Application of Petri Nets to Evaluation of Grid Applications Efficiency

Wojciech Rząsa[1], Marian Bubak[2,3]

(1) Rzeszow University of Technology, Poland
(2) Institute of Computer Science AGH, Krakow, Poland
(3) ACC Cyfronet, AGH, Krakow, Poland

Outline

- Motivation and the goal
- Related work
- Analysis method
  - The models
  - Enabling the simulation
- Results
- Conclusions and future work

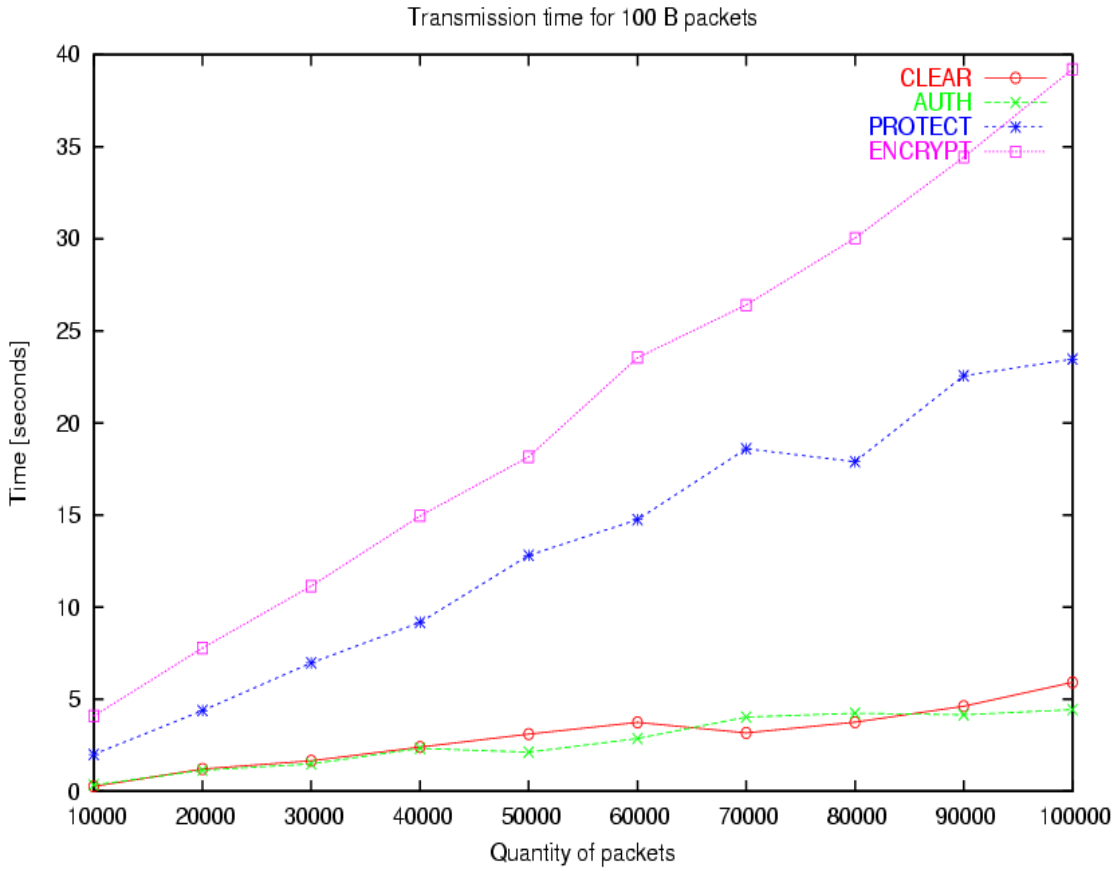
Motivation
GT2 security overhead

[Baliś, Bubak, Rząsa, Szepieniec 2004]

- Secured connection enables
  - Authentication
  - Data integrity
  - Confidentiality
- Connection establishment
- Data transmission

Connections	Requested in 1 second	Established in 1 second	Failed
Secured	896	30	4

Clear	1692	1691	0



Motivation
Communication overhead examples

- Secure communication
  - GT2 GSI
  - GT3/GT4 transport-level and message-level security
  - SSL/TLS
  - VPN
- Communication layers
  - TCP/IP
  - HTTP over TCP/IP
  - SOAP over HTTP over TCP/IP
- Network link parameters

Motivation
Distributed application efficiency

- Communication overhead
- Delay caused by the other resources
  - e.g. CPU
- Application logic and topology
  - implies resource usage

The goal

- Analyze efficiency of distributed application
- Depending on
  - Parameters of resources
  - Application design
- Method for application developers

Simulations of the Grid – examples

- Replica management simulators
  - `OptorSim` DataGrid project, 2004
  - `GridNet` [Lamehamedi, Shentu, Szymanski, Deelman 2003]
  - `ChicagoSim` [Ranganathan, Foster 2001]
- Scheduling algorithms simulators
  - `SimGrid` [Casanova, Legrand, Marchal 2003]
  - `GridSim` [Buyya 2005]
- Grid security simulator – `G3S` [Naqvi, Riguidel 2005]
- Grid application simulators
  - `Performance Prophet` [Fahringer 2005]

Simulations of the Grid – tools and engines exploited

- General purpose discrete-event simulators
  - PARSEC [Bagrodia]
    - `ChicagoSim`
  - SimJava [McNab,Howell 1998]
    - `GridSim`
  - CSIM [Schwetman 1998]
    - `Performance Prophet`
- Network Simulator (ns)
  - `GridNet`

The method

- Model the environment and the application
- Perform simulation to obtain statistics

