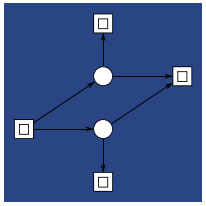


Brandenburg
University of Technology
Cottbus

Colored Petri Nets for Modeling and Analyzing Biological Systems

Fei Liu

Department of Computer Science
Brandenburg University of Technology Cottbus

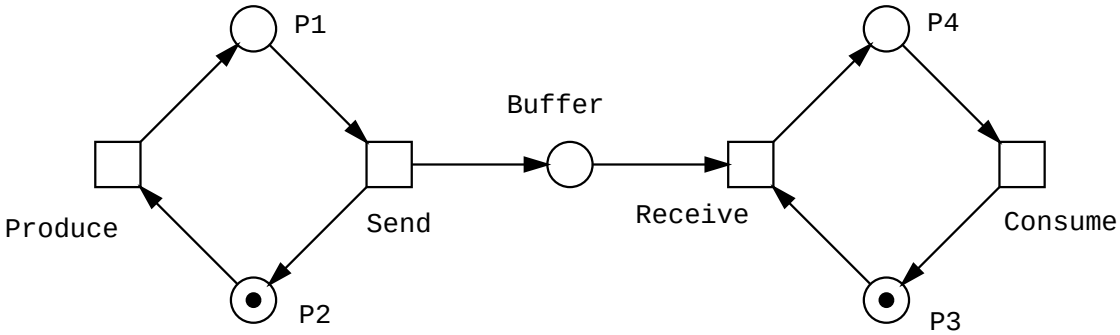
Dagstuhl Seminar
April 10 2011

Outline

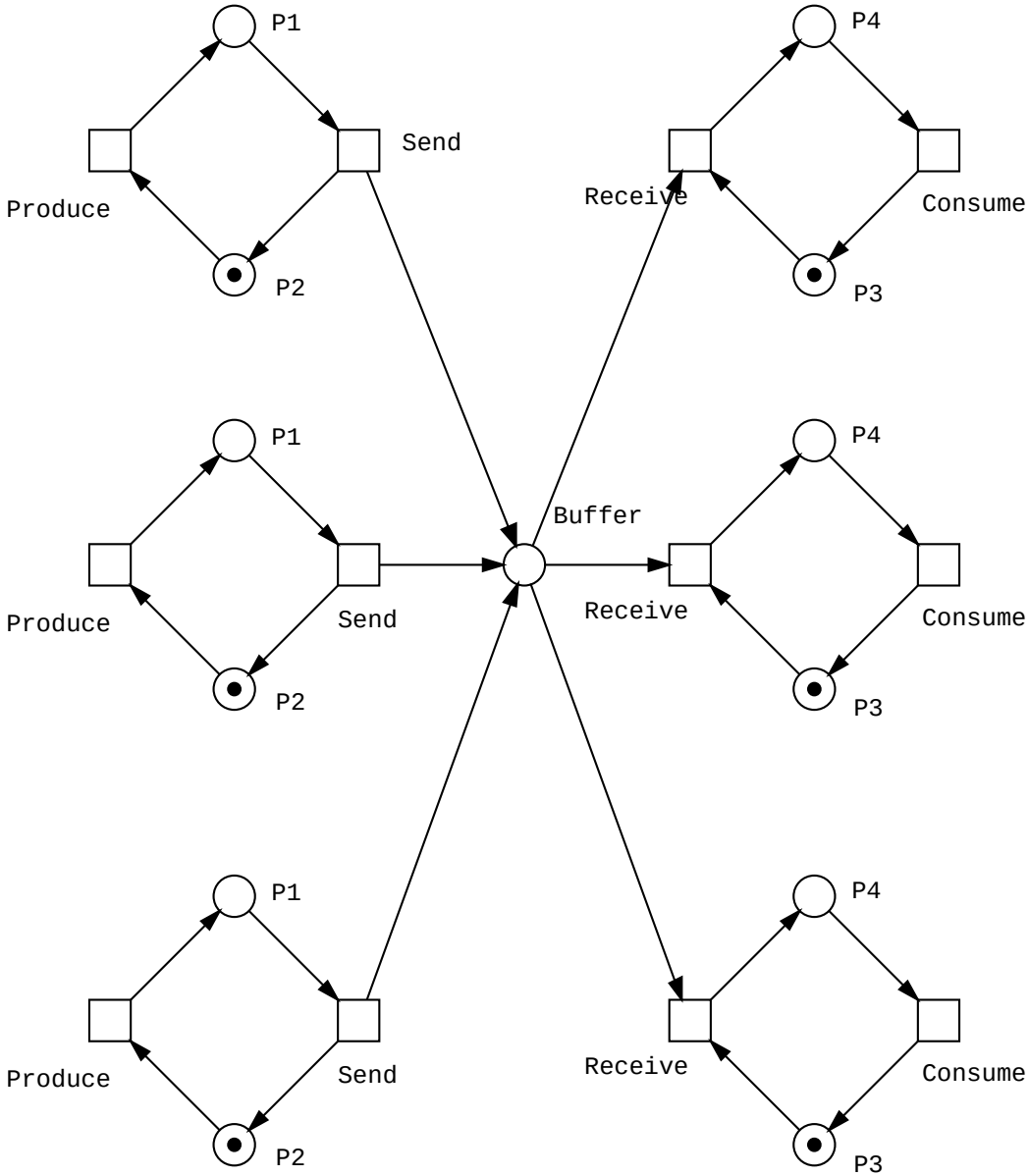
- Motivation
- Snoopy

Motivation

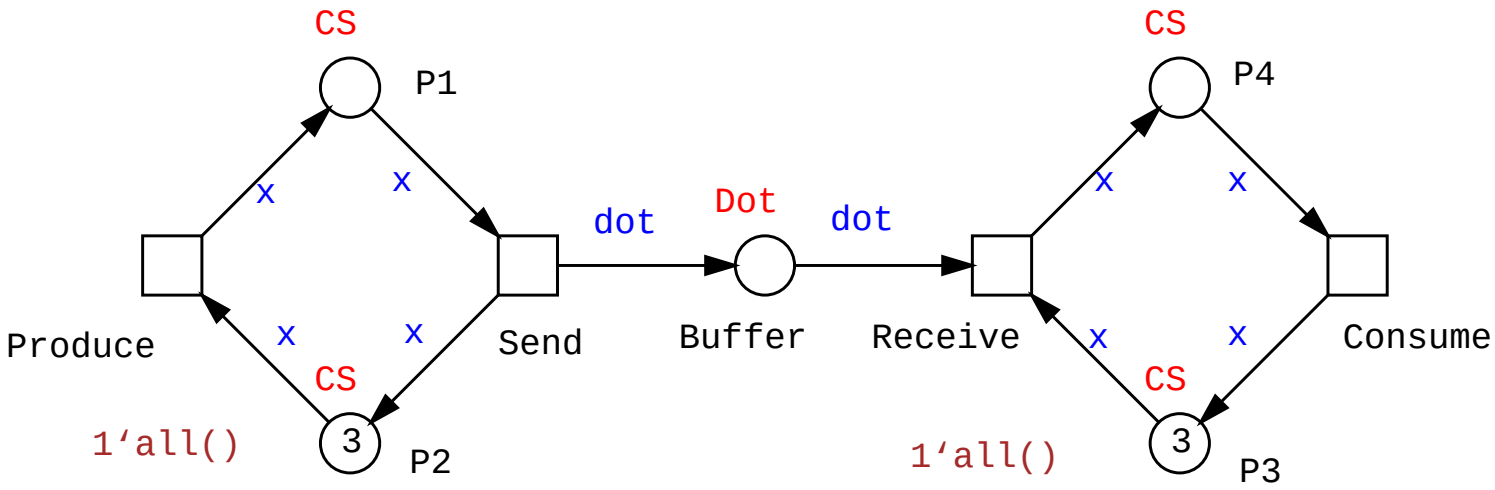
Producer-Consumer



Producer-Consumer



Producer-Consumer



Why use colored Petri nets

Why use colored Petri nets

- Compact and readable representation,

Why use colored Petri nets

- Compact and readable representation,
- Scalable models,
- Increasing net size = increasing color sets,

Why use colored Petri nets

- Compact and readable representation,
- Scalable models,
- Increasing net size = increasing color sets,
- Analysis techniques of low-level Petri nets by automatic unfolding,
  - Animation/Simulation,
  - Structural analysis,
  - State space analysis.

