# Stochastic modelling and evaluation using GreatSPN

Elvio G. Amparore
Università di Torino

Tools for Stochastic Modelling and Evaluation (TOSME)
12 Nov 2021

UNIVERSITÀ
DEGLI STUDI
DI TORINO

# GreatSPN v3.0: an overview

The **G**raphical **E**ditor and **A**nalyzer for **T**imed and **S**tochastic **P**etri **N**ets (**GreatSPN**) is a framework for modeling systems as:

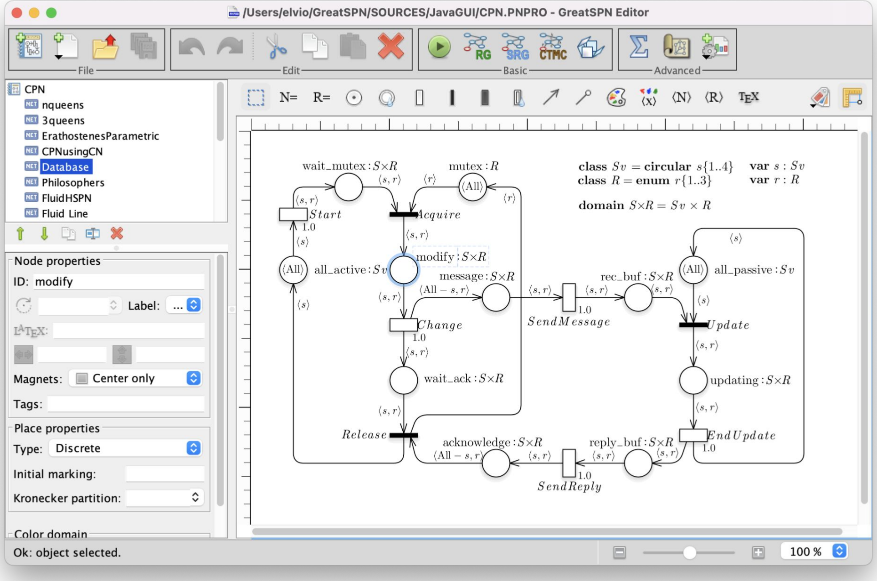
- **G**eneralized **S**tochastic **P**etri **N**ets
- **S**tochastic **W**ell-formed **N**ets
- **D**eterministic and **S**tochastic **P**etri **N**ets

Collection of separate tools.

Tools have a command line interface, and are accessible through a GUI.

Designed around a *modeling workflow*.

Open source.



/Users/elvio/GreatSPN/SOURCES/JavaGUI/CPN.PNPRO - GreatSPN Editor

CPN
- nqueens
- 3queens
- ErathostenesParametric
- CPNusingCN
- Database
- Philosophers
- FluidHSPN
- Fluid Line

Node properties

ID: modify

Label: ...

LaTeX:

Magnets: Center only

Tags:

Place properties

Type: Discrete

Initial marking:

Kronecker partition:

Color domain

Ok: object selected.

100 %

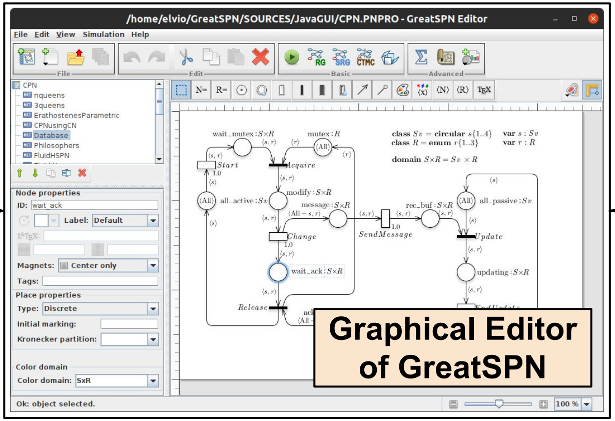
# Tools of GreatSPN

The GreatSPN framework tool collection covers these topics:

- **GUI** for drawing and composing models
- **Translating** various formats
- **Model composition**: model algebra, unfolding
- Analysis of **structural** model properties
- **Markov decision** process support (MDPN)
- **Model checking** using temporal logic specifications
- **Stochastic solutions** using:
    - Reachability Graph (explicit, symbolic or MDD)
    - Simulation
    - Differential equations (ODE/SDE)

# Tools of GreatSPN - overview

PNML model



Node properties

ID: wait_ack

Label: Default

Magnets: Center only

Tags:

Place properties

Type: Discrete

Initial marking:

Kronecker partition:

Color domain

Color domain: SxR

Ok: object selected.

