Colored Petri Nets for Multiscale Systems Biology

Fei Liu
Control and Simulation Center
Harbin Institute of Technology
Outline

- Multiscale Systems Biology
- Colored Petri nets
- Colored Petri net framework
- Key techniques
- Analysis techniques
- Applications
- Summary
Outline

- Multiscale Systems Biology
- Colored Petri nets
- Colored Petri net framework
- Key techniques
- Analysis techniques
- Applications
- Summary
Systems Biology

- understand biology at a system level [Kitano 2002]

- take into account complex interactions at different scales of biological organization [Popel et al. 2009]

<table>
<thead>
<tr>
<th>Biological scale</th>
<th>Some hallmark phenomena</th>
</tr>
</thead>
<tbody>
<tr>
<td>Molecules</td>
<td>Signaling network</td>
</tr>
<tr>
<td></td>
<td>Metabolic network</td>
</tr>
<tr>
<td></td>
<td>Compartmentalization</td>
</tr>
<tr>
<td>Cells</td>
<td>Multicellularity</td>
</tr>
<tr>
<td></td>
<td>Variant (mutants)</td>
</tr>
<tr>
<td></td>
<td>Cell-cell communication</td>
</tr>
<tr>
<td></td>
<td>Cell movement</td>
</tr>
<tr>
<td></td>
<td>Cellular differentiation</td>
</tr>
<tr>
<td></td>
<td>(Hierarchical) organization</td>
</tr>
<tr>
<td>Tissues</td>
<td>Pattern formation</td>
</tr>
<tr>
<td>Organs</td>
<td>(Hierarchical) organization</td>
</tr>
<tr>
<td>Organisms</td>
<td>(Hierarchical) organization</td>
</tr>
</tbody>
</table>
Multiscale challenges

- Repetition
- Variation
- Organization
- Hierarchical organization
- Communication
- Movement
- Replication
- Deletion
- Differentiation
- Dynamic (variable) grid size
- Pattern formation

[Gao et. al. 2011]
Modeling challenges

How to
- Design & construct models
- Simulate models
- Analyze models
- Visualize results

To use
- Colored Petri nets
Outline

- Multiscale Systems Biology
- Colored Petri nets
- Colored Petri net framework
- Key techniques
- Analysis techniques
- Applications
- Summary
Colored Petri nets

Chemical reactions:

- Prey $\rightarrow$ 2Prey
- Prey + Predator $\rightarrow$ 2Predator
- Predator $\rightarrow$ $\lambda$
Colored Petri nets

![Diagram of colored Petri nets with two subsystems: sub-system1 and sub-system2. Each subsystem includes transitions for reproduction of prey, predator death, consumption of prey, and two sets of prey and predator tokens.](image)
Colored Petri nets

sub-system1

reproduction_of_prey

predator_death

Prey1

Predator1

consumption_of_prey

50

100

2

sub-system2

reproduction_of_prey

predator_death

Prey2

Predator2

consumption_of_prey

50

100

2

Tom and Jerry cartoon image
Colored Petri nets

Declarations:
- colorset CS = red, blue;
- variable x : CS

- Changing color sets adapts the model to various scenarios
Colored Petri nets – basic notions

Declarations:
- colorset CS = red, blue;
- variable x : CS

Notions:
- Multiset
  - a set in which there can be several occurrences for the same element
  - denoted by \( m(s1)\times s1 ++ m(s2)\times s2 ++ \cdots \)
  - Example: \{a, a, b, b, b\} = \( 2\times a ++ 3\times b \)
Colored Petri nets – basic notions

Declarations:
- colorset CS = red, blue;
- variable x : CS

Notions:
- Place/transition/arc
- Color sets
- Guards: Boolean expressions
- Arc expressions: result type = type of connected place
- Initial marking: initialization expressions
**Colored Petri nets – basic notions**

**Declarations:**
- colorset CS = red, blue;
- variable x : CS

**Notions:**
- Color sets
  - Define how many components/subsystems (cells) in a colored Petri net model
  - Each component (cell) is a color

- Simple types: dot, integer, string, Boolean, enumeration, index
- Compound types: product, union
Colored Petri nets – basic notions

Declarations:
- colorset CS = a,b,c;
- variable x : CS

Notions:
- Guard
  - Boolean expression (true/false)
  - Select those uncolored transitions where the guard is evaluated to true
Colored Petri nets – basic notions

Declarations:
- colorset CS = red, blue;
- variable x : CS

Notions:
- Arc expressions
  - usually variables of color sets
  - e.g., x, 2\`x
  - result type = type of connected place
- Initial marking
  - A multiset expression
  - 50`all() == 50`a ++ 50`b
Colored Petri nets – basic notions

