

WSIZ, 2023

Evaluation of System Performance by Colored Petri Nets

Dmitry Zaitsev

<http://daze.ho.ua>

Modeling by Petri and Sleptsov nets

- Verification of protocols by Petri nets
- Model analysis methods. Composition of clans
- Analysis of Computational Grids and Clouds by Infinite Petri Nets
- **Evaluation of System Performance by Colored Petri Nets**
- Computing on Sleptsov networks

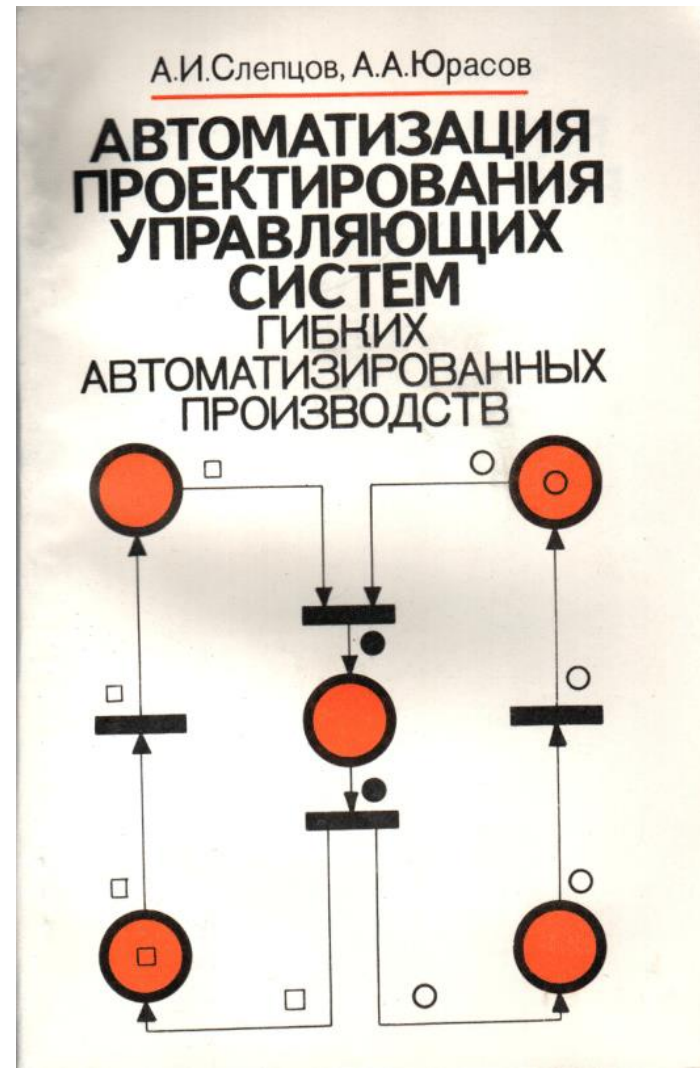
Simulation

- **Computing process resembling real-life process in essential features**
- **Model-driven development**
- **Express-evaluation of performance and QoS**
- **Technology of fast modification of model**
- **Technique for evaluation of performance directly in the process of simulation**

Modeling by Sleptsov nets

- **Loaded net: token – an object of abstract data type; functions in an algorithmic language are associated with elements of net**
- **Timed net with multichannel transitions – multiple firing a transition in a few instances at a step**

Sleptsov A.I., Yurasov A.A.
AUTOMATION OF DESIGN OF
CONTROL SYSTEMS OF FLEXIBLE
AUTOMATED PRODUCTION /
Edited by B.N. Malinovsky. - K.:
Tekhnika, 1986. -160s. in English





Login/Create Account

Language: English



US



China



Search title, author, ISBN...

All Products



Books

Journals

e-Collections

Articles/Chapters

Publish with Us

Resources

Catalogs

About Us

Newsroom

Special Offers

newsroom

Email Updates

RSS



Search

Dmitry Zaitsev provides the history and the latest research on Sleptsov net computing, which is showcased in IGI Global's upcoming Encyclopedia release

The Highest Standard: Sleptsov Software

By Elizabeth Leber on May 23, 2017

As Oprah Winfrey said "Real integrity is doing the right thing, knowing that nobody's going to know whether you did it or not." When it comes to Sleptsov nets, integrity is of the utmost importance, and those who have been conducting the research are determined to fight to make it known. Sleptsov nets are a new graphical language of parallel programming which when implemented in hardware will yield hyperperformance. You simply draw programs instead of writing them. Circles, rectangles and arcs connect everything, and only numbers can be written inside the circles.

"Our nets bring integrity to programming on place-transition networks. No other concepts, no other facilities. Only nets, which represent a program, run on nets, which represent a processor. They run fast and massively parallel, so let us implement them!" states Dr. Anatolii Illich Sleptsov.

Dr. Anatolii Illich Sleptsov received his Ph.D. in automation & telecommunications from Donetsk Polytechnic Institute (Ukraine) in 1974 and a doctor of sciences degree in information technologies in 1989 at the Institute of Cybernetics of National Academy of Sciences of Ukraine. Prof. Sleptsov taught undergraduate and graduate information system courses for computer and software engineering students in Donetsk Polytechnic Institute and Donetsk National University for more than 30 years. Since 2007, he has been teaching data mining



Dr. Anatolii Illich Sleptsov

Interested in Additional Content & Publications in Similar Topics



Receive the Latest Research Findings, Publications, Publishing News & Complimentary Content, Directly to Your Inbox.

Sign Up to Receive Our Newsletter & Publication Updates

Keep Me Updated

Increase Your Paper's Conditional Acceptance & Save Time through IGI Global's Author Services

- English Language Copy Editing

- Scientific & Scholarly Editing

Activate Windows

Go to Settings to activate Windows.



45°F Fog



Practical application

- **PN-DPI** – integrated graphical simulation environment, 1988
- **Opera-Topaz** – production control and management system, 1990; Topaz, Motorsich
- **Nevod** - system for modeling discrete parallel processes and embedding in automated systems, 1991; Minsk Woodworking Plant

Concept of colored Petri net

- Petri net graph – bipartite directed graph
- Functional programming language ML for loading graph elements
- Tokens as elements of abstract data type (color set)
- Hierarchical structure – substitution of a transition by a net
- Timed delays associated with transitions and arcs
- Kurt Jensen, 1991: Design/CPN, CPN Tools

Tools to investigate a model

- **Observe token game – for understanding and debugging**
- **Build model state space – for correctness check of modeled system (verification)**
- **Simulation on prolonged intervals of time – for performance evaluation**

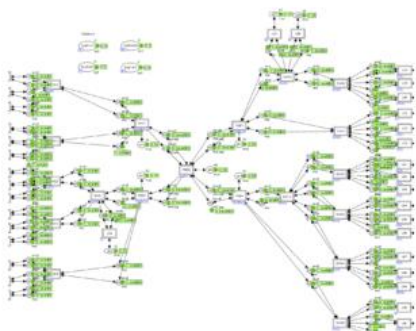
CPN TOOLS

[DOWNLOADS](#) ▾[GETTING STARTED](#) ▾[DOCUMENTATION](#) ▾[SUPPORT](#) ▾[CONTACT](#)

Declare
Hierarchy
Monitoring
Net
Simulation
State space

Examples contributed by Dmitry A. Zaitsev

Provider Backbone Bridge



This model is contributed by Dmitry Zaitsev. We list models without going thru them in too much detail, so listing does not imply that we endorse them or any papers listed, only that we provide them as inspiration. This model implements a Provider Backbone Bridge (PBB) network: PBB interior switch, PBB edge switch —with the dynamic filling up of address tables. This model is described in: [...]

