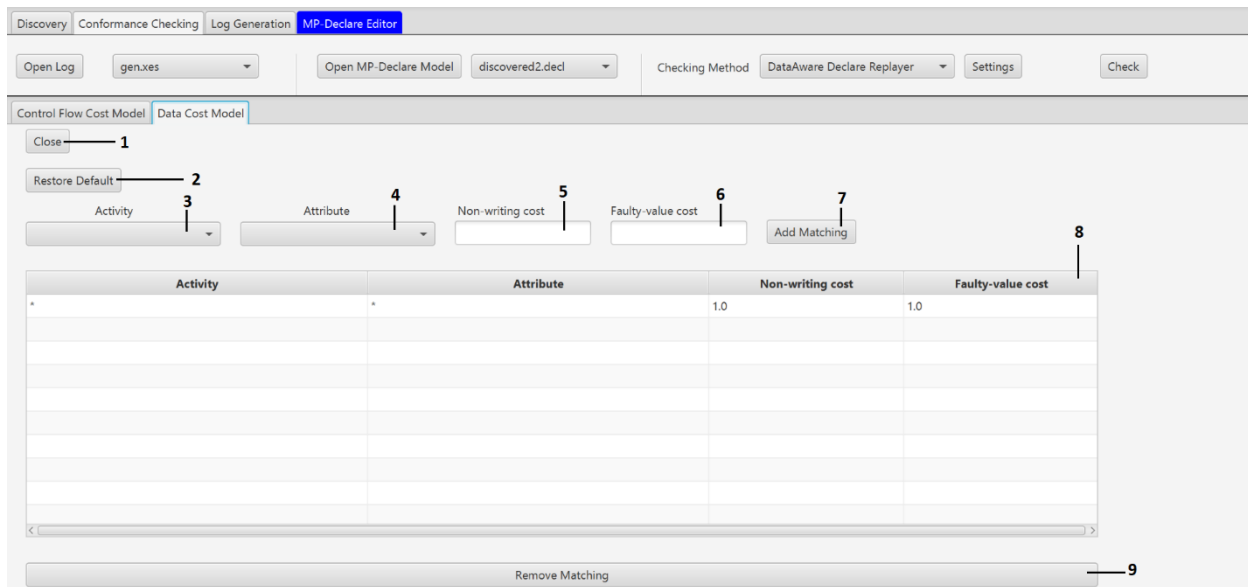
Figure 12 – Data Cost Model setting for DataAware Declare Replayer

Figure 12 shows Data Cost Model setting for Data Aware Declare Replayer. Here:

(1) – to close the setting screen

(2) – to restore default settings

(3) – to select an activity for a cost model

(4) – to select an attribute for a cost model

(5) – to define a non-writing cost value

(6) – to define a faulty-value cost value

(7) – to define a data cost model

(8) – defined data cost models

(9) – to remove a selected data cost model in (8)

4. Log Generation tab
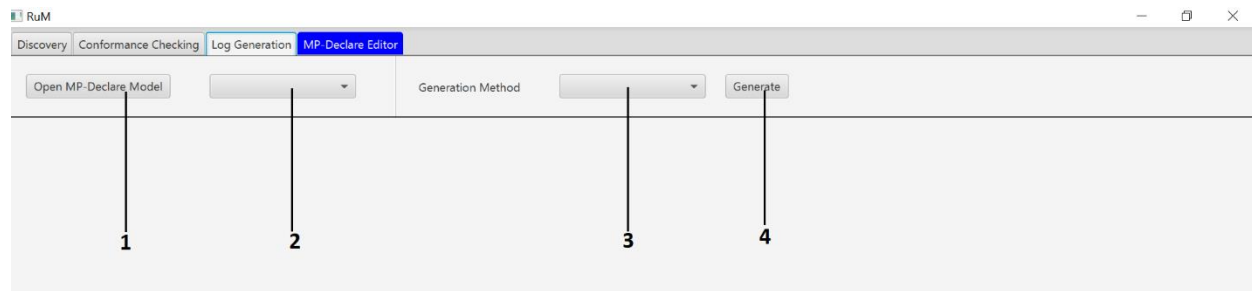
(1) – to open a MP-Declare model in DECL format in the file system

(2) – opened models can be chosen later here

(3) – a method to generate a log: Alloy Log Generator [7] and Minerful Log Generator [8]