- Folding/unfolding
- Place/transition instances

Questions:
- Automatic folding
- Automatic unfolding
Outline

- Multiscale Systems Biology
- Colored Petri nets
- Colored Petri net framework
- Key techniques
- Analysis techniques
- Applications
- Summary
Colored Petri net framework

Colored Petri net – implementation
Colored qualitative Petri net (QPNC)

- Colored extension of QPN
- Special arcs: inhibitor, read, equal, reset
- Animation built in Snoopy
- Export (unfold) to external analysis tools
Colored stochastic Petri net (SPNC)

- Colored extension of SPN
- Transition: stochastic rate with an exponential probability distribution
- Semantics: continuous time Markov chains (CTMCs)
- Special arcs, modifier arcs
- Special transitions: immediate, deterministic, scheduled
- Animation/stochastic simulation built in Snoopy
- Export (unfold) to external analysis tools
Colored continuous Petri net (CPNC)

- Colored extension of CPN
- Place: real values
- Transition: deterministic rates
- Semantics: a set of ordinary differential equations (ODEs)
- Special arcs: inhibitor, reader, modifier
- Continuous simulation built in Snoopy
- Export (unfold) to external analysis tools
Colored hybrid Petri net (HPNC)

- Colored extension of GHPN
- Place: real/integer values
- Transition: deterministic/stochastic rates
- Semantics: ODEs/CTMC
- Special arcs: inhibitor, reader, equal, reset, modifier
- Hybrid simulation built in Snoopy
- Export (unfold) to external analysis tools
Outline

- Multiscale Systems Biology
- Colored Petri nets
- Colored Petri net framework
- Key techniques
- Analysis techniques
- Applications
- Summary
Key techniques

- Annotation language of colored Petri nets

\[
\begin{align*}
\langle \text{Type} \rangle & ::= \langle \text{SimpleType} \rangle \mid \langle \text{CompoundType} \rangle \\
\langle \text{SimpleType} \rangle & ::= \langle \text{TypeIdentifier} \rangle \mid \langle \text{StructuredType} \rangle \\
\langle \text{TypeIdentifier} \rangle & ::= \langle \text{UnsignedInteger} \rangle \mid \langle \text{Boolean} \rangle \mid \langle \text{String} \rangle \\
\langle \text{UnsignedInteger} \rangle & ::= \text{“int”} \\
\langle \text{Boolean} \rangle & ::= \text{“bool”} \\
\langle \text{String} \rangle & ::= \text{“string”} \\
\langle \text{StructuredType} \rangle & ::= \langle \text{Enumeration} \rangle \mid \langle \text{Index} \rangle \\
\langle \text{Enumeration} \rangle & ::= \langle \text{IdentifierList} \rangle \\
\langle \text{IdentifierList} \rangle & ::= \langle \text{Identifier} \rangle \mid \langle \text{IdentifierList} \rangle \text{“,”} \langle \text{Identifier} \rangle \\
\langle \text{Index} \rangle & ::= \langle \text{Identifier} \rangle \text{“[”} \langle \text{IndexSpecifier} \rangle \text{“]”} \\
\langle \text{IndexSpecifier} \rangle & ::= \text{“int”} \\
\langle \text{CompoundType} \rangle & ::= \langle \text{Product} \rangle \mid \langle \text{Union} \rangle \\
\langle \text{Product} \rangle & ::= \langle \text{Type} \rangle \text{“×”} \langle \text{Type} \rangle \mid \langle \text{Product} \rangle \text{“×”} \langle \text{Type} \rangle \\
\langle \text{Union} \rangle & ::= \langle \text{Type} \rangle \mid \langle \text{Union} \rangle \text{“,”} \langle \text{Type} \rangle
\end{align*}
\]
Key techniques