Posted: January 9, 2018

Under: Examples contributed by Dmitry A. Zaitsev, Real-life examples, Sample CPN Models

[Read Me](#)[Leave comment](#)

Activate Windows
Go to Settings to activate Windows.



46°F Cloudy



6:54 AM
11/22/2021



Palettes and tools

CPN Tools (Version 4.0.1, February 2015)

Tool box

- Auxiliary
- Create
- Declare
- Hierarchy
- Monitoring
- Net
- Simulation
- Statistics
- View

Save Net As

Undo

Close Net

Declarations

Monitors

New Page

Save Net

Redo

Hier

Mon

Aux

Create

Net

Sim

SS

Style

View

New Page

Binder 0

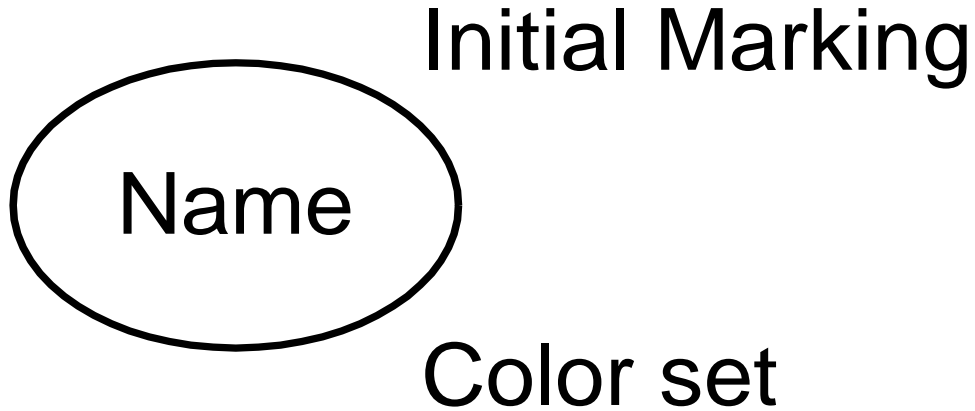
None

CPN

Активация Windows
Чтобы активировать Windows, перейдите в раздел "Параметры".

11:53
30.05.2017

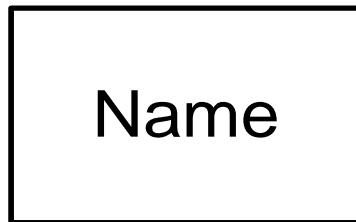
Place



Transition

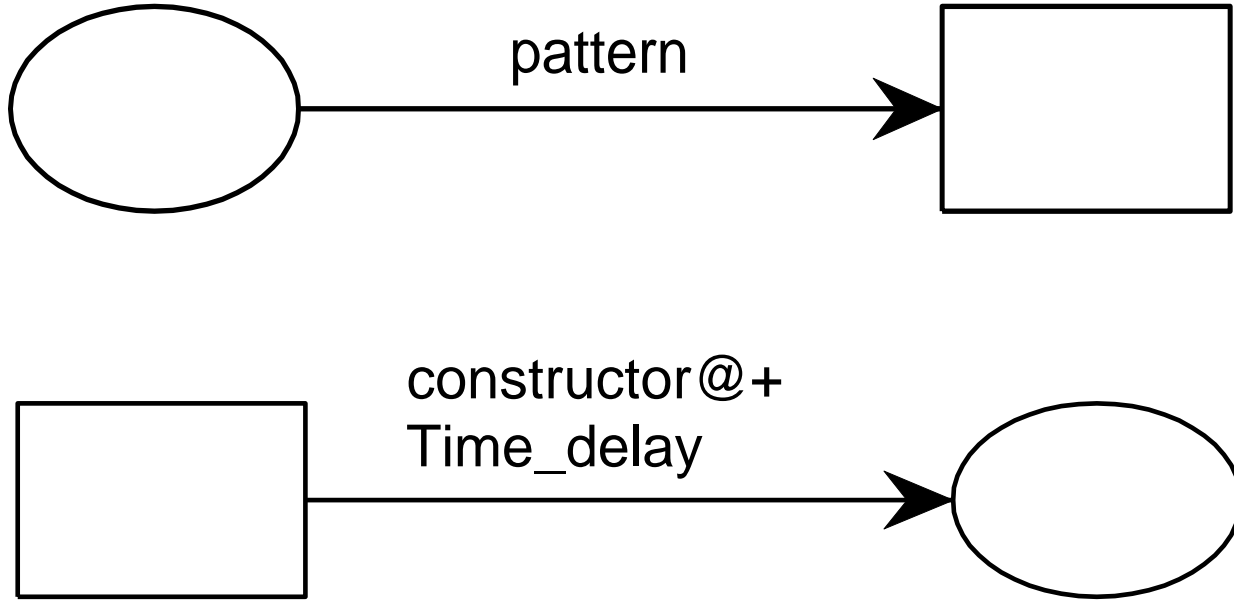
[Guard predicate]

@+Timed Delay



input ();
output ();
action
(A Program);

Arc

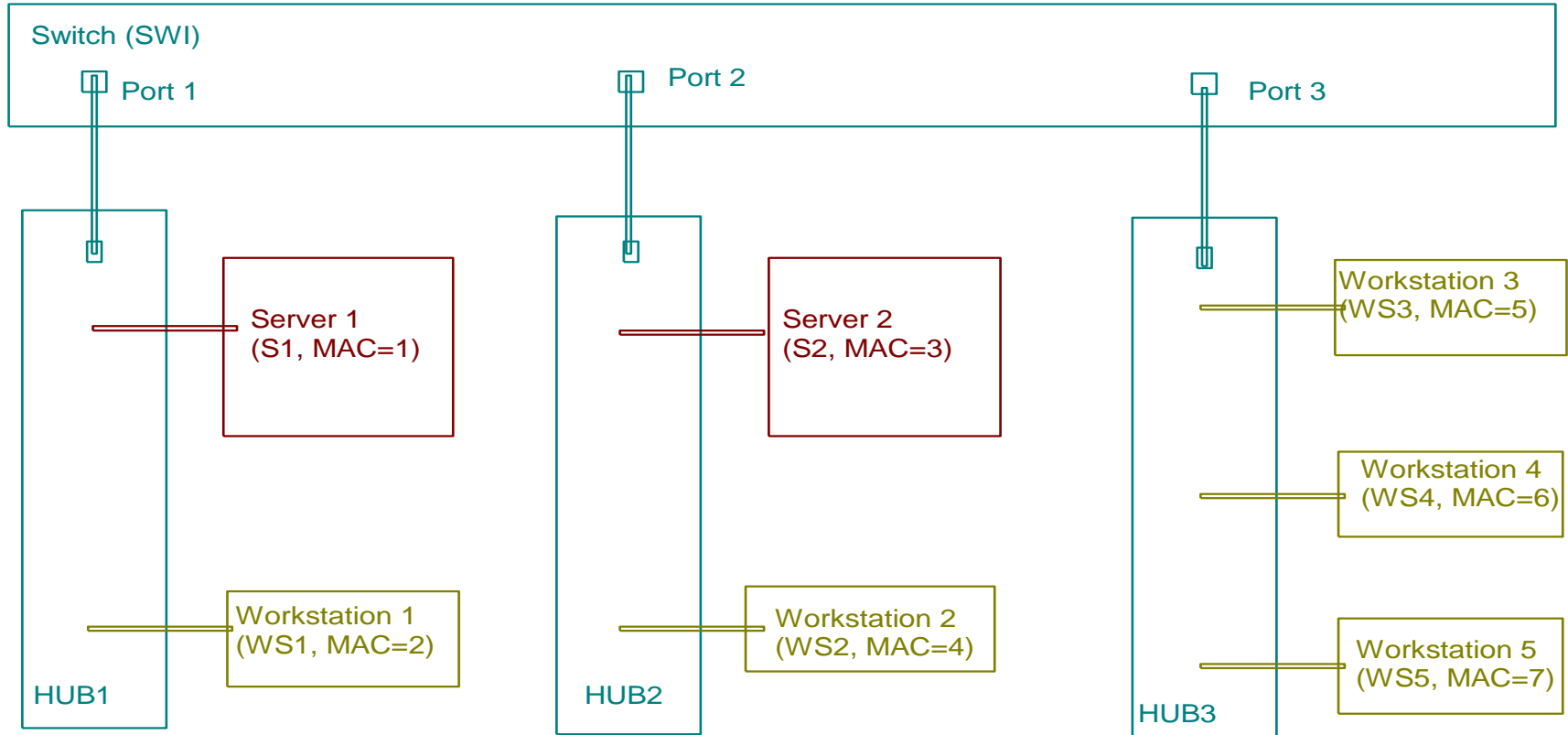


Declarations

- **colset** – color set (data type)
- **val** – constant
- **var** – variable
- **fun** – function