Why use colored Petri nets

- Compact and readable representation,
- Scalable models,
- Increasing net size = increasing color sets,
- Analysis techniques of low-level Petri nets by automatic unfolding,
  - Animation/Simulation,
  - Structural analysis,
  - State space analysis.
- Analysis techniques of high-level Petri nets.
  - Symbolic simulation,
  - State space analysis.

Why use colored Petri nets

- Compact and readable representation,
- Scalable models,
- Increasing net size = increasing color sets,
- Analysis techniques of low-level Petri nets by automatic unfolding,
  - Animation/Simulation,
  - Structural analysis,
  - State space analysis.
- Analysis techniques of high-level Petri nets.
  - Symbolic simulation,
  - State space analysis.

Scenarios

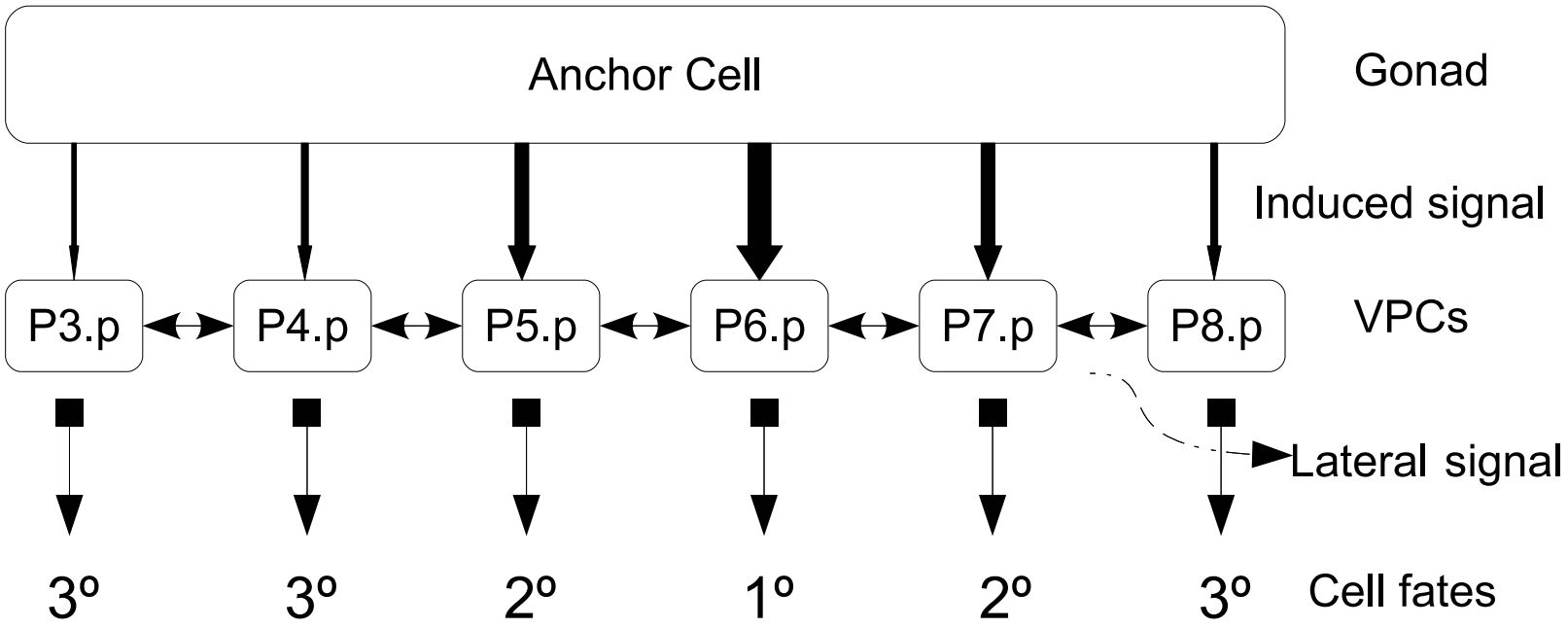
Scenarios

- biological systems with similar objects
  - cells
  - genes
  - receptors
  - transducers

