

PINT: Formal and Scalable Analysis for Transient Dynamics of Qualitative Networks

Loïc Paulevé

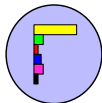
CNRS/LRI, Univ. Paris-Sud, Univ. Paris-Saclay – BioInfo team

`loic.pauleve@lri.fr`

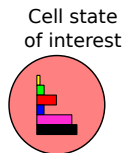
`http://loicpauleve.name`

Colomoto meeting - July 2017

Cellular Dynamics

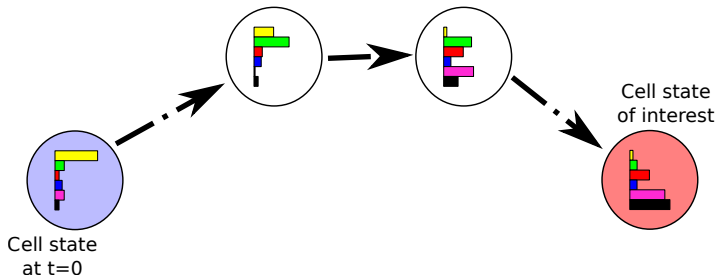


Cell state
at $t=0$



Initial state(s)/Goal state(s)

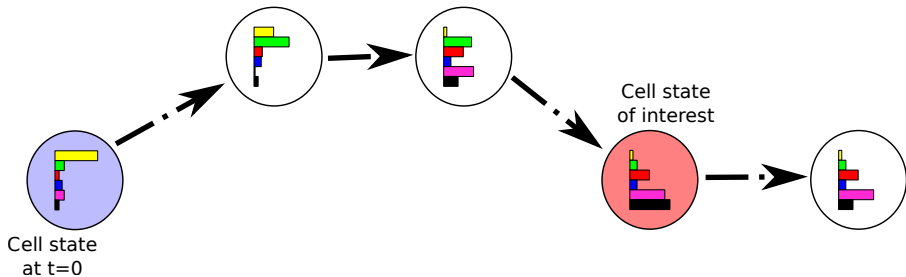
Cellular Dynamics



Initial state(s)/Goal state(s)

- Trajectory existence (reachability)

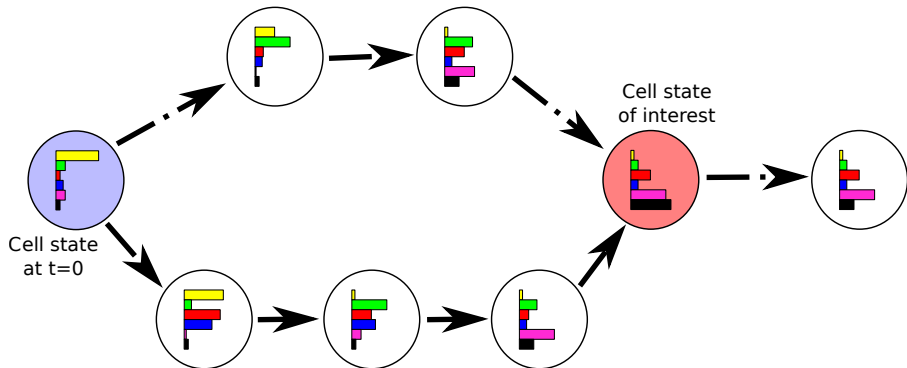
Cellular Dynamics



Initial state(s)/Goal state(s)

- Trajectory existence (reachability)

Cellular Dynamics

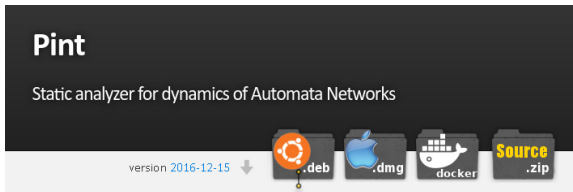


Initial state(s)/Goal state(s)

