MASSPA-Modeller: A Spatial Stochastic Process Algebra modelling tool

ICCSW 2011

Marcel C. Guenther, Jeremy T. Bradley

Department of Computing, Imperial College London

September 26, 2011

Introduction

Spatial population modelling:

- Systems Biology, Ecology, Performance Analysis, ...

Introduction

Spatial population modelling:

- Systems Biology, Ecology, Performance Analysis, ...

Problem:

- CTMCs with enormous state spaces

Introduction

Spatial population modelling:

- Systems Biology, Ecology, Performance Analysis, ...

Problem:

- CTMCs with enormous state spaces

Solution:

- High-level modelling languages: process algebras, stochastic Petri nets, ...

Introduction

Spatial population modelling:

- Systems Biology, Ecology, Performance Analysis, ...

Problem:

- CTMCs with enormous state spaces

Solution:

- High-level modelling languages: process algebras, stochastic Petri nets, ...
- Moments approximating ODEs: $\mathbb{E}[Prey]$, $Var[Predator]$ [1, 2]

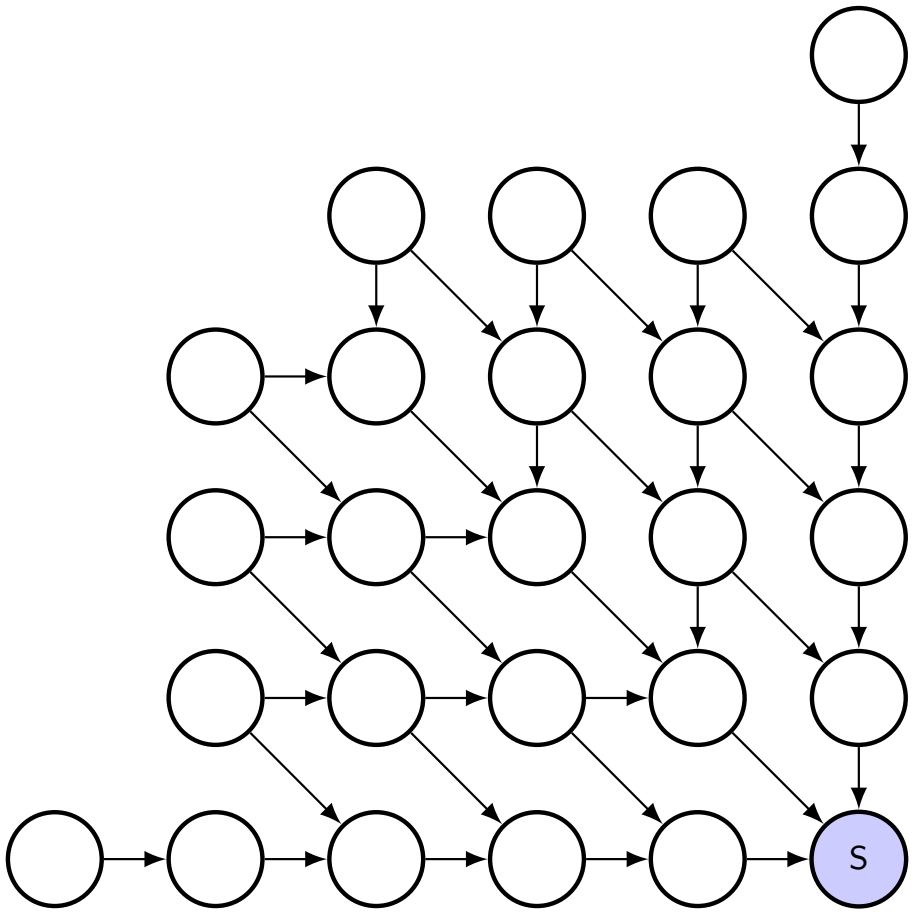
Introduction

- What if high-level descriptions become tedious?

```
Agent OnOff {
  On  = !(1.0,M,1.0).Off;
  Off = ?(M,1.0).On;
};

Locations = {A,B,C,D,E,F,...};

On@A  = 450; Off@B = 450;
Off@C = 300; Off@D = 300;
...

Channel(On@A,Off@B,M) = 1/450;
Channel(On@B,Off@C,M) = 1/300;
...
```

Introduction

Visual modelling:



Markovian agent models (MAM)s in MASSPA

```
Agent OnOff {



};
```

Markovian agent models (MAM)s in MASSPA

```
Agent OnOff {
   On  = !(actionName?,1.0,M,1.0).Off;


};
```