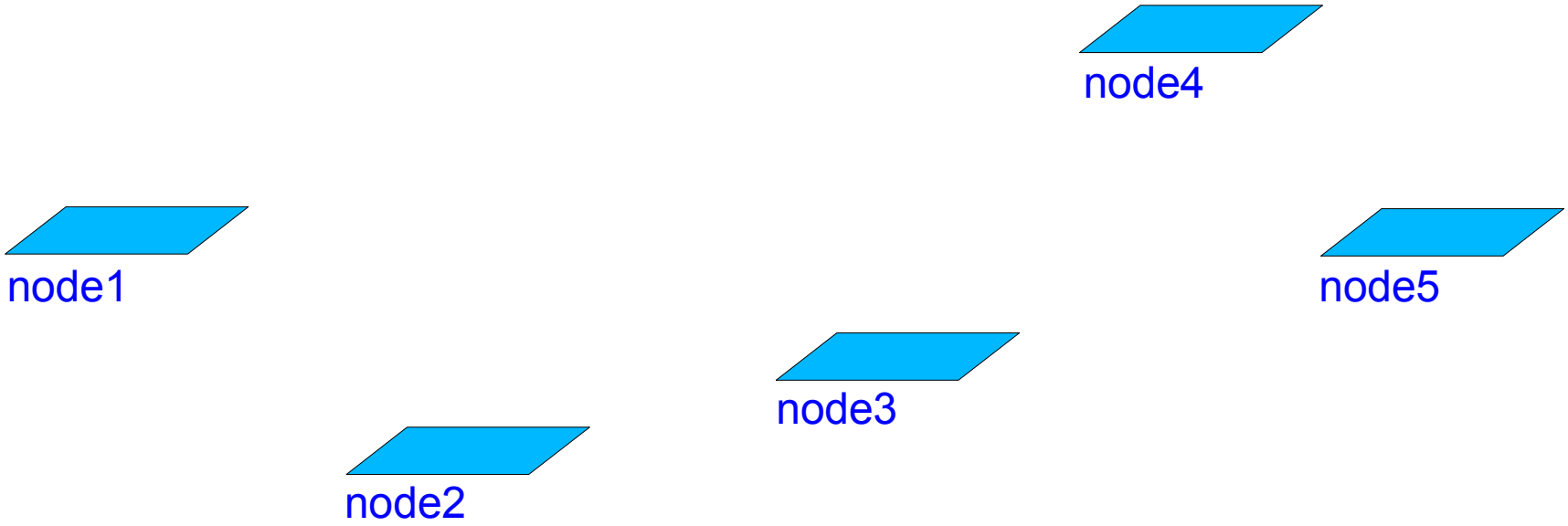
Parts of the model

- Model of the Grid resources
- Model of the application

The model

Nodes:

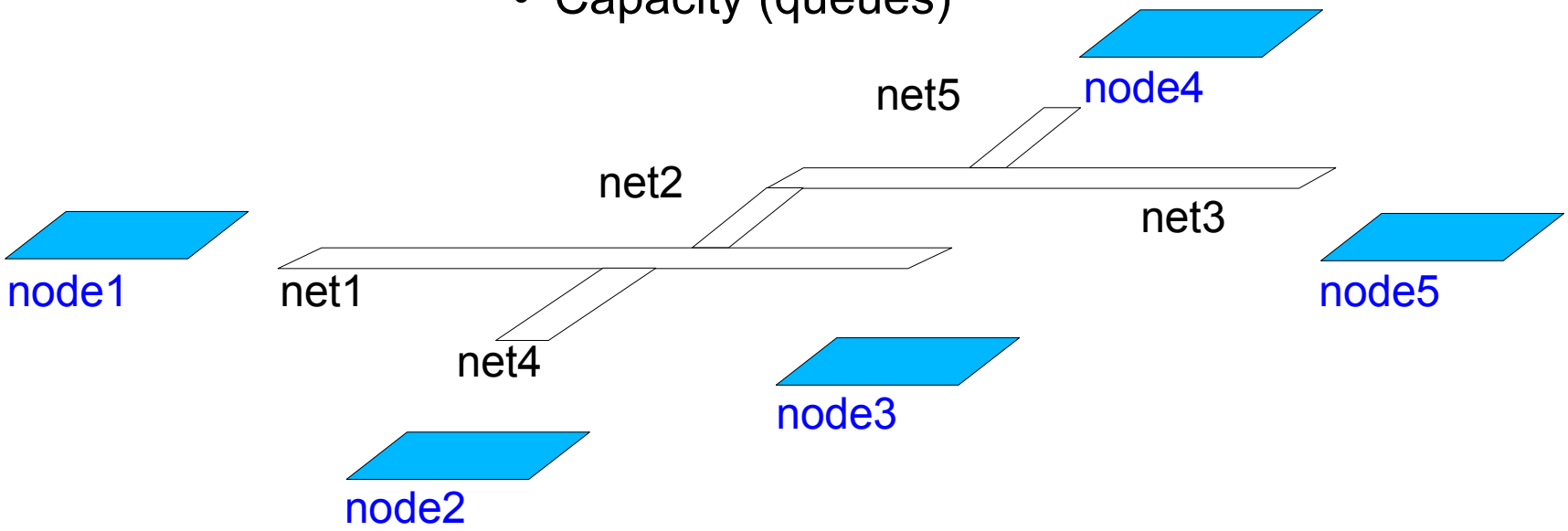
- CPU(s)



The model

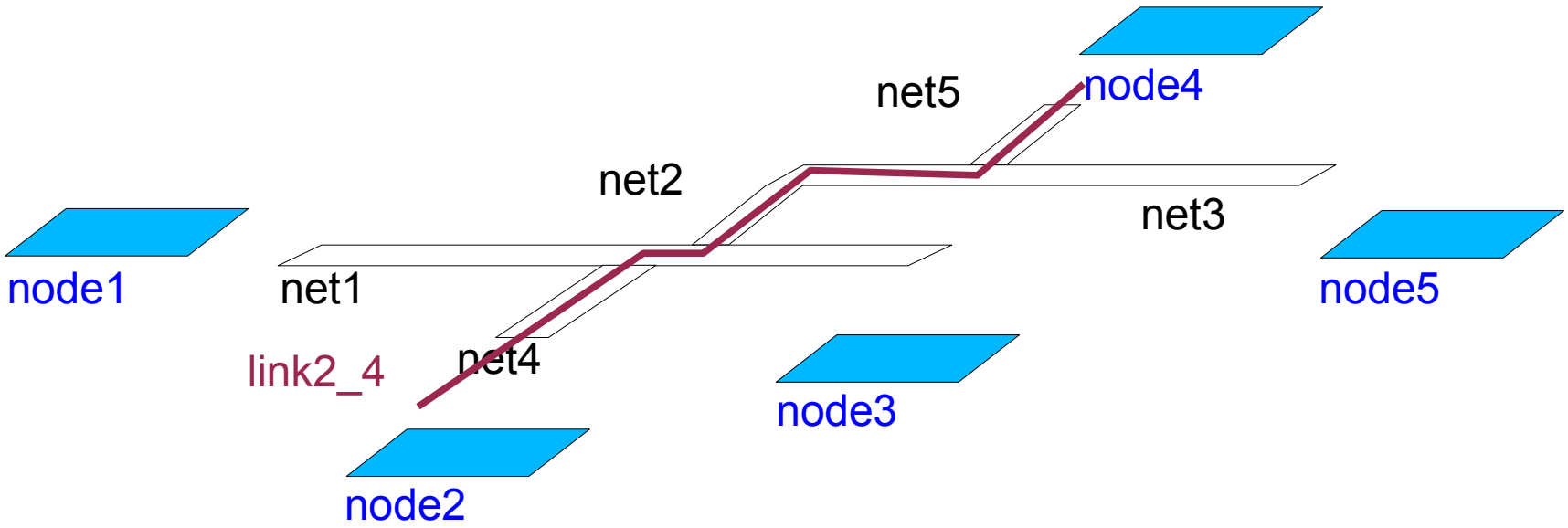
Parameters of the **net segments**:

