High Performance Clusters: A Promising Environment for Parallel Discrete Event Simulation

C.D. Pham

RHDAC
Université Claude Bernard, Lyon 1
43 bld du 11 novembre 1918
69622 Villeurbanne cedex
France

Wednesday, June 30th, 1999

PDPTA'99
Las Vegas, USA

RHDAC

Some complex problems

- Parallel programs, large and complex code
  - Performances, Correctness,
  - Verification, Scalability.
- Mobile Communication, thousands of mobiles
  - Handoff probability,
  - Frequency assignment.
- ATM networks, hundreds of nodes
  - Buffer size,
  - Routing overhead,
  - End-to-End delays.
- And much more...

Performance Evaluation

How to handle these large systems?

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

Reduce the simulation time

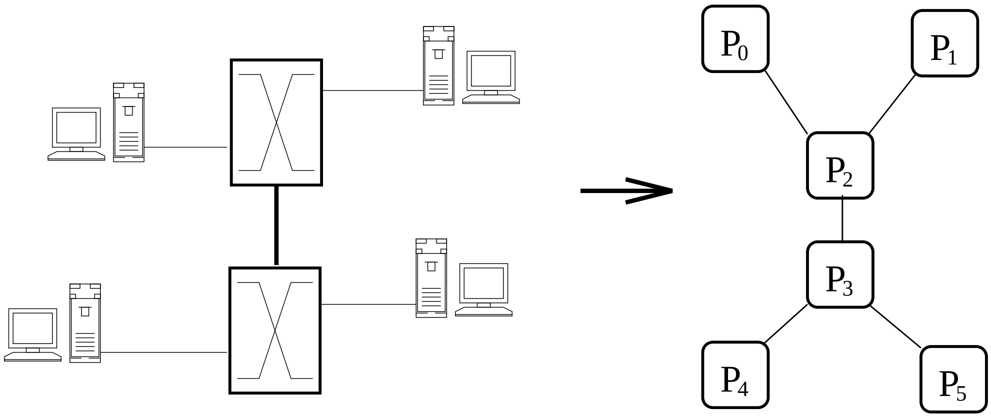
- Faster sequential computer,
- Hybrid simulation,
- Parallel architectures.

Parallel and Distributed Simulation

- On parallel computers, networks of workstations, clusters of workstations,
- Focus mainly on synchronization problems.

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.

