An Improved Simulation of Hybrid Biological Models with many stochastic events and quasi-disjoint subnets

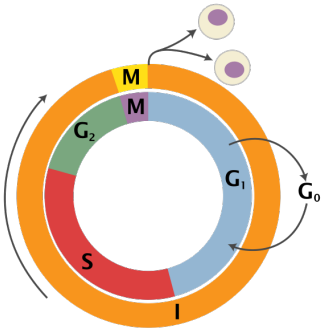
Mostafa Herajy [1] and Monika Heiner [2]

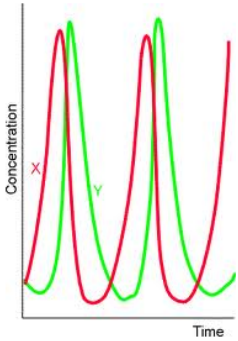
[1] Department of Mathematics and Computer Science,
Faculty of Science, Port Said University, Egypt

[2] Computer Science Institute, Faculty 1
Brandenburg University of Technology, Cottbus, Germany

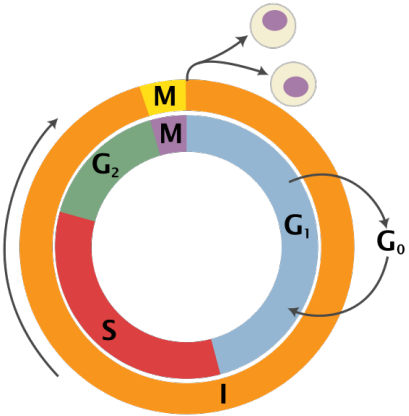
Gothenburg, 10 December 2018

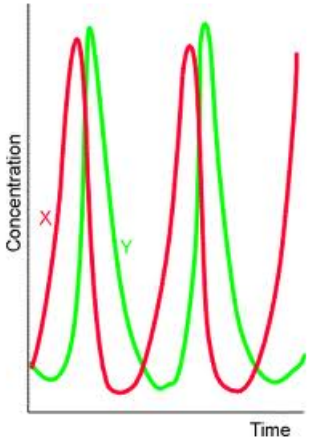
Outline





Prologue





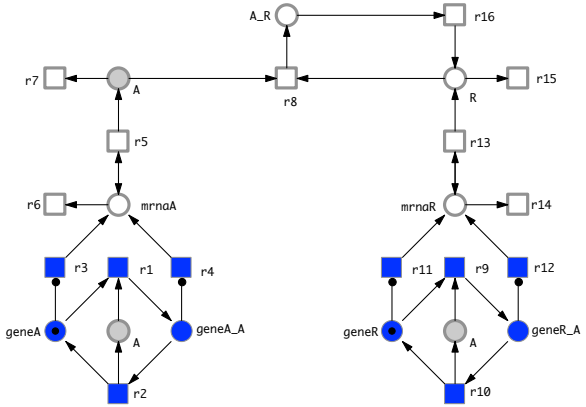
Prologue

We use Generalised Hybrid Petri Nets ($\mathcal{GHPN}$) to

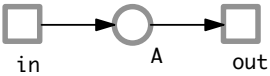
- construct and
- graphically visualise

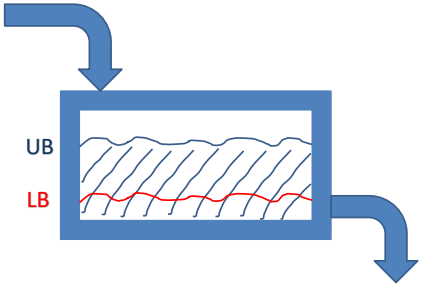
our models.

But, our approach can be applied to any representation of (biochemical reaction) networks.