Scenarios

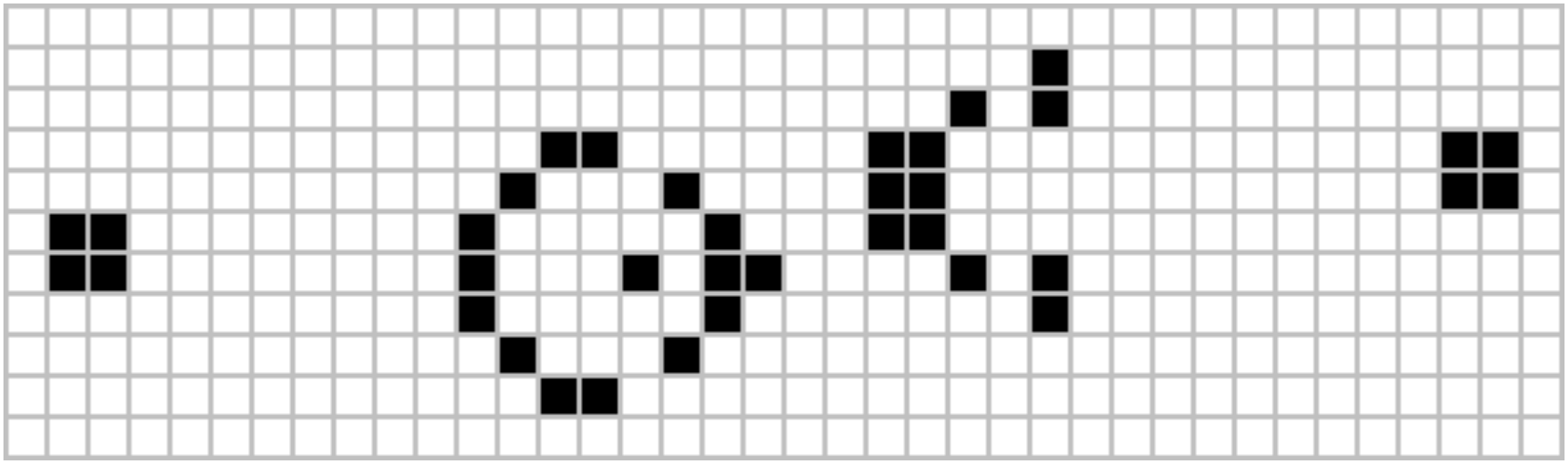
- biological systems with similar objects
  - cells
  - genes
  - receptors
  - transducers
- biological systems with spatial aspects
  - grid-based agent systems
  - membrane systems