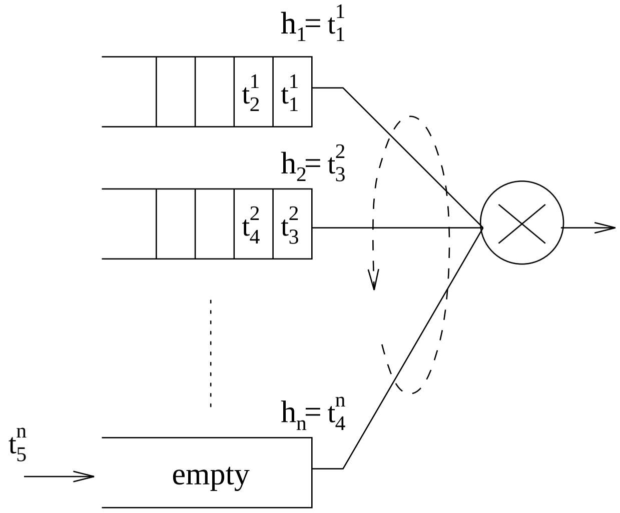
⬇

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

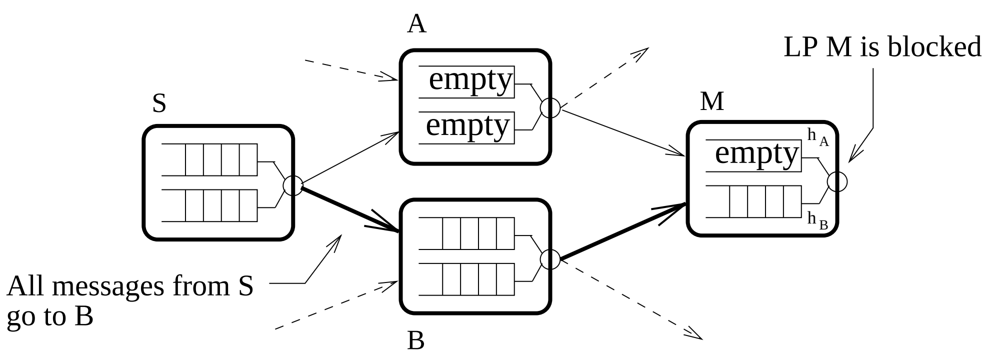
Conservative Approaches

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

The basic conservative algorithm

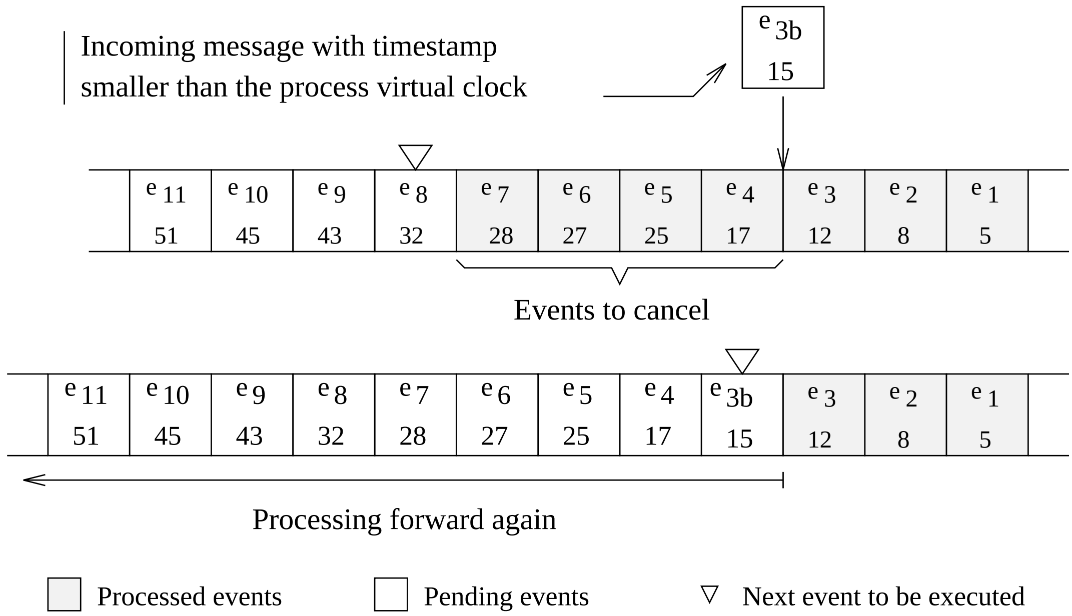


Deadlock → null-messages and lookahead



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.

Migration to Cluster Computing

Parallel simulations have traditionally been performed on parallel computers to keep the communication overhead low

* * *

Network technology for clusters of workstations have dramatically improved making clusters very efficient with a high performance/cost ratio

⬇

Great opportunies for parallel simulation to become more accessible to the end user

⬇

More people will use parallel simulation

⬇

We will be very happy !

Cluster technologies

Fast Ethernet, ATM

- Disapointing performances.