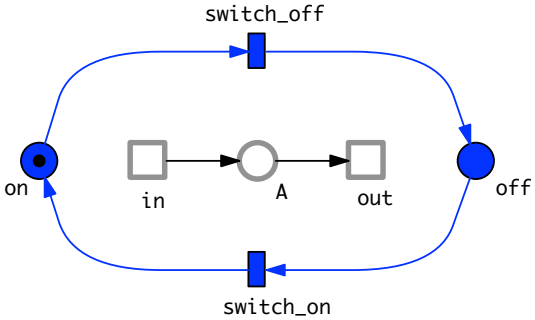


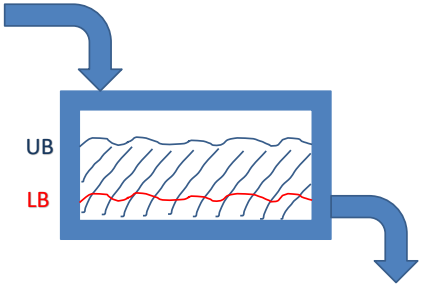
$\mathcal{GHPN}$ example



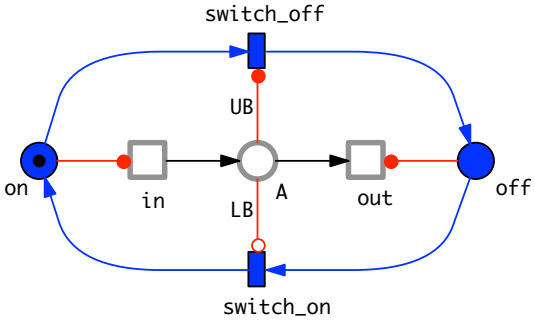


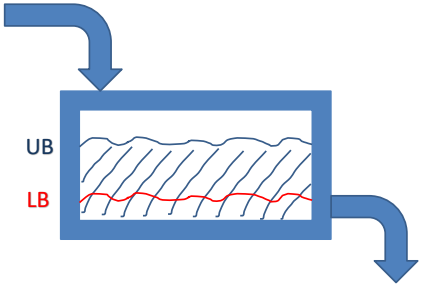
$\mathcal{GHPN}$ example with immediate transitions



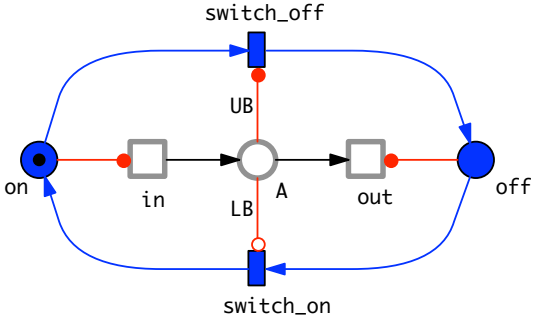


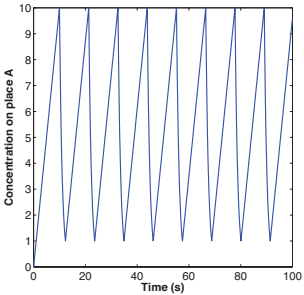
$\mathcal{GHPN}$ example with immediate transitions



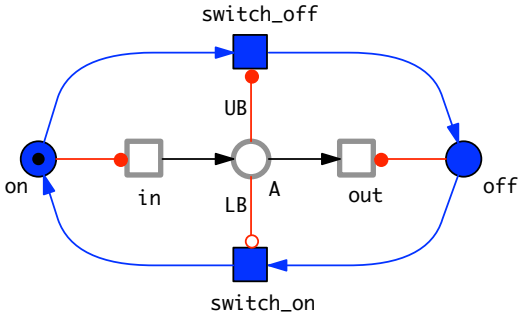


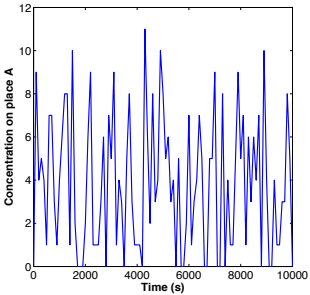
$\mathcal{GHPN}$ example with immediate transitions



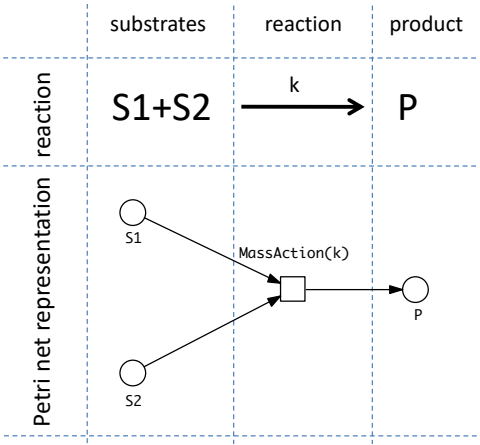


$\mathcal{GHPN}$ example with stochastic transitions





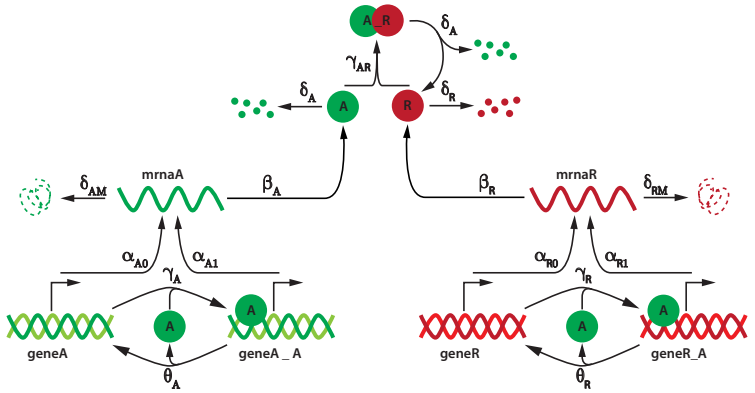
From reaction to Petri net



notations

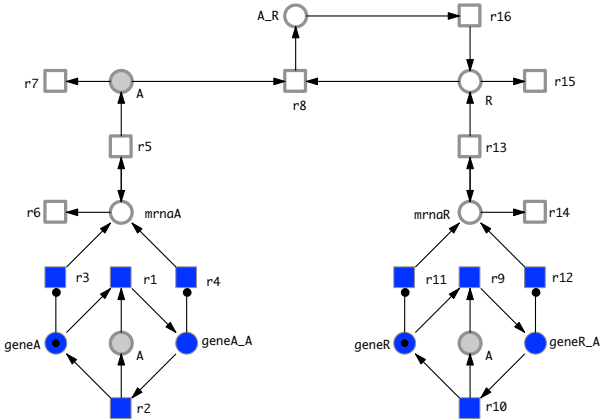
- species $=\{S_1, S_2, P\}$
- substrates $=\{S_1, S_2\}$
- products $=\{P\}$
- rate constants $=\{k\}$
- reaction rate:
  $MassAction(k) = k \cdot S_1 \cdot S_2$

