TP Cosmos

Benoît Barbot Alban Linard Francis Hulin-Hubard

August 27, 2013

1 Calling the tool

1.1 Loading Models

In this first part, we will load predefined example models and call Cosmos on them. First launch Coloane, an empty window will appear (see figure 1). On the left part, left click, select *new* and *Modeling project* (see figure 2) and give a name to the new project.

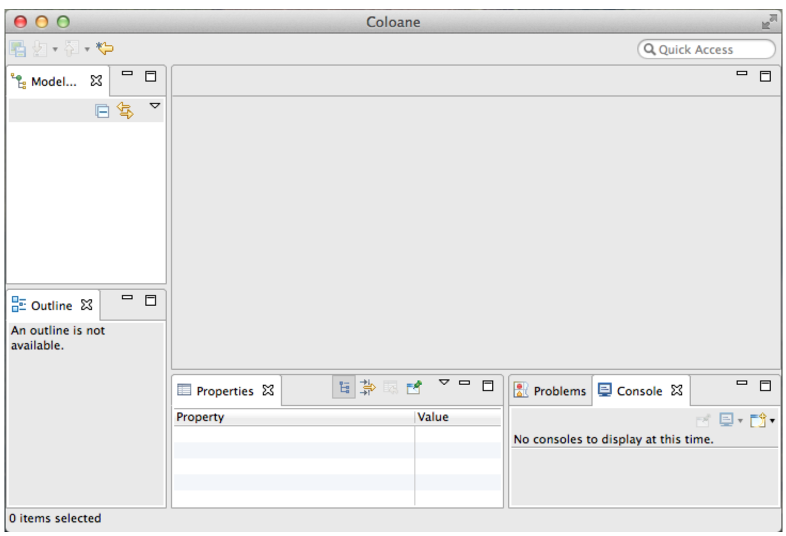

Figure 1: Coloane starting window

To add new models, left click on the new project and select *Model* (see 3). A list of available formalisms will be shown, select *COSMOS*. In the next dialog window, select *Tandem* as a default model (see 4). This model is a tandem queues with two queues with a Poisson arrival rate and an exponentially distributed service time.

Repeat the same operation and choose *COSMOSLHA* as formalism and *Tandem(lha)* for the default model. The synchronized product of this automaton

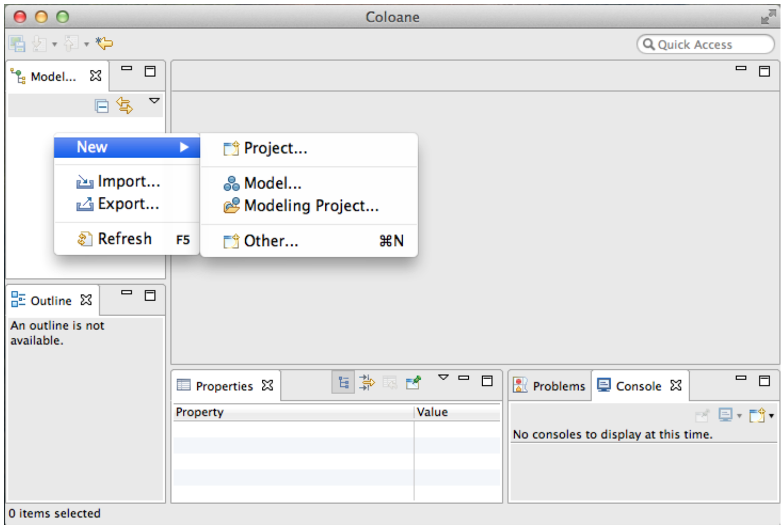

Figure 2: New project menu

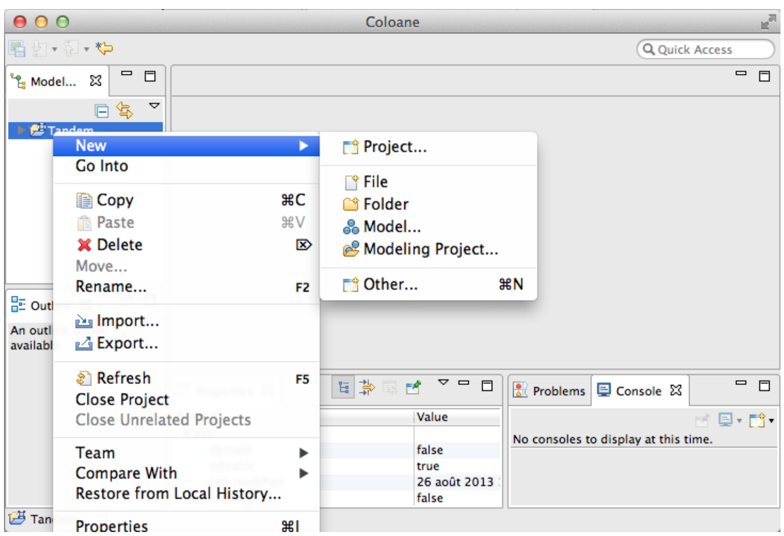

Figure 3: New model menu

with the Petri net model computes the probability that 5 clients are in the two queues before there is no client in the two queues.

