Coloured Hybrid Petri Nets for Systems Biology

Mostafa Herajy [1], Fei Liu [2] and Christian Rohr [3]

Department of Mathematics and Computer Science, Faculty of Science, Port Said University - Port Said, Egypt

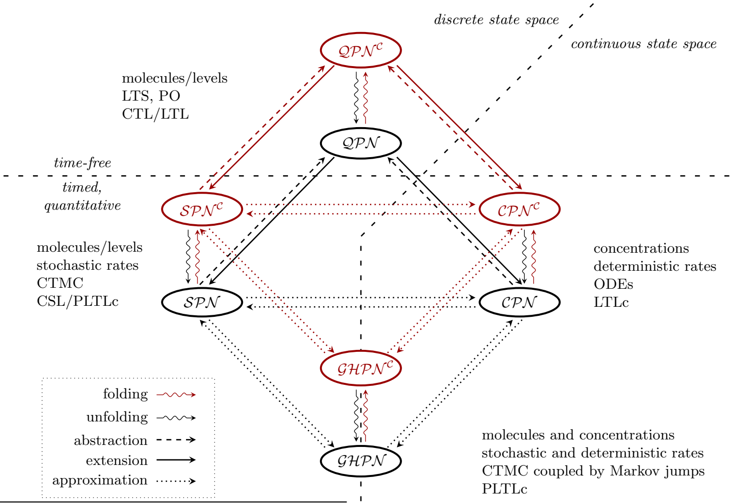
Control and Simulation Center, Harbin Institute of Technology

Computer Science Institute, Brandenburg University of Technology Cottbus, Germany

Tunis 2014

Agenda

Petri Nets in Snoopy [1]



[1] Heiner et al. 2012

Motivations for $\mathcal{GHPN}^{\mathcal{C}}$

- The rabid change of the size of biological models
- Certain biological phenomena necessitates the existence of discrete and continuous variables as well as continuous and stochastic processes in one and the same model
- Coloured Petri nets can easily be used to model biological systems with repetition of components

Motivations for $\mathcal{GHPN}^{\mathcal{C}}$

- The rabid change of the size of biological models
- Certain biological phenomena necessitates the existence of discrete and continuous variables as well as continuous and stochastic processes in one and the same model
- Coloured Petri nets can easily be used to model biological systems with repetition of components

Motivations for $\mathcal{GHPN}^{\mathcal{C}}$

- The rabid change of the size of biological models
- Certain biological phenomena necessitates the existence of discrete and continuous variables as well as continuous and stochastic processes in one and the same model
- Coloured Petri nets can easily be used to model biological systems with repetition of components

Features

- Can easily be used to model scalable biological systems
- Combines both CPN and GSPN into one class
- Different transition types→ different reaction types can be modelled using GHPN
- Stiff biochemical networks can be easily modelled and simulated using GHPN
- The final model can be simulated using either static or dynamic partitioning
- Biological switch can be easily represented

Features

- Can easily be used to model scalable biological systems
- Combines both CPN and GSPN into one class
- Different transition types$\longrightarrow$ different reaction types can be modelled using GHPN
- Stiff biochemical networks can be easily modelled and simulated using GHPN
- The final model can be simulated using either static or dynamic partitioning
- Biological switch can be easily represented

Features

- Can easily be used to model scalable biological systems
- Combines both CPN and GSPN into one class
- Different transition types→ different reaction types can be modelled using GHPN
- Stiff biochemical networks can be easily modelled and simulated using GHPN
- The final model can be simulated using either static or dynamic partitioning
- Biological switch can be easily represented

Features

- Can easily be used to model scalable biological systems
- Combines both CPN and GSPN into one class
- Different transition types→ different reaction types can be modelled using GHPN
- Stiff biochemical networks can be easily modelled and simulated using GHPN
- The final model can be simulated using either static or dynamic partitioning
- Biological switch can be easily represented

Features

- Can easily be used to model scalable biological systems
- Combines both CPN and GSPN into one class
- Different transition types→ different reaction types can be modelled using GHPN
- Stiff biochemical networks can be easily modelled and simulated using GHPN
- The final model can be simulated using either static or dynamic partitioning
- Biological switch can be easily represented

Features

- Can easily be used to model scalable biological systems
- Combines both CPN and GSPN into one class
- Different transition types→ different reaction types can be modelled using GHPN
- Stiff biochemical networks can be easily modelled and simulated using GHPN
- The final model can be simulated using either static or dynamic partitioning
- Biological switch can be easily represented