- Bandwidth
- Delay
- Capacity (queues)



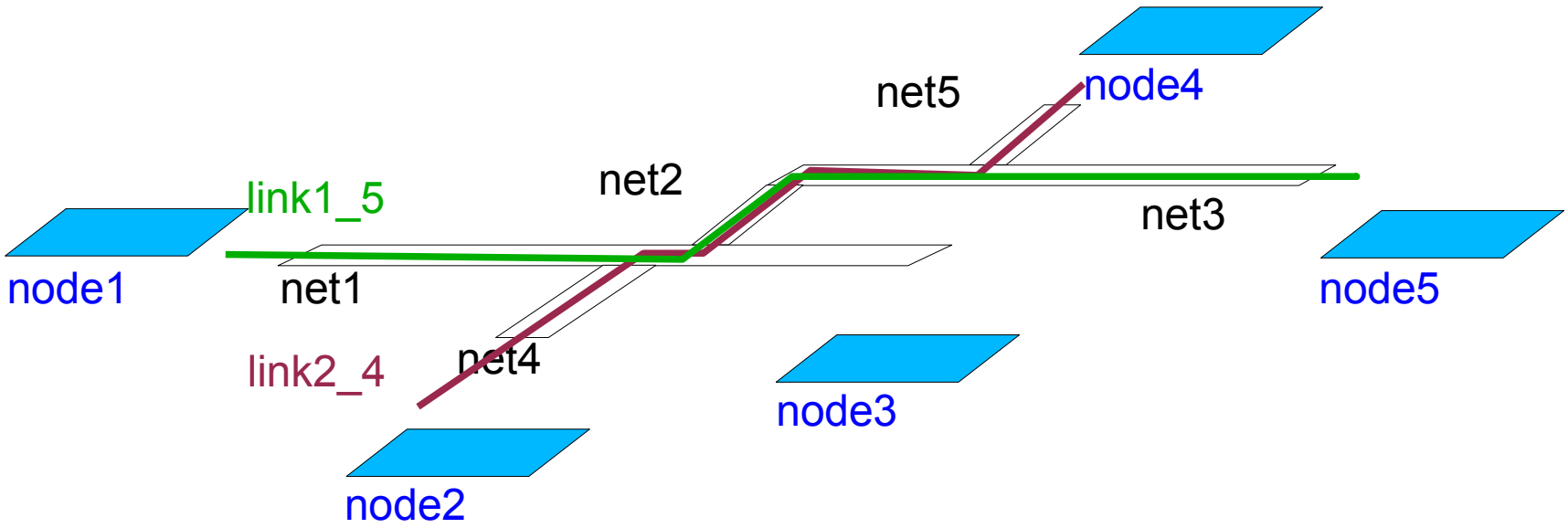
The model

Link – route between the nodes



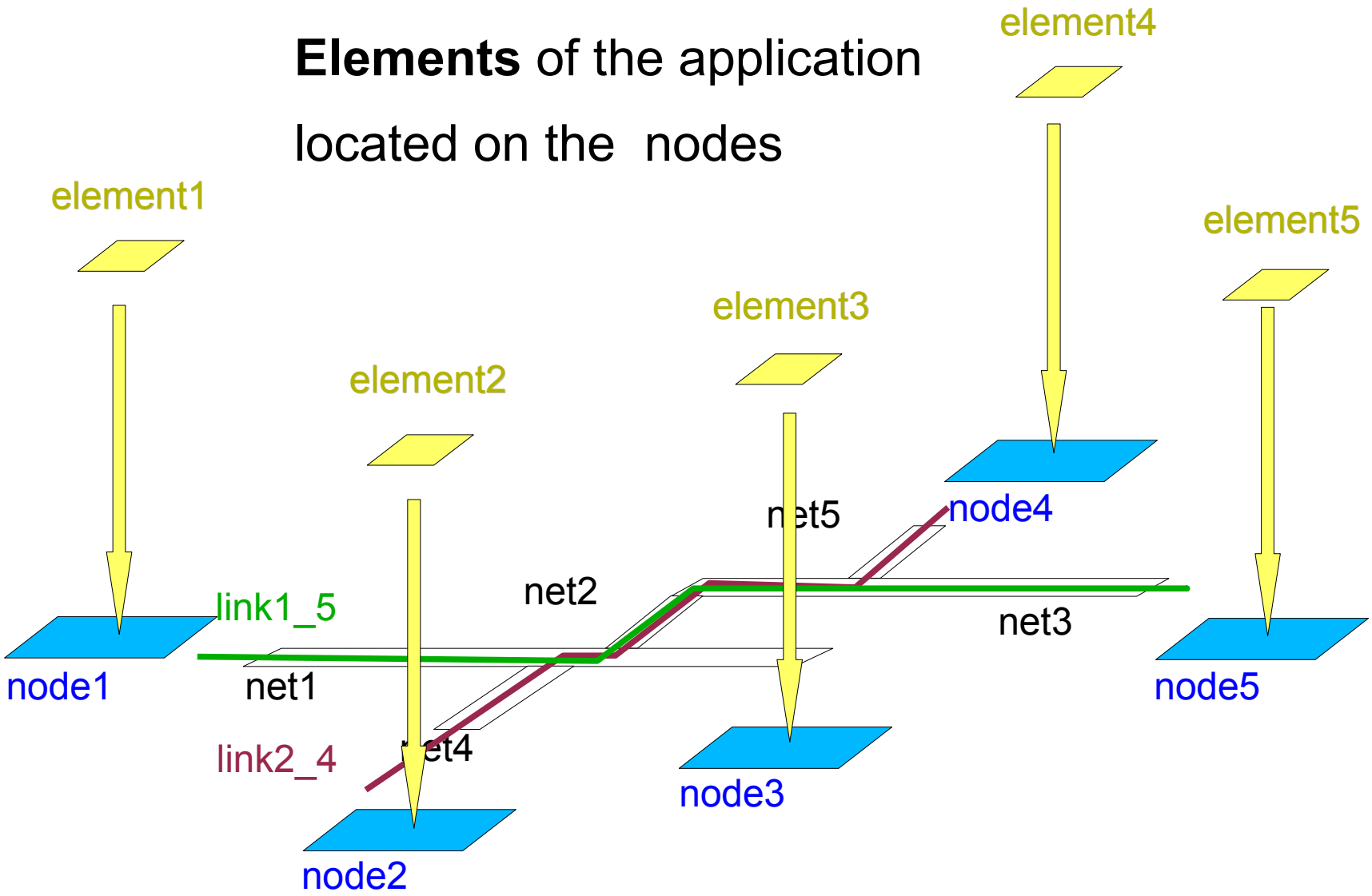
The model

Link – route between the nodes



The model

Elements of the application located on the nodes

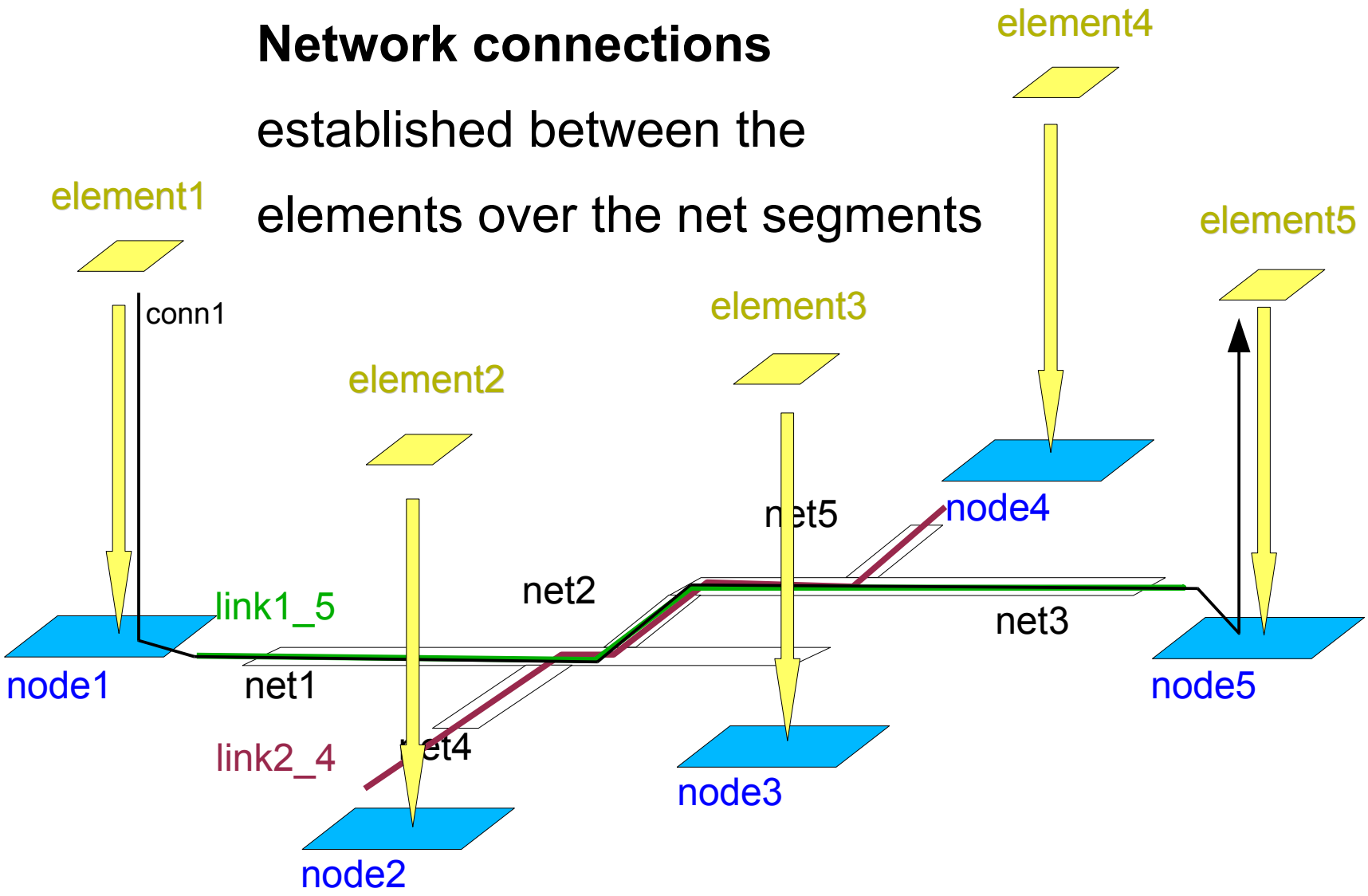


The model

Network connections

established between the

elements over the net segments

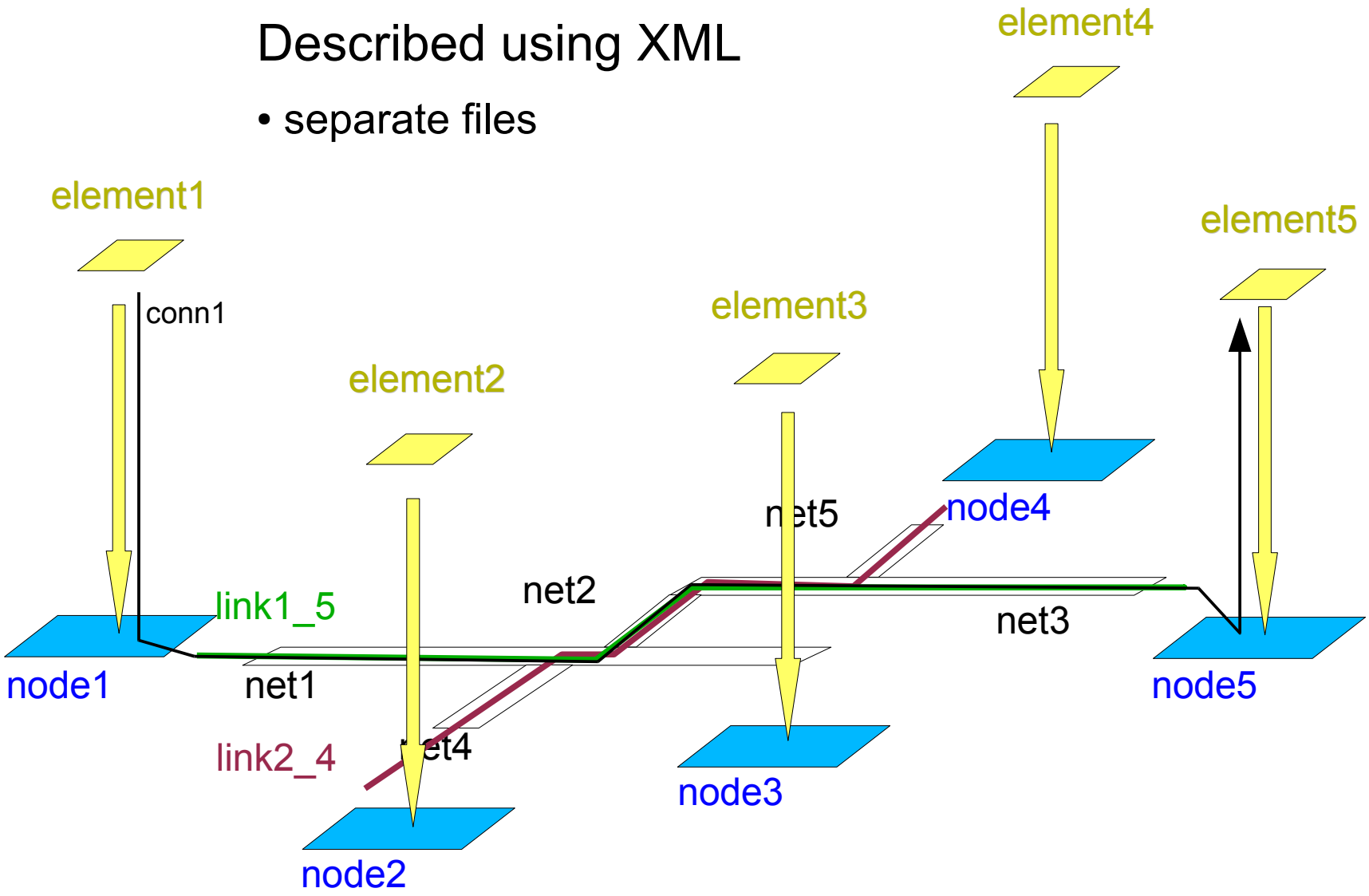


The model