Reaction network - example - cartoon



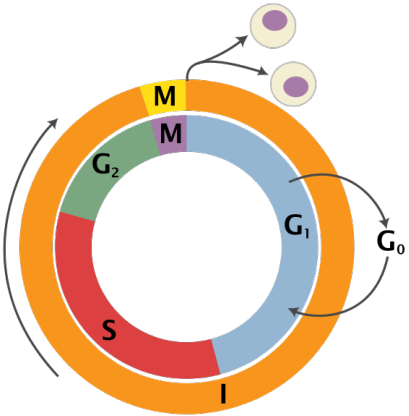
Blätke, Heiner & Marwan, BioModel Engineering with Petri Nets, Elsevier (2015)

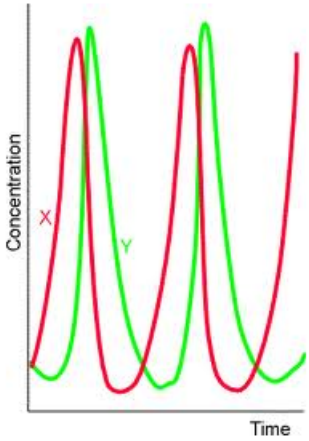
Reaction network - example - Petri net



Blätke, Heiner & Marwan, BioModel Engineering with Petri Nets, Elsevier (2015)

The Problem





The Problem

Hybrid simulation (of biological reaction networks)

- combines deterministic and stochastic regime;
- used as an alternative approach when
  - pure stochastic simulation is too expensive,
  - pure deterministic simulation neglects inherently discrete and/or stochastic effects;
- provides a trade-off between accuracy and efficiency.

Hybrid simulation: How does it work?

repeat the following steps:

- (re)initialise the ODE solver
- numerically integrate the system of ODEs until a stochastic event is to occur
- find the index of the stochastic reaction to occur
- fire this stochastic reaction
- update the propensities of the stochastic reactions

Hybrid simulation: How does it work?

repeat the following steps:

- (re)initialise the ODE solver
- numerically integrate the system of ODEs until a stochastic event is to occur
- find the index of the stochastic reaction to occur
- fire this stochastic reaction
- update the propensities of the stochastic reactions

Hybrid simulation: How does it work?

repeat the following steps:

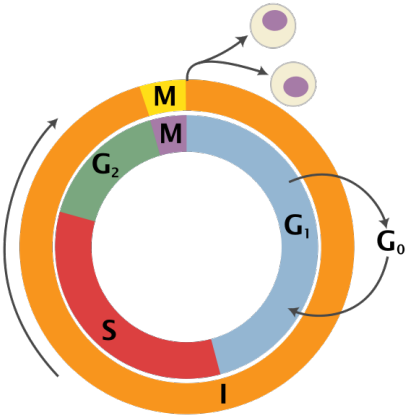
- (re)initialise the ODE solver
- numerically integrate the system of ODEs
  until a stochastic event is to occur → jump equation
- find the index of the stochastic reaction to occur
- fire this stochastic reaction
- update the propensities of the stochastic reactions

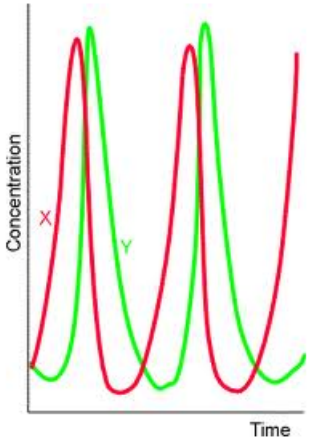
Hybrid simulation: two performance issues

Additional computational overhead caused by

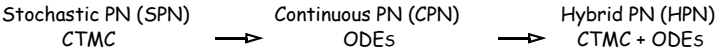
1. jump equation
   - to decide when to go to the stochastic regime
   - needs to be done during deterministic simulation
   - decision depends on propensities of stochastic transitions
2. re-initialisation of ODE solver
   - when coming back from the stochastic regime discontinuities may occur
   - may be rather frequent
   - expensive for multiple step-size ODE solvers recording accuracy and history information