(4) – to start log generation with a model in (2) and method configuration

4.1 Configuration screen for Alloy Log Generator

(1) – for destination of the output log

(2) – to select a location in the filesystem for the output log

(3) – for minimum trace length of the output log

(4) – for maximum trace length of the output log

(5) – for number of traces of the output log

(6) – button to increment by 1 the value (3), (4) or (5)

(7) – button to decrease by 1 the value (3), (4) or (5)

(8) – to enable vacuity detection during the generation

(9) – to enable to generate negative traces during the generation

(10) – to ensure even length distribution during the generation

(11) – constraints in the input model

In Minerful Log Generator, the configuration screen does not have (8), (9) and (10); but has the remaining elements in Figure 14.

5. MP-Declare Editor tab
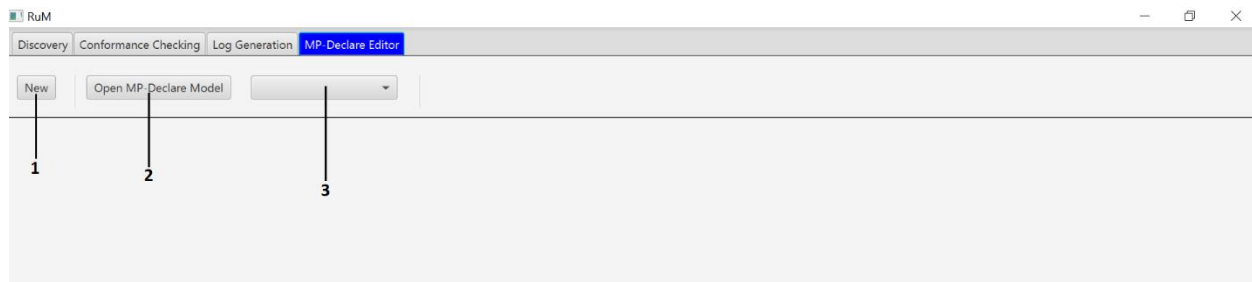
(1) – to open a new editor with three tabs

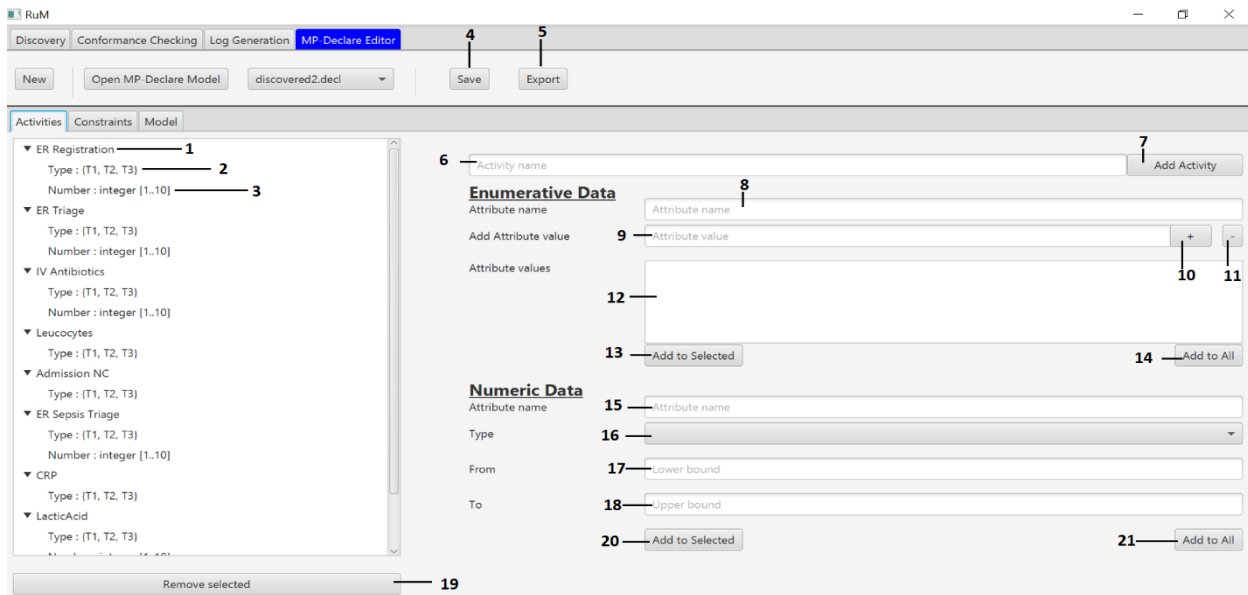(2) – to open a MP-Declare model in DECL format from the file system.