Model of switched Ethernet

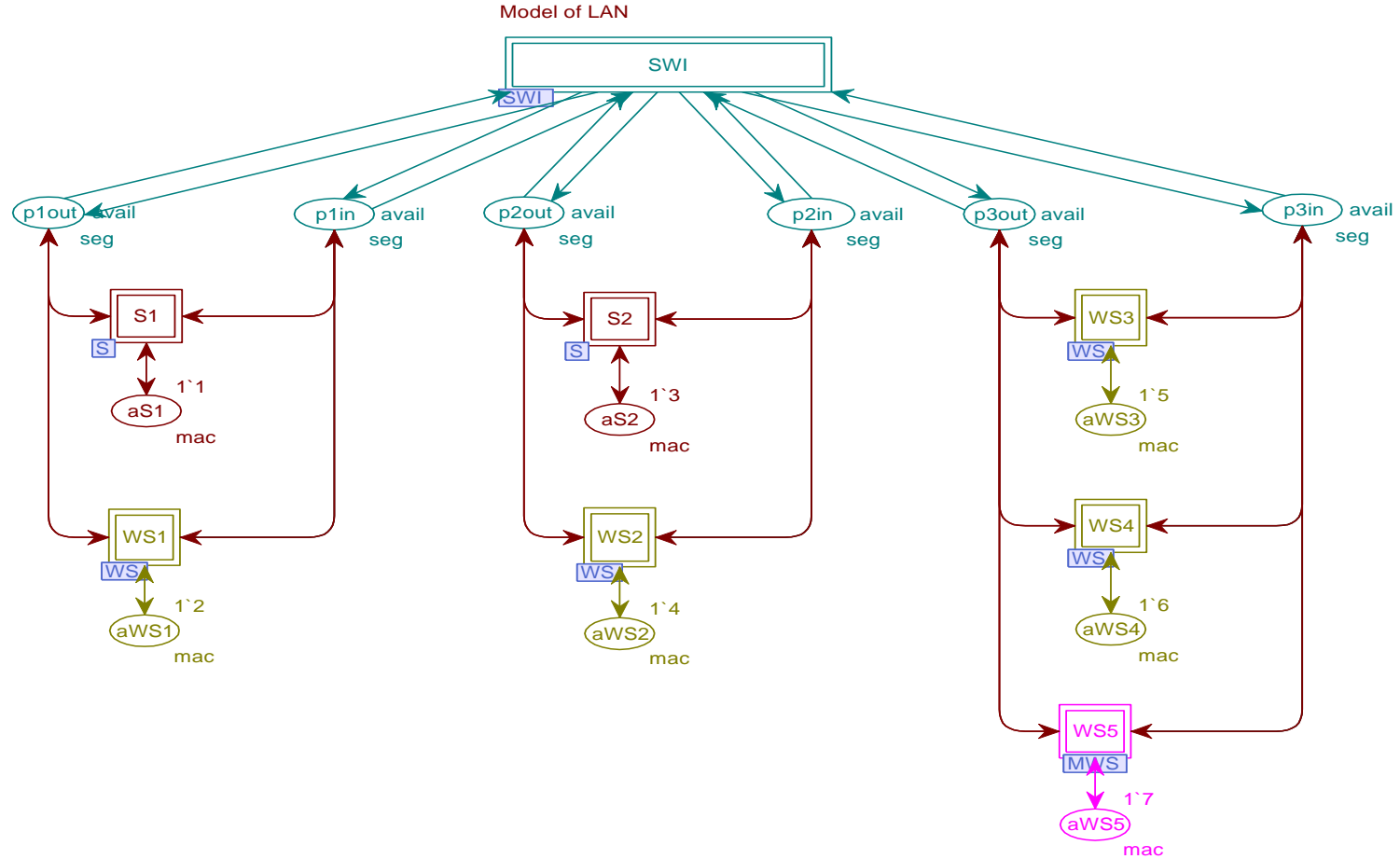
Scheme of sample switched LAN



Ethernet switch (Catalyst, Cisco)