application.xml mapping.xml resources.xml

Described using XML

- separate files



Parameters of the application element

- Network connection parameters
- Delay caused by processing
- Parameters of generated communication

- All parameters described using expressions

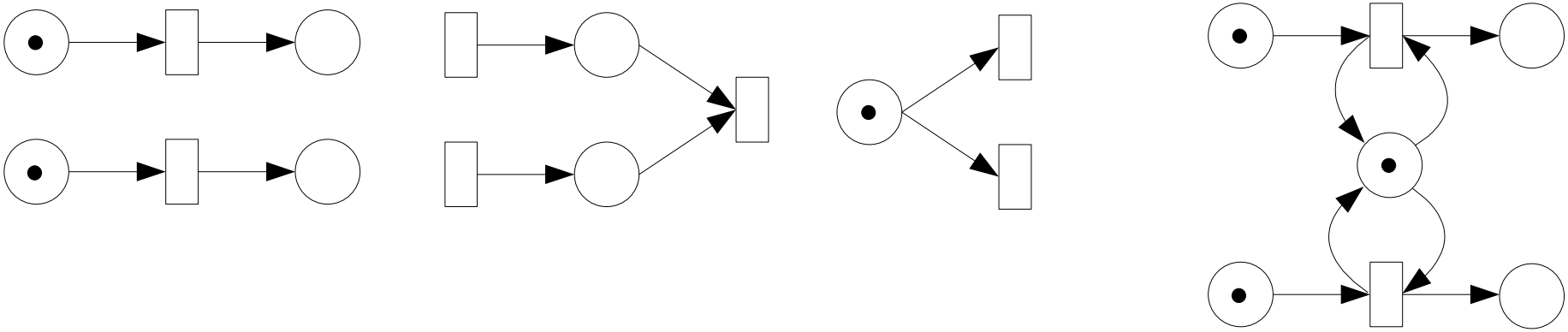
```
if ($time<1000) { 100*$incomingVolume }
else { pow($incomingVolume,3) }
```

Reliable simulation

- High level model – convenient for application developers
- Executable model for reliable simulation