**Graphical Editor of GreatSPN**

model as GSPN SWN MRGP

specification

**gsol**
PNML rev.2009 to net/def translation

**unfolding2**
SWN to P/T net translation

**ArgoSPE**
Eclipse UML to Petri net translation

**algebra**
Composition by P/T superposition

*Model translation / composition*

**struct / bounds**
Place bounds, conflict sets

**ilp-bounds**
Place bounds using integer LP

**invariants2**
P/T flows, semiflows, basis, traps, siphons

*Structural Analysis*

**MDWN**
MDP from Markov Decision Petri nets

**mdwnsolve**
MDP solution

**mdwn2prism**
MDWN to Prism conversion

*Decision Processes*

**MC4CSLTA**
Stochastic model checker for CSLTA

**STARMC**
Symbolic CTL* model checker

*Model checking*

**(GSPN|WN|WNS)RG**
Reachability graph generation

**(GSPN|WN|WNS)SIM**
Monte Carlo simulation

**sc / trs**
Stationary/Transient solution computation

**DSPN-Tool**
Steady-state MRGP solution

**PN2ODE**
Ordinary/Stochastic Differential Equations solver

*Stochastic solution*

# Input formats

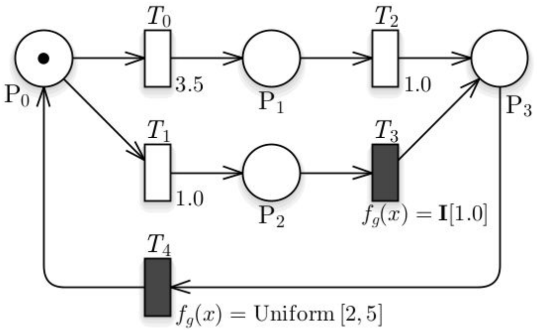
- Imports and exports **PNML**[1] models.
  - Supports both P/T and Symmetric Nets
  - Unfortunately PNML does not encode stochastic info

- Core format: **net/def** and **PNPRO** (Petri Net Project)
- Other model formats: **APNN, GrML, NetLogo, UML, …** and **PDF/PNG**

[1] E. Kindler. *The Petri Net Markup Language and ISO/IEC 15909-2: Concepts, status, and future directions*. Entwurf komplexer Automat., 2006.

# The modelling workflow

Designed to simplify the learning curve of the framework.



## 1. Draw model

using one of the supported extensions (GSPN, SWN, DSPN) + model composition

## 2. Verify structural integrity

P/T invariants and (semi)flows,
siphons, traps,
token game,
deadlock analysis,
CTL/LTL/CTL*

## 3. Specify & compute performance indices

run solver with a set of target measures

## 4. Export data or draw plot

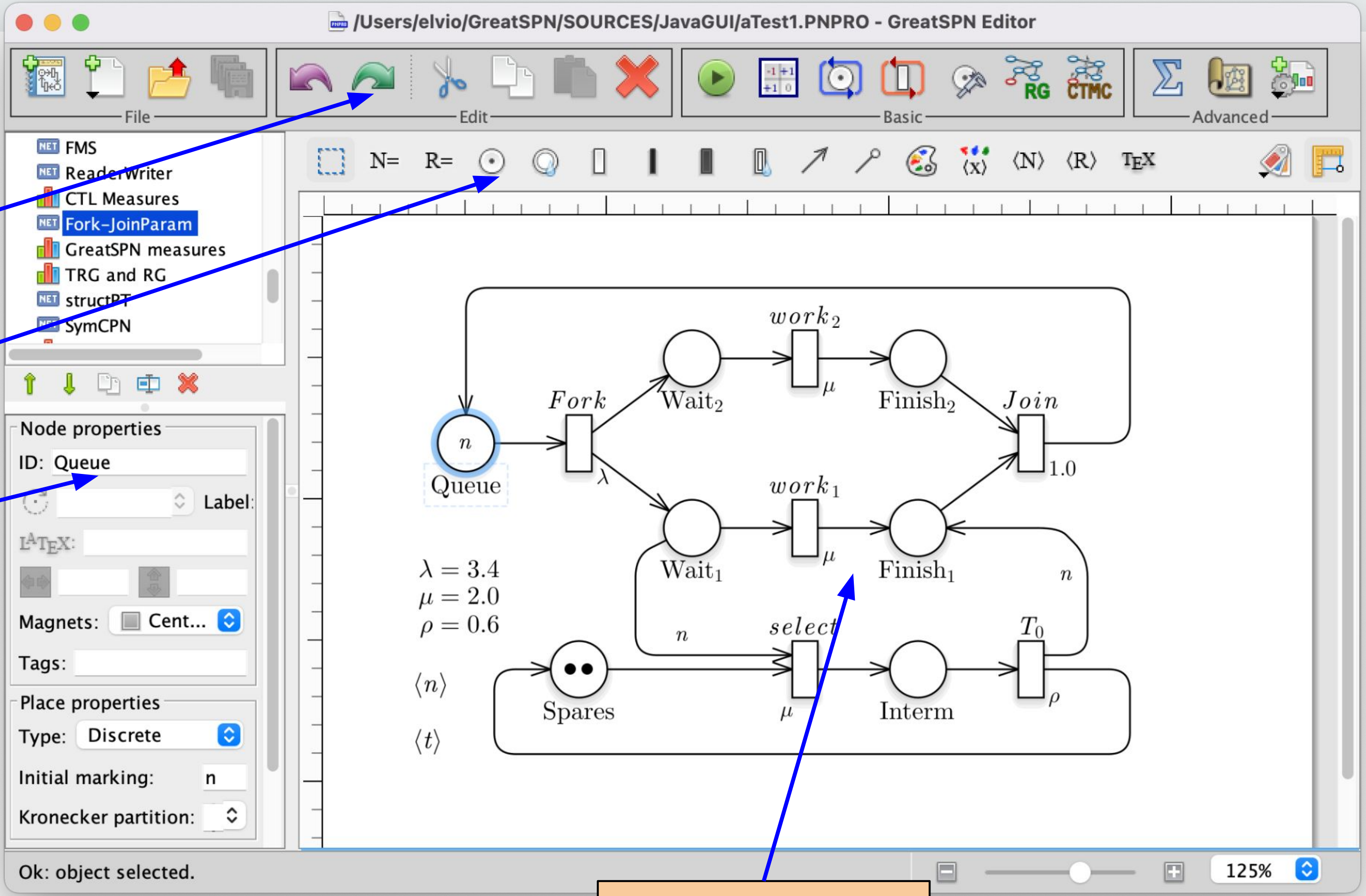
CSV, Excel,
PDF, PNG, ...