Main page of model

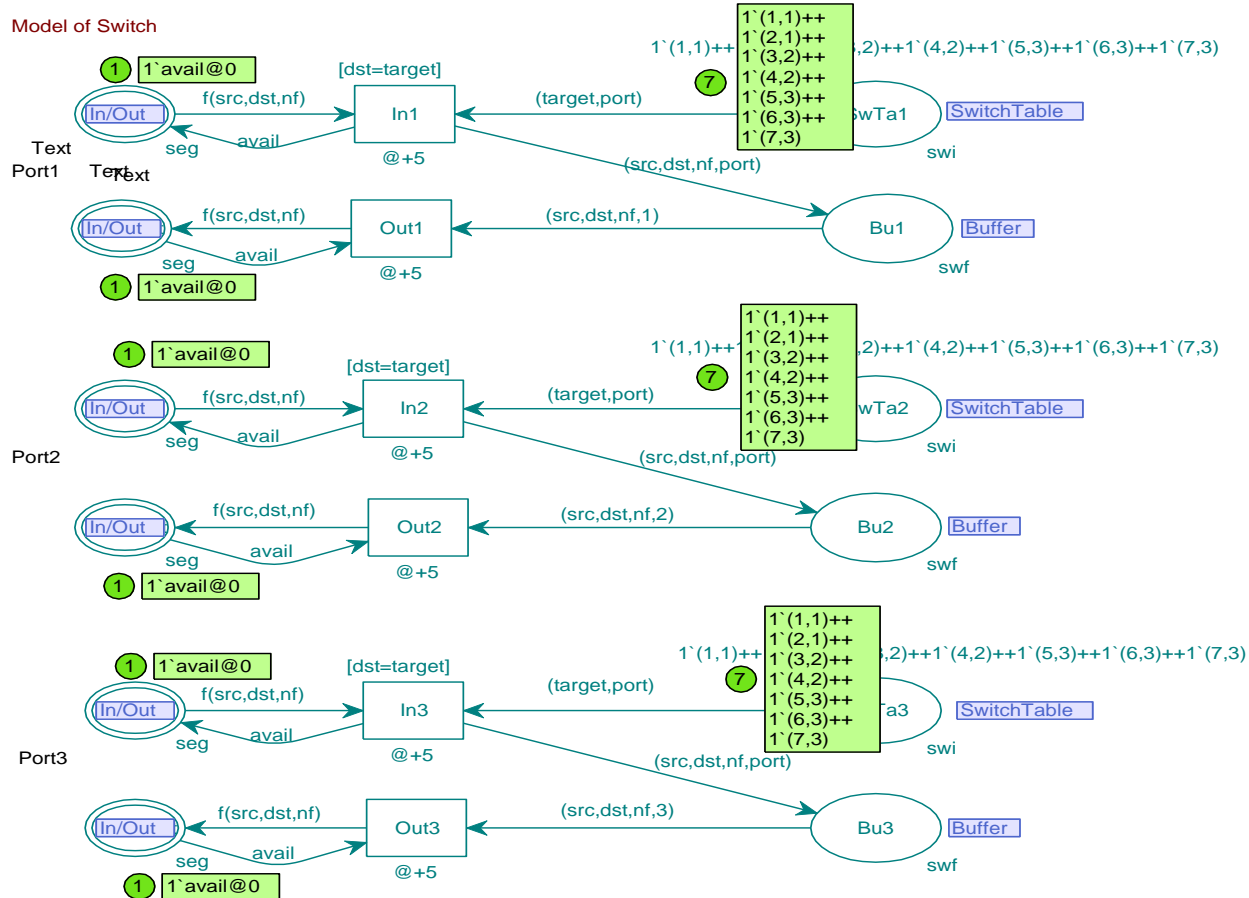


Declarations

```
colset mac = INT timed;  
colset portnum = INT;  
colset nfrm = INT;  
colset sfrm = product nfrm * INT timed;  
colset frm = product mac * mac * nfrm timed;  
colset seg = union f:frm + avail timed;  
colset swi = product mac * portnum;  
colset swf = product mac * mac * nfrm * portnum timed;  
colset remsv = product mac * nfrm timed;  
var src, dst, target: mac;  
var port: portnum;  
var nf, rnf: nfrm;  
var t1, t2, s, q, r: INT;  
colset Delta = int with 1000..2000;  
fun Delay() = Delta.ran();  
colset dex = int with 100..200;  
fun Dexec() = dex.ran();  
colset dse = int with 10..20;  
fun Dsend() = dse.ran();  
colset nse = int with 10..20;  
fun Nsend() = nse.ran();  
fun cT()=IntInf.toInt(!CPN'Time.model_time)
```

