

MaBoSS Tutorial

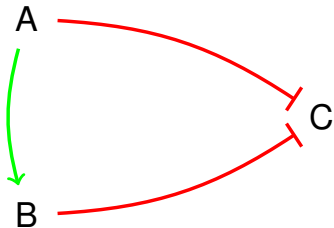
Vincent Noël¹

¹Computational Systems Biology of Cancer,
Institut Curie, INSERM U900, Université PSL, Paris

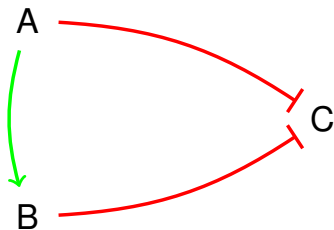
19th European Conference on Computational Biology

September 1, 2020

Introduction

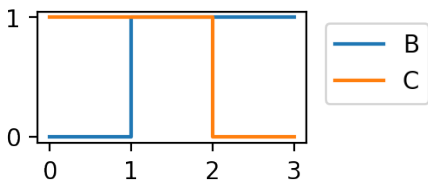


Introduction

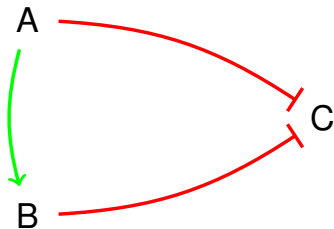


Boolean

- › A: input
- › B: A
- › C: !A & !B

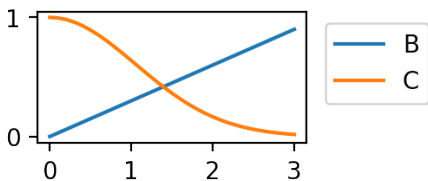


Introduction



ODEs


- › $\frac{dA}{dt} = 0$
- › $\frac{dB}{dt} = k_1 \cdot A$
- › $\frac{dC}{dt} = -k_2 \cdot A \cdot B \cdot C$



Methodology article | [Open Access](#) | Published: 29 August 2012

Continuous time boolean modeling for biological signaling: application of Gillespie algorithm

[Gautier Stoll](#) , [Eric Viara](#), [Emmanuel Barillot](#) & [Laurence Calzone](#)

[BMC Systems Biology](#) 6, Article number: 116 (2012) | [Download Citation](#) 
6306 Accesses | 32 Citations | 1 Altmetric | [Metrics](#) 



Gautier Stoll



Eric Viara



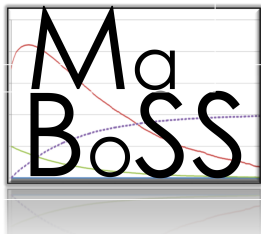
Laurence Calzone



Emmanuel Barillot

Introduction

Markovian Boolean Stochastic Simulator



<https://maboss.curie.fr/>

- › Boolean
- › Model asymptotic and transient behavior
- › Physical time
- › Handle different time scale processes (transcription, phosphorylation, etc.)

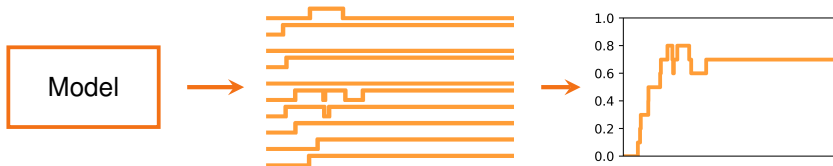
⇒ Fills the gap between ODE and Boolean modeling

Introduction

Continuous time Markov process applied on a Boolean network state space

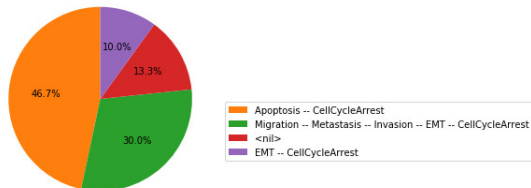
Transition rate :

$$p(S \rightarrow S') = \begin{cases} R_{up}(S), & \text{if } S_i = 0 \\ R_{down}(S), & \text{if } S_i = 1 \end{cases}$$

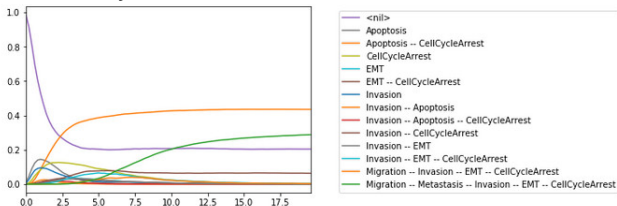


Introduction

› Final states distribution



› State distribution trajectories



Introduction

BND file : Network definition

```
1 node A {
2     logic = 1;
3     rate_up = @logic ? 1 : 0;
4     rate_down = 0;
5 }
6
7 node B {
8     logic = A & !C;
9     rate_up = @logic ? $act_B : 0;
10    rate_down = 0;
11 }
12
13 node C {
14     logic = A & !B;
15     rate_up = @logic ? $act_C : 0;
16     rate_down = 0;
17 }
18
```

CFG file: Simulation settings

```
1 A.istate = FALSE;
2 B.istate = FALSE;
3 C.istate = FALSE;
4
5 time_tick = 0.1;
6 max_time = 20.0;
7 sample_count = 10000.0;
8 use_physrandgen = 1.0;
9 thread_count = 1.0;
10
11 $act_B = 1;
12 $act_C = 1;
```

pyMaBoSS

- › Initially developped by Nicolas Levy
- › Maintained by Aurelien Naldi, Loic Pauleve, me
- › Available on Pypi:
\$ pip install maboss
- › Available on Conda:
\$ conda install -c colomoto pymaboss



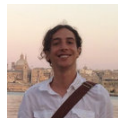
Nicolas Levy



Aurelien Naldi



Loïc Paulevé



Vincent Noël

What if we want to use MaBoSS to create population models ?