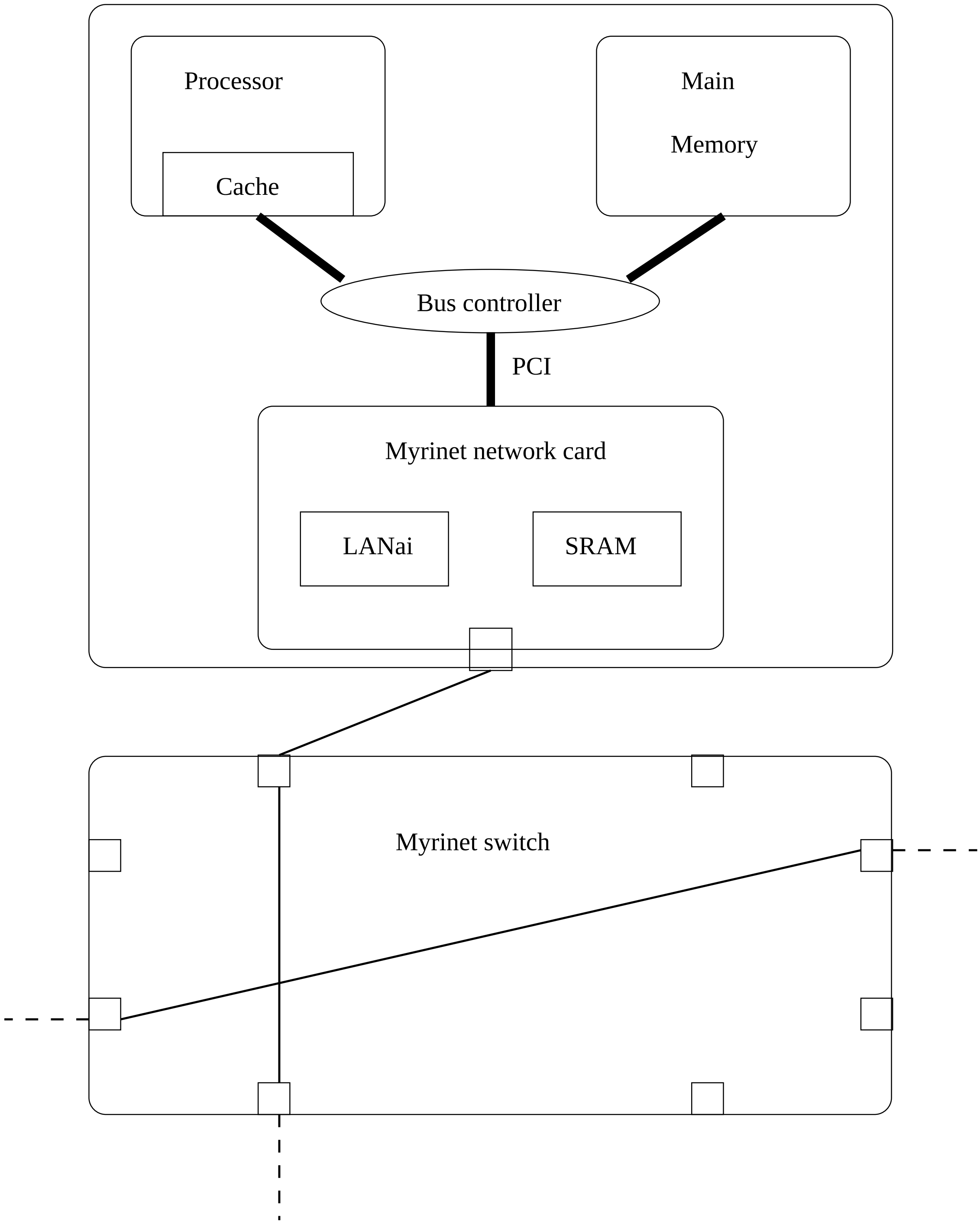
Myrinet

- Promising performances.

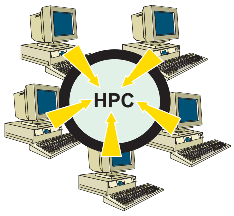
Myrinet's main features

- Gigabits/s rate, Very low latencies,
- Hardware end-to-end flow control that guarantees a reliable delivery. Message losses are exceptional.
- Embedded processor for an efficient overlapping of communication/computation,
- Programmable processor for custom communication strategies,
- Memory up to 1 MBytes on the card to isolate transfers on the network from transfers on the main bus.

