Simulation of a Routing Algorithm Using Distributed Simulation Techniques

C.D. Pham J. Essmeyer S. Fdida

Laboratoire LIP6
Université Pierre et Marie Curie
4 place Jussieu 75252 Paris Cedex 05

Friday 29th August, 1997
Euro-Par'97

Laboratoire LIP6

Goals

Performance Evaluation

- Analytical methods are too limited
- Simulations are very time-consuming

Reduce the simulation time

- Parallel and Distributed Simulations

ATM Network Models

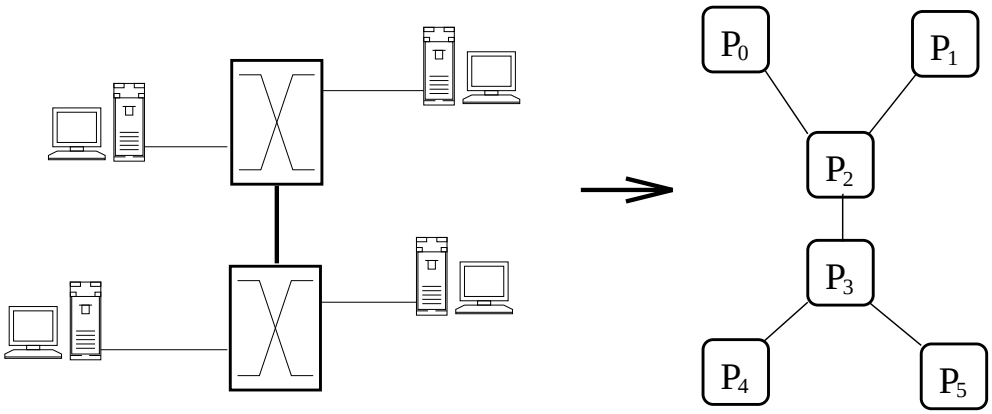
- Hot problem
- Cooperation with the Dret

Our contribution

- Distributed Simulator for ATM Models
- Simulation of Large and Complex Models

Parallel Simulation and The Process Synchronization Problem

Spatial partitioning and Logical Process (LP)



Distribution of event lists and virtual clocks.

⬇

Causality constraints and time errors.

⬇

A synchronization algorithm is needed to respect causality.

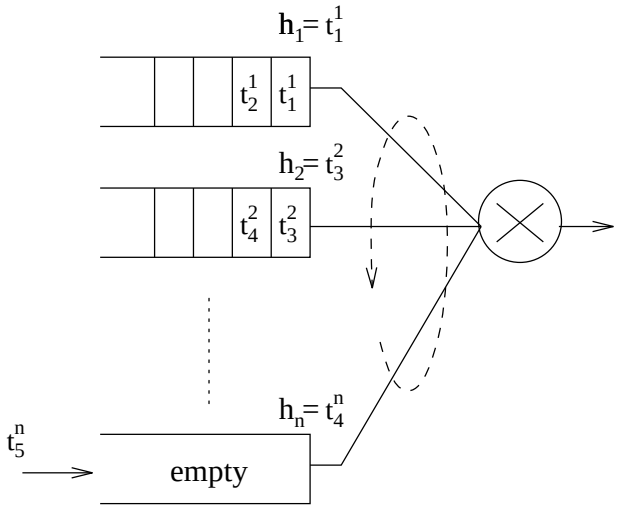
⬇

These synchronization mechanisms fall in two categories: **conservative** and **optimistic**.

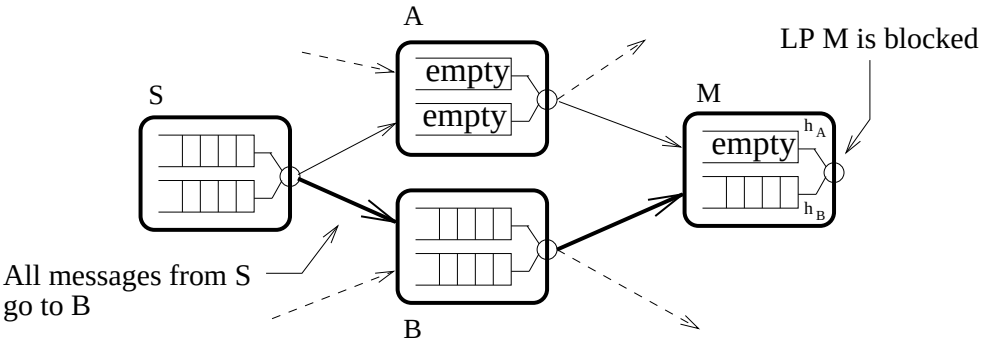
Conservative Approaches

Conservative algorithms (Chandy-Misra-Bryant) avoid all cases of time errors by blocking LPs.