  - Based on a formalism
  - Properly reflecting activities of concurrent, distributed applications
- Automatic transformation, transparent for the user

Enabling the simulation
Petri Nets

[Murata 1989]



Concurrency | Synchronization | Conflict | Mutual exclusion

- Reliable formal model of concurrent processes
- Simulation
  - efficient
  - interactive

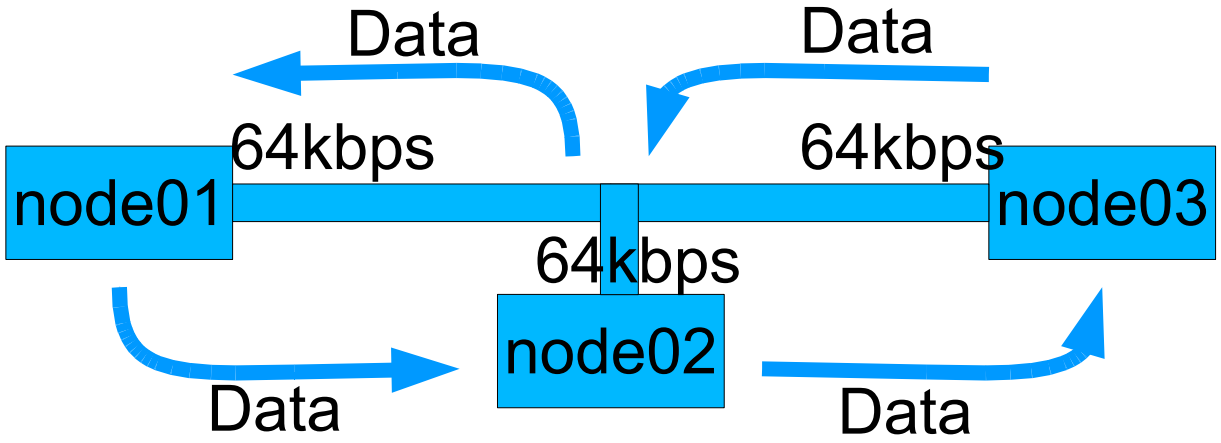
Timed Colored Petri Net

[Jensen 1994] [Jensen 1995/96]