Markovian agent models (MAM)s in MASSPA

```
Agent OnOff {
   On  = !(actionName?,1.0,M,1.0).Off;
   Off = ?(actionName?,M,1.0).On;

};
```

Markovian agent models (MAM)s in MASSPA

```
Agent OnOff {
   On  = !(actionName?,1.0,M,1.0).Off;
   Off = ?(actionName?,M,1.0).On;
   Off =  (actionName?,1.0).On;
};
```

Markovian agent models (MAM)s in MASSPA

```
Agent OnOff {
   On  = !(actionName?,1.0,M,1.0).Off;
   Off = ?(actionName?,M,1.0).On;
   Off =  (actionName?,1.0).On;
};

Locations = A,B,C,D,E,F,...;
...
```

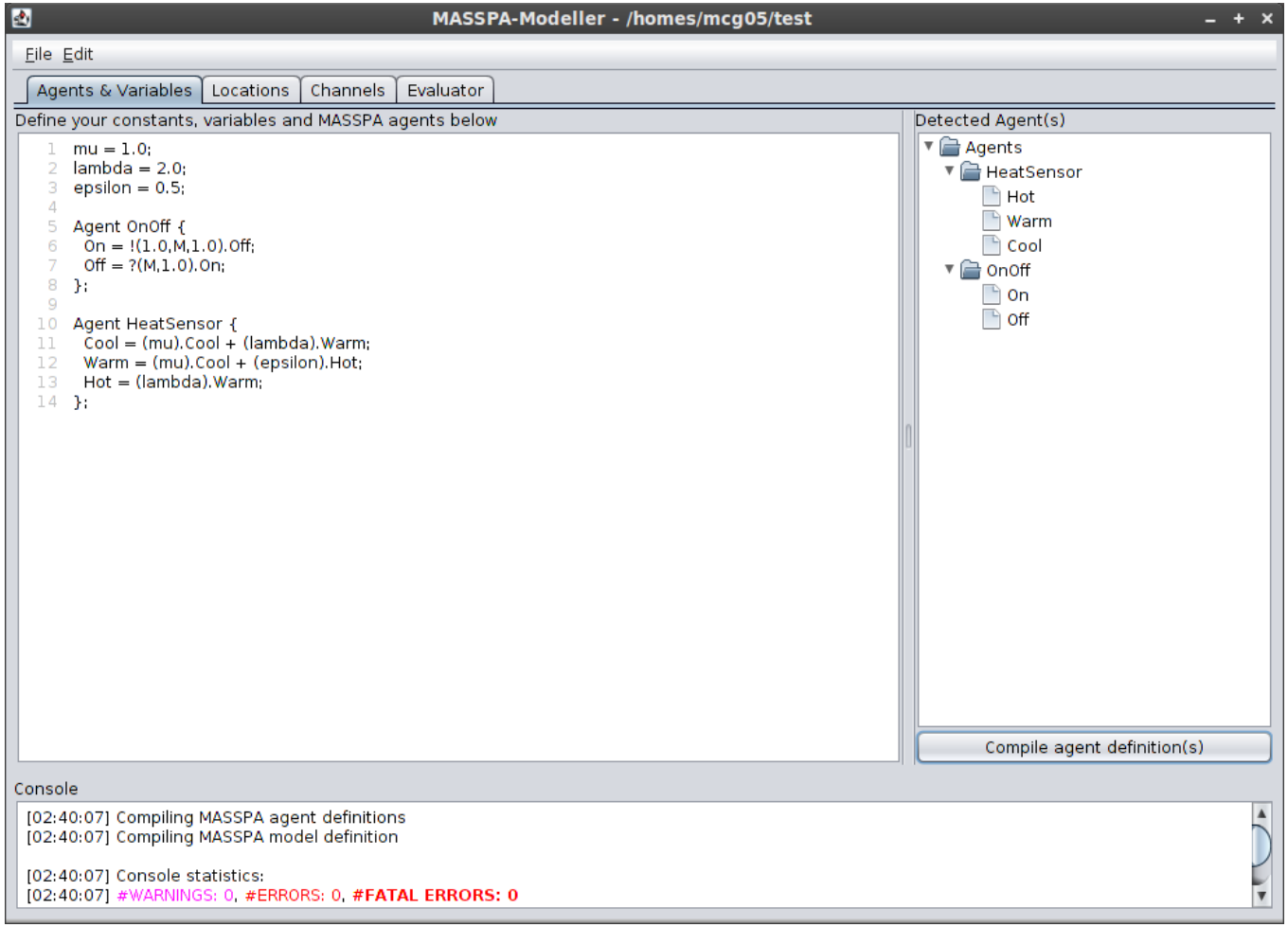
Markovian agent models (MAM)s in MASSPA

```
Agent OnOff {
   On  = !(actionName?,1.0,M,1.0).Off;
   Off = ?(actionName?,M,1.0).On;
   Off =  (actionName?,1.0).On;
};

Locations = A,B,C,D,E,F,...;
...

Channel(On@A,Off@B,M) = 1/450;
...
```