The basic conservative algorithm

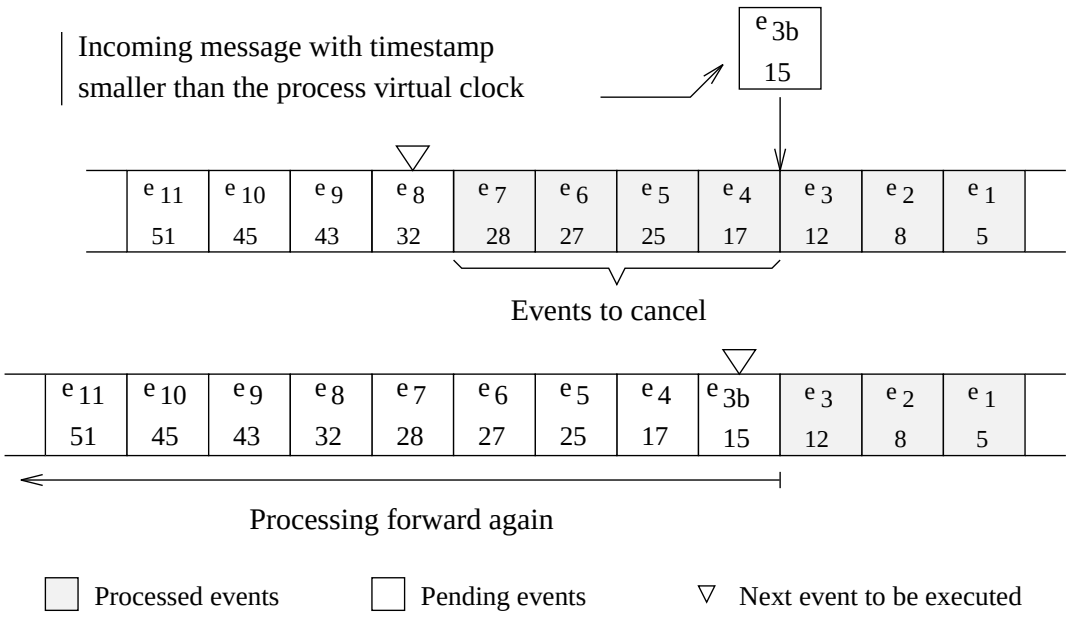


Deadlock → null-messages and lookahead



Optimistic Approaches

Optimistic algorithms (Time Warp) implement a rollback mechanism.



Need of periodic state saving and anti-messages to:

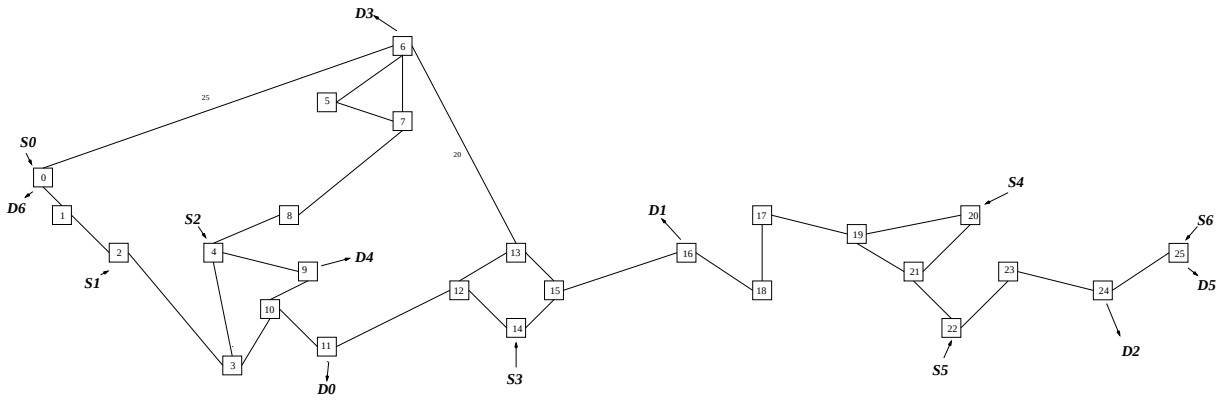
- Restore the process state,
- Send anti-messages to other processes,
- Re-execute events in the right order.