- Trajectory existence (reachability)
- Reasoning on all trajectories: e.g., common features
- **Control:** perturbations to avoid/enforce goal reachability

Software: Pint

<http://loicpauleve.name/pint>



Model input

- Main formalism: automata networks (\approx Petri nets)
- Asynchronous Boolean/multi-valued networks
- Formats: any supported by bioLQM; biocham

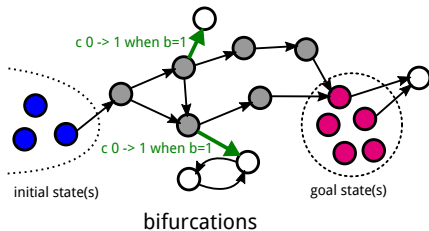
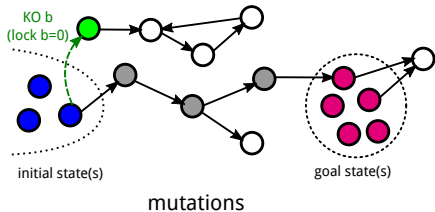
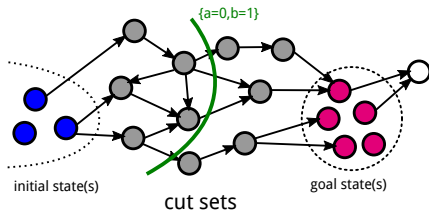
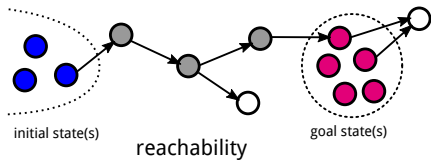
Interfaces

- Command line tools
- Python module; integration with Jupyter notebook web interface

Documentation: <http://loicpauleve.name/pint/doc>

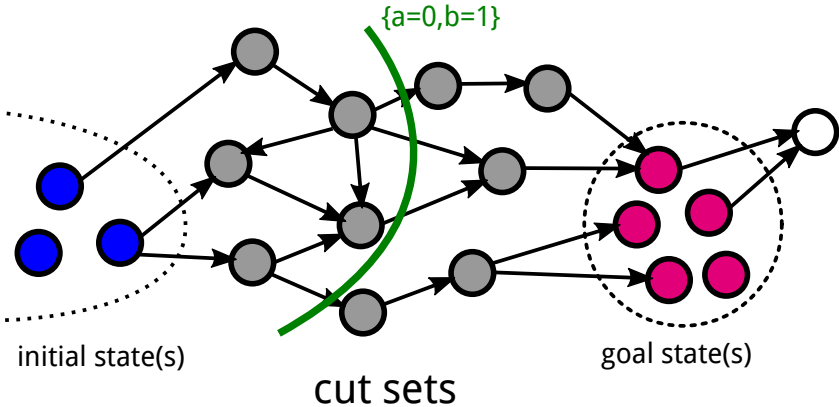
⇒ will be presented at CMSB 2017 in Darmstadt