MASSPA-Modeller work-flow

- Step 1: Define sequential agents

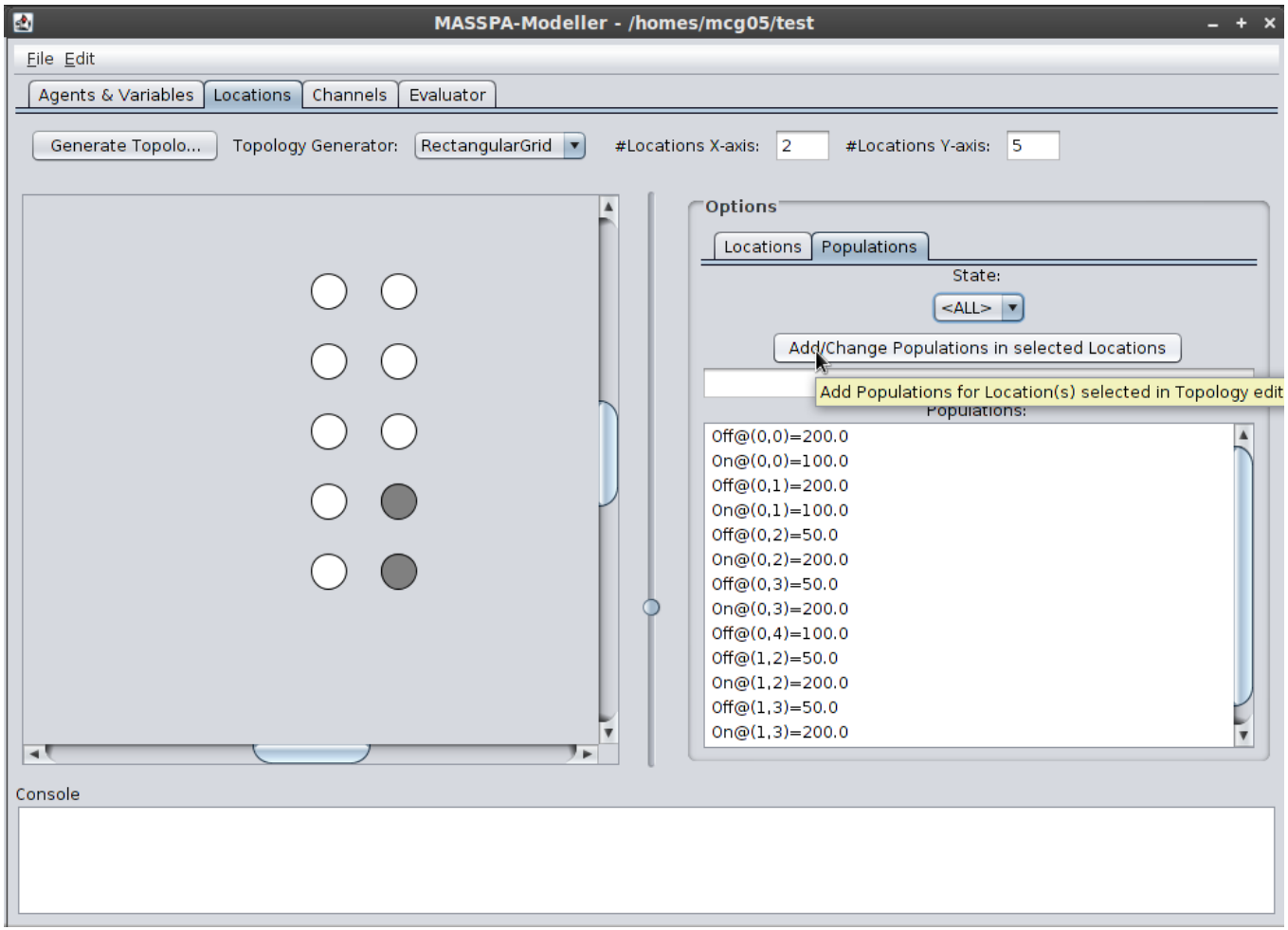
MASSPA-Modeller work-flow



MASSPA-Modeller work-flow