Network model

Complex network with routing algorithm

Simulation at the cell level

Connection management and VPI/VCI commutation



Network model with 32 switches

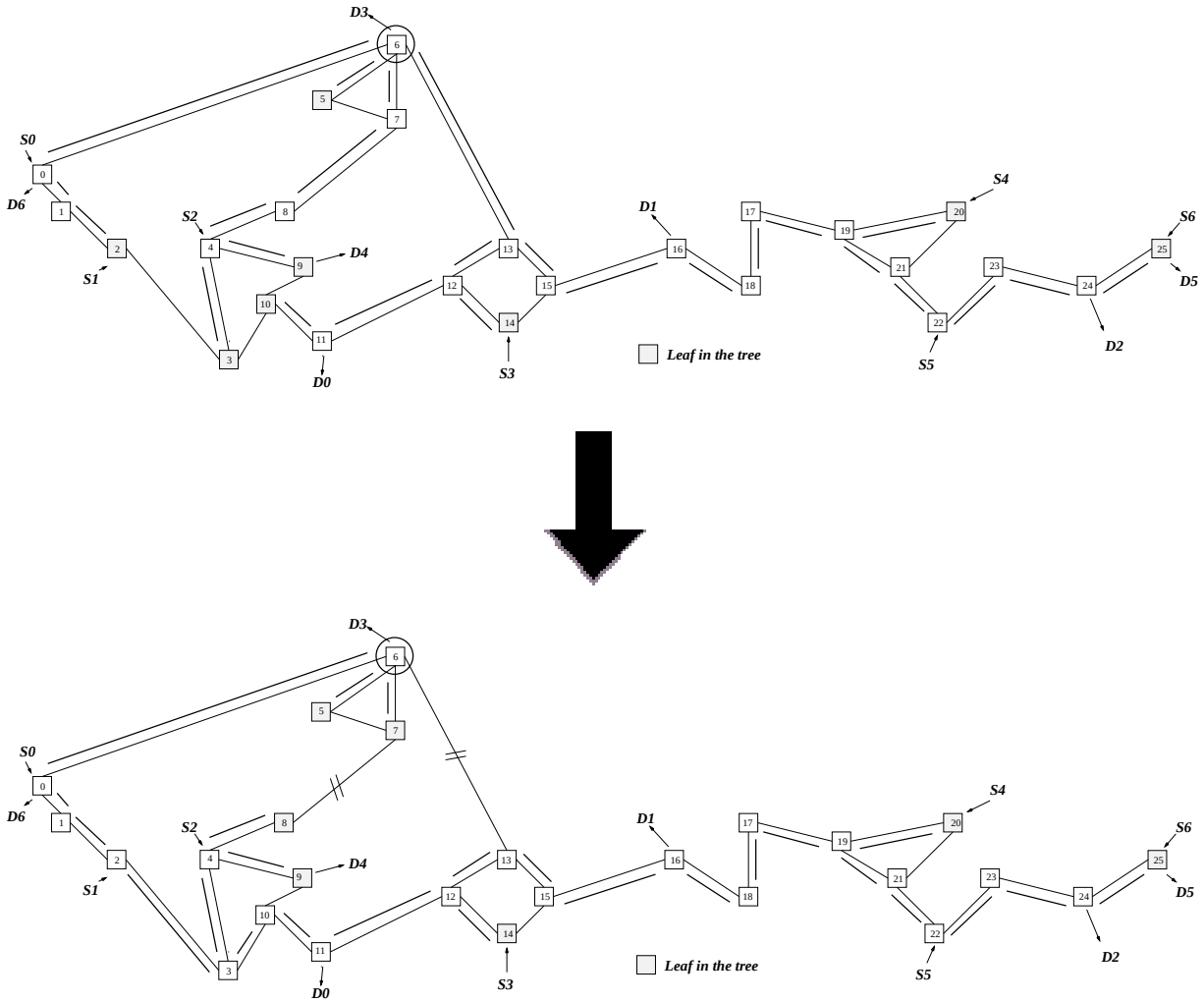
- ☞ Loops
- ☞ Load imbalance

Simulation of a Routing Algorithm

The Jaffe and Moss algorithm

Vector-Distance Protocol

Construction and Update of Routing Tables