New Results

[Comment on this paper](#)

UPMaBoSS: a novel framework for dynamic cell population modeling

Gautier Stoll, Aurélien Naldi, Vincent Noël, Eric Viara, Emmanuel Barillot, Guido Kroemer,
 Denis Thieffry, Laurence Calzone

doi: <https://doi.org/10.1101/2020.05.31.126094>



Gautier Stoll



Eric Viara



Laurence Calzone



Emmanuel Barillot



Aurelien Naldi



Vincent Noël



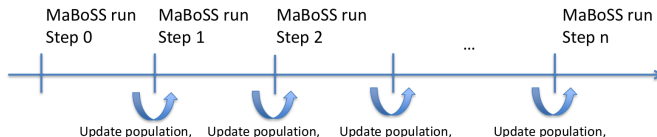
Denis Thieffry



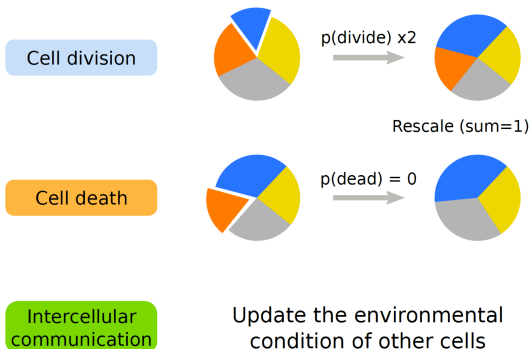
Guido Kroemer

UPMaBoSS

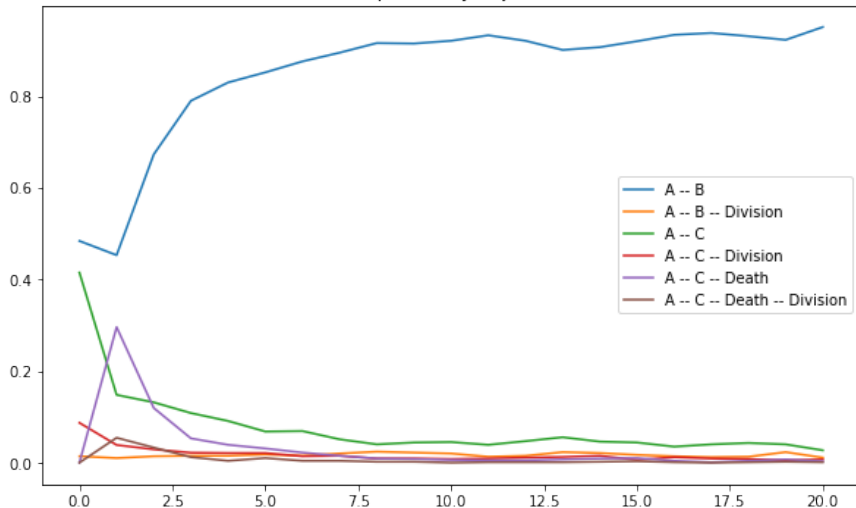
- › We start from an existing MaBoSS model
- › We add model nodes for division and death
- › We add inter-cellular communication
- › Chaining MaBoSS simulations, updating population and communication between each



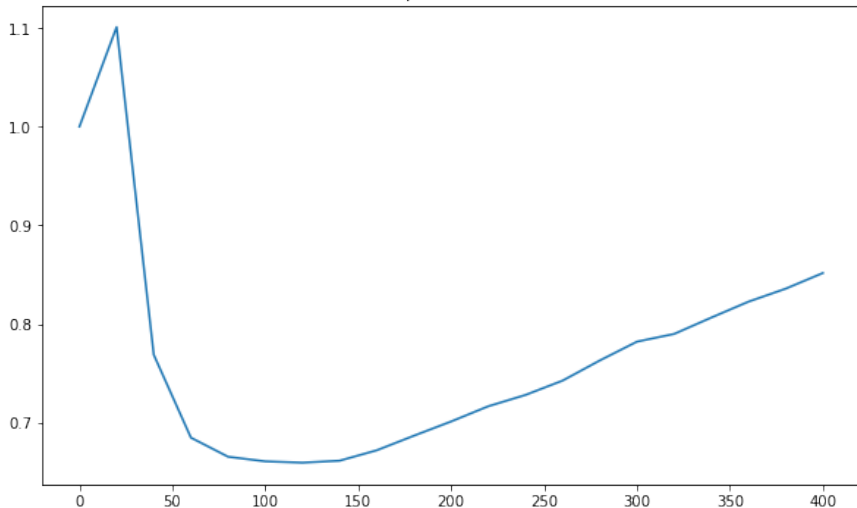
Updating population:



State probability trajectories



Population size



Aknowledgments



MaBoSS early team



Gautier Stoll



Eric Viara



Laurence
Calzone



Emmanuel
Barillot

MaBoSS ecosystem team



Aurelien
Naldi



Denis
Thieffry



Loïc
Paulevé



Nicolas
Levy



Jonathan
Pol



Arnau
Montagud



Vincent
Noël



Gaëlle
Letort



Mihaly
Koltai



Andrei
Zinovyev



Randy
Heiland



Paul
Macklin



Jonas
Beal



Stéphanie
Chevalier



Andrea
Checcoli



Lorenzo
Pantolini



Marco
Ruscone



Gerard
Pradas