# Drawing models

functions organized in groups

net drawing tools

selected object's properties



ID: Queue

Magnets: Cent...

Tags:

Type: Discrete

Initial marking: n

Kronecker partition:

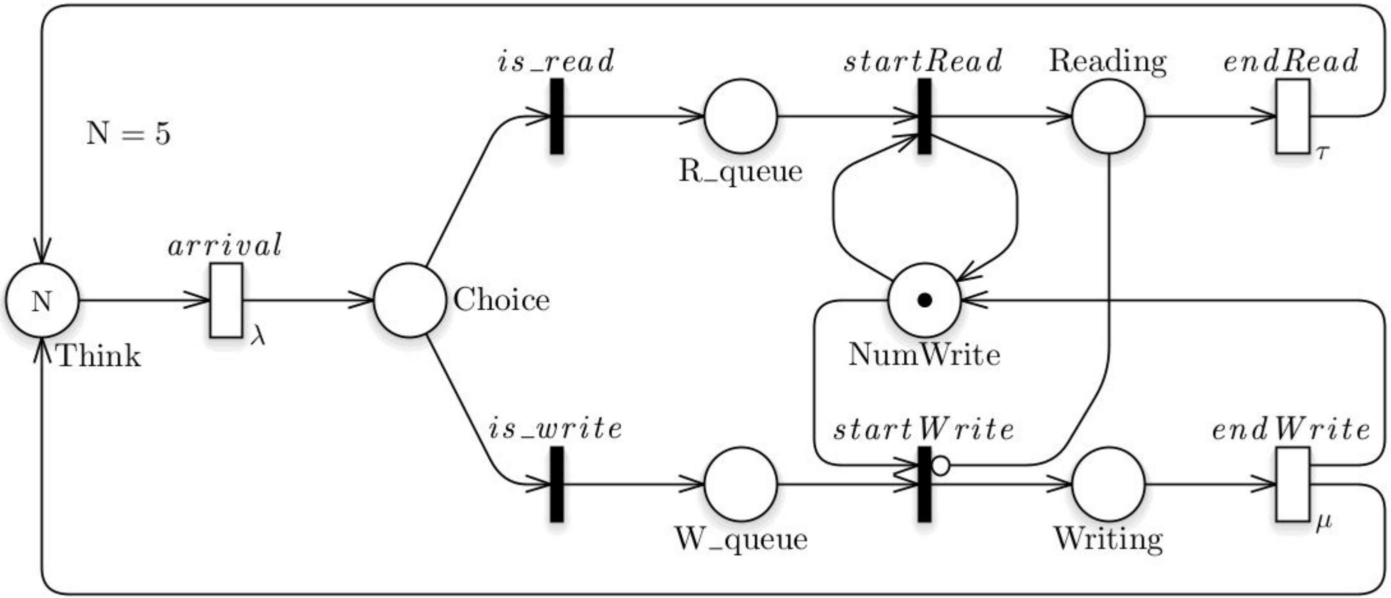
Ok: object selected.

