

# Tutorial:

## Dynamics computation plugin and Chart plotting plugin for Metaplab

Luca Marchetti      Alberto Castellini

Verona University, Computer Science Department,  
Strada Le Grazie 15, 37134 Verona, Italy.

{luca.marchetti, alberto.castellini}@univr.it

March 4, 2009

In this tutorial we introduce the main features of two fundamental tools employed by MetaPlab, respectively, to compute the time evolution of and MP system and to plot the related charts. The two tools presented in the following are called *Dynamics computation plugin* and *Chart plotting plugin*. The first one can be run when an MP graph has been drawn and initial conditions and regulation functions have been defined. The second tool is usually launched after the system dynamics has been computed (or imported from external files) and it enables the user to graphically visualize and to analyze time evolution trends.

### Dynamics computation plugin

In order to process MP models we employ plugin tools performing various tasks. We launch the plugin manager by clicking on the rightmost button of the top-toolbar, highlighted in Figure 1, or by using the related command of the `MetaPlab` menu. The plugin manager of MetaPlab is the tool which permits to get advantage of the plugin architecture of the software introduced in [1]. Whoever can write a plugin in order to perform a wide variety of operations on MP model data.

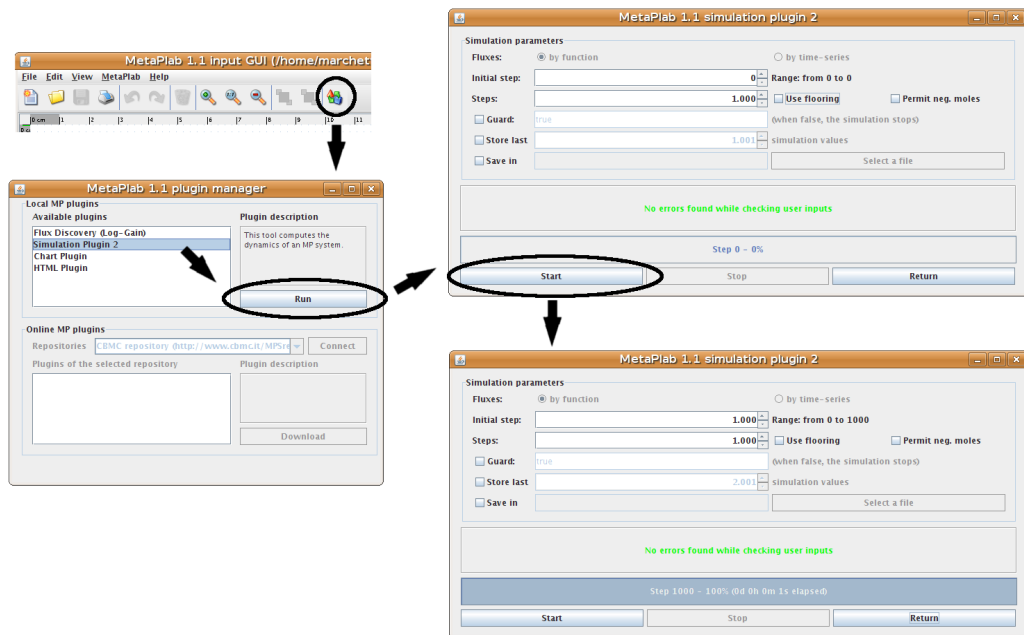


Figure 1: The procedure to simulate the dynamics of an MP model.

In this section we employ two plugins which are distributed within the software. The first one is called dynamics computation and it is used to compute the dynamics of the model currently loaded. To launch the plugin we select its entry in the plugin list prompted by the plugin manager and then we launch it by pressing the **Run** button. Before computing the dynamics, the plugin visualizes a window where the user can insert some simulation parameters, such as the number of simulation steps which we set to 1000. Then we start the simulation by pressing the **Start** button. After few seconds (execution time depends on the number of steps performed), the simulation procedure ends and we can go back to the plugin manager by clicking on the **Return** button. All the steps described before are depicted in Figure 1. The simulation plugin of MetaPlab offers to the user many functionalities which can be used to manage the simulation process. Among them we point out the possibilities of:

- stopping or resuming the dynamics computation by using the **Start** and **Stop** buttons;
- changing the initial step from which to start the computation of the dynamics, if possible;
- setting the number of steps to compute;

- defining a boolean expression which must be true during the simulation procedure (when it becomes false, a window is prompted to the user and the simulation is paused);
- changing the number of steps which the user wants to store in memory (if this number is less than the number of simulation steps, then only the last values will be maintained in memory)<sup>1</sup>;
- storing to file the time-series computed by the software.

## Chart plotting plugin

When the dynamics of our model has been computed, we plot it by means of another plugin called **Chart plotting plugin** which can be launched at the same way of the previous one (Figure 2). This plugin visualizes a window which enables to create different types of chart. The plugin window is divided in two parts: in the upper side it is depicted the chart that the user wants to study, in the bottom side there are three panels which enables to create line charts, phase charts and to set some general options.