- Step 1: Define sequential agents
- Step 2: Create topology

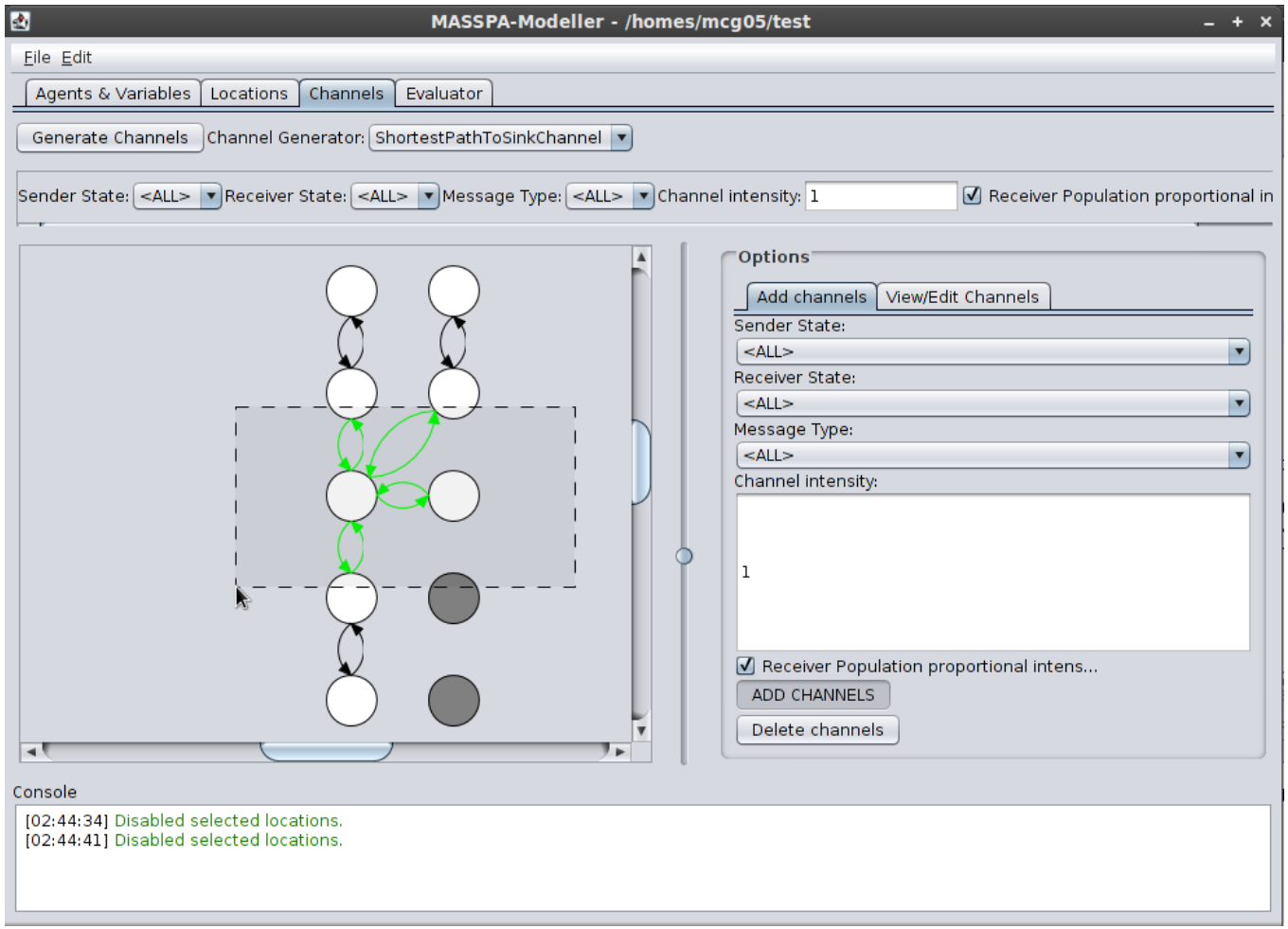
MASSPA-Modeller work-flow



MASSPA-Modeller work-flow

