Extending libSedML to support CellML models

Dongxue Amy You
David Nickerson

**AUCKLAND
BIOENGINEERING INSTITUTE**
THE UNIVERSITY OF AUCKLAND
NEW ZEALAND

Te Whare Wānanga o Tāmaki Makaurau

MIASE

...is a community effort to identify the Minimum Information About a Simulation Experiment, to help enabling the reproducible description of simulation experiments.

- The common set of information a modeller needs to provide in order to enable the execution and reproduction of a numerical simulation experiment, derived from a given set of quantitative models.
- Designed to help modelers and software tools to exchange their simulation settings.
- Information about the models to use; the simulation steps; and the output.

SED-ML

The Simulation Experiment Description Markup Language is an XML-based format for encoding simulation experiments, following the requirements defined in the MIASE guidelines. SED-ML allows to define the model to use, the experimental task to run and which result to produce. SED-ML currently only covers part of MIASE compliant simulations. On the other hand, the formalisation of SED-ML forces the information to be more precise than the bare MIASE requirements.

<http://sed-ml.org>

SED-ML object model

- Model
 - Used to reference the models used in the simulation experiment
 - Independent of model encoding format
 - Defines methods to change source model prior to simulation
- Simulation
 - Simulation settings and steps taken during simulation
 - Type of simulation; algorithm to be used (KiSAO); associated parameters
- Task
 - Combination of a model and a simulation

SED-ML object model

- DataGenerator
 - Simulation result
 - Post-processing description
- Output
 - Defines the output of a simulation
 - 2D or 3D plot; data table
 - Described using DataGenerator's

SED-ML tool support

- jlibsedml – a Java library to facilitate SED-ML support for applications to read, validate, edit, and write SED-ML documents in compliance with MIASE guidelines. Also contains modules to execute simulation tasks and produce outputs
 - Virtual Cell and Centre for Systems Biology at Edinburgh
- libSedML – C# / .NET libraries and tools supporting SED-ML
 - libSedML; libSedMLRunner
 - libSedMLScript – simple script format designed for human use; python based

SED-ML Script

SED-ML Script - an application for constructing SED-ML files

The SED-ML Script application constructs SED-ML documents using all three libraries, `libSedML`, `libSedMLRunner` and `libSedMLScript`. Thus, not only can SED-ML files be created or modified, they can also be executed right then and there.

The application is currently implemented using the WinForms API. It won't look nice on non-windows platforms, but at least it will work just fine on OS X and Linux (unless you refuse to install MONO).

[File Releases](#) | [Source Code](#) | [Examples](#) | [Specification](#)

maintained by Frank Bergmann

Workshop Progra... x SBSI Visual Web I... x jlibsedml | Downl... x Simulation Tool x BioModels.net: Ml... x

libsedml.sourceforge.net/libSedML/Simulation_Tool.html

Welcome Screenshots Libraries SedMLScript **Simulation Tool**

Simulation Tool

Simulation Tools - Carrying out interactive Simulation Experiments

The Simulation Tool was originally developed as part of [SBW](#). It not only provides a easy to use interface for [RoadRunner](#), even third party simulators have been enabled to run with it, and thus COPASI can be used from it just as well as the SBML ODE Solver (though to be fair, some of the advanced features are only provided by RoadRunner).

What has all that to do with SED-ML? Since the Simulation Tool has been opened up, to provide a plug-in API, one such plug-in provides support for the use of SED-ML. Currently, it just runs the SED-ML simulation experiment. Additionally, all models (or rather all SBML models) encoded within the SED-ML file, can be further analyzed with the remaining simulation experiments: time course simulation, steady state analysis, pulse experiments, (steady state) parameter scans and more.

[File Releases](#) | [Source Code](#) | [Examples](#) | [Specification](#)

maintained by Frank Bergmann

Modifications for CellML

- Wrap two command line utilities with C# modules
 - UniqueID: map variables specified using XPath to source variables in a CellML model
 - MyRunCellML: performs a single SED-ML task (model+simulation)
- Then behaves as any other simulation tool utilised by libSedMLRunner
- Created a specialised wizard for CellML models

SED-ML Script

```
AddTimeCourseSimulation('simulation', 'KISAO:0000019', 0, 0, 1500, 1500)

AddModel('BR', '../models/1977_beeler/experiments/periodic_stimulus.xml')

AddModel('BREJ',
'../models/1977_beeler/experiments/1980_ebihara_johnson.xml')

AddModel('BRDR',
'../models/1977_beeler/experiments/1987_drouhard_roberge.xml')

AddTask('BR-task', 'simulation', 'BR')
AddTask('BREJ-task', 'simulation', 'BREJ')
AddTask('BRDR-task', 'simulation', 'BRDR')

AddColumn('BR-time', [['BR-time', 'BR-task', 'time']])
AddColumn('BR-Vm', [['BR-Vm', 'BR-task', 'Vm']])
AddColumn('BREJ-time', [['BREJ-time', 'BREJ-task', 'time']])
AddColumn('BREJ-Vm', [['BREJ-Vm', 'BREJ-task', 'V']])
AddColumn('BRDR-time', [['BRDR-time', 'BRDR-task', 'time']])
AddColumn('BRDR-Vm', [['BRDR-Vm', 'BRDR-task', 'V']])

AddPlot('plot1', 'Action Potentials', [['BR-time', 'BR-Vm'], ['BREJ-
time', 'BREJ-Vm'], ['BRDR-time', 'BRDR-Vm']]);
```

Simulation outputs

Time course experiment wizard

The image shows a software dialog box titled "Create Simple Time Course Experiment". It is divided into three main sections: "Specify Model:", "Define Simulation:", and "Define Output:". The "Specify Model:" section has a "Model Source:" text field and a "Browse" button. The "Define Simulation:" section contains four input fields: "Initial Time:" (0.0), "Output Start Time:" (0.0), "Number of Points:" (1000), and "Output End Time:" (10.0). Below these is a "Number of Copies:" field set to 1. The "Define Output:" section contains six checkboxes: "Create Output For Floating Species" (checked), "Create Output For Boundary Species" (unchecked), "Create Output For Compartments" (unchecked), "Create Output For Parameters" (unchecked), "Generate Plot" (checked), "Use separate plot for each Graph" (unchecked), and "Generate Report" (unchecked). At the bottom right are "Cancel" and "Create" buttons.

Create Simple Time Course Experiment

Specify Model:

Model Source:

Browse

Define Simulation:

Initial Time: Output Start Time:

Number of Points: Output End Time:

Number of Copies:

Define Output:

☒ Create Output For Floating Species

☐ Create Output For Boundary Species

☐ Create Output For Compartments

☐ Create Output For Parameters

☒ Generate Plot

☐ Use separate plot for each Graph

☐ Generate Report

Cancel Create

CellML time course wizard

CreateTimeCourseExperimentForCellML

Define Simulation

Initial Time: 0.0 Output Start Time: 0.0

Number of Points: 1000 Output End Time: 10.0

Add Models

Add New Simulation

Define Simulation

Initial Time: 0.0 Output Start Time: 0.0

Number of Points: 1000 Output End Time: 10.0

Add Models

Model: periodic_stimulus.xml
Source:

Add New Simulation

Define Output:

☒ Generate Plot for Each Simulation ☐ Generate Report

Cancel Create

Graphical output