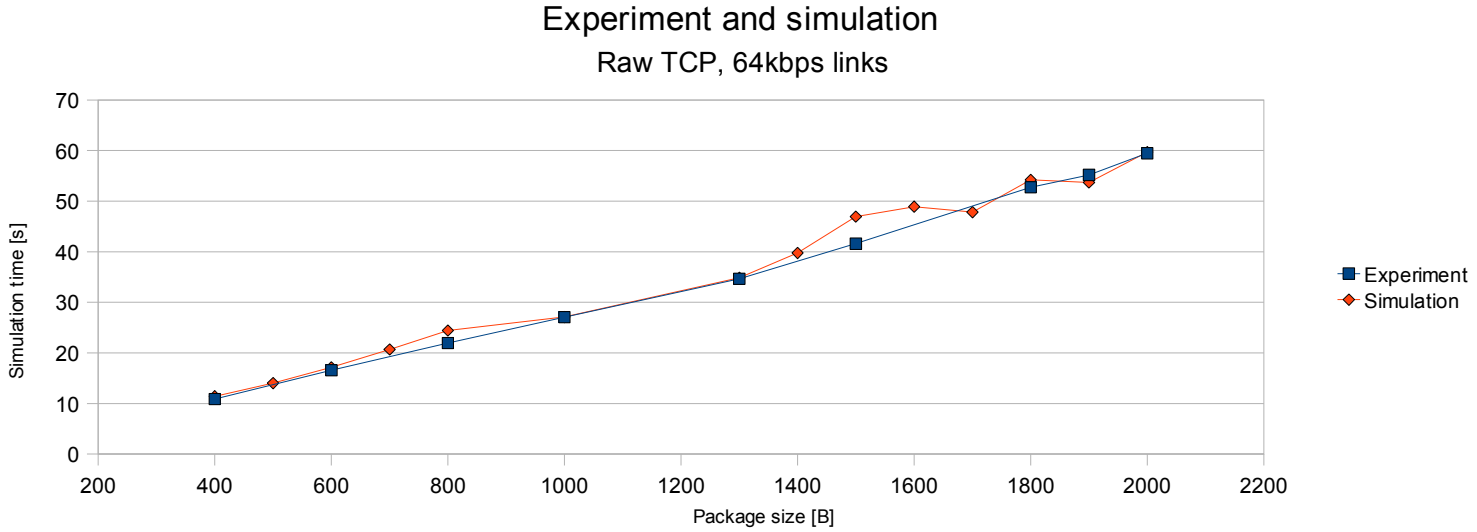
- Classical PN extended by
  - Color sets – data types
  - Colors of tokens – values
  - Guards defined for transitions
  - Arc expressions
- More compact and transparent model
- Time
  - Tokens with timestamps
  - Timestamps modified by transitions
  - Timestamps affect availability of tokens

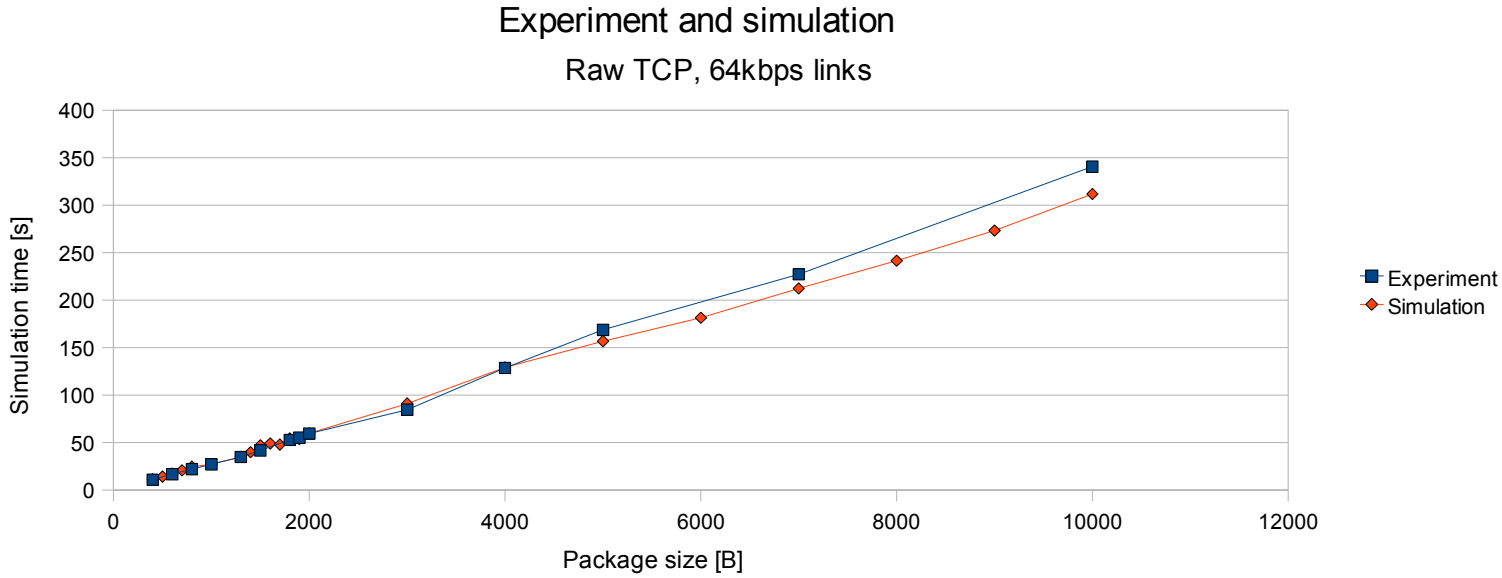
Experiment

- At most one persistent connection between each two nodes
- Transmission of 100 packages of data
- No other data processing
- Measurement of wall time of whole experiment
- Nodes: AMD Athlon 64 1.8GHZ, 2GB RAM

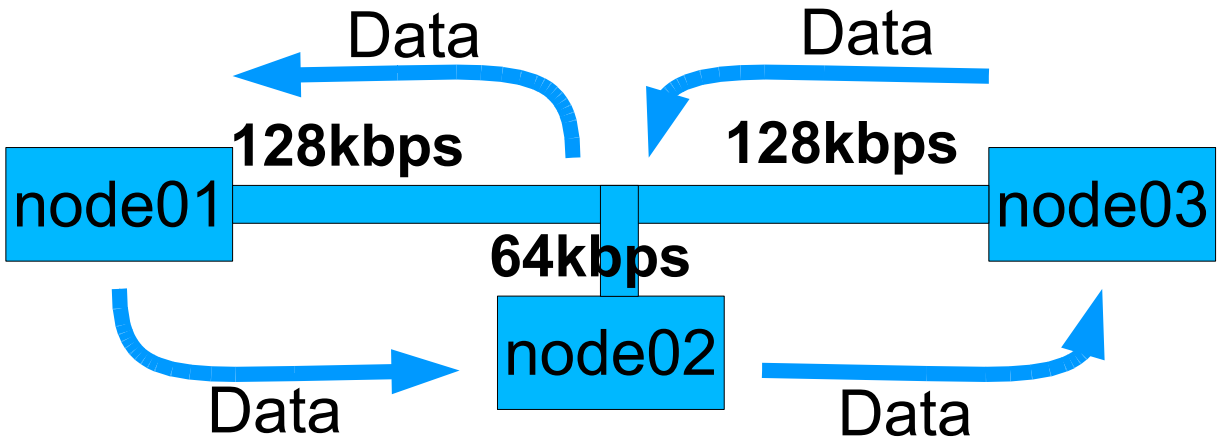


Results (64kbps links)