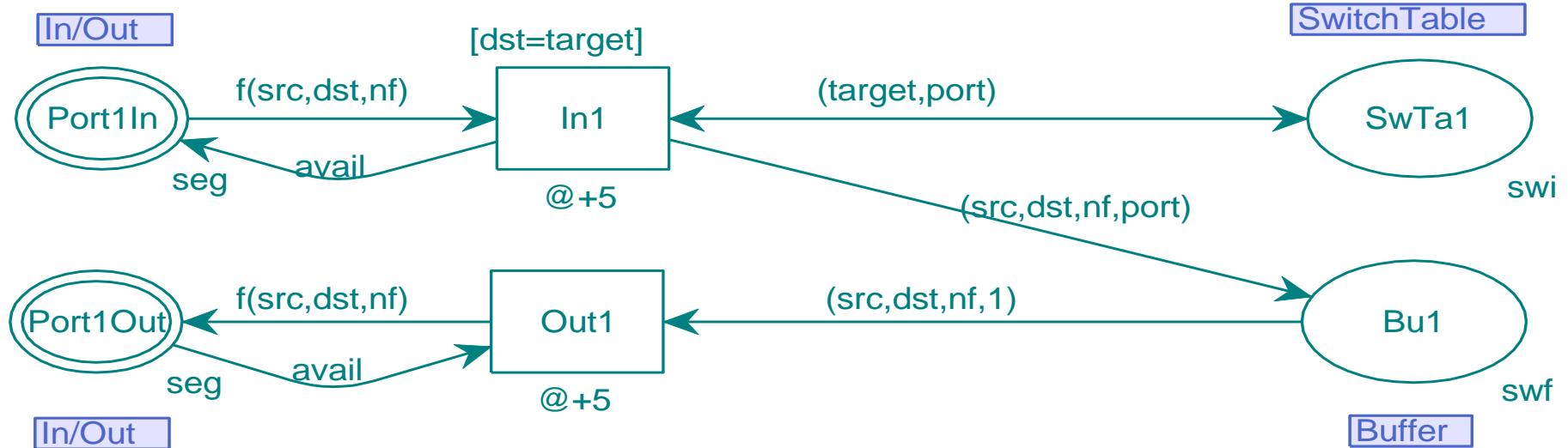
Ethernet switch model

Model of Switch

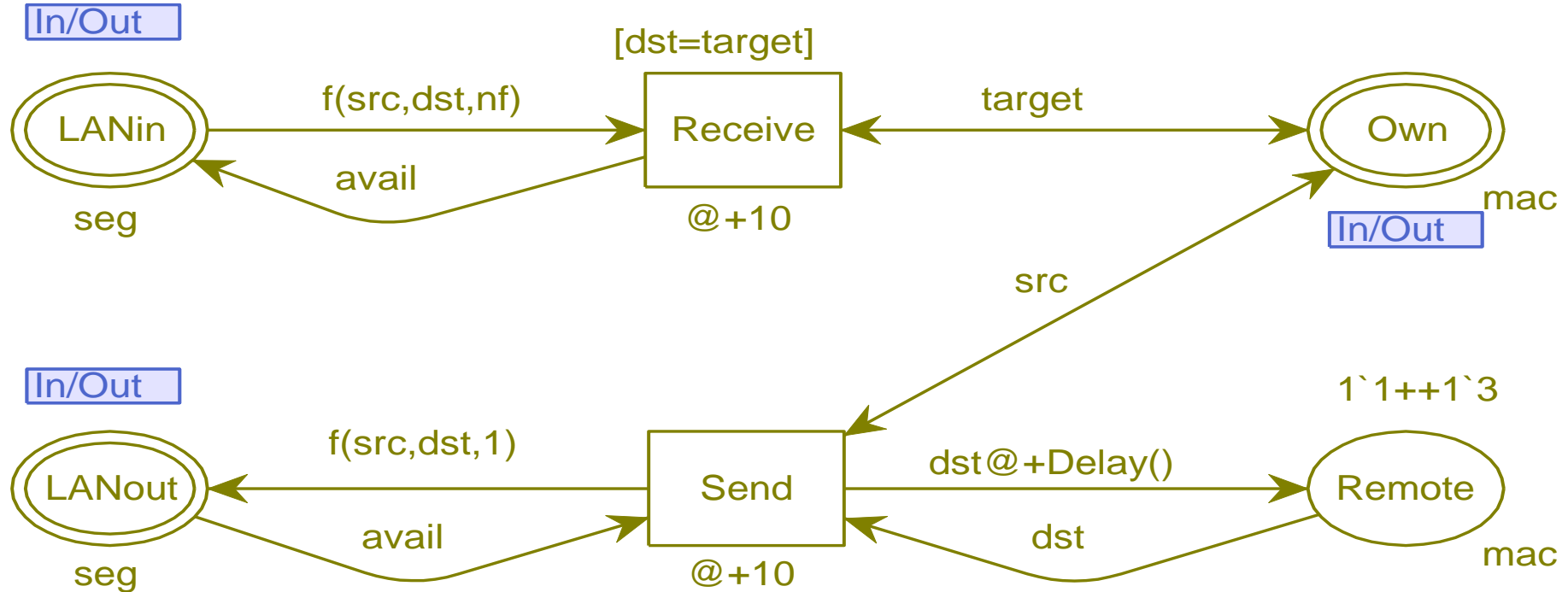


Switch port model

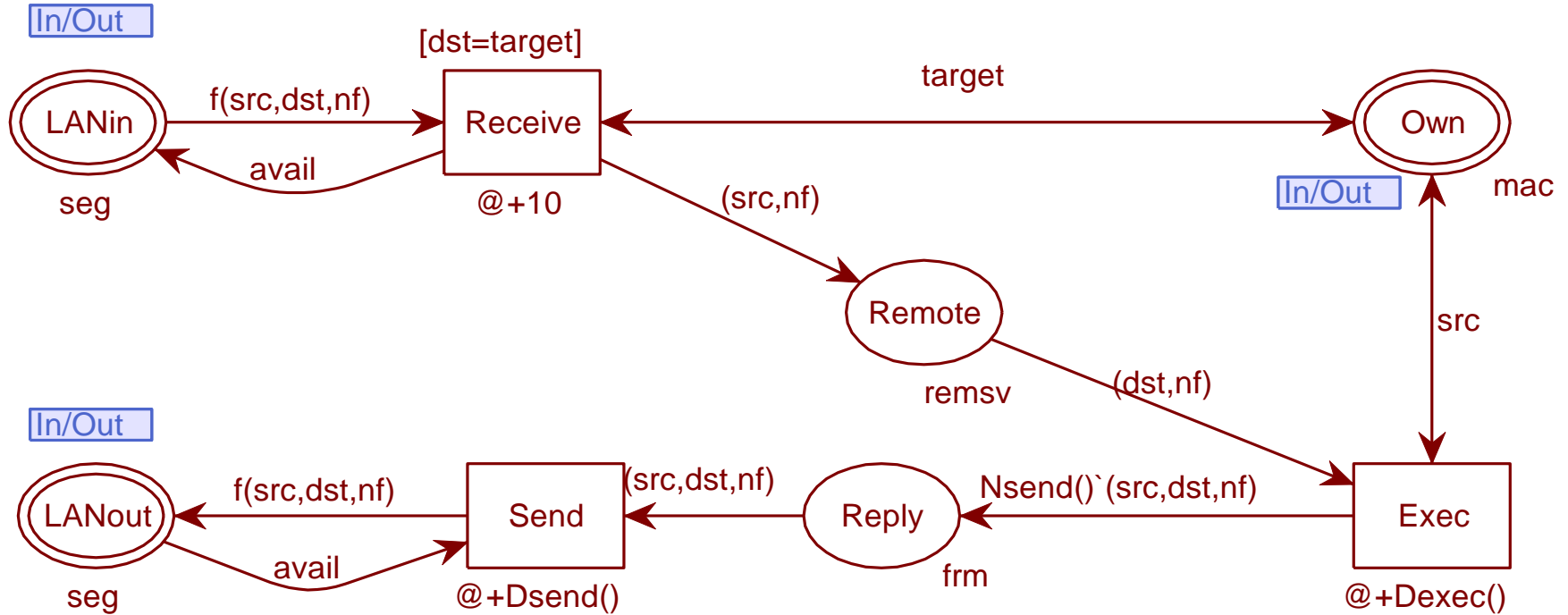
$1^`(1,1)++1^`(2,1)++1^`(3,2)++$
 $1^`(4,2)++1^`(5,3)++1^`(6,3)++1^`(7,3)$