- Annotation language of colored Petri nets

\[
\begin{align*}
\langle \text{ColorExpr} \rangle & ::= \langle \text{MultiSetExpr} \rangle \\
\langle \text{MultiSetExpr} \rangle & ::= \langle \text{Predicate} \rangle \mid \langle \text{MultiSetExpr} \rangle \langle \text{MSAdditionOp} \rangle \langle \text{Predicate} \rangle \\
\langle \text{MSAdditionOp} \rangle & ::= \text{	extquoteleft++	extquoteright} \\
\langle \text{Predicate} \rangle & ::= \langle \text{SeparatorExpr} \rangle \mid \text{[} \langle \text{OrExpr} \rangle \text{]} \langle \text{SeparatorExpr} \rangle \\
\langle \text{SeparatorExpr} \rangle & ::= \langle \text{TupleExpr} \rangle \mid \langle \text{SeparatorExpr} \rangle \langle \text{SeparatorOp} \rangle \langle \text{TupleExpr} \rangle \\
\langle \text{SeparatorOp} \rangle & ::= \text{	extquoteright\textquoteright} \\
\langle \text{TupleExpr} \rangle & ::= \langle \text{OrExpr} \rangle \mid \text{("} \langle \text{CommaExpr} \rangle \text{")} \\
\langle \text{CommaExpr} \rangle & ::= \langle \text{TupleExpr} \rangle \mid \langle \text{CommaExpr} \rangle \langle \text{CommaOp} \rangle \langle \text{TupleExpr} \rangle \\
\langle \text{CommaOp} \rangle & ::= \text{,} \\
\langle \text{OrExpr} \rangle & ::= \langle \text{AndExpr} \rangle \mid \langle \text{OrExpr} \rangle \langle \text{OrOp} \rangle \langle \text{AndExpr} \rangle \\
\langle \text{OrOp} \rangle & ::= \text{	extquoteleft	extquoteright} \\
\langle \text{AndExpr} \rangle & ::= \langle \text{EqualExpr} \rangle \mid \langle \text{AndExpr} \rangle \langle \text{AndOp} \rangle \langle \text{EqualExpr} \rangle \\
\langle \text{AndOp} \rangle & ::= \text{	extquoteright&	extquoteright} \\
\langle \text{EqualExpr} \rangle & ::= \langle \text{RelationExpr} \rangle \mid \langle \text{EqualExpr} \rangle \langle \text{EqualOp} \rangle \langle \text{RelationExpr} \rangle \\
\langle \text{EqualOp} \rangle & ::= \text{	extquoteleft=\textquoteright} \mid \text{	extquoteleft<>\textquoteright} \\
\langle \text{RelationExpr} \rangle & ::= \langle \text{AddExpr} \rangle \mid \langle \text{RelationExpr} \rangle \langle \text{RelationOp} \rangle \langle \text{AddExpr} \rangle \\
\langle \text{RelationOp} \rangle & ::= \text{	extquoteleft<\textquoteright} \mid \text{	extquoteleft<=\textquoteright} \mid \text{	extquoteleft>=\textquoteright} \mid \text{	extquoteleft>\textquoteright} \\
\langle \text{AddExpr} \rangle & ::= \langle \text{MultiplicityExpr} \rangle \mid \langle \text{AddExpr} \rangle \langle \text{AddOp} \rangle \langle \text{MultiplicityExpr} \rangle \\
\langle \text{AddOp} \rangle & ::= \text{	extquoteleft+\textquoteright} \mid \text{	extquoteleft-\textquoteright}
\end{align*}
\]
Key techniques

- Annotation language of colored Petri nets
- Flex scanner & Bison parser
- C++ vs ML (functional programming) language

F Liu, M Heiner and C Rohr: Manual for Colored Petri Nets in Snoopy; Technical report 02-12, Brandenburg University of Technology Cottbus, Department of Computer Science, March 2012.
Key techniques

- Unfolding algorithm
- Reuse analysis techniques and tools for standard Petri nets
- Reuse stochastic or continuous simulation algorithms

Key techniques

- Unfolding algorithm
  - compute all instances (bindings) for every transition
  - bind every variable to each color of its color set
  - the combination of color sets if the guard is always true

Declarations:
colorset cs1=int with 1,2,3;
colorset cs2=int with 1,2;
variable x:cs1;
variable y:cs2;
Key techniques

- Unfolding algorithm
  - the number of instances for a transition is constrained by its guard
- A constraint satisfaction approach
  - the guard is not always true
  - all the variables in the guard have finite integer domains

Declarations:
- colorset CS1=int with 1,2,3;
- colorset CS2=int with 1,2;
- variable x:CS1;
- variable y:CS2;
Key techniques

- Unfolding algorithm
  - get the guard of a transition
  - get all variables in this guard
  - define the color set of each variable as its domain in CSP
  - define the guard as a constraint of CSP
  - solve CSP using constraint solvers, e.g. Gecode

```plaintext
Declarations:
colorset CS1=int with 1,2,3;
colorset CS2=int with 1,2;
variable x:CS1;
variable y:CS2;
```
Key techniques

Unfolding algorithm

<table>
<thead>
<tr>
<th>Size</th>
<th>Unfolding time</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>without CSP</td>
</tr>
<tr>
<td>$M \times N$</td>
<td>Places</td>
</tr>
<tr>
<td>10x10</td>
<td>100</td>
</tr>
<tr>
<td>50x50</td>
<td>2,500</td>
</tr>
<tr>
<td>100x100</td>
<td>10,000</td>
</tr>
<tr>
<td>200x200</td>
<td>40,000</td>
</tr>
</tbody>
</table>

*done on PC, Intel(R) Xeon(R) CPU 2.83GHz, RAM 4.00GB. ◇ we did not get the result within 24 hours.
Outline

- Multiscale Systems Biology
- Colored Petri nets
- Colored Petri net framework
- Key techniques
- Analysis techniques
- Applications
- Summary
Analysis capabilities