Our Approach

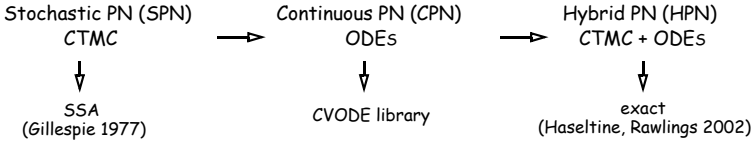




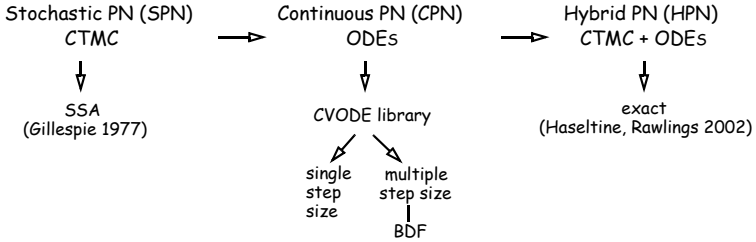
A brief history of . . .



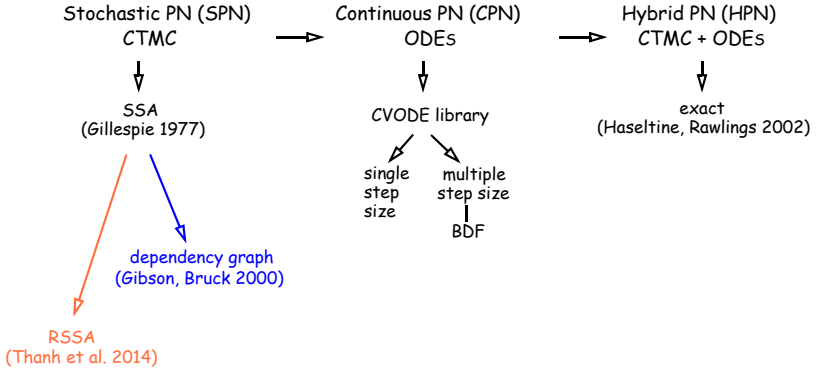
A brief history of . . .



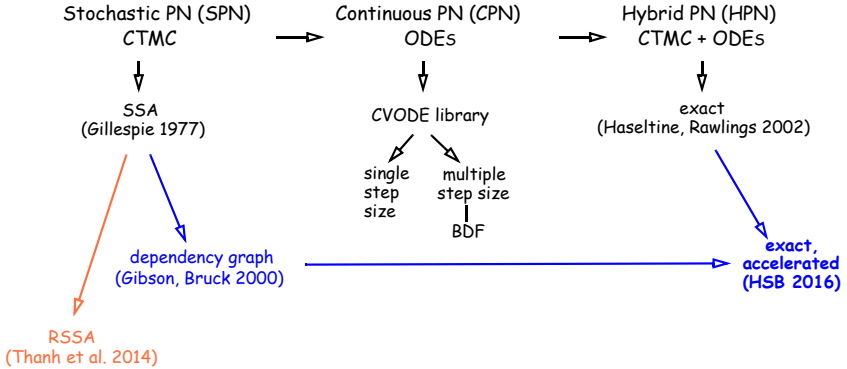
A brief history of . . .



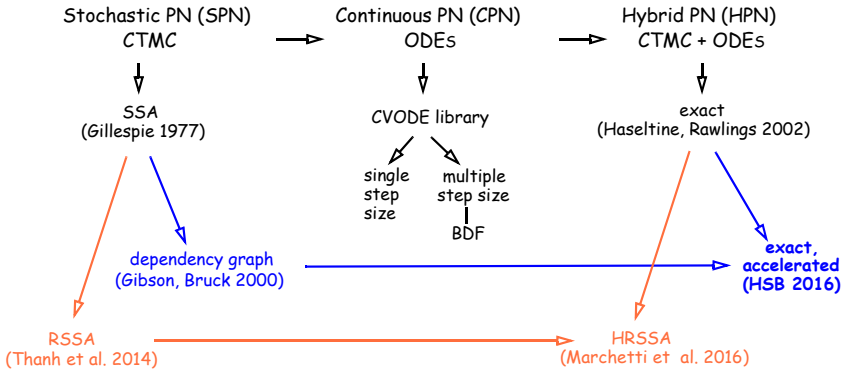
A brief history of . . .



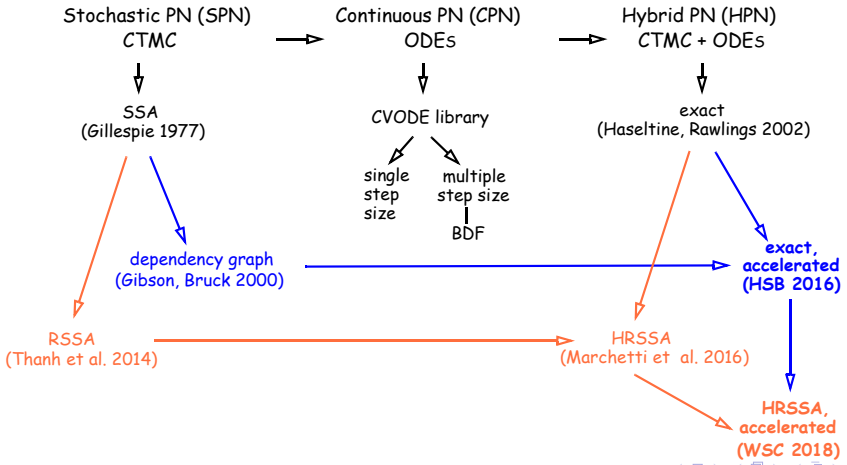
A brief history of . . .