Example: C. elegans



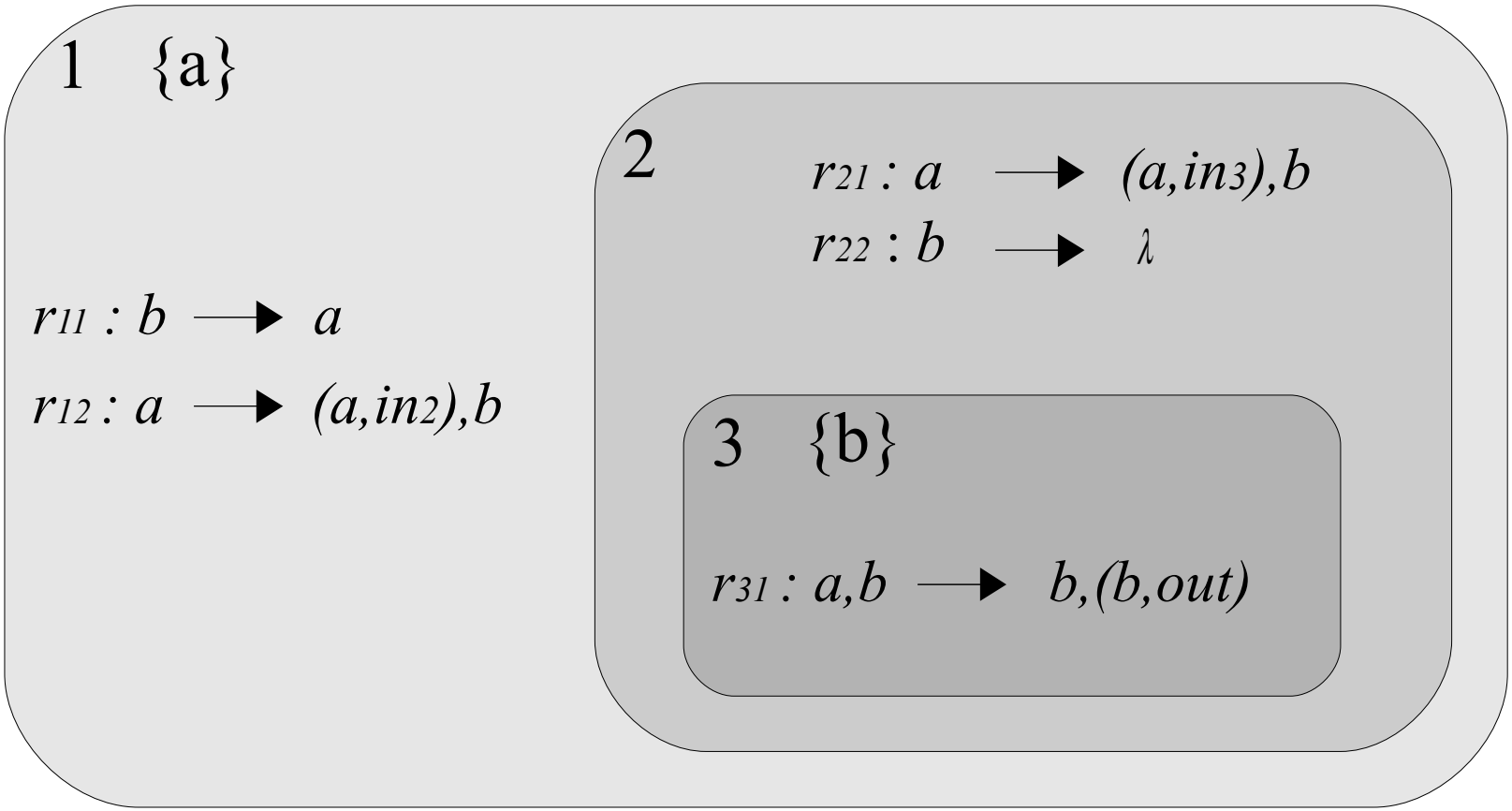
- Colset CS = integer with 3-8;