- built in Snoopy
  - animation ($QPN^c/SPN^c$)
  - simulation ($SPN^c/CPN^c/HPN^c$)

- external analysis tools
  - structural analysis (all net classes): Charlie
  - CTL model checking ($QPN^c$): Marcie
  - numerical CSL model checking ($SPN^c$): Marcie
  - simulative PLTLc model checking ($SPN^c$): Marcie
  - simulative PLTLc model checking ($SPN^c/CPN^c/HPN^c$): MC2 tool
Outline

- Multiscale Systems Biology
- Colored Petri nets
- Colored Petri net framework
- Key techniques
- Analysis techniques
- Applications
- Summary
Ex1- C. Elegans Vulval Development
Ex1- C. Elegans Vulval Development

based on [Li et al. 2009]
Ex1 - C. Elegans Vulval Development
Q Gao, D Gilbert, M Heiner, F Liu, D Maccagnola and D Tree: Multiscale Modelling and Analysis of Planar Cell Polarity in the Drosophila Wing; IEEE/ACM Transactions on Computational Biology and Bioinformatics, 10(2):337-351, 2013
Ex1- Planar Cell Polarity in Drosophila Wing
A cluster: a group of strongly coupled channels

An array of weakly coupled clusters

Ex3 - Ca2+ release sites

- Ignoring/considering the effect of neighboring clusters
- Few/frequent waves
Ex3 – Phase variation in a cell colony

Ex4 – the Brusselator

The PDEs of the Brusselator

\[
\frac{\partial U}{\partial \tau} = A - (B + 1)U + U^2V + \nabla^2 U
\]

\[
\frac{\partial V}{\partial \tau} = BU - U^2V + D\nabla^2 V
\]

\(B = (\mu + 1) \times (1 + \eta)^2\)

The chemical reactions and diffusion

\[
\phi \xrightarrow{A} U_{xy}
\]

\[
U_{xy} \xrightarrow{B} V_{xy}
\]

\[
2U_{xy} + V_{xy} \xrightarrow{1} 3U_{xy}
\]

\[
U_{xy} \xrightarrow{1/h^2} U_{ab}
\]

\[
V_{xy} \xrightarrow{D/h^2} V_{ab}
\]

\[
U_{xy} \xrightarrow{1} \phi
\]

F Liu, MA Blätke, M Heiner and M Yang:
Modelling and simulating reaction–diffusion systems using coloured Petri nets;
Computers in Biology and Medicine, 53:297–308, October 2014
Ex4 – the Brusselator

- The colored Petri net model of the Brusselator

<table>
<thead>
<tr>
<th>transition</th>
<th>rate function</th>
</tr>
</thead>
<tbody>
<tr>
<td>$t_{16}$</td>
<td>$A$</td>
</tr>
<tr>
<td>$t_{17}$</td>
<td>$B \cdot U$</td>
</tr>
<tr>
<td>$t_{18}$</td>
<td>$U \cdot U \cdot V$</td>
</tr>
<tr>
<td>$t_{19}$</td>
<td>$U$</td>
</tr>
<tr>
<td>$t_{20}$</td>
<td>$U/(h \cdot h)$</td>
</tr>
<tr>
<td>$t_{21}$</td>
<td>$D \cdot V/(h \cdot h)$</td>
</tr>
</tbody>
</table>
Ex4 – the Brusselator

The colored Petri net model of the Brusselator
Outline

- Multiscale Systems Biology
- Colored Petri nets
- Colored Petri net framework
- Key techniques
- Analysis techniques
- Applications
- Summary
Multiscale challenges vs colored Petri nets

- Repetition
- Variation
- Organization
- Hierarchical organization
- Communication
- Movement
- Replication
- Deletion
- Differentiation
- Dynamic (variable) grid size
- Pattern formation

- Colors
- Choose a group of colors
- Ordered color sets
- Ordered product color sets
- Exchange colors
- Change a color to another
- Create a new color in a color set
- Remove a color from a color set
- Change a color to another
- Dynamic color sets
- Combination
Acknowledgements

- Federal Ministry of Education and Research, Germany (0315449H)
- National Natural Science Foundation of China (61273226)

- Data Structures and Software Dependability Chair, BTU Cottbus, Germany
- Molecular Network Group, Otto-von-Guericke-Universität, Germany
- Computational Systems and Synthetic Biology Research Group, Brunel University, UK
- Department of Computer Science, Port Said University, Egypt
Thank you for your attention!
References

• C. Li et al. Simulation-Based Model Checking Approach to Cell Fate Specification During Caenorhabditis Elegans Vulval Development by Hybrid Functional Petri Net with Extension. BMC Systems Biology.
• F Liu and M Heiner: Multiscale modelling of coupled Ca2+ channels using coloured stochastic Petri nets; IET Systems Biology, 2013.