First of all, we create a line chart which displays the concentration of the three substances involved in our test case model ( $A$ ,  $B$  and  $C$ ). The result of this operation is depicted in the right side of Figure 2. Moreover we can create phase charts using the second panel of the plugin window as depicted in Figure 3. Charts created by the plugin can be exported to file,

<sup>1</sup>This functionality can be useful when we want to investigate the behavior of an oscillatory phenomenon because it permits to run long simulations without needing a big quantity of memory.

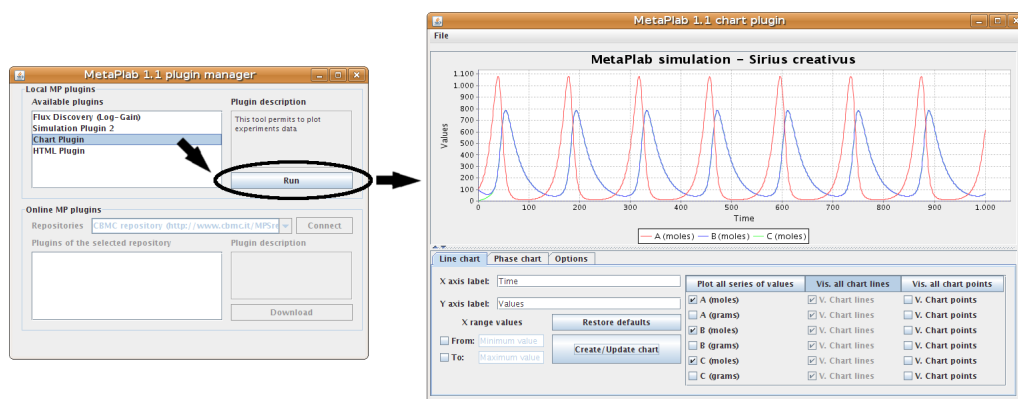


Figure 2: The procedure to display the simulation values of an MP model.

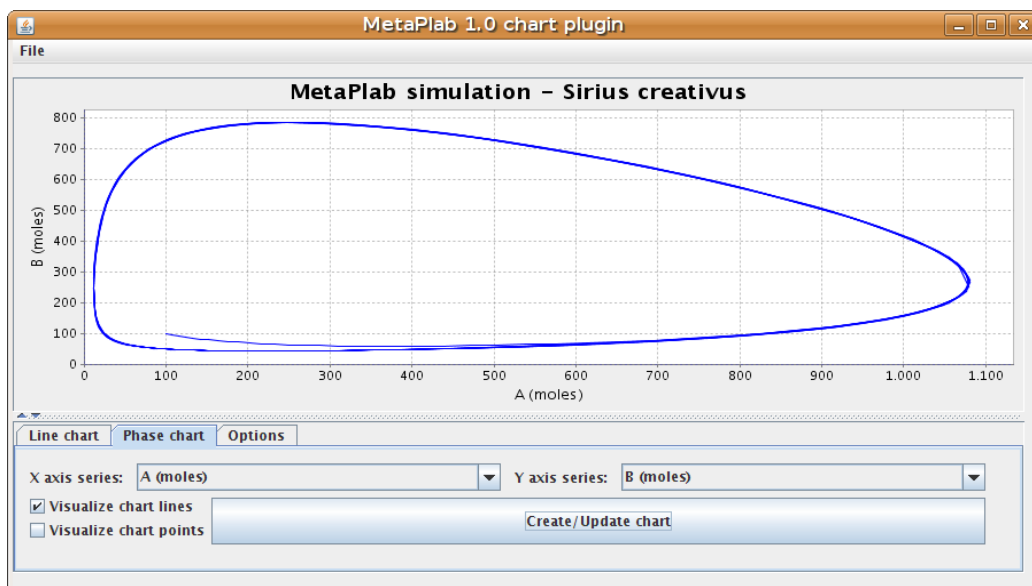


Figure 3: A phase chart of the substance  $A$  and  $B$  of the Sirius model.

printed and zoomed. In order to reduce memory load, the creation of each chart is performed by means of an algorithm which uses data sampling. In the third panel it is possible to set a parameter which modifies the accuracy of the chart plotting. Its default value is set to 1, but it can be changed to any rational number<sup>1</sup>. Figure 4 displays the same line chart plotted using different parameter values for data sampling.

## Result analysis

When we close the plugin and the plugin manager a window is prompted asking to update the current model by new values computed during the plugins execution. Confirming the operation, the update is performed in a few seconds. If we click on substance node  $A$ , in order to check its properties, we find the new time-series computed during the simulation, as displayed in Figure 5.

Now we try to improve the readability of the MP graph by adding some frames and labels. The MetaPlab input GUI offers also the possibility to hide a part of the MP graph without deleting its nodes, as depicted in Figure 6, a useful feature for managing complex graphs. This operation can be performed by a window which can be launched by clicking on the button

<sup>1</sup>Values greater than 1 reduce the accuracy of the plot, while smaller values increase it.