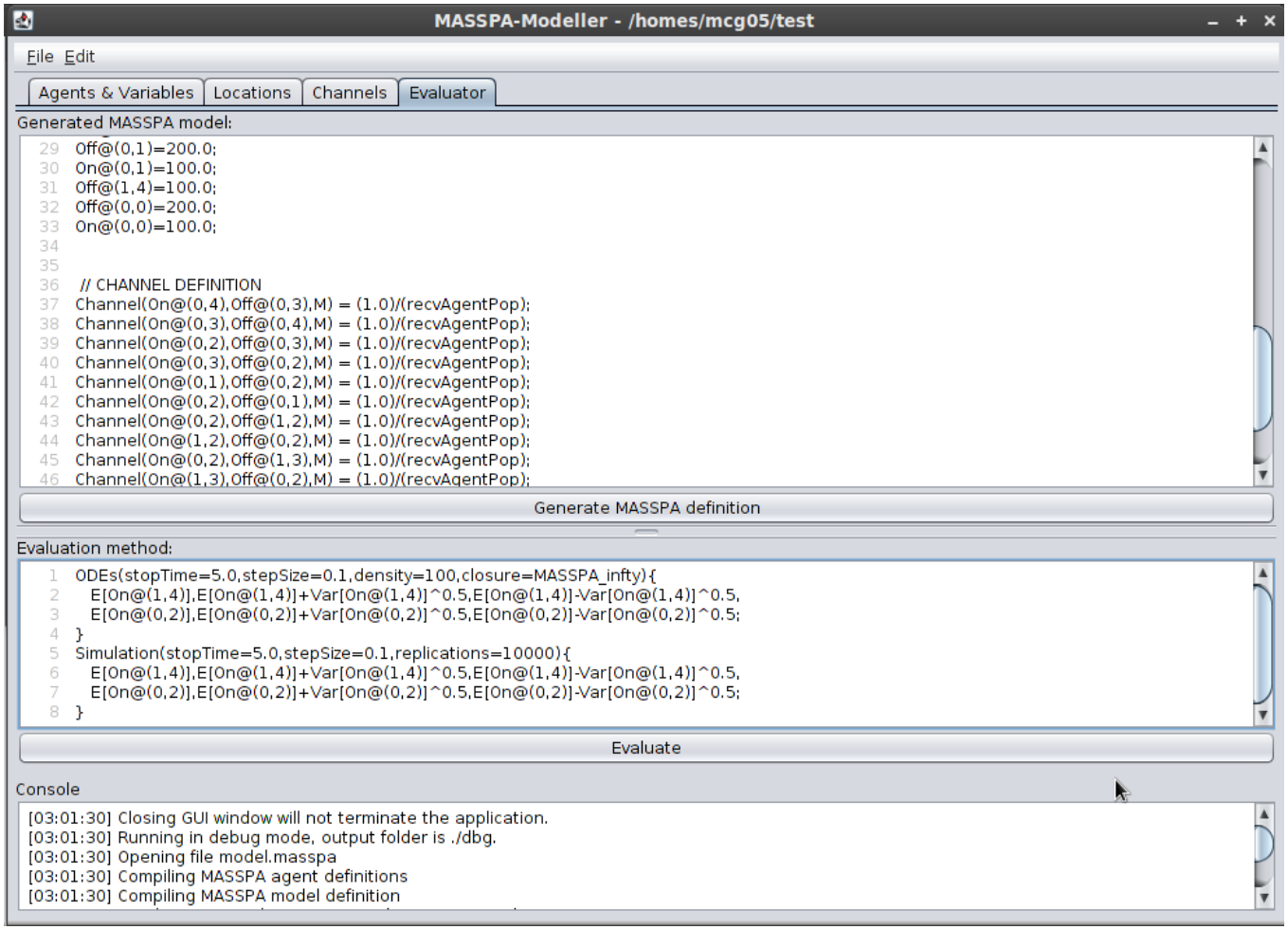
- Step 1: Define sequential agents
- Step 2: Create topology
- Step 3: Create communication patterns