(3) – opened models can be chosen later from here

5.1 Activities tab

(1) – An activity

(2) – An enumerative data

(3) – A numeric data

(4) – to save the model inside the tool

(5) – to save the model both inside the tool and in the file system

(6) – to insert new activity name

(7) – button to add new activity

(8) – to insert a name for enumerative attribute

(9) – a value for the enumerative attribute

(10) – to add the value to the enumerative attribute

(11) – to remove the value from the enumerative attribute

(12) – defined values for the enumerative attribute

(13) – button to insert the enumerative attribute to the selected activities

(14) – button to add the enumerative attribute to the all activities

(15) – to insert a name for numeric attribute

(16) – to select a type for the numeric attribute: integer or float

(17) – to set a lower bound for the numeric attribute

(18) – to set an upper bound for the numeric attribute

(19) – to remove a selected activity or attribute

(20) – button to insert the numeric attribute to the selected activities

(21) – button to add the numeric attribute to the all activities

5.2 Constraints tab
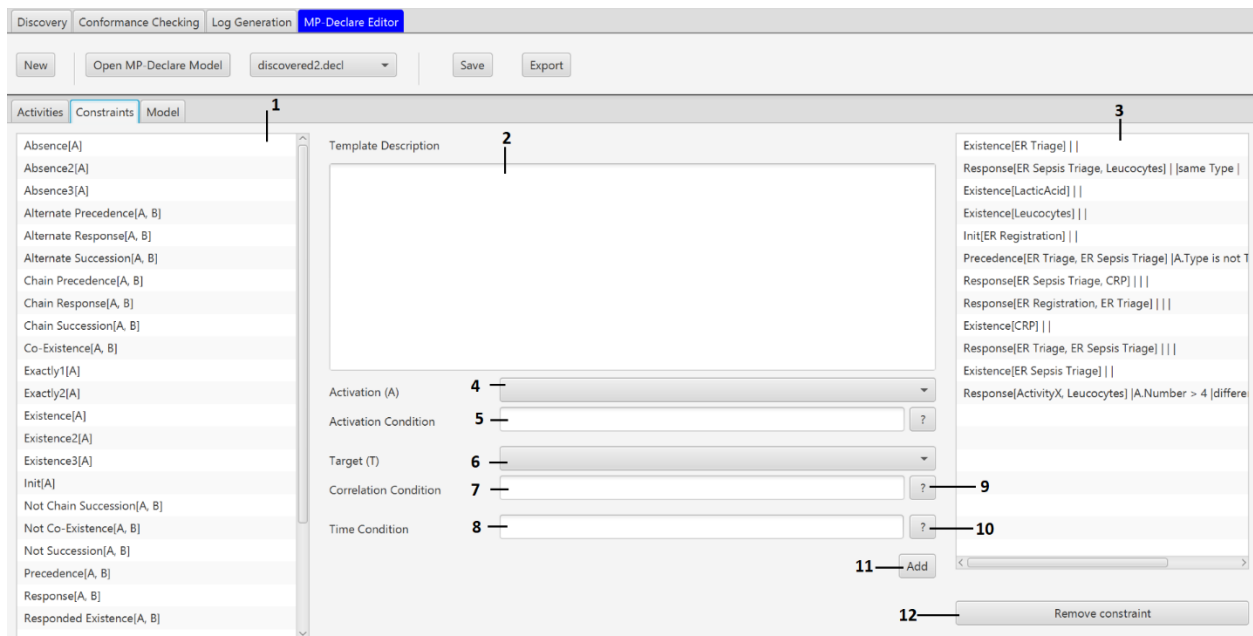
(1) – Declare templates list

(2) – Description of a Declare template selected in (1)

(3) – Existing constraints list in the input model

(4) – to select an activity for the activation

(5) – to insert an activation condition

(6) – to select an activity for the target

(7) – to define a correlation condition

(8) – to insert a time condition

(9) – tutorial for defining activation and correlation conditions

(10) – tutorial for defining time conditions

(11) – to add a constraint in (3)

(12) - remove a selected constraint in (3)