Elements

Places

Transitions

Arcs

Elements

Places

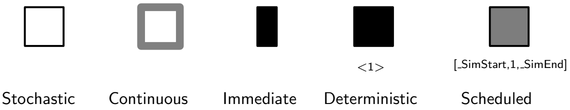
Transitions

Arcs

Elements

Places

Transitions

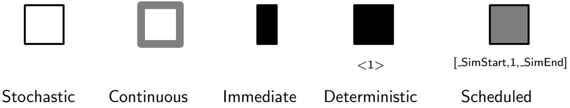


Arcs

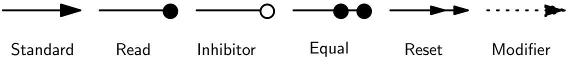
Elements

Places

Transitions

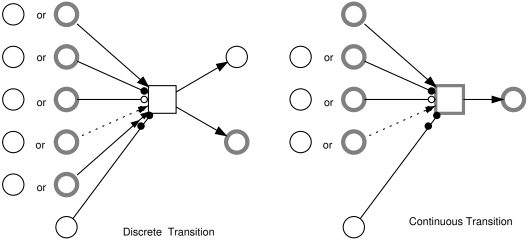


Arcs



M. Herajy and M. Heiner, NAHS (2012)

Connectivity

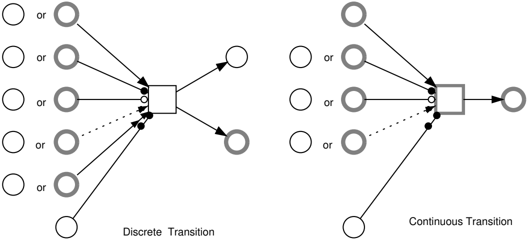


The restrictions are:

- discrete places cannot be connected with continuous transitions using standard arcs,
- continuous places cannot be tested with equal arcs,
- and continuous transitions cannot use reset arcs.

M. Herajy and M. Heiner, NAHS (2012)

Connectivity

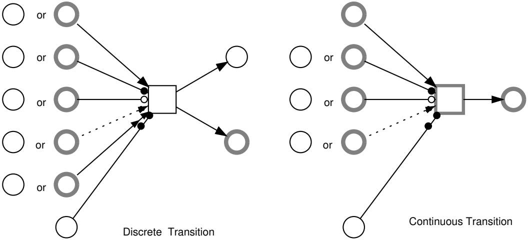


The restrictions are:

- discrete places cannot be connected with continuous transitions using standard arcs,
- continuous places cannot be tested with equal arcs,
- and continuous transitions cannot use reset arcs.

M. Herajy and M. Heiner, NAHS (2012)

Connectivity



The restrictions are:

- discrete places cannot be connected with continuous transitions using standard arcs,
- continuous places cannot be tested with equal arcs,
- and continuous transitions cannot use reset arcs.

M. Herajy and M. Heiner, NAHS (2012)

Simulation of $\mathcal{GHPN}^{\mathcal{C}}$

Unfolding

- Coloured hybrid Petri nets can be unfolded into uncoloured ones where many analysis and simulation techniques can be applied.
- The conversion between uncoloured and coloured Petri nets changes the style of representation, but does not change the actual net structure of the underlying biological reaction network.

Unfolding

- Coloured hybrid Petri nets can be unfolded into uncoloured ones where many analysis and simulation techniques can be applied.
- The conversion between uncoloured and coloured Petri nets changes the style of representation, but does not change the actual net structure of the underlying biological reaction network.

Unfolding (Cont.)

- If the colour set of each variable in a transition guard has a finite integer domain, a constraint satisfaction approach is used to obtain all valid transition instances.
- Otherwise, a general unfolding algorithm is adopted, in which some optimization techniques like partial binding partial test and pattern matching are used

Unfolding (Cont.)

- If the colour set of each variable in a transition guard has a finite integer domain, a constraint satisfaction approach is used to obtain all valid transition instances.
- Otherwise, a general unfolding algorithm is adopted, in which some optimization techniques like partial binding partial test and pattern matching are used

Simulation of $\mathcal{GHPN}^{\mathcal{C}}$

Two key issues are very important in simulating $\mathcal{GHPN}^{\mathcal{C}}$:

- The partitioning of the net transitions into stochastic and continuous ones.
- The synchronization between the stochastic and continuous regimes.

Simulation of $\mathcal{GHPN}^{\mathcal{C}}$

Two key issues are very important in simulating $\mathcal{GHPN}^{\mathcal{C}}$:

- The partitioning of the net transitions into stochastic and continuous ones.
- The synchronization between the stochastic and continuous regimes.

Simulation of $\mathcal{GHPN}$ (Cont.)

- Static Partitioning
- Dynamic Partitioning

Simulation of $\mathcal{GHPN}$ (Cont.)

- Static partitioning: partitioning is done off-line before the simulation starts.
- Dynamic partitioning: partitioning is done on-line during the simulation.

Simulation of $\mathcal{GHPN}$ (Cont.)

- Static partitioning: partitioning is done off-line before the simulation starts.
- Dynamic partitioning: partitioning is done on-line during the simulation.