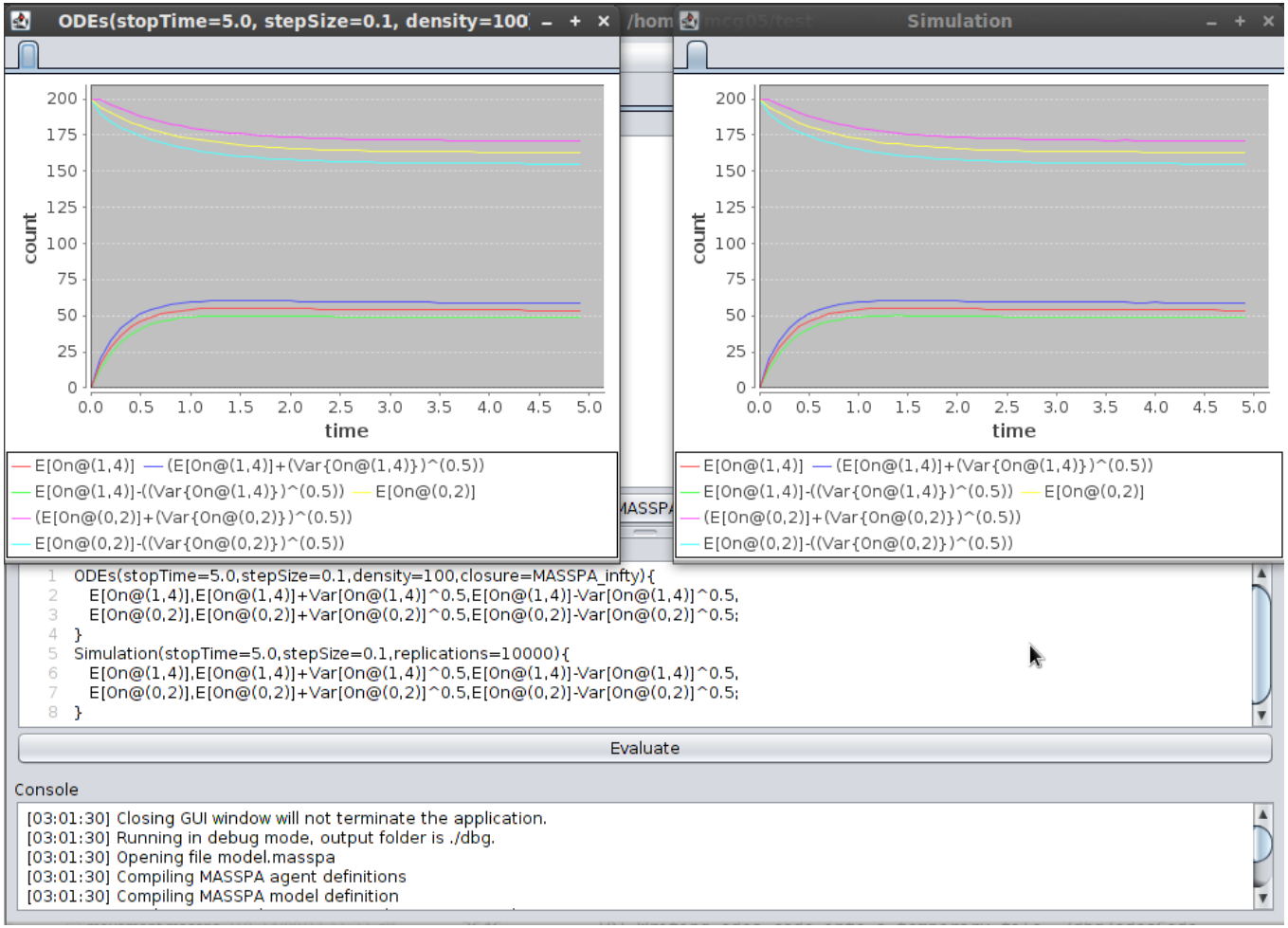
MASSPA-Modeller work-flow



MASSPA-Modeller work-flow

- Step 1: Define sequential agents
- Step 2: Create topology
- Step 3: Create communication patterns
- Step 4: Generate MASSPA and evaluate using GPA [3]