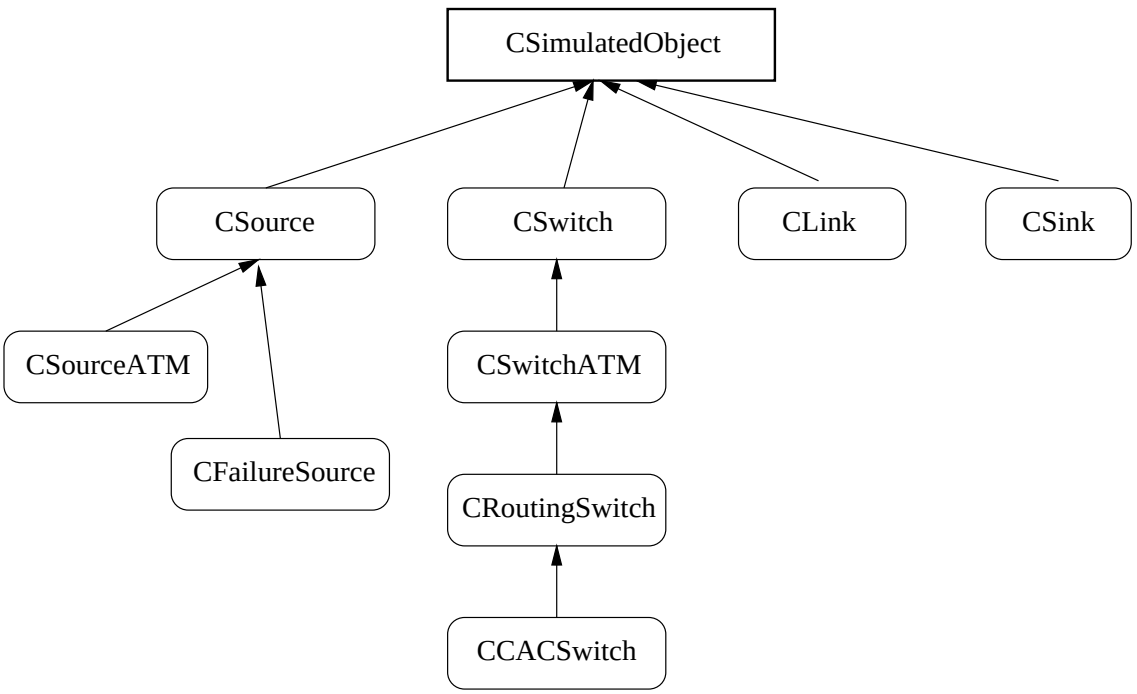
The Distributed Simulator

Unified Framework for Sequential and Distributed Simulations

Object-Oriented Design, Written in C++

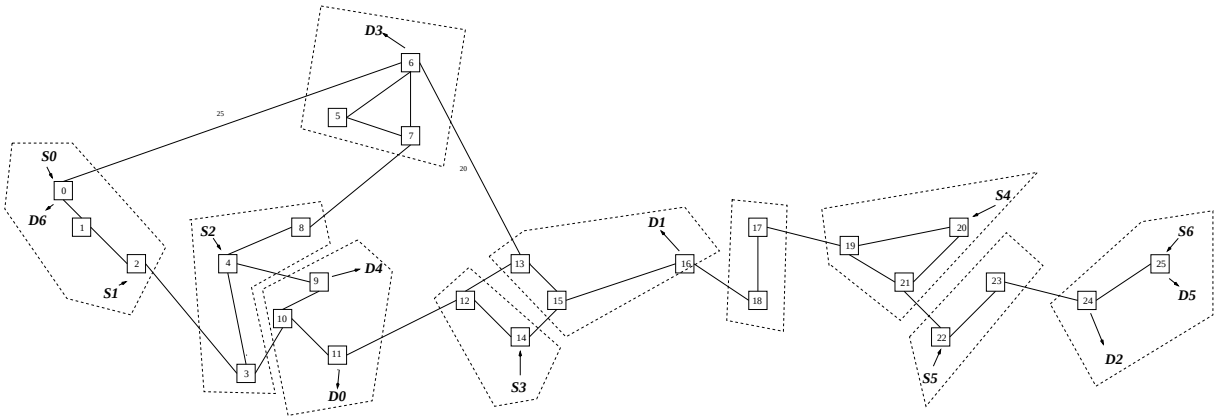
Conservative Synchronization

Tested on Sun Workstation, Intel Pentium, CM-5, Cray T3E

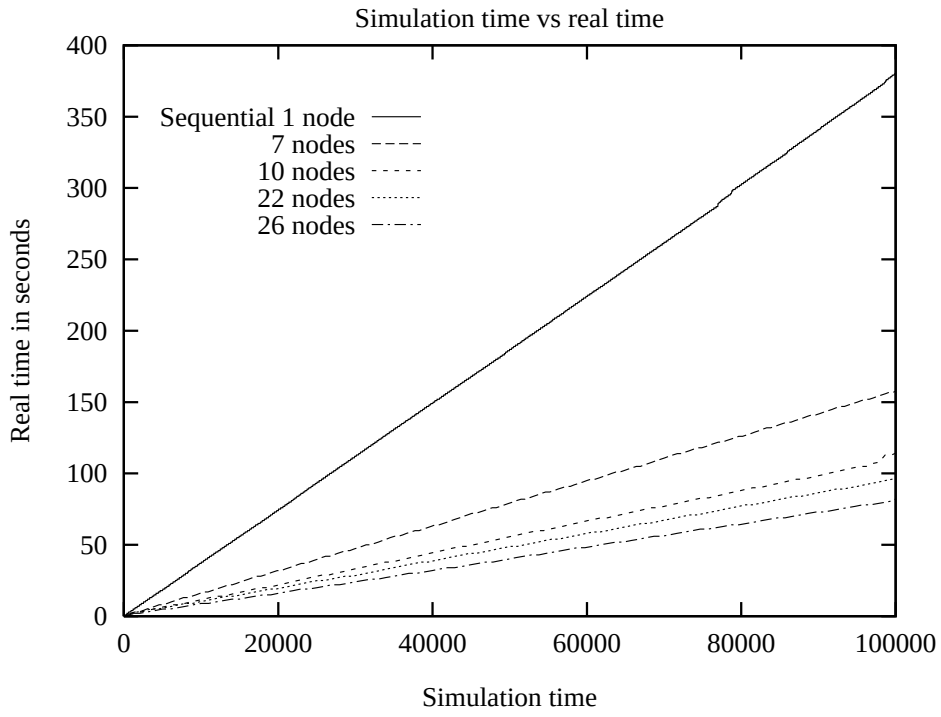


Simulation time

7, 10, 22 and 26 processor mapping