5.3 Model tab
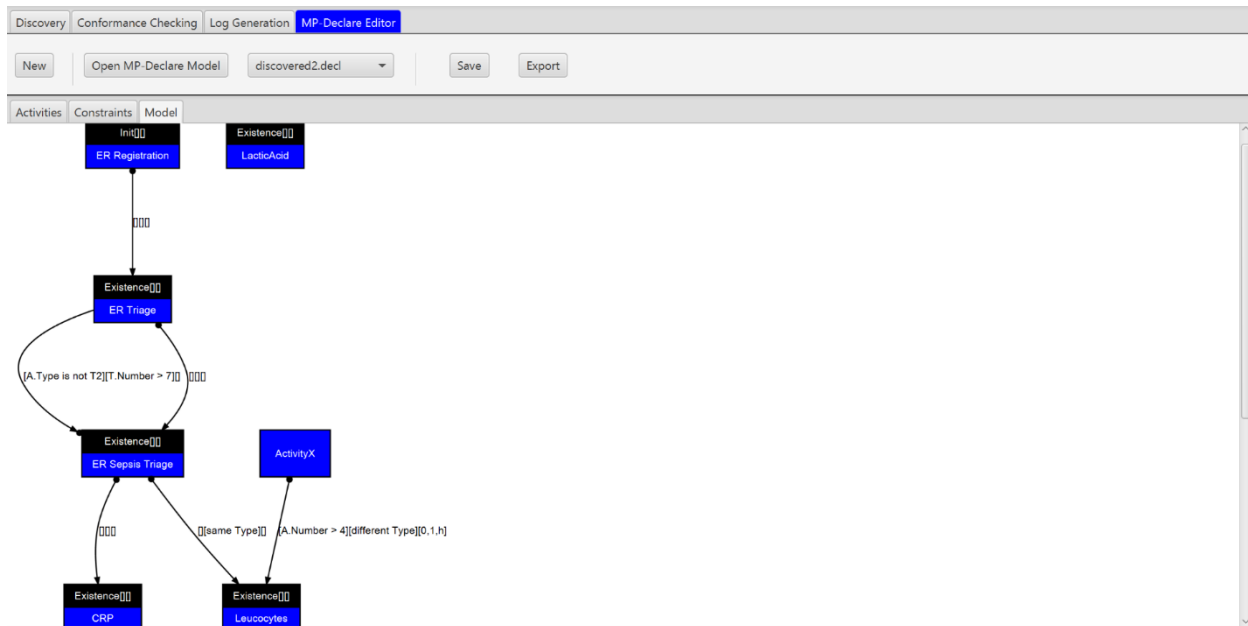
The screen is prepared by the existing constraints in **Constraints tab**.

1- Black boxes represent Existence templates: [][] refers to activation and time conditions

2- Arrows are binary Declare templates: [][][] refers to activation, correlation and time conditions

3 – Blue boxes are activities from Activities tab.

6. References

[1] F. M. Maggi, C. Di Ciccio, C. Di Francescomarino & T. Kala. "Parallel algorithms for the automated discovery of declarative process models". *Information Systems* pp. 136-152, 2018.

[2] C. Di Ciccio & M. Mecella. "On the discovery of declarative control flows for artful processes". *ACM Transactions on Management Information Systems (TMIS), 5(4)*, p. 24, 2015

[3] V. Skydanienko, "Data-aware Synthetic Log Generation for Declarative Process Models," *Masters Thesis, University of Tartu*, 2018.

[4] A. Burattin, F. M. Maggi and A. Sperduti, "Conformance checking based on multiperspective declarative process models," *Expert Systems with Applications*, pp. 194-211, 2016.

[5] C. Mawoko, "Aligning Data-Aware Declarative Process Models and Event Logs" *Masters Thesis, University of Tartu*, 2019.

[6] M. de Leoni, F. M. Maggi and W. M. van der Aalst, "An alignment-based framework to check the conformance of declarative process models and to preprocess event-log data," *Information Systems*, pp. 258-277, 2015.

[7] V. Skydanienko, "Data-aware Synthetic Log Generation for Declarative Process Models," *Masters Thesis, University of Tartu*, 2018.

[8] C. Di Ciccio, M. L. Bernardi, M. Cimitile and F. M. Maggi, "Generating Event Logs Through the Simulation of Declare Models," *Lecture Notes in Business Information Processing*, pp. 20-36, 2015.