The myrinet technology



The Cluster test-bed



MYRINET-BASED CLUSTER OF 12 PCS

PENTIUM PRO 200MHZ, 64 MBYTES RAM MEMORY, 440FX CHIPSETS

LANAI 4.1 AND 256KBYTES OF MEMORY

LINUX OPERATING SYSTEM

BIP and MPI-BIP library

LOW OVERHEAD, MINIMIZES MEMORY COPIES

HIGH-PERFORMANCE COMMUNICATION

SMALL AND LONG MESSAGES STRATEGIES

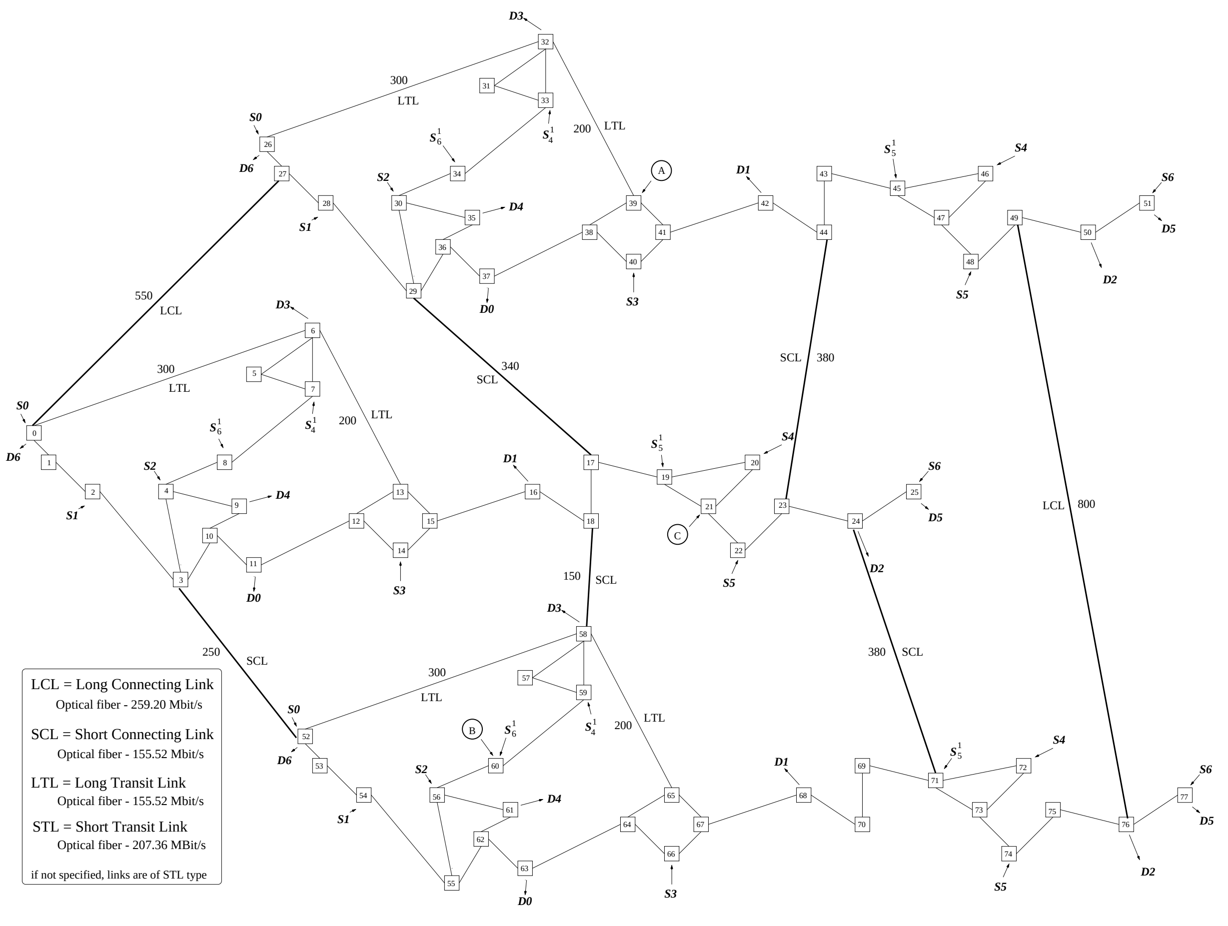
$5\mu s$ OF LATENCY FOR BIP, $11\mu s$ FOR MPI-BIP

The ATM network model

78-switch network with routing facilities

Routing algorithm with Dynamic link costs

Connection management and Admission control



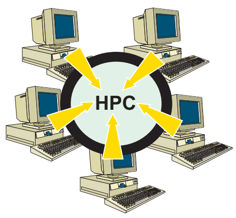
Model's properties

- **Load imbalance between links**
  - a large number of null-messages have to be sent.
- **Bottlenecks in the simulation model**
  - some processors can be overloaded.
- **Complex load-balance patterns**
  - partitioning and load-balancing are difficult.
- **Very small granularity**
  - high level of message synchronization,
  - very small computation/communication ratio