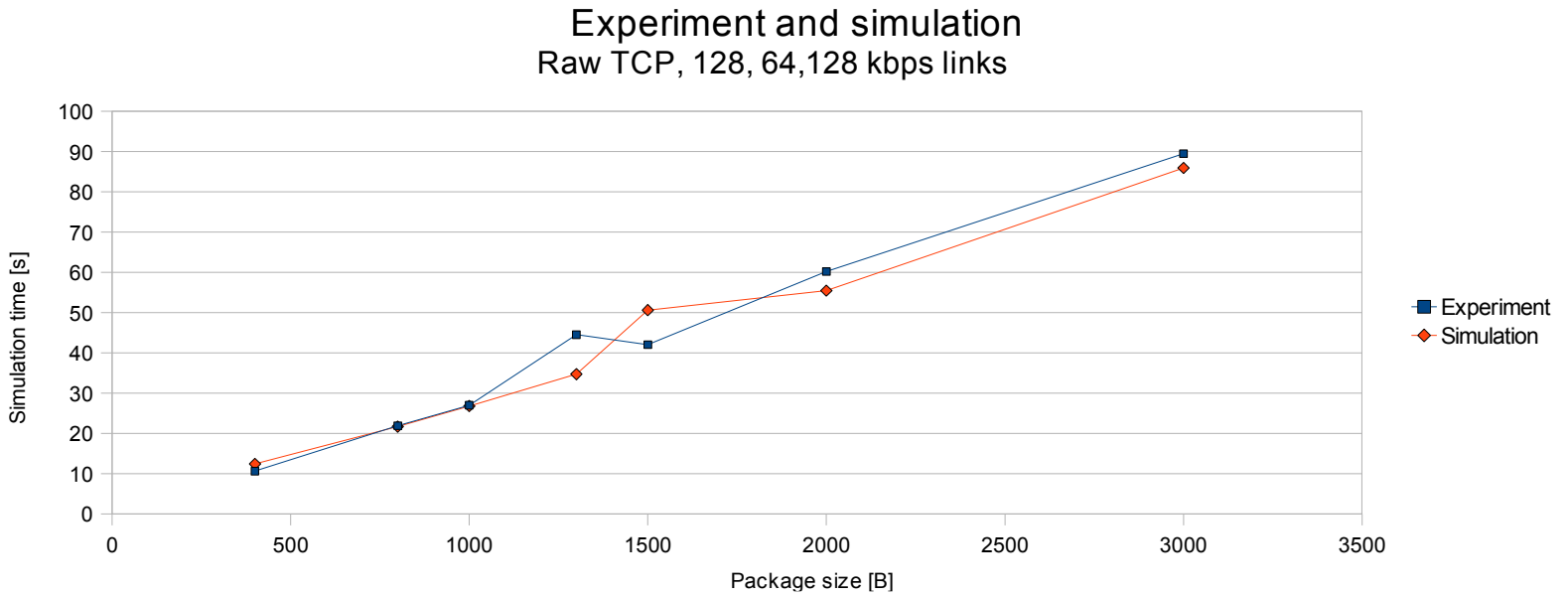


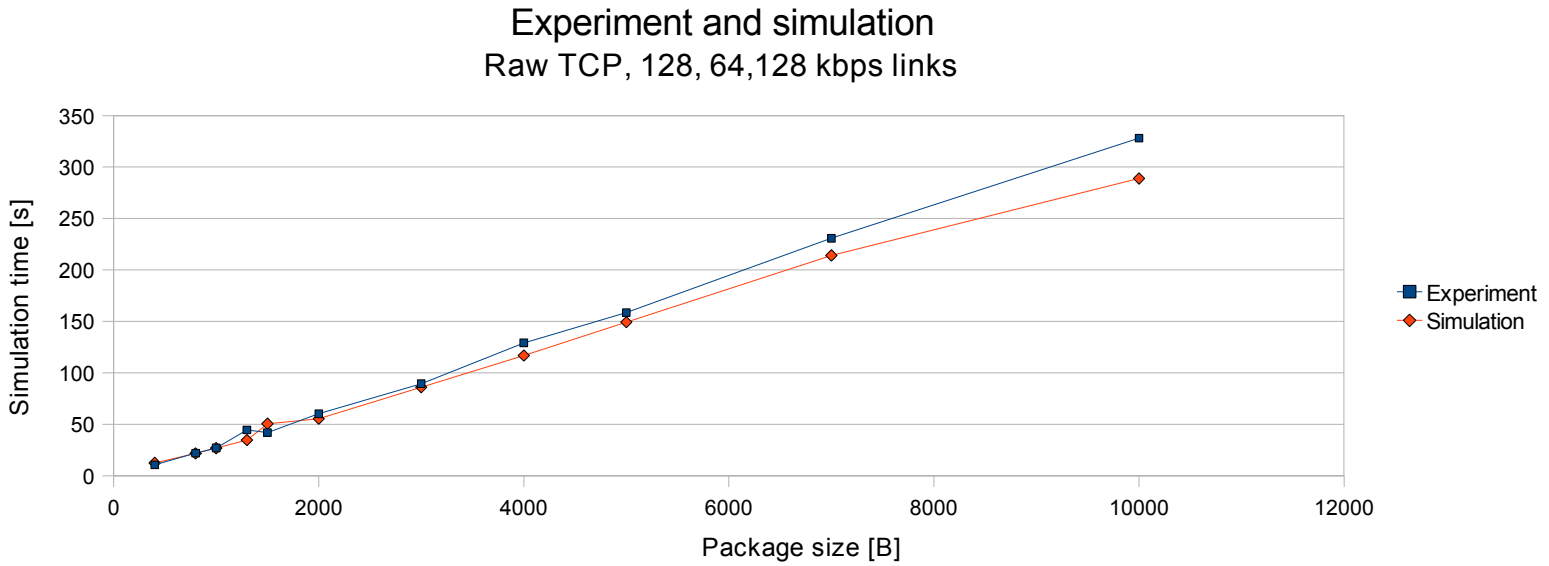


128 and 64kbps links

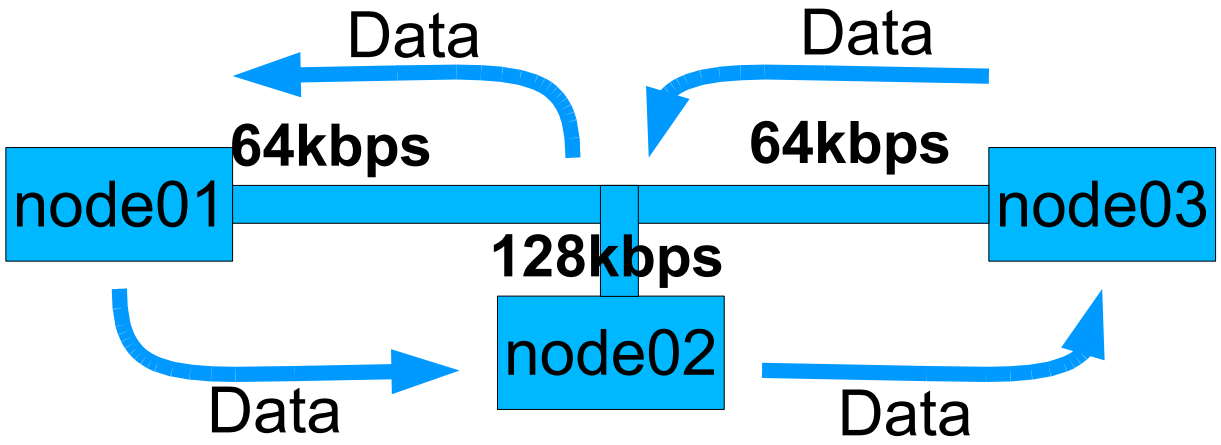


Results (128 and 64kbps links)