125%

model being drawn

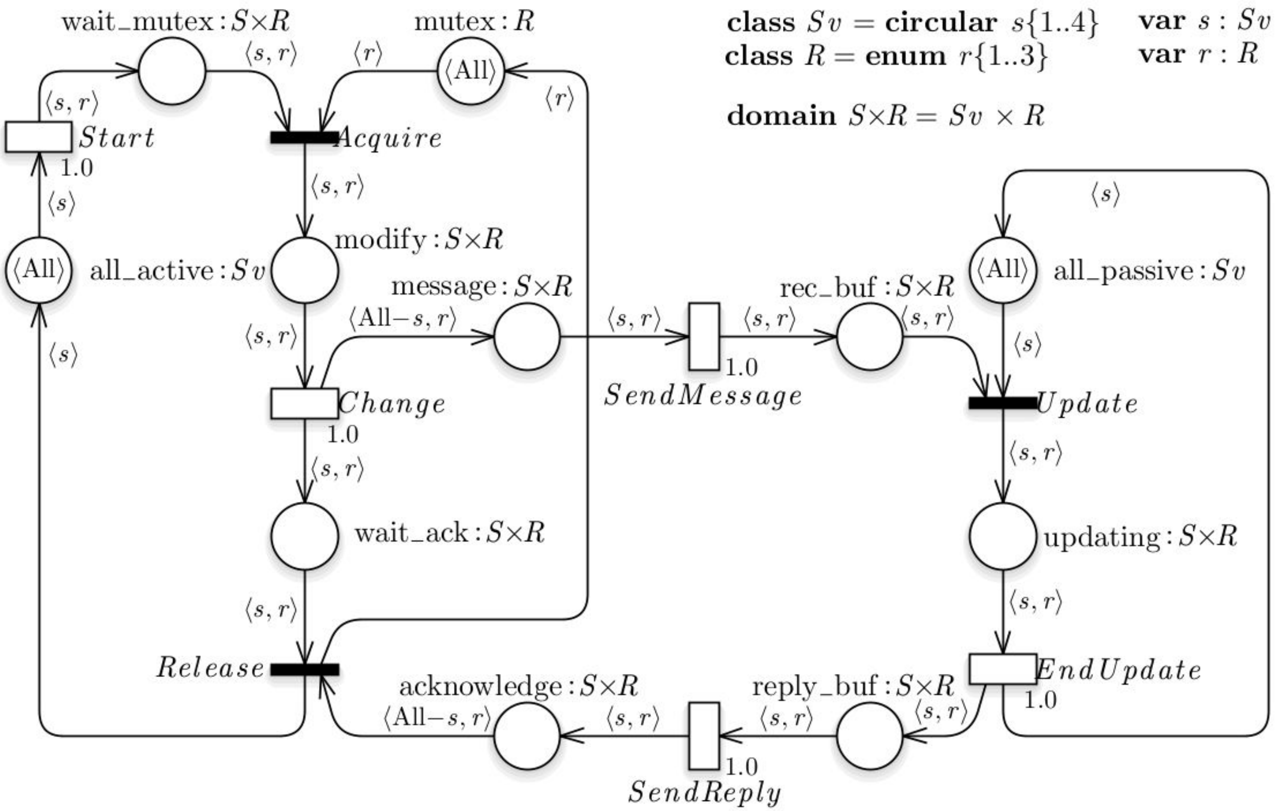
# GSPN Example

## Reader-Writer synchronization model



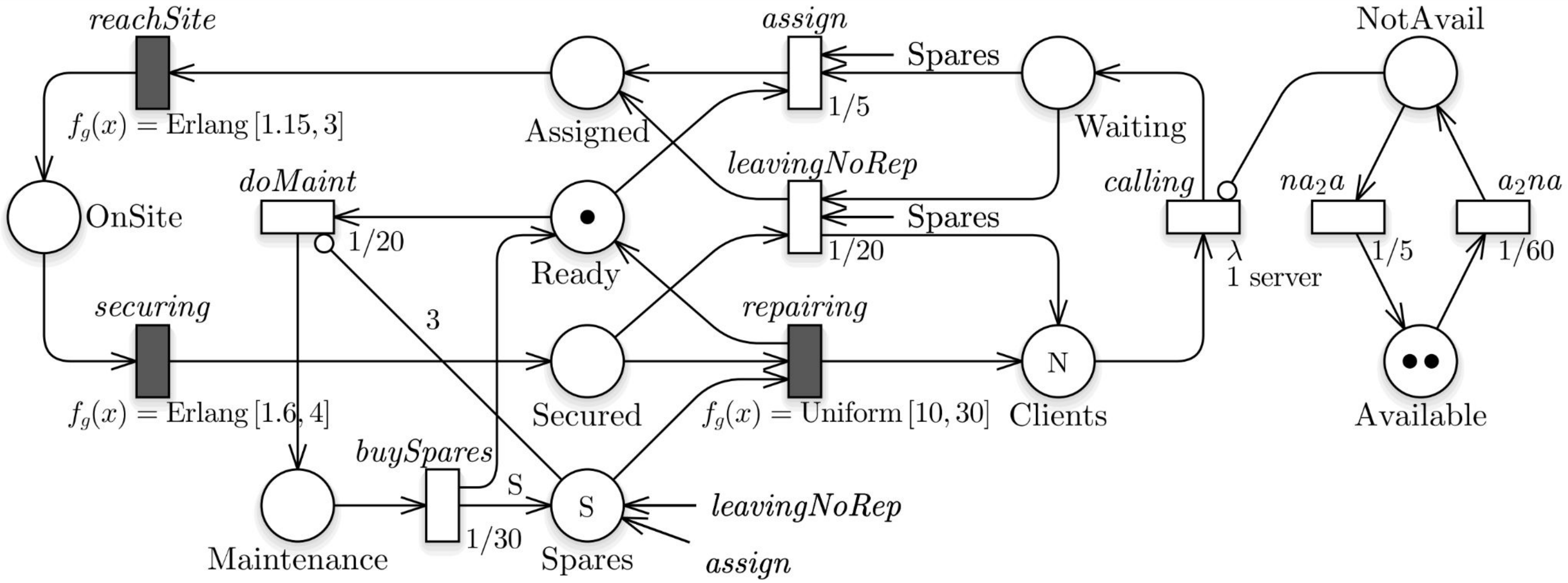
# SWN Example

Database server model



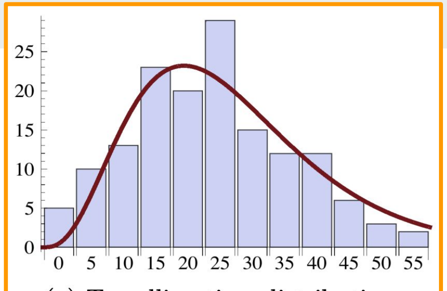
# DSPN Example

## Model of a multi-utility company

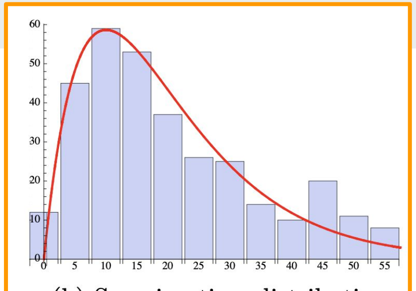


# DSPN Example

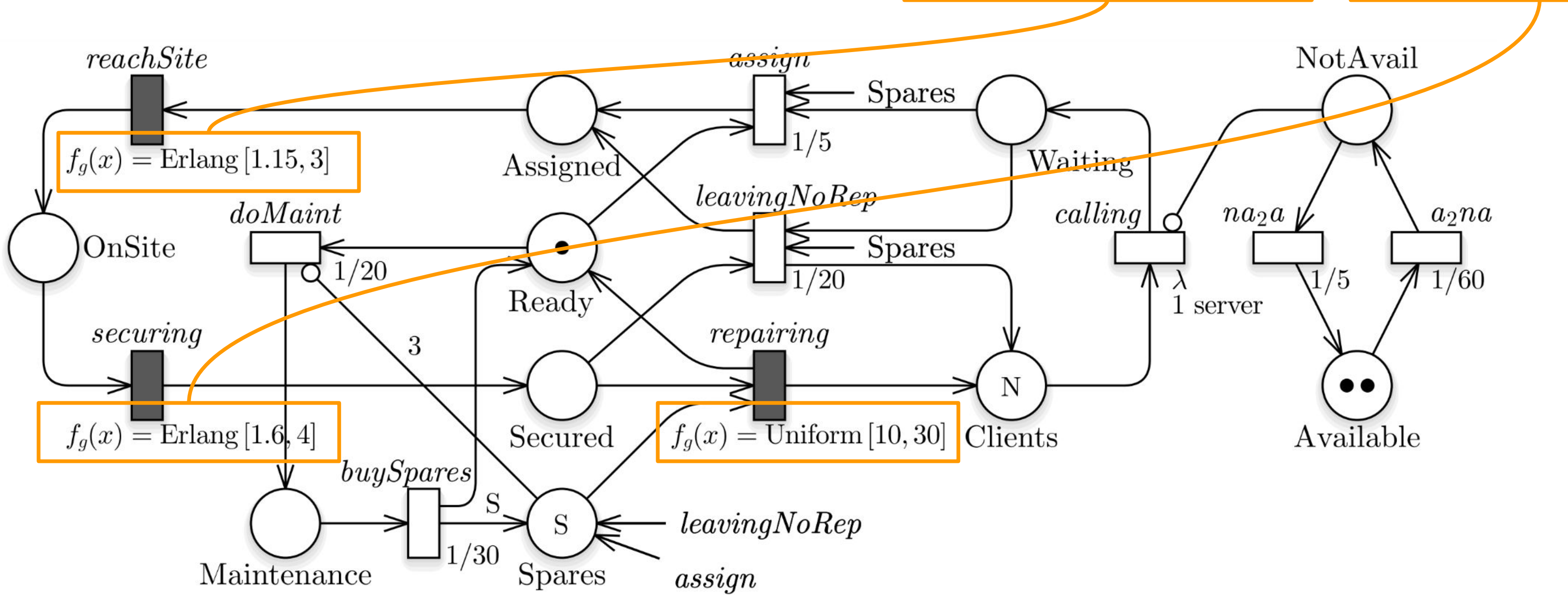