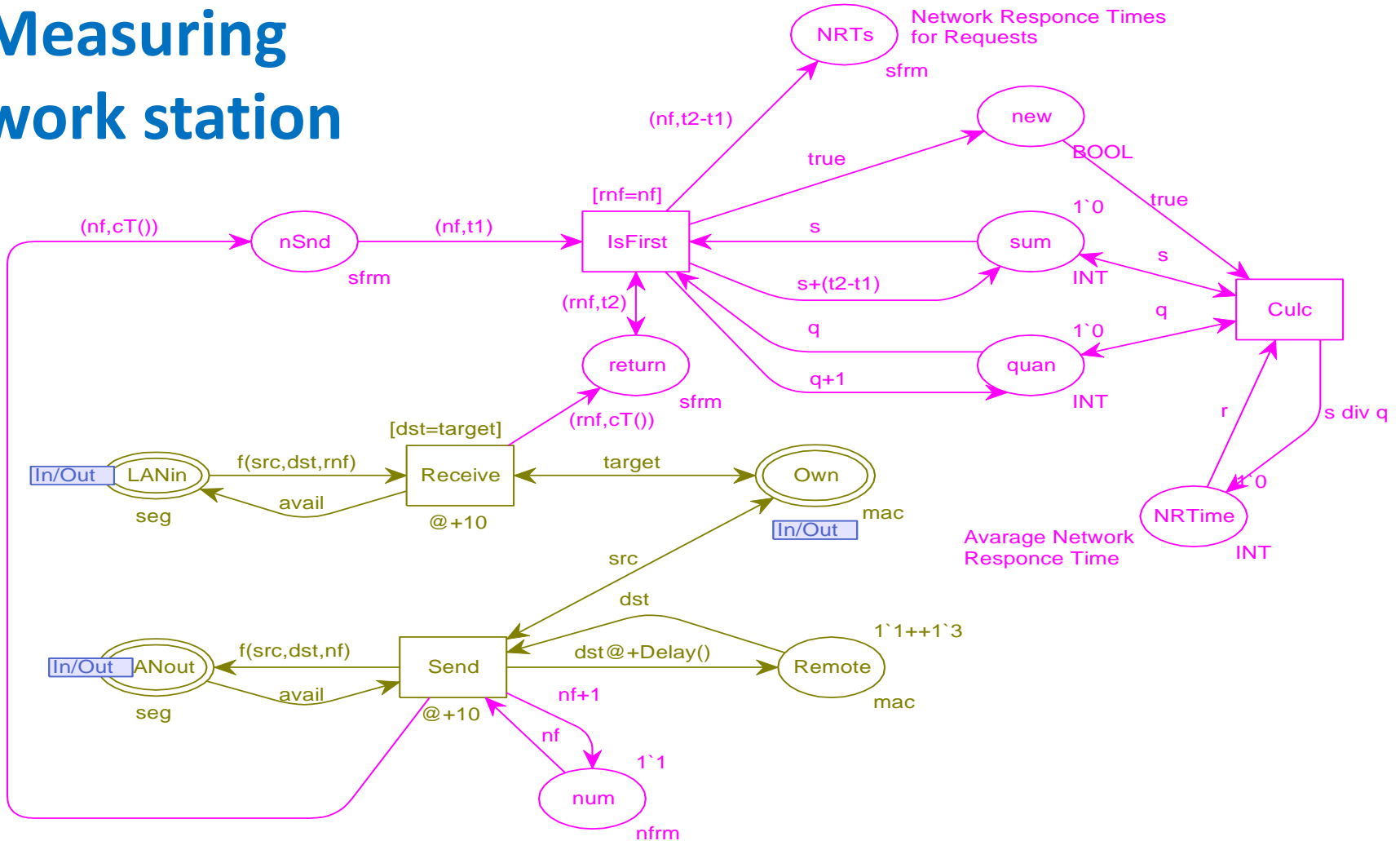
Working station



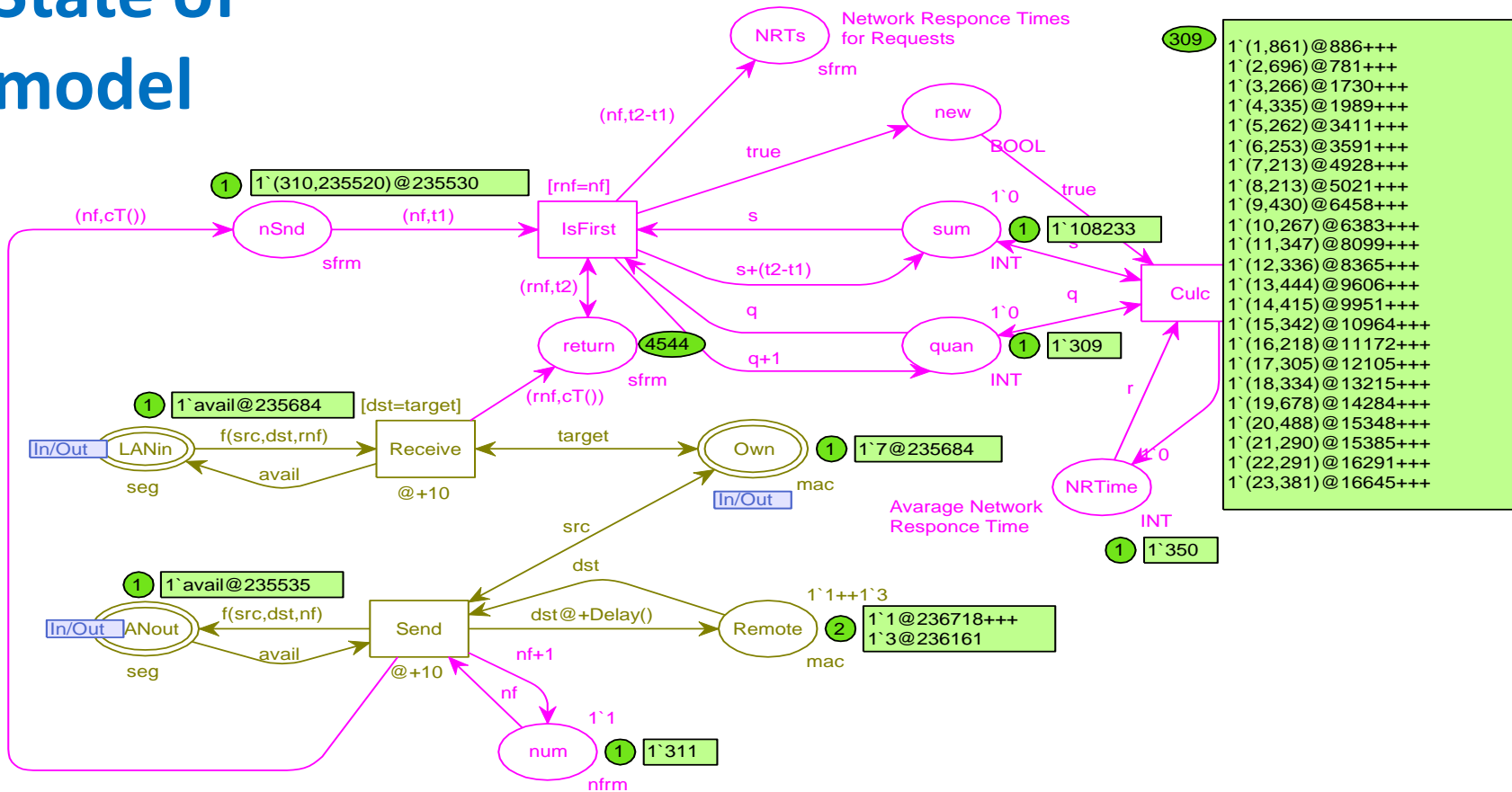
Server



Measuring work station



State of model

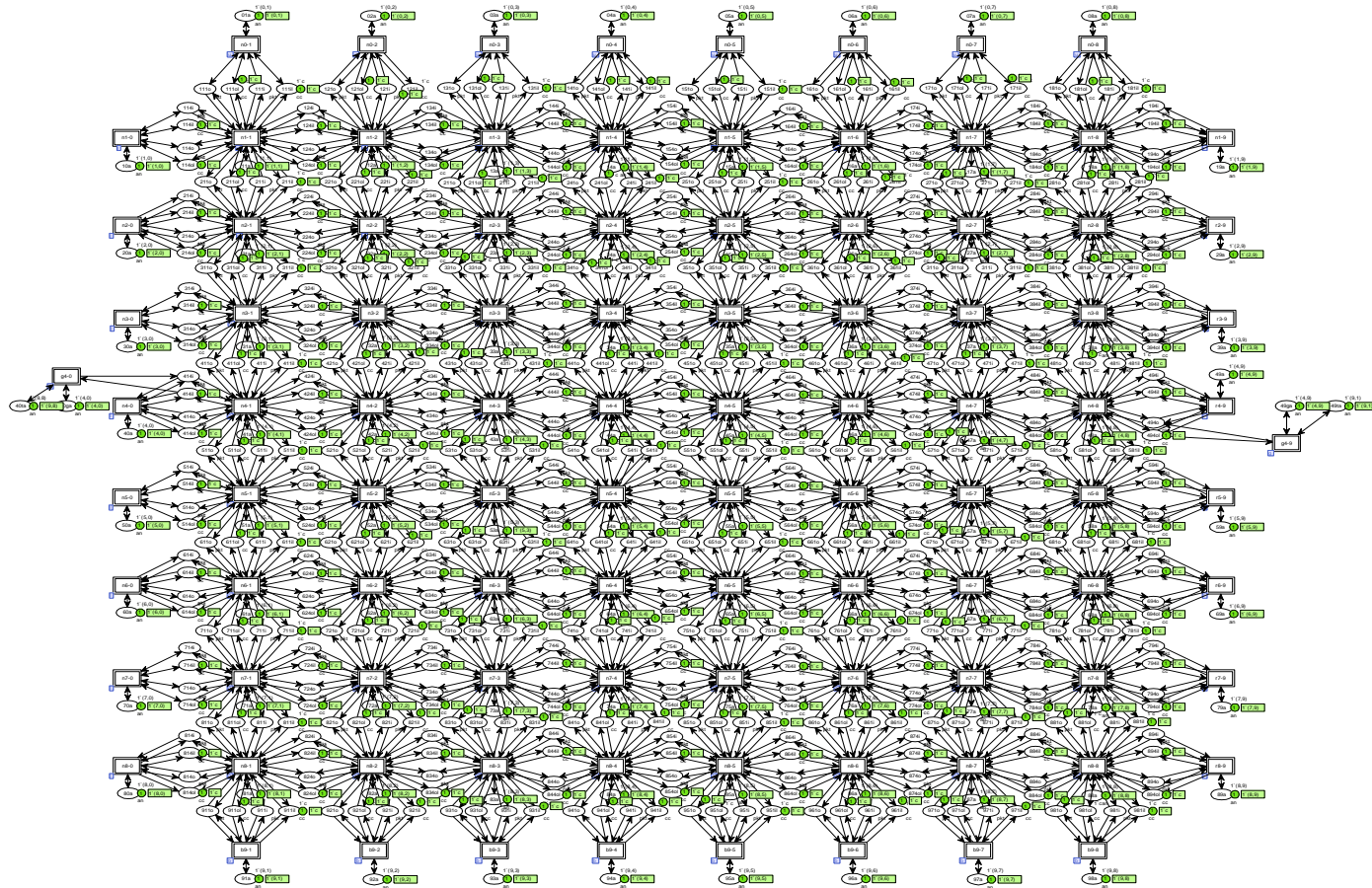


Scaling of time

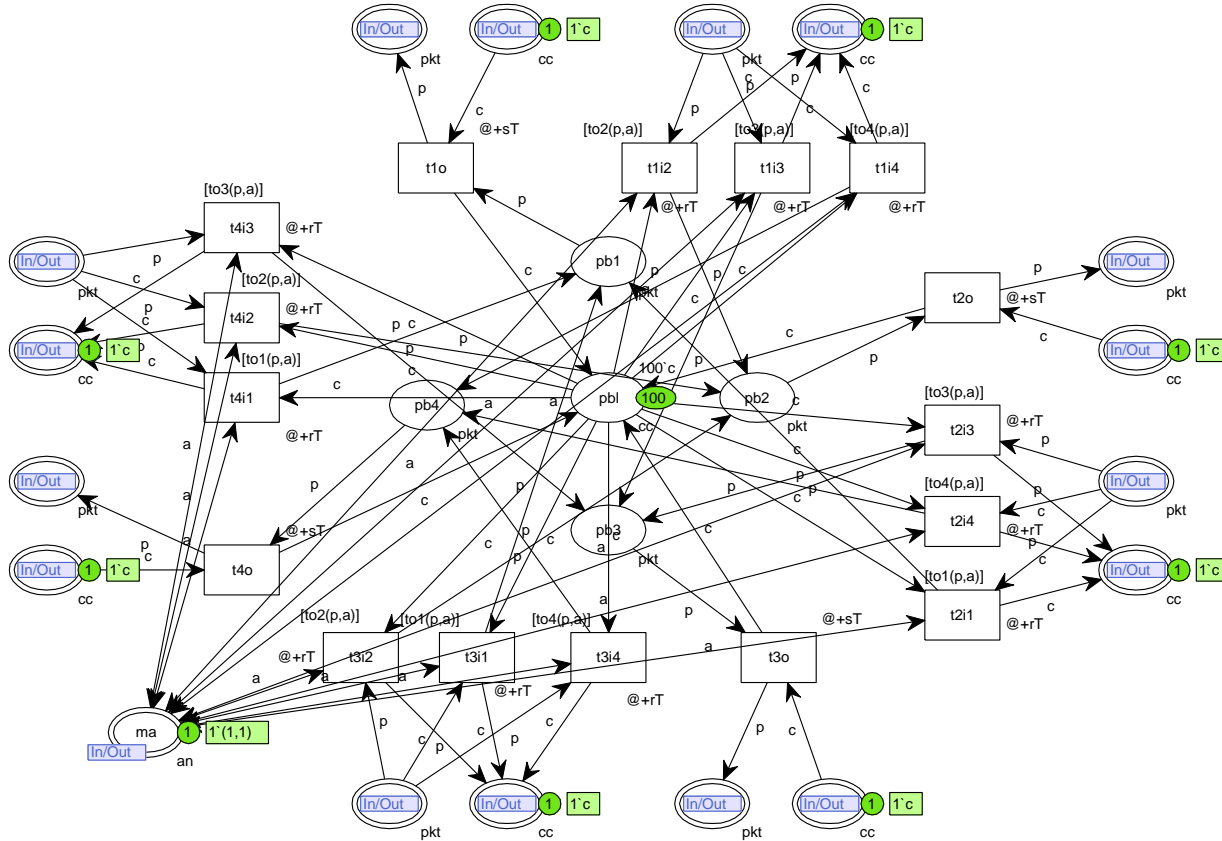
Table 2. Parameters of model

Parameter	Variable/Element	Real value	Model value
LAN switch read frame delay	In*	500 ns	5
LAN switch write frame delay	Out*	500 ns	5
LAN adapter read frame delay	Receive	1 ms	10
LAN adapter write frame delay	Send	1 ms	10
Server's time of request processing	Dex	10-20 ms	100-200
Client's delay between requests	Delta	100-200 ms	1000-2000
Length of request		1.2 Kb	1
Length of response	Nse	15-30 Kb	10-20