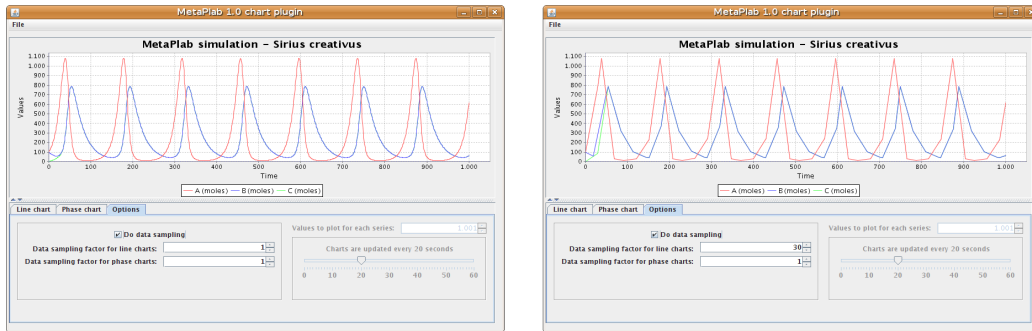


Figure 4: The same line chart depicted with different data sampling values (1 on the left and 30 on the right).

of the side-bar having a pair of binoculars as icon, or by selecting the right command of the **Edit** menu. The area of the MP graph to be visualized or hidden have to be selected before launching the visualization window or it can be defined specifying a particular type of node or edge.

## References

- [1] A. Castellini and V. Manca. MetaPlab: A computational framework for metabolic P systems. In *LNCS*, volume 5391, pages 157–168. Springer-Verlag, 2009.

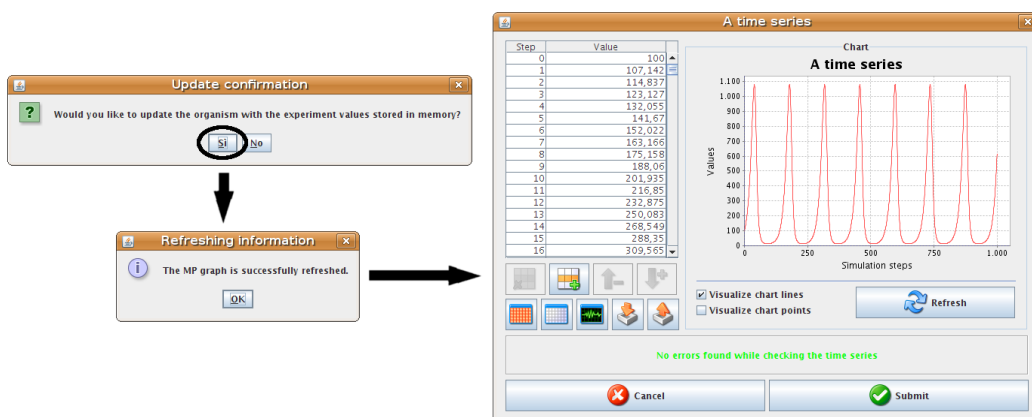


Figure 5: The update of the MP graph after the closure of the plugin manager: windows (on the left) and the new time-series for A (on the right).

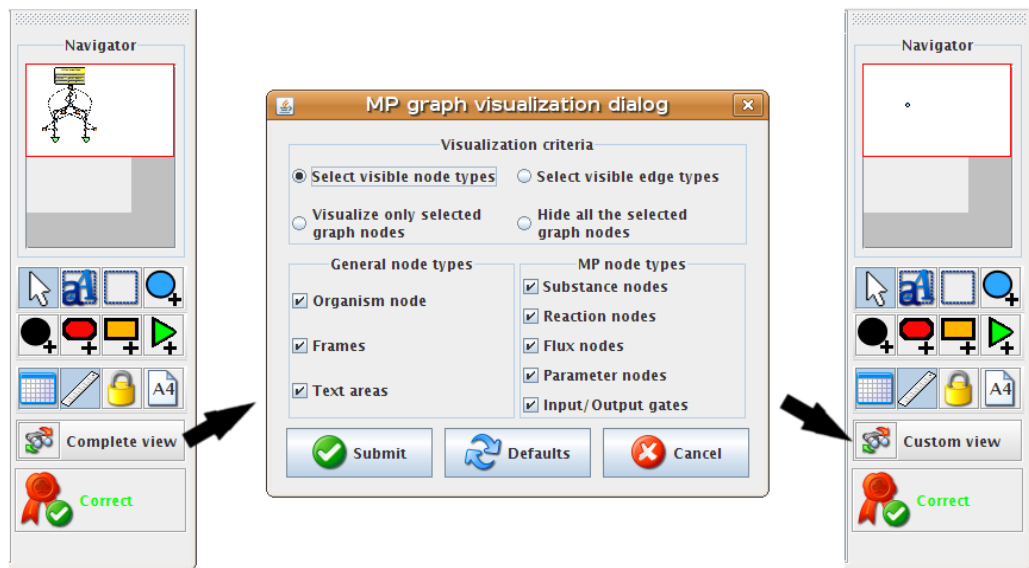


Figure 6: The window employed to change the visualization of the current MP graph.