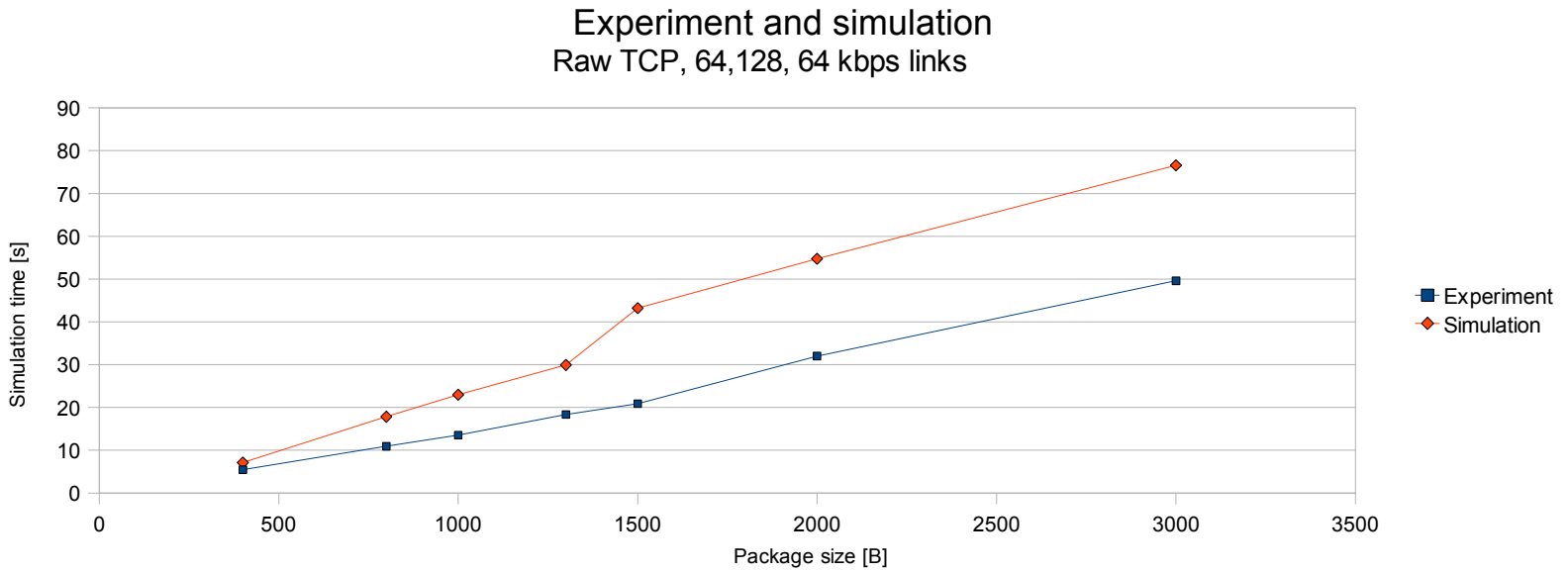


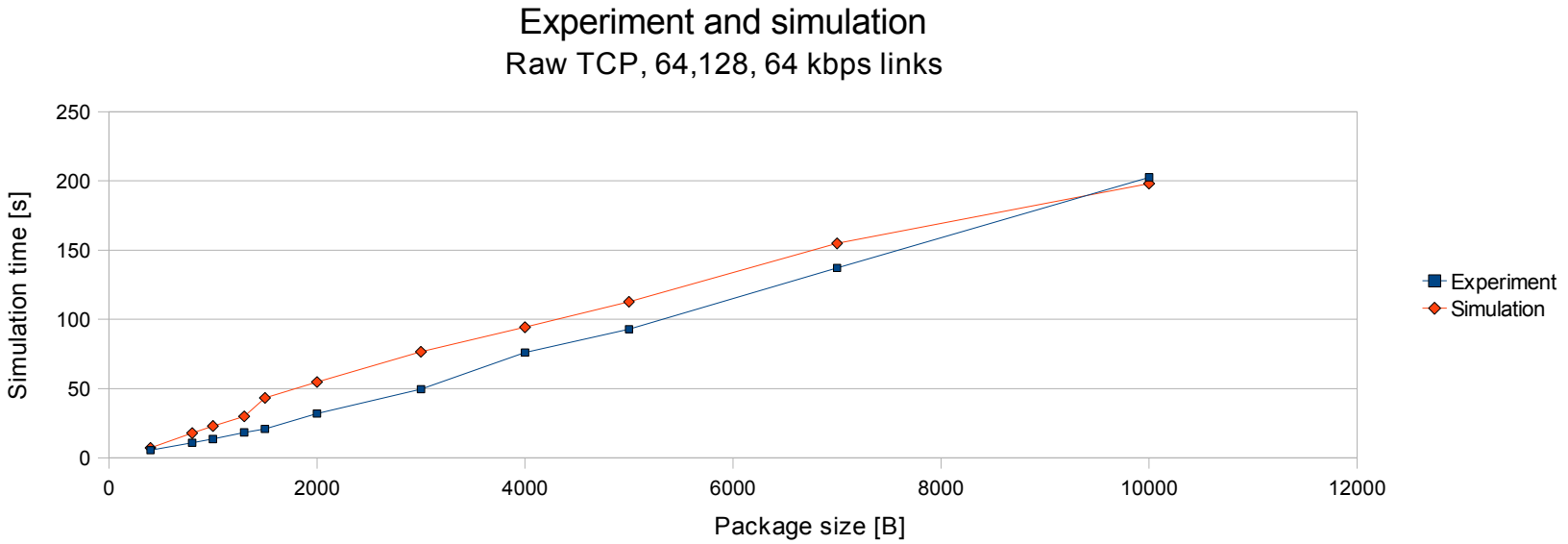


64 and 128kbps links



Results (64 and 128kbps links)





Conclusions and future work

- Conclusions
  - Correct results if TCP model works correctly
  - The Petri net based model works correctly
  - Inaccuracy caused mostly by incorrect TCP flow control implementation
- Future work
  - Correct TCP model
  - Improve High-level application model
  - Feasibility study – ATLAS TDAQ soft real time system [Korcyl, Szymocha, Kitowski, et al. 2008]