Model of a multi-utility company



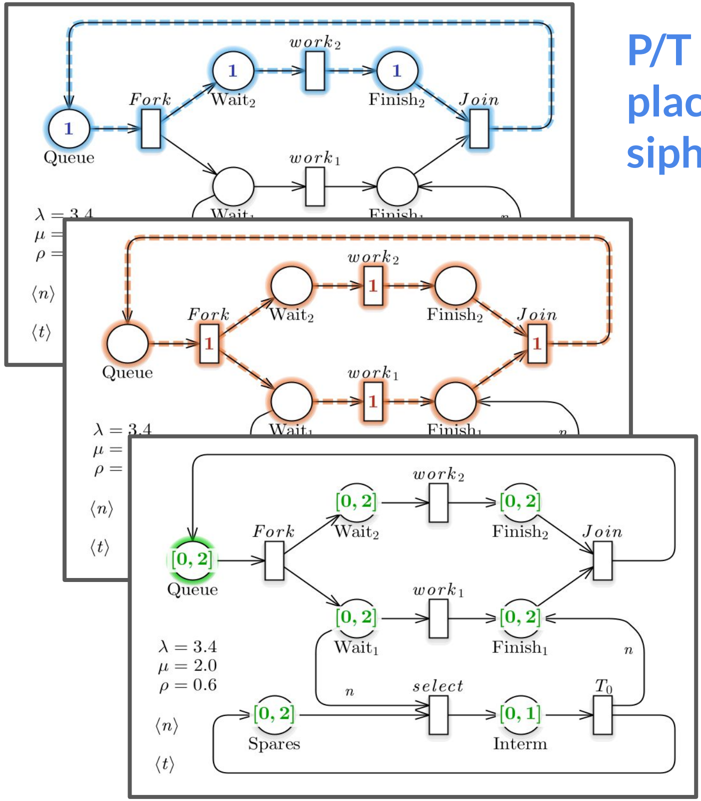
(a) Travelling time distribution.



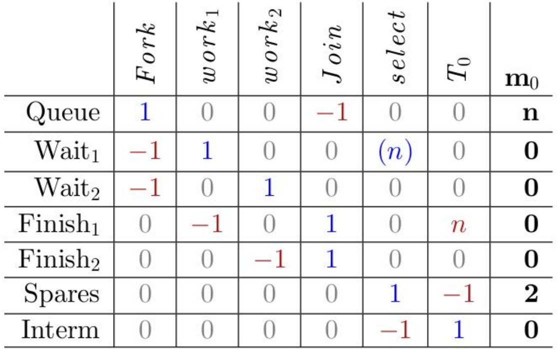
(b) Securing time distribution



# Verifying structural properties

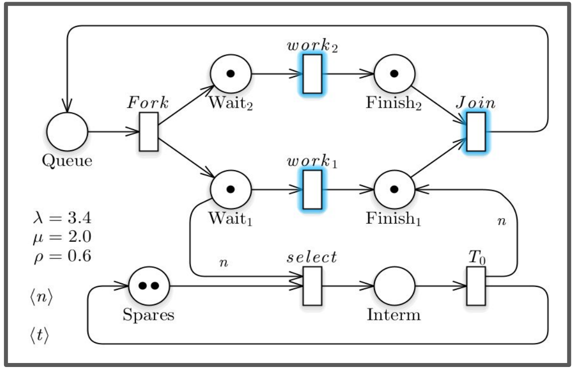


P/T (semi)flows, place bounds, siphons, traps, basis



| | $Fork$ | $work_1$ | $work_2$ | $Join$ | $select$ | $T_0$ | $\mathbf{m}_0$ |
|---|---|---|---|---|---|---|---|
| Queue | 1 | 0 | 0 | −1 | 0 | 0 | **n** |
| Wait$_1$ | −1 | 1 | 0 | 0 | $(n)$ | 0 | **0** |
| Wait$_2$ | −1 | 0 | 1 | 0 | 0 | 0 | **0** |
| Finish$_1$ | 0 | −1 | 0 | 1 | 0 | $n$ | **0** |
| Finish$_2$ | 0 | 0 | −1 | 1 | 0 | 0 | **0** |
| Spares | 0 | 0 | 0 | 0 | 1 | −1 | **2** |
| Interm | 0 | 0 | 0 | 0 | −1 | 1 | **0** |