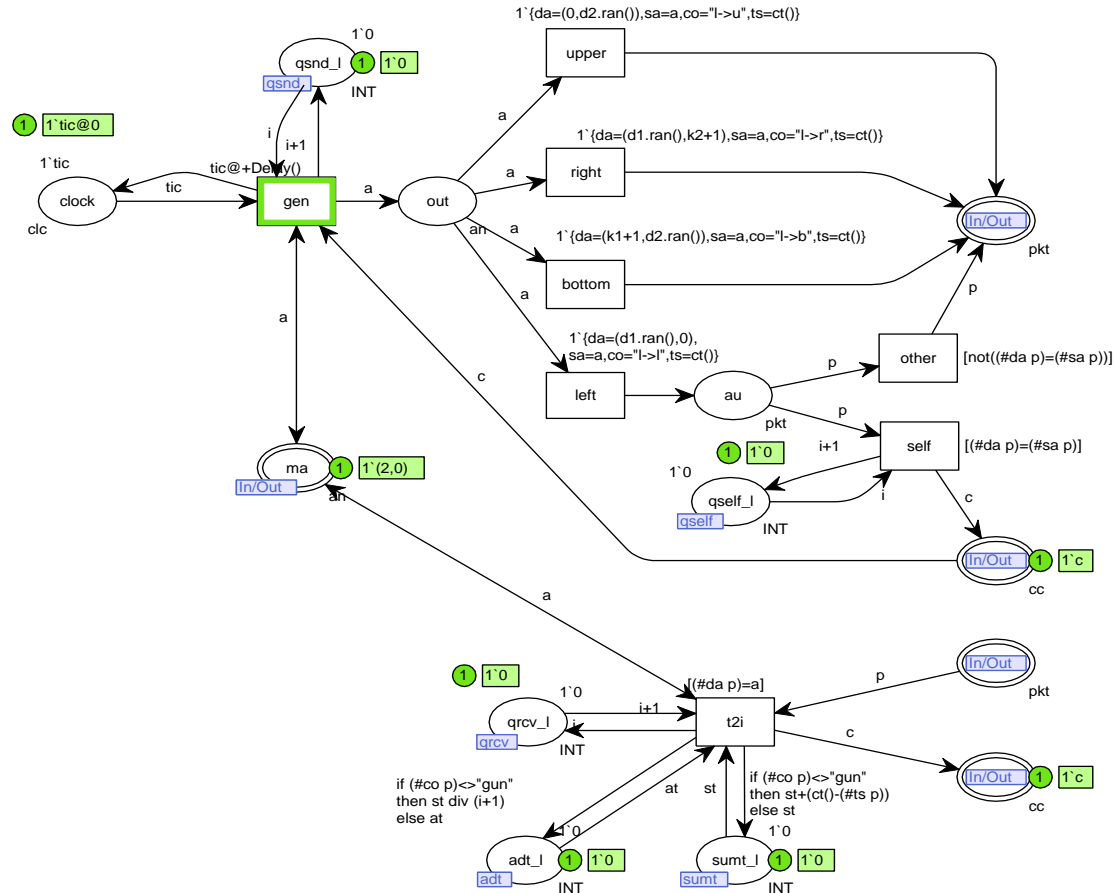
Square grid model 8x8



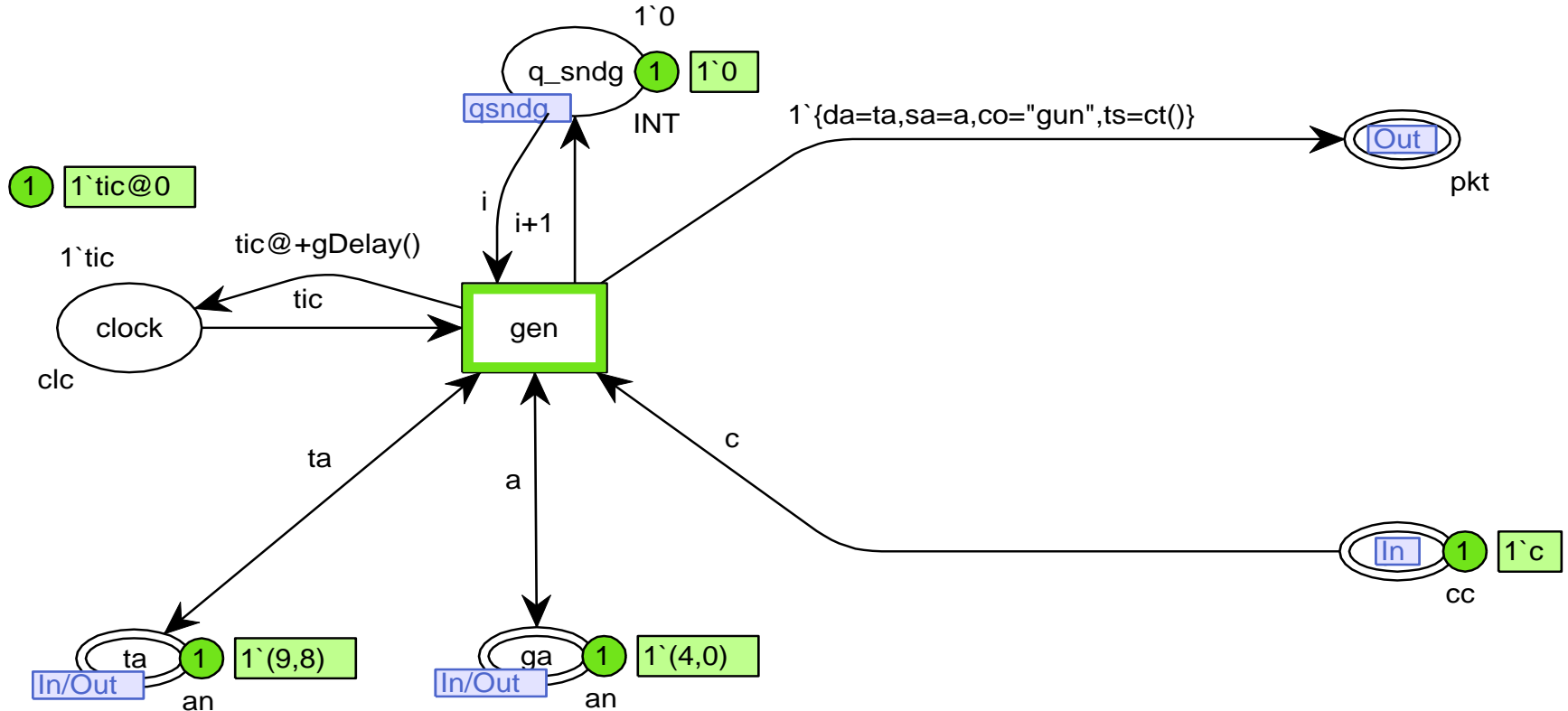
Grid node model



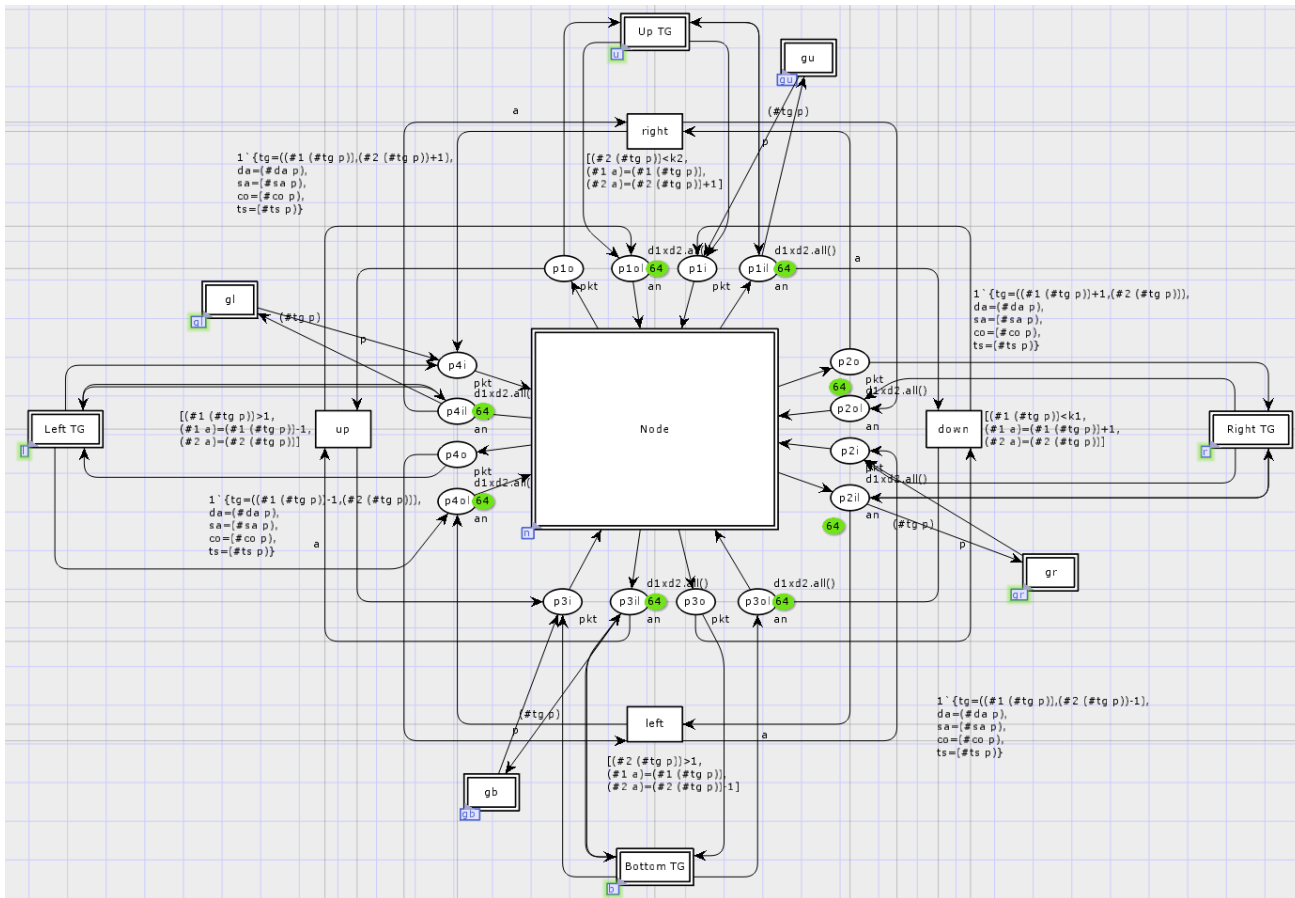
Model of (left) boundary device



Traffic gun model



Reenterable model of grid



Library of models

- **Ethernet** – switched local area network
- **IP** – Internet protocol network
- **MPLS** – label switched network
- **Bluetooth** – wireless pico-network
- **PBB** – provider backbone bridge
- **E6** – network with original family of protocols
- **Grids** – computing and communication grids

Conclusions

- **Colored Petri net – convenient and powerful tool for modeling networks, grids, and clouds**
- **Build state space for verification**
- **Simulate on prolonged intervals of time for performance evaluation**
- **Flexibility and vivid graphical representation**
- **Recommended for design of real-time systems and novel networking technology**

Problem

- **Heterogeneous environment for parallel and distributed programming (OpenMP, MPI, CUDA/OpenCL), processor-memory bottleneck, difficulties of concurrent programs verification**
- **Novel paradigm of computation and graphical concurrent programming on Sleptsov nets**

References

- Zaitsev D.A. and Shmeleva T.R. [Modeling With Colored Petri Nets: Specification, Verification, and Performance Evaluation of Systems \(pp. 378-404\)](#) Chapter 14 in T. Shmelova, N. Rizun, D. Kucherov and K. Dergachov (Ed.) Automated Systems in the Aviation and Aerospace Industries. IGI-Global: USA, 2019.
- D.A. Zaitsev, T.R. Shmeleva, A.I. Sleptsov: Reenterable Colored Petri Net Models of Networks, Grids, and Clouds: Case Study for Provider Backbone Bridge. [26th Telecommunications Forum \(TELFOR 2018\), November 20-21, 2018, Belgrade, Serbia.](#)
- Zaitsev D.A. Switched LAN simulation by colored Petri nets, [Mathematics and Computers in Simulation, Volume 65, Issue 3, 22 April 2004, 245–249.](#)