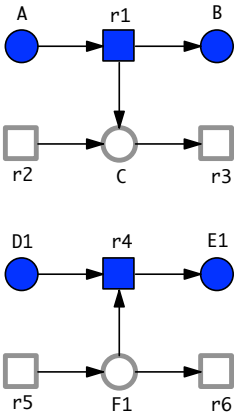
A brief history of . . .



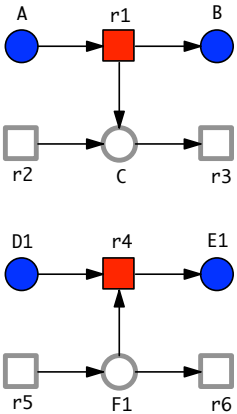
A brief history of . . .



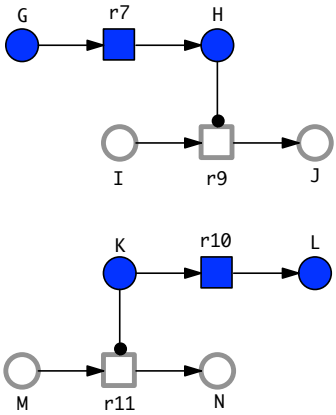
Structural analysis - interface transitions



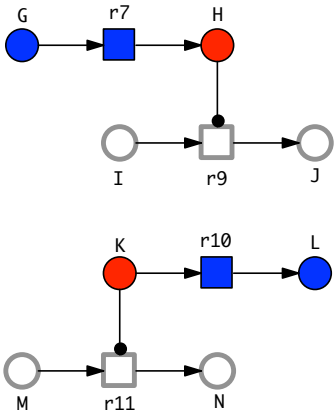
Structural analysis - interface transitions