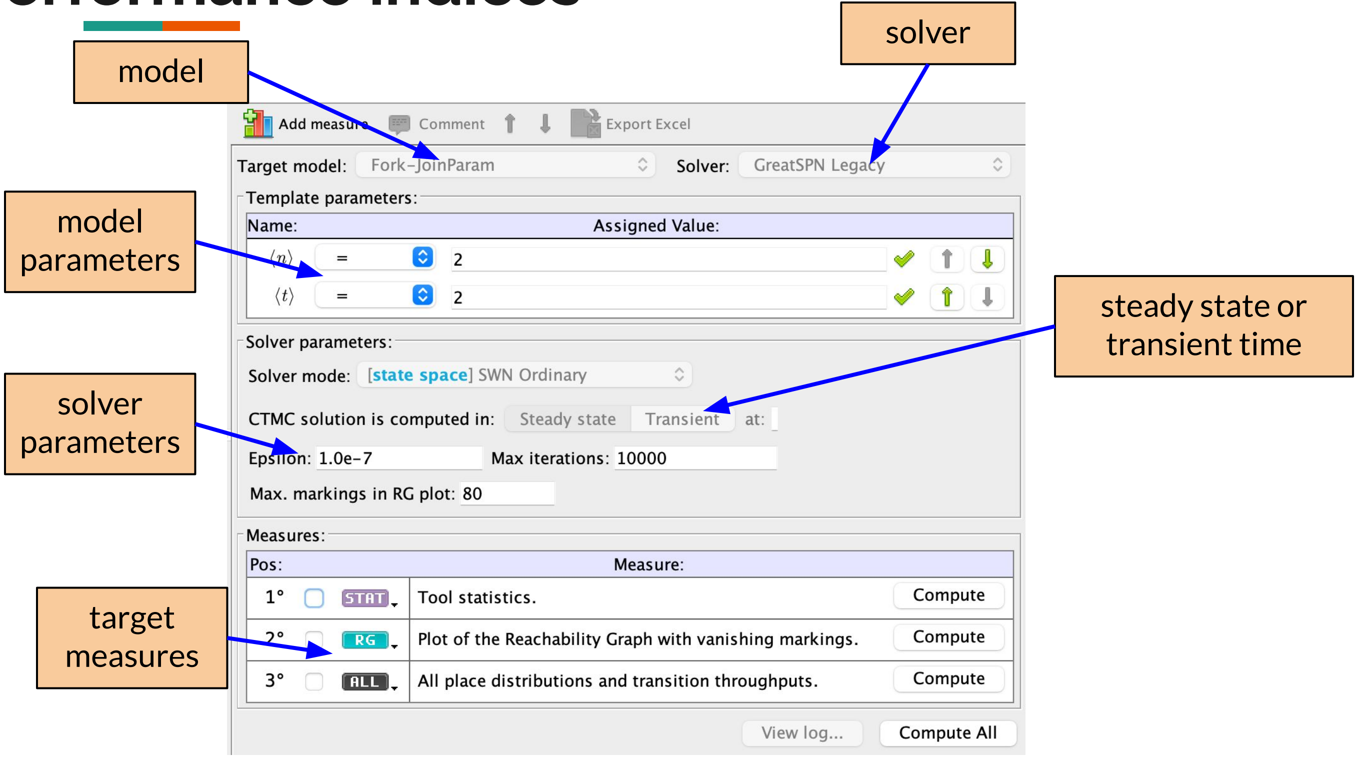
incidence matrix



interactive token game

Model checking: deadlock, reachability, CTL, LTL, fair CTL, CTL*

# Specify & compute performance indices



model

model parameters

solver parameters

target measures

solver

steady state or transient time

Target model: Fork-JoinParam

Solver: GreatSPN Legacy

Template parameters:

| Name: | | Assigned Value: |
|---|---|---|
| $\langle n \rangle$ | = | 2 |
| $\langle t \rangle$ | = | 2 |

Solver parameters:

Solver mode: [state space] SWN Ordinary

CTMC solution is computed in: Steady state | Transient at:

Epsilon: 1.0e-7 Max iterations: 10000

Max. markings in RG plot: 80

Measures:

| Pos: | | | Measure: | |
|---|---|---|---|---|
| 1° | □ | STAT | Tool statistics. | Compute |
| 2° | □ | RG | Plot of the Reachability Graph with vanishing markings. | Compute |
| 3° | □ | ALL | All place distributions and transition throughputs. | Compute |

# Core Solvers

Solvers that generate the RG:

- GSPNRG: basic RG of GSPN models.
- WNRG: RG of SWN models, i.e. supporting colored transition firings.
- WNSRG: Symbolic RG of SWN models, exploiting model symmetries.
- DSPN-Tool: steady state solution of DSPN models..
- STARMC: RG encoded using Decision Diagrams. Only limited stochastic solutions (steady state with Jacobi method) are possible, but very large state spaces ($10^{100}$ and beyond) can be encoded.

Non-RG solvers:

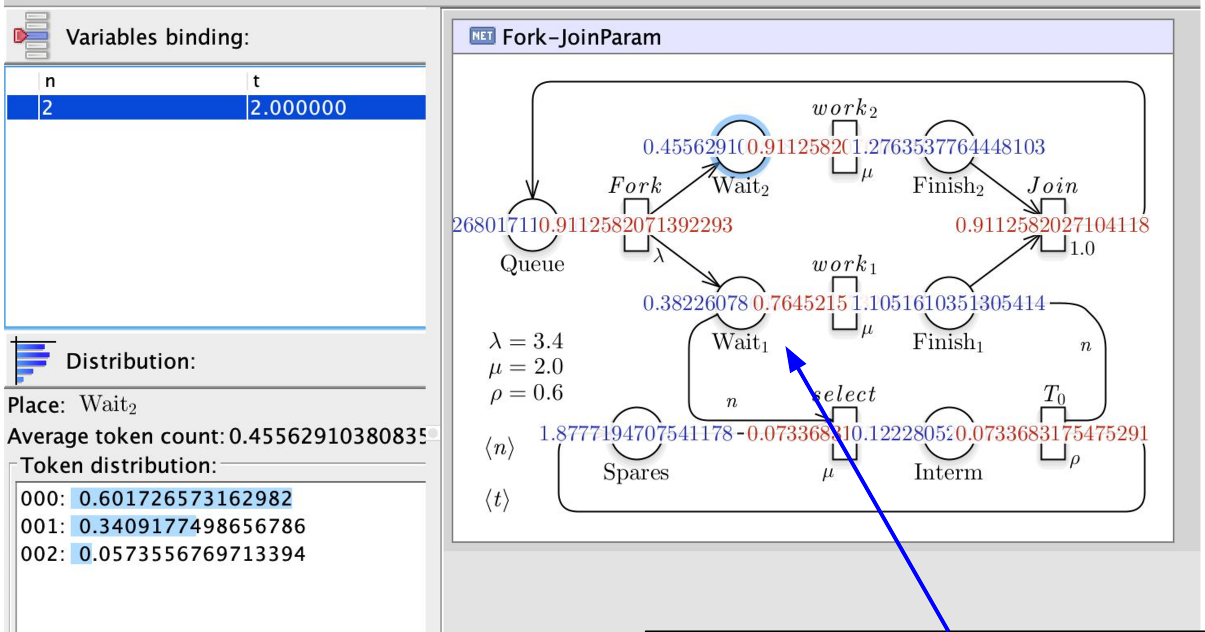
- WNSIM: Monte Carlo simulation using batch simulation.
- WNSSYM: simulation using symbolic markings.
- PN2ODE: solution using ODE/SDE.