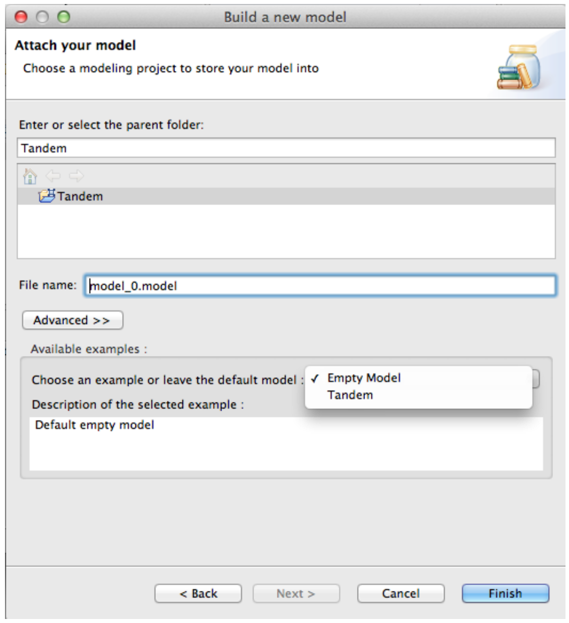

Figure 4: New model dialog box

1.2 Calling the tool

The first simple analysis we can perform on the tandem queues system is to simulate one trajectory. To do this, select one alligator server in the service menu and then select *Cosmos*, *Generate one trajectory of the model* (see figure 5).

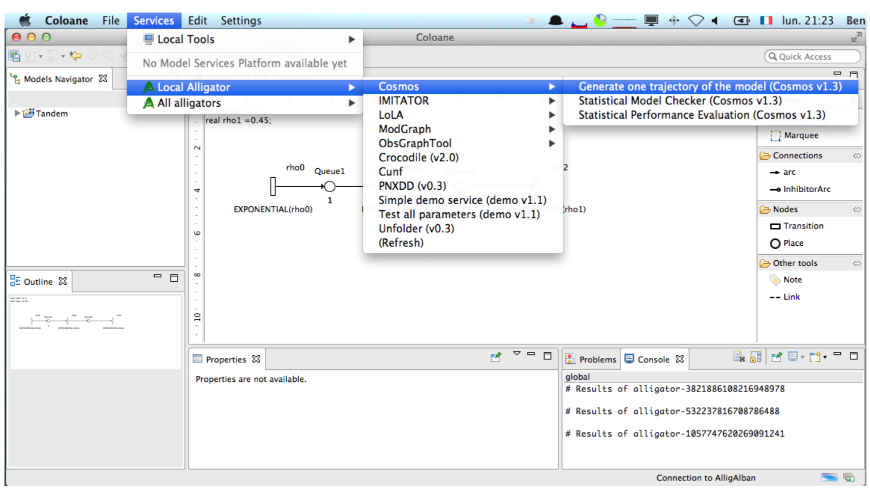

Figure 5: Lauching Cosmos

A window will pop asking for some parameters. The first page of parameters contains only the length of the trajectory, the second page requires to set the model, check the box of the tandem queue model (see figure 6). Once Cosmos

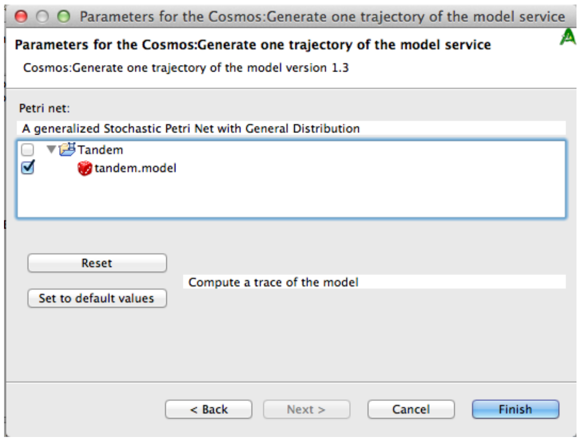

Figure 6: Model selection box

finishes the simulation a graphics showing the number of tokens in each place is displayed.