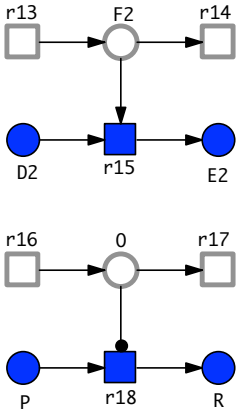
Structural analysis - interface places



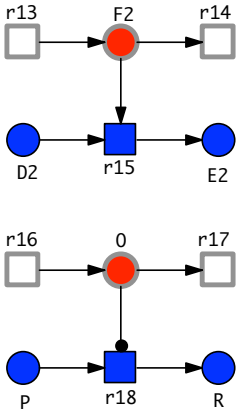
Structural analysis - interface places



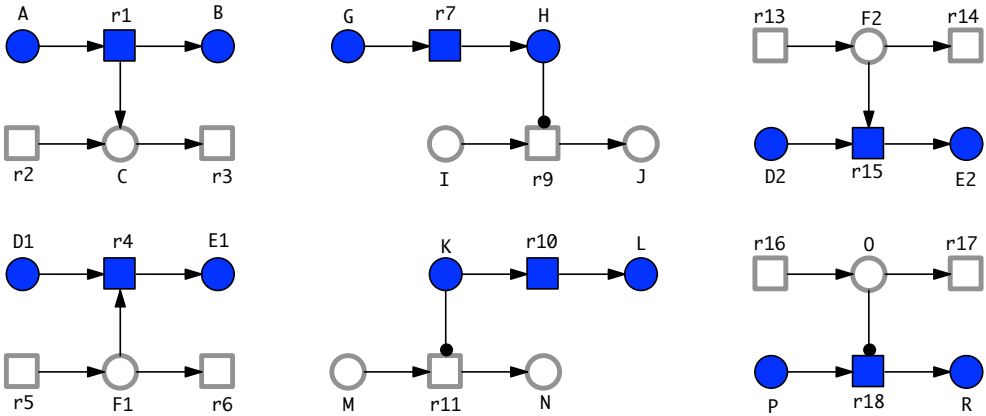
Structural analysis - monitored places



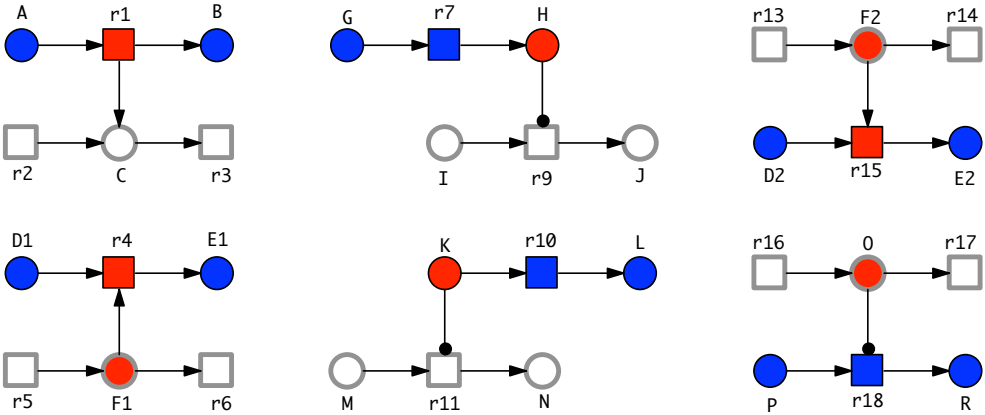
Structural analysis - monitored places



Structural analysis - summary



Structural analysis - summary



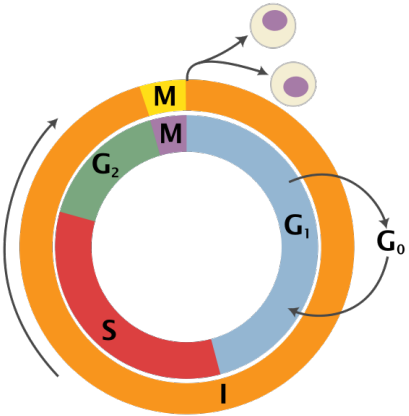
Structural analysis - terminology

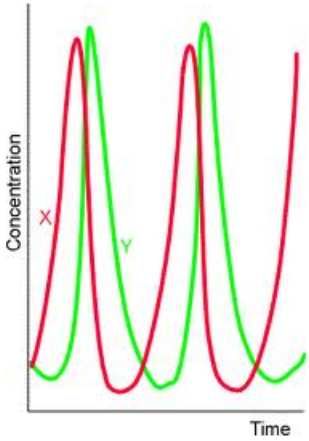
- interface transitions, direct dependencies → ODEs initialisation
  - stochastic transitions
  - sharing pre/post-places with continuous transitions
- interface places, indirect dependencies → ODEs initialisation
  - stochastic places
  - side conditions of continuous transitions
- monitored places → jump equation
  - continuous places
  - pre-places of stochastic transitions

Structural analysis - summary

- done only once
  during initialisation of hybrid simulation
- speed-up
  depends on interactions between stochastic and continuous model parts
- preserves accuracy
  accuracy of accelerated HRSSA = accuracy of HRRSA
- immediate transitions
  often stand for rare events, thus not considered (so far)

Evaluation



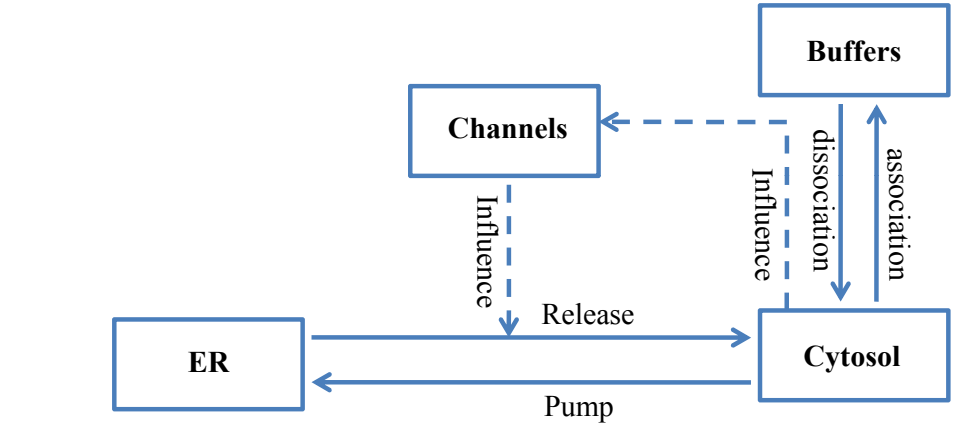


Case Studies

We use three case studies to test or approach:

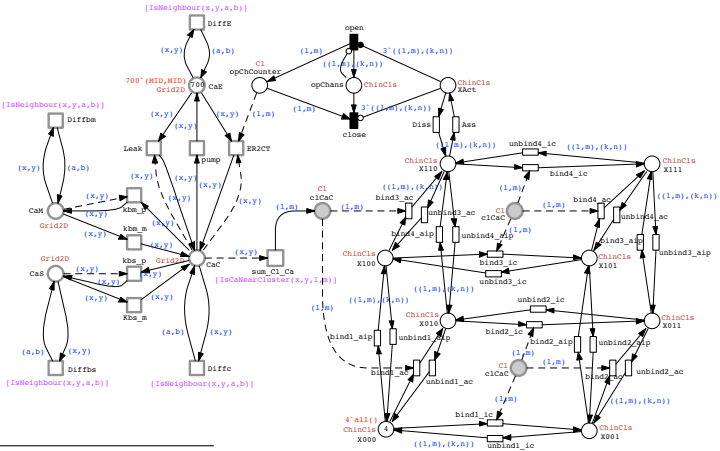
1. Calcium model (coloured $\mathcal{GHPN}$)
   Ismail, Herajy & Heiner, ARSBM (2018), in print
2. Eukaryotic cell cycle ($\mathcal{GHPN}$)
   Herajy, Schwarick & Heiner, ToPNoC (2013)
3. Yeast cell cycle (coloured $\mathcal{GHPN}$)
   Herajy, Liu & Heiner, J. Nonlinear Analysis: Hybrid Systems (2018)