External solvers:

- Cosmos (cosmos.lacl.fr/) - statistical model checker
- NSolve (ls4-www.cs.tu-dortmund.de/APNN-TOOLBOX/) - Kronecker representation (GSPN/DSPN)
- Storm (www.stormchecker.org/) - model checker

# Visualize computed results

Rechability Graph



Variables binding:

| n | t |
|---|---|
| 2 | 2.000000 |

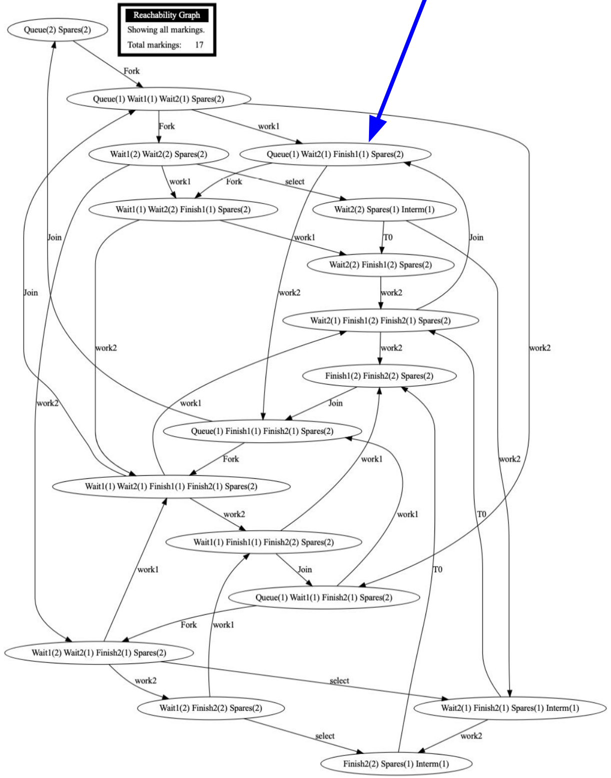
Distribution:

Place: $\mathrm{Wait}_2$

Average token count: 0.45562910380835

Token distribution:

```
000: 0.601726573162982
001: 0.340917749865678
002: 0.0573556769713394
```



Reachability Graph
Showing all markings.
Total markings: 17
Queue(2) Spares(2)
Fork
Queue(1) Wait1(1) Wait2(1) Spares(2)
Fork
work1
Wait1(2) Wait2(2) Spares(2)
Queue(1) Wait2(1) Finish1(1) Spares(2)
work1
Fork
select
Wait1(1) Wait2(2) Finish1(1) Spares(2)
Wait2(2) Spares(1) Interm(1)
Join
work1
T0
Join
Wait2(2) Finish1(2) Spares(2)
Join
work2
work2
Wait2(1) Finish1(2) Finish2(1) Spares(2)
work2
work2
work2
Finish1(2) Finish2(2) Spares(2)
work2
work1
Join
Queue(1) Finish1(1) Finish2(1) Spares(2)
Fork
work1
work2
Wait1(1) Wait2(1) Finish1(1) Finish2(1) Spares(2)
work2
work1
T0
Wait1(1) Finish1(1) Finish2(2) Spares(2)
work1
Join
T0
Queue(1) Wait1(1) Finish2(1) Spares(2)
Fork
work1
Wait1(2) Wait2(1) Finish2(1) Spares(2)
work2
select
Wait1(2) Finish2(2) Spares(2)
Wait2(1) Finish2(1) Spares(1) Interm(1)
select
work2
Finish2(2) Spares(1) Interm(1)

results computed by the selected solver shown on the net objects

# Recent GreatSPN features

While GreatSPN started its development in 1986, 35 years ago, it is still actively developed.
Some features added in the last few years:

- The model checker **STARMC** - Decision Diagrams to verify properties expressed in CTL/LTL/fair CTL /CTL*.
- The new Java **GUI** with the streamlined workflow.
- The **DSPN** solver.
- Support for the **PNML** format.
- **GreatMod**: dedicated platform for System Biology models (https://qbioturin.github.io/epimod/).

# Availability

- GreatSPN is **open source** (GPLv2)
- Runs on all major platforms (Linux, Windows, macOS)
- **Sources**: https://github.com/greatspn/SOURCES
- **Pre-installed VM**: http://www.di.unito.it/~greatspn/VBox/