Synchronization Mechanism

One option is to use (1) [2]:

$$g(\mathbf{x}) = \int_{t}^{t+\tau} a_0^s(\mathbf{x})dt - \xi = 0\,, \qquad (1)$$

where ξ is a random number exponentially distributed with a unit mean, and $a_0^s(\mathbf{x})$ is the cumulative rate of all the stochastic transitions.

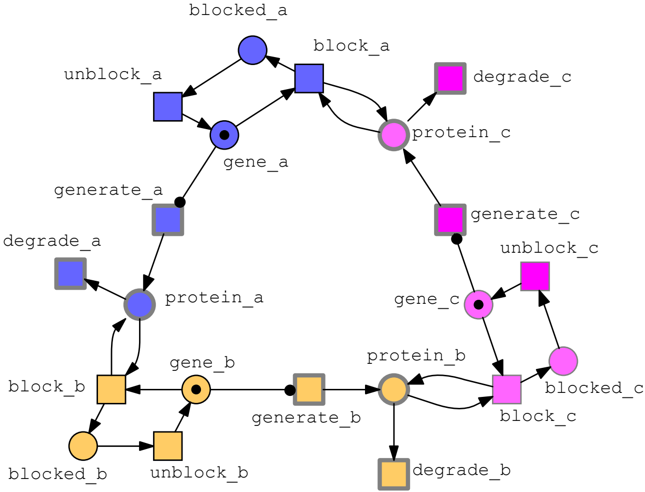
[2] Gillespie 91, Markov Processes

Case Study

Repressilator

- It is an example of a synthetic circuit.
- The repressilator system is a regulatory cycle of three genes (gene a, gene b and gene c).
- Each gene represses its successor, namely, gene a inhibits gene b, gene b inhibits gene c, and gene c inhibits gene a.
- This negative regulation is realized by the repressors, protein a, protein b and protein c.

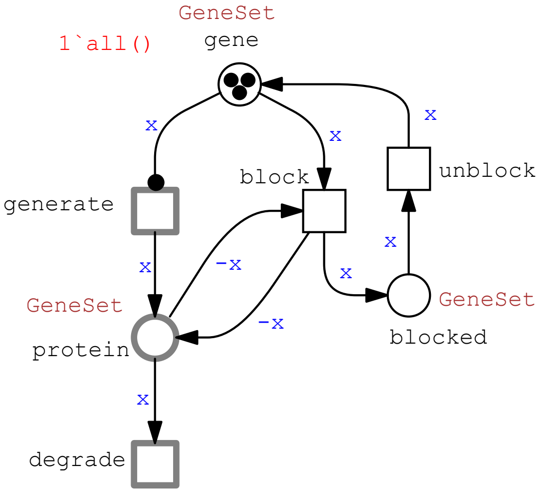
$\mathcal{GHPN}$ Representation



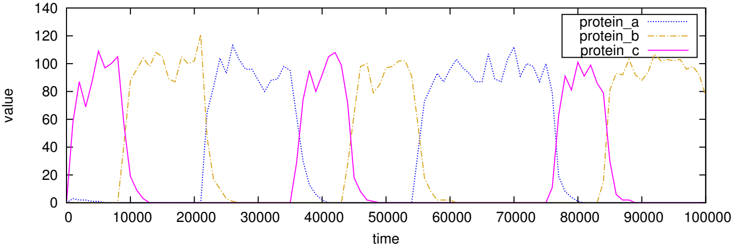
$\mathcal{GHPN}^{\mathcal{C}}$ Net Partitioning

- The 1-bounded places as determined by P-invariant analysis and the related transitions as determined by T-invariant analysis are kept discrete.
- The unbounded places and related transitions are approximated by continuous places and transitions, respectively.

$\mathcal{GHPN}^{\mathcal{C}}$ Representation



$\mathcal{GHPN}^{\mathcal{C}}$ Simulation Result



Implementation

All the features of $\mathcal{GHPN}^{\mathcal{C}}$ are implemented in:

- Snoopy – a unifying Petri net editing tool.
- S^4 – Snoopy Steering and Simulation Server

Conclusion

- In this paper we have introduced a class of coloured Petri nets called Coloured Generalised Hybrid Petri Nets ($\mathcal{GHPN}^{\mathcal{C}}$).
- $\mathcal{GHPN}^{\mathcal{C}}$ are particularly tailored to systems biologists needs to model and analyse multiscales models.
- $\mathcal{GHPN}^{\mathcal{C}}$ provide the interplay between stochastic and continuous regimes on the coloured level

Future work

- Adding marking-dependent arc weights.
- Simulation of $\mathcal{GHPN}^{\mathcal{C}}$ on the coloured level
- Implementing more complex case studies.

Acknowledgement

The authors would like to acknowledge the help of Monika Heiner during the manuscript preparation and the model construction

Thank You