Case study: Calcium model - cartoon



Ismail, Herajy & Heiner, ARSBM (2018)

Case study: Calcium model - coloured $\mathcal{GHPN}$



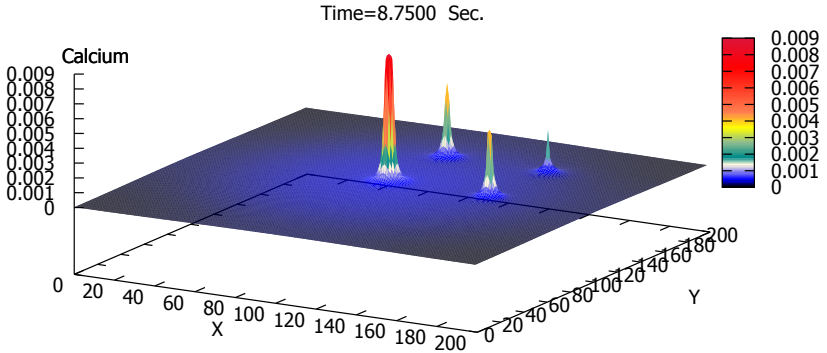
Ismail, Herajy & Heiner, ARSBM (2018)

Case study: Calcium model - model size

- 10,252 species (places) generated out of 16 coloured places
  discrete: 251, continuous: 10001
- 57,401 reactions (transitions) generated out of 40 transitions
  stochastic + immediate: 650 + 50, continuous: 56701
- structural analysis
  - interface transitions: 0
  - interface places: 0 (opChCounter, modified by immediate transitions)
  - monitored places: 1 (clCaC)
- accelerated HRSSA about **two times faster** than exact simulation

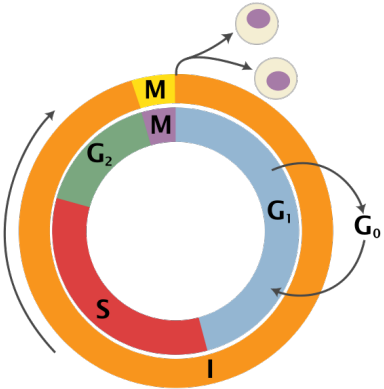
Case study: Calcium model - 2D trace

Simulation results, example:

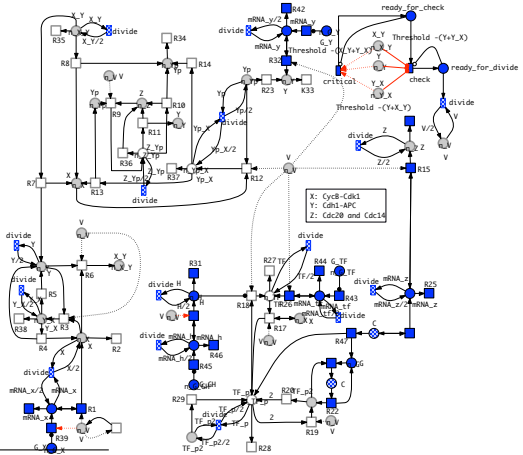


Case studies: cell cycle regulation

- S phase (synthesis)
- G2 gap
- M phase (mitosis)
- G1 gap



Case study: Eukaryotic cell cycle - $\mathcal{GHPN}$



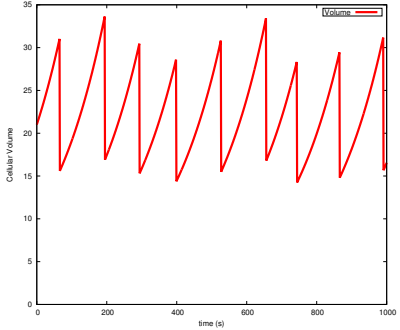
Herajy, Schwarick & Heiner, ToPNoC (2013)

Case study: Eukaryotic cell cycle - model size

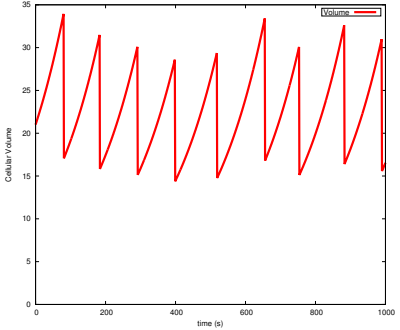
- 26 species (places)
  discrete: 14, continuous: 12
- 51 reactions (transitions)
  stochastic + immediate: 20 + 3, continuous: 28
- structural analysis
  - interface transitions: 8
  - interface places: 1
  - monitored places: 3
- accelerated HRSSA about **five times faster** than exact simulation