10 processor mapping



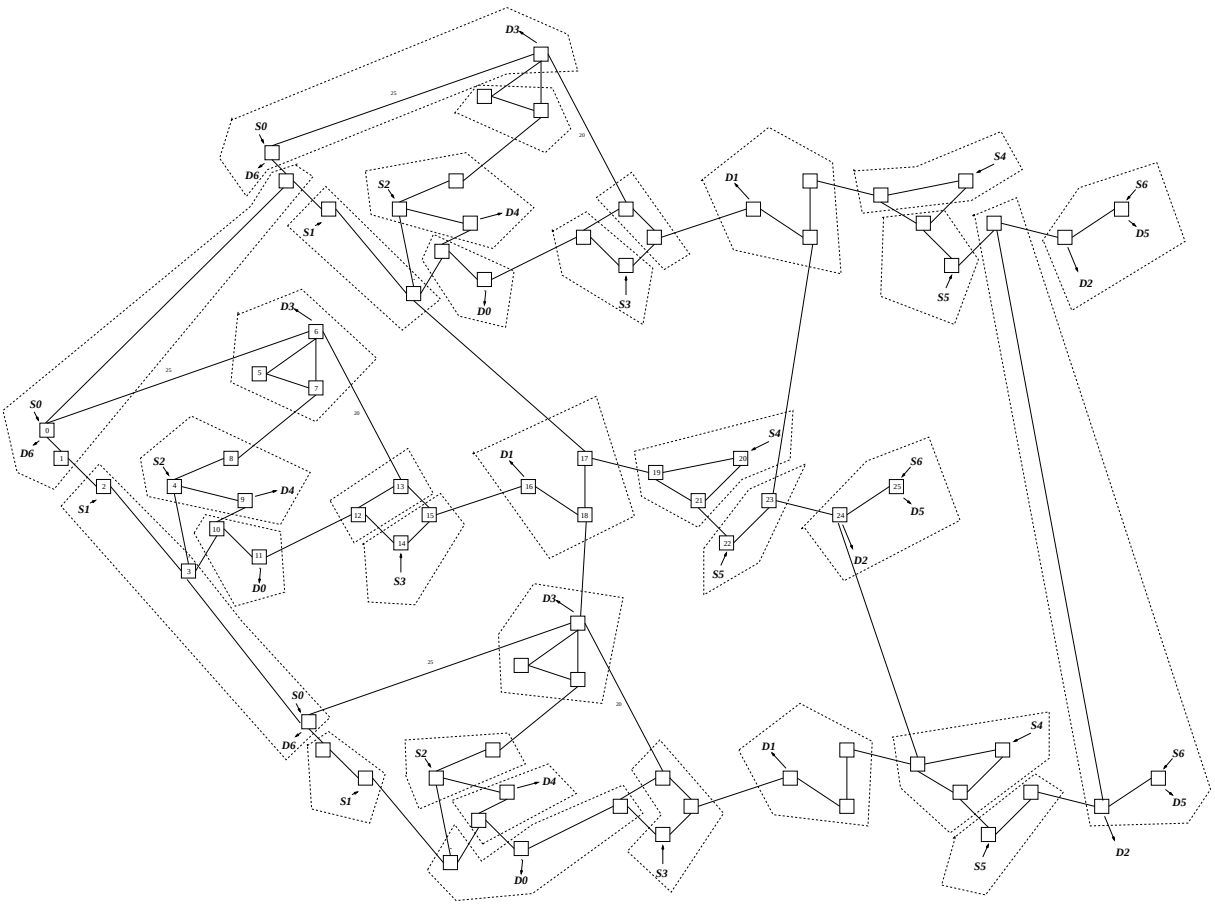
Speedup

Speedup and Efficiency

nproc	*real time*	*speedup*	*efficiency*
1	380	1.0	1.0
7	158	2.41	0.34
10	115	3.30	0.33
22	97	3.92	0.18
26	84	4.52	0.26

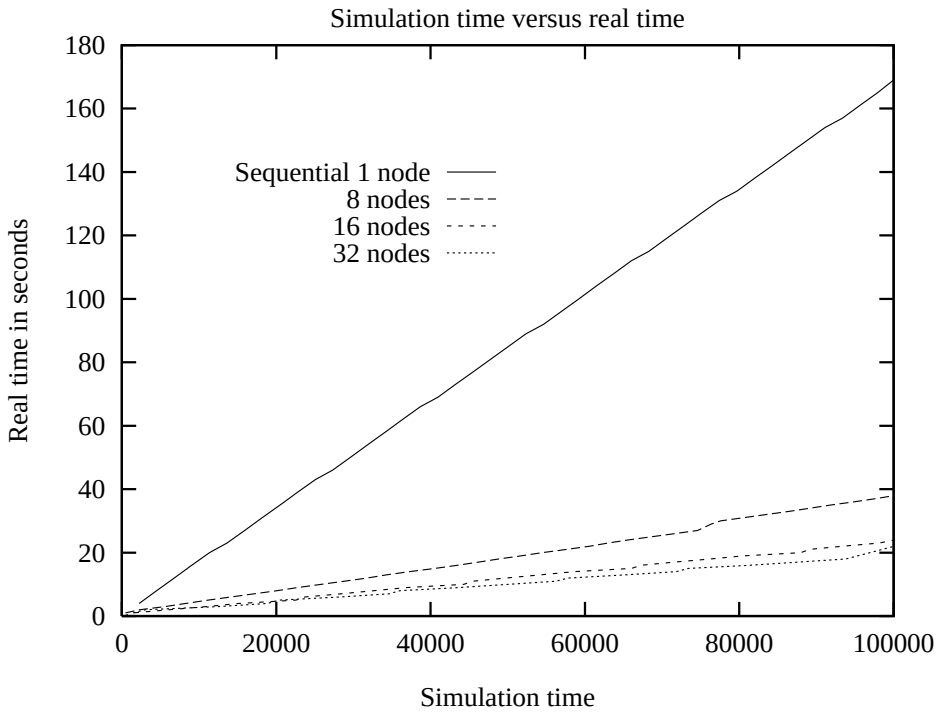
Increasing the network size

Model with 78 switches



32 processor mapping.

Speedup



nproc	*Real time*	*speedup*	*efficiency*
1	169	_	1.0
8	38	4.44	0.55
16	24	7.04	0.44
32	22	7.68	0.24

Speedup and Efficiency

Conclusion and Perspectives

- ✓ Distributed simulation techniques can successfully be applied to simulate large and complex models.
- ✓ A speedup of 7.68 can be achieved with our distributed simulator.
- ✗ Implementation of more models.
- ✗ Load balancing features.