A more detailed analysis can be performed by choosing the service *Cosmos Statistical Perfomance Evaluation* in the service menu. This service requires more parameters. You can first keep the default values and launch the service on the tandem queues system. After some time, the result will be displayed. For each place, the tool returns an estimation of the mean number of tokens looking like:

```
MeanToken_Queue1:
Estimated value:0.286221703319812
Confidence interval:[0.282671106477023 , 0.289772300162601]
Width:0.00710119368557882
```

For each transition, an estimation of the throughput is returned:

```
Throughput_rho0:
Estimated value:0.0999094999999996
Confidence interval:[0.0985010508695943 , 0.101317949130405]
Width:0.00281689826081077
```

You can modify the model by changing the rate *rho0* to a value above 0.45 and observe that the system is unstable (meaning the number of tokens in a place diverges) by simulating a trajectory or computing mean number of tokens. You can also try to change the distribution of the first transition from *EXPONENTIAL(rho0)* to *DETERMINISTIC(10)* and observe the change on the trace of the simulation. You can try other distributions, the available distributions are (GAMMA, UNIFORM, EXPONENTIAL, DETERMINISTIC, LOGNORMAL, TRIANGLE, GEOMETRIC and ERLANG).

To compute more complex indexes, we will use an automaton to describe them. The LHA model represents an automaton which waits for the system to reach either a state where the total number of clients in the system is equal to 0 or a state where it equals N. In the latter case, the variable x is set to 1. Finally, the expression *AVG(last(x))* makes the tool returns an estimation of the expected value of x in final states.

Launch Cosmos statistical model checker service from the service menu. This service requires an LHA, a model and several parameters which can be kept at there default values. The returned result is a confidence interval of an estimation of the expected value of x. A graph depicting the convergence of the estimation is also returned.

You can try to modify parameters of the LHA or of the Model to see the impact on the variable x. You can also try to change the calling parameters of the service to obtain tighter confidence intervals.

2 Building A Model

In this section, you will model a network of queues. Create a new empty model in the Cosmos formalism and build the model depicted in figure 7.

You need to name all places and transitions for the tool to accept the model.

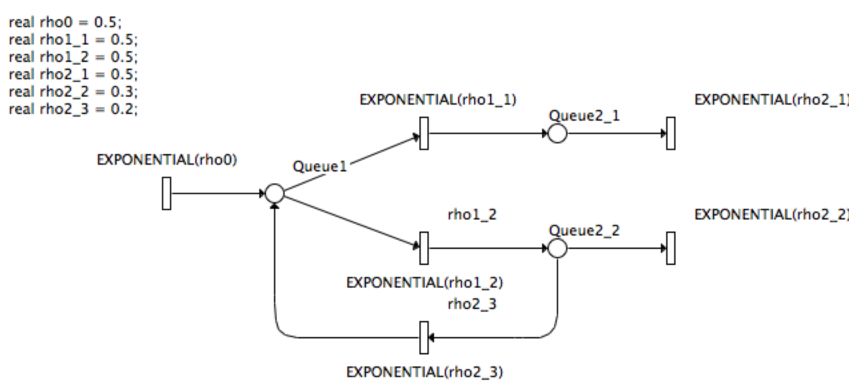

Figure 7: an example of network of waiting queues

Then you can now adapt the LHA of the previous example to work with this one, computing the probability for the system to reach a state containing N clients before reaching a state with no client.

You can also build an LHA computing the average time between the first arrival of a client and the first time the system is empty. *The notation for the flow of a variable is* $x' = c$, *where* x *is the variable and* c *an expression depending only of the number of tokens is some place.* Assuming the variable accounting for the time is t, by replacing the HASL formula by *AVG(last(t)); CDF(last(t),0.1,0.0,50.0);*, the Cumulative Density Function of the random variable t is returned with the result (see Figure 8).

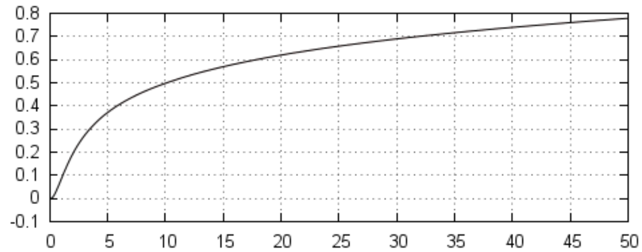

Figure 8: an example of a computation of CDF