Experimentations

Cray T3E with 256 processors
SHMEM and MPI library

and



Myrinet-based cluster of 12 Pentium Pro
MPI-BIP library

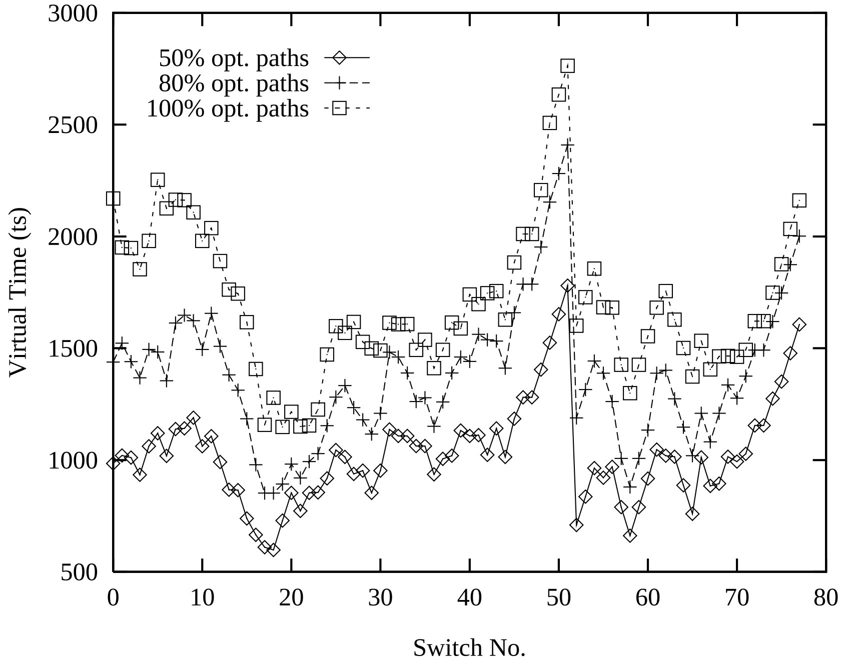
We simulate 500,000 time units

that represent 0.31s of the real system

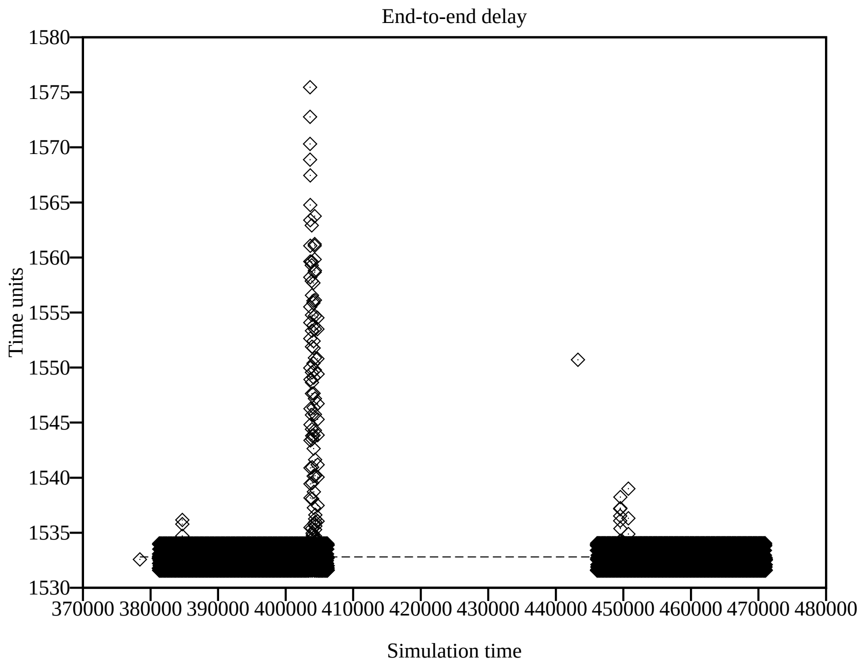
↓

more than 53 millions of events simulated

Performance of the model



Time to find optimal paths



End-to-end delays

Performance of the simulator (1)

Table 1: Cray T3E with SHMEM and MPI.

np.	*time SHMEM (s)*	*speedup SHMEM*
1	488	-
4	131	3.72
8	72	6.77
16	46	10.60
32	33	14.78
64	31	15.74

np.	*time MPI (s)*	*speedup MPI*
1	488	-
4	253	2.07
8	133	3.66
16	82	5.95
32	59	8.27
64	55	8.87

- SHMEM yields very good performances.

Performance of the simulator (2)

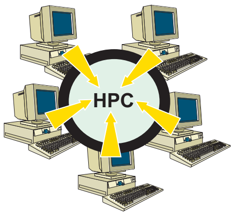