Main features

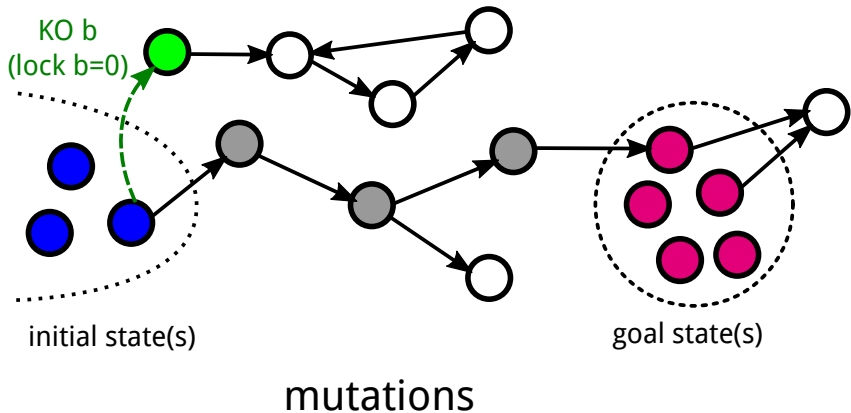


Scalability: networks with 100 - 10,000 components

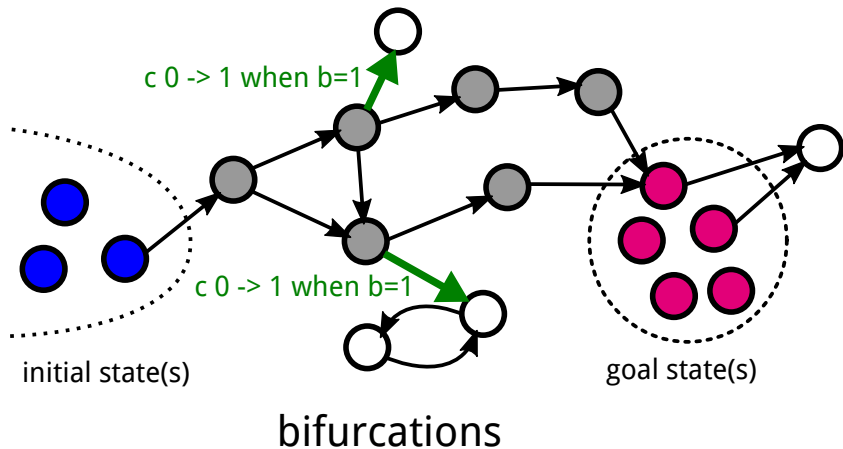
Main features



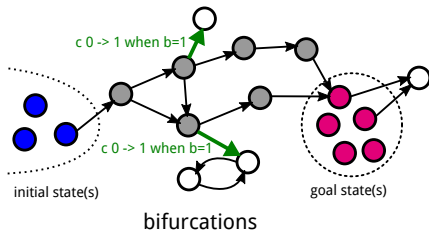
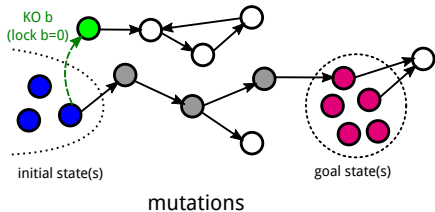
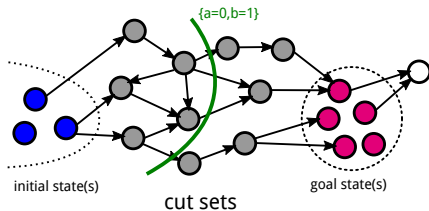
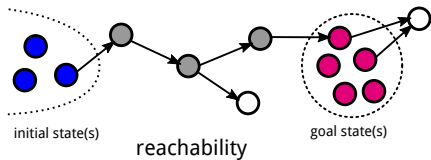
Main features



Main features



Main features



Scalability: networks with 100 - 10,000 components

Behind the scenes

Computational challenge

- Mentioned features are **PSPACE-complete**
- **Avoid screening** of mutations/cut-sets/bifurcations candidates

PINT technology

- Static analysis by **abstract interpretation**
- Avoid building the state transition graph (even symbolically)
⇒ compute abstractions (**Local Causality Graphs**)
- **Formal approximations** (under-/over-approximations);
reduce to NP (SAT) solving.

[Math. Struct. for Comput. Sci. 2012; CAV'13; TCS 2015; CMSB 2016]

IPython/Jupyter integration:

- makes easier the [reproductibility and documentation of workflows](#)
- becomes standard in data science/bioinformatics
- easy access to [statistics/visualisation tools](#) (scipy, pandas, matplotlib, R, ...)

Python interface for your tools:

- allow easy interaction with other tools;
- no much effort required (simple implementation: call executable from python; use JSON)

Who's in for building the **Computational Systems Biology notebook?**