Example: agent-based models with a grid



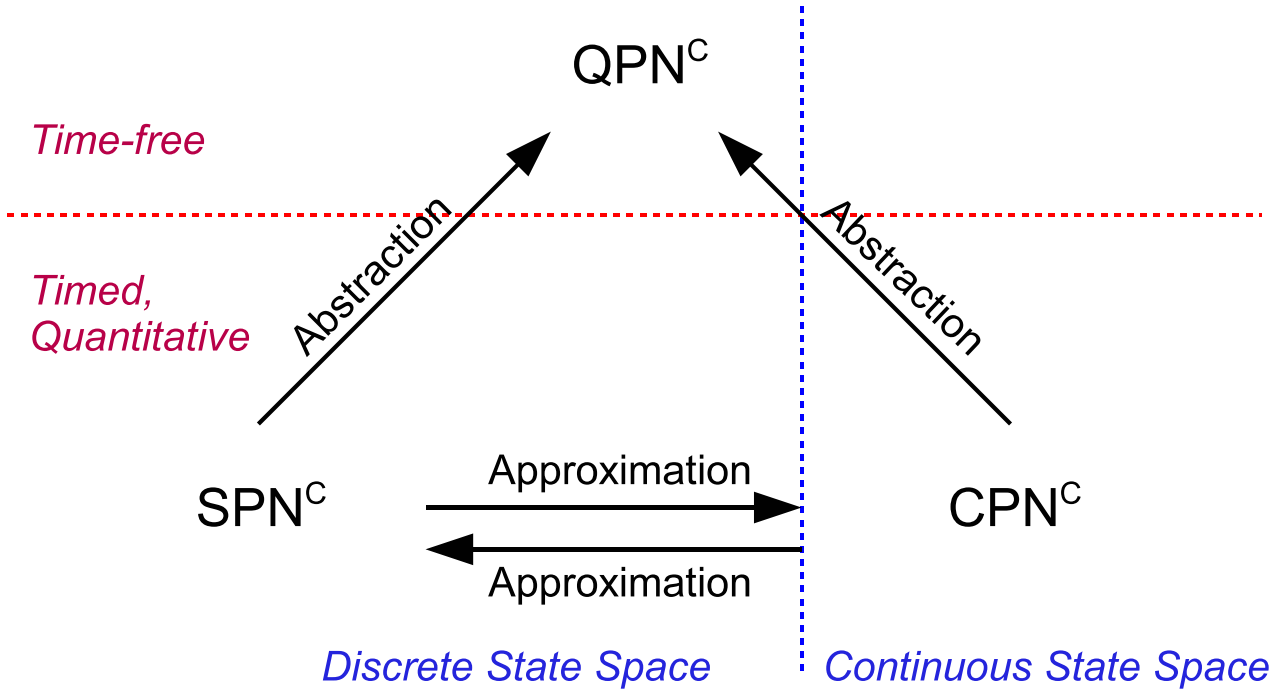
Ref: http://en.wikipedia.org/wiki/File:Game_of_life_glider_gun.svg

Example: membrane systems



Snoopy

Colored Petri nets-based framework



Features for modeling

- Drawing of the Petri net graph as usual.
- Rich data types for color set definition: dot, int, string, bool, enum, index, product, union.
- User-defined functions.

Features for modeling

- Drawing of the Petri net graph as usual.
- Rich data types for color set definition: dot, int, string, bool, enum, index, product, union.
- User-defined functions.
- Several extended arc types, such as inhibitor arc, read arc, equal arc, reset arc, and modifier arc.
- Several special transitions: stochastic transitions with freestyle rate functions, immediate transitions, deterministic transitions, and scheduled transitions.

Features for modeling

- Drawing of the Petri net graph as usual.
- Rich data types for color set definition: dot, int, string, bool, enum, index, product, union.
- User-defined functions.
- Several extended arc types, such as inhibitor arc, read arc, equal arc, reset arc, and modifier arc.
- Several special transitions: stochastic transitions with freestyle rate functions, immediate transitions, deterministic transitions, and scheduled transitions.
- Concise specification of initial marking for larger color sets.
- Rate function definition for each transition instance.
- Highlighting the markings, color sets, guards, and expressions.

Features for animation (for QPN^C/SPN^C)

- Automatic animation,
- Single-step animation by manually choosing a binding.

Features for simulation (for SPN^C / CPN^C)

- Simulation is done on an automatically unfolded Petri net.
- Show or export simulation results for colored or uncolored places/transitions separately or together.

Features for simulation (for SPN^C/CPN^C)

- Simulation is done on an automatically unfolded Petri net.
- Show or export simulation results for colored or uncolored places/transitions separately or together.
- Several simulation algorithms to simulate SPN^C, including the Gielespie stochastic simulation algorithm (SSA) [Gil77].
- Several simulation algorithms to simulate CPN^C, including the Euler algorithm, Runge-Kutta algorithm etc.

Ref: [Gil77] Gillespie D. T.: Exact stochastic simulation of coupled chemical reactions. Journal of Physical Chemistry 1977, 81(25):2340-2361.

Features for export

- QPN^C, SPN^C and CPN^C are exported to different net formalisms within Snoopy,
- Export/import beyond Snoopy, e.g., export to CPN tools [CPN].

Ref: [CPN] http://cpntools.org

Thank You for Your Attention!