MASSPA-Modeller - /homes/mcg05/test
File Edit
Agents & Variables
Locations
Channels
Evaluator
Generated MASSPA model:
29 Off@(0,1)=200.0;
30 On@(0,1)=100.0;
31 Off@(1,4)=100.0;
32 Off@(0,0)=200.0;
33 On@(0,0)=100.0;
34
35
36 // CHANNEL DEFINITION
37 Channel(On@(0,4),Off@(0,3),M) = (1.0)/(recvAgentPop);
38 Channel(On@(0,3),Off@(0,4),M) = (1.0)/(recvAgentPop);
39 Channel(On@(0,2),Off@(0,3),M) = (1.0)/(recvAgentPop);
40 Channel(On@(0,3),Off@(0,2),M) = (1.0)/(recvAgentPop);
41 Channel(On@(0,1),Off@(0,2),M) = (1.0)/(recvAgentPop);
42 Channel(On@(0,2),Off@(0,1),M) = (1.0)/(recvAgentPop);
43 Channel(On@(0,2),Off@(1,2),M) = (1.0)/(recvAgentPop);
44 Channel(On@(1,2),Off@(0,2),M) = (1.0)/(recvAgentPop);
45 Channel(On@(0,2),Off@(1,3),M) = (1.0)/(recvAgentPop);
46 Channel(On@(1,3),Off@(0,2),M) = (1.0)/(recvAgentPop);
Generate MASSPA definition
Evaluation method:
1 ODEs(stopTime=5.0,stepSize=0.1,density=100,closure=MASSPA_infty){
2 E[On@(1,4)],E[On@(1,4)]+Var[On@(1,4)]^0.5,E[On@(1,4)]-Var[On@(1,4)]^0.5,
3 E[On@(0,2)],E[On@(0,2)]+Var[On@(0,2)]^0.5,E[On@(0,2)]-Var[On@(0,2)]^0.5;
4 }
5 Simulation(stopTime=5.0,stepSize=0.1,replications=10000){
6 E[On@(1,4)],E[On@(1,4)]+Var[On@(1,4)]^0.5,E[On@(1,4)]-Var[On@(1,4)]^0.5,
7 E[On@(0,2)],E[On@(0,2)]+Var[On@(0,2)]^0.5,E[On@(0,2)]-Var[On@(0,2)]^0.5;
8 }
Evaluate
Console
[03:01:30] Closing GUI window will not terminate the application.
[03:01:30] Running in debug mode, output folder is ./dbg.
[03:01:30] Opening file model.masspa
[03:01:30] Compiling MASSPA agent definitions
[03:01:30] Compiling MASSPA model definition


ODEs(stopTime=5.0, stepSize=0.1, density=100
Simulation
count
time
E[0n@(1,4)]
(E[0n@(1,4)]+(Var{0n@(1,4)})^(0.5))
E[0n@(1,4)]-((Var{0n@(1,4)})^(0.5))
E[0n@(0,2)]
(E[0n@(0,2)]+(Var{0n@(0,2)})^(0.5))
E[0n@(0,2)]-((Var{0n@(0,2)})^(0.5))
ODEs(stopTime=5.0,stepSize=0.1,density=100,closure=MASSPA_infty){
E[0n@(1,4)],E[0n@(1,4)]+Var[0n@(1,4)]^0.5,E[0n@(1,4)]-Var[0n@(1,4)]^0.5,
E[0n@(0,2)],E[0n@(0,2)]+Var[0n@(0,2)]^0.5,E[0n@(0,2)]-Var[0n@(0,2)]^0.5;
}
Simulation(stopTime=5.0,stepSize=0.1,replications=10000){
E[0n@(1,4)],E[0n@(1,4)]+Var[0n@(1,4)]^0.5,E[0n@(1,4)]-Var[0n@(1,4)]^0.5,
E[0n@(0,2)],E[0n@(0,2)]+Var[0n@(0,2)]^0.5,E[0n@(0,2)]-Var[0n@(0,2)]^0.5;
}
Evaluate
Console
[03:01:30] Closing GUI window will not terminate the application.
[03:01:30] Running in debug mode, output folder is ./dbg.
[03:01:30] Opening file model.masspa
[03:01:30] Compiling MASSPA agent definitions
[03:01:30] Compiling MASSPA model definition

Thank you!

[1] J. Hillston, "Fluid flow approximation of PEPA models," *Second International Conference on the Quantitative Evaluation of Systems QEST05*, pp. 33–42, 2005.

[2] R. A. Hayden and J. T. Bradley, "A fluid analysis framework for a Markovian process algebra," *Theoretical Computer Science*, vol. 411, no. 22-24, pp. 2260–2297, 2010.

[3] A. Stefanek, R. Hayden, and J. Bradley, "A new tool for the performance analysis of massively parallel computer systems," *Eighth Workshop on Quantitative Aspects of Programming Languages QAPL 2010 March 2728 2010 Paphos Cyprus*, 2010.