Case study: Eukaryotic cell cycle - traces

Cellular volume - simulation results, single runs:

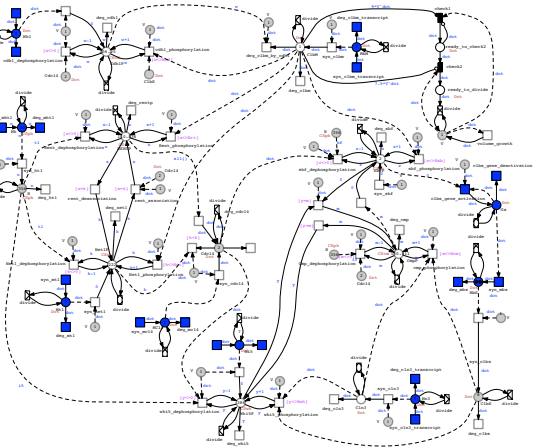


exact



accelerated HRSSA

Case study: Yeast cell cycle - coloured $\mathcal{GHPN}$



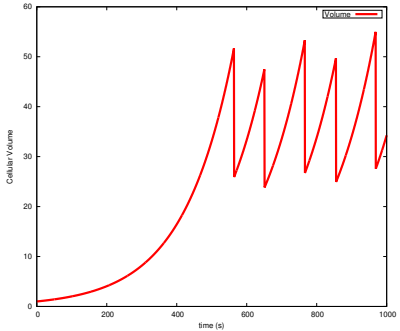
Herajy, Liu & Heiner, NAHS (2018)

Case study: Yeast cell cycle - model size

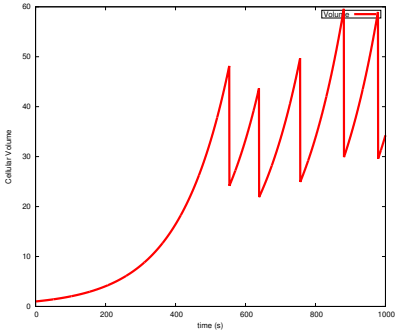
- 62 species (places) generated out of 23 coloured places
  discrete: 13, continuous: 49
- 194 reactions (transitions) generated out of 62 coloured transitions
  stochastic + immediate: 22 + 3, continuous: 169
- structural analysis
  - interface transitions: 0
  - interface places: 10
  - monitored places: 2
- accelerated HRSSA about **six times faster** than exact simulation

Case study: Yeast cell cycle - traces

Cellular volume - simulation results, single runs:



exact



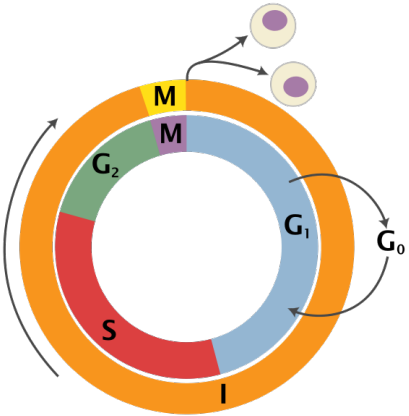
accelerated HRSSA

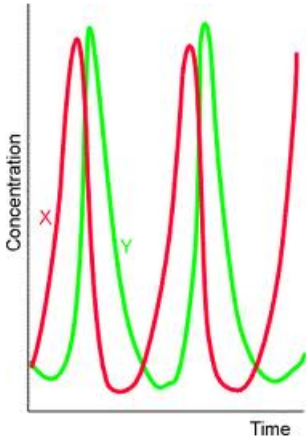
Case studies: summary

Models/ Algorithms	Haseltine &Rawlings	Accelerated	HRSSA	Accelerated HRSSA	Simulation Time
Yeast Cell Cycle	984.5	423	435.2	163.3	10,000 min.
Eukaryotic Cell Cycle	780	276.2	240	164.2	1,000 min.
Calcium model	22,620	14,460	30,980	10,800	10 sec.

For each case study, the number of stochastic events generated by the four algorithms are comparable.

Conclusions





Implementation - tools supporting $\mathcal{GHPN}$

- Snoopy (2010) - GUI tool for design and simulation of Petri nets

- S4 (2014) - Snoopy Simulation and Steering Server

- Spike (2018) - CLI tool for simulation of Petri nets, using config scripts

All available at http://www-dssz.informatik.tu-cottbus.de/DSSZ/Software/Software

Conclusions

- We have presented an approach for improving the performance of hybrid simulation algorithms → accelerated HRSSA
  - exploiting structural knowledge.
- The improvements
  - minimise the work done by the jump equation,
  - minimise the re-initialisation of the ODEs simulation,
  - see paper for details.
- The suggested improvements will be useful to cope with the rapid growth of (biological) models.

Future work

- Extend structural analysis to immediate transitions (optional).
- Better understand why performance gain may depend on the chosen ODEs solver.
- User guidance to find a partitioning minimising the interface between the stochastic and deterministic subnet.
- Investigating dynamic partitioning in combination with accelerated HRSSA.
- Performance comparison for more case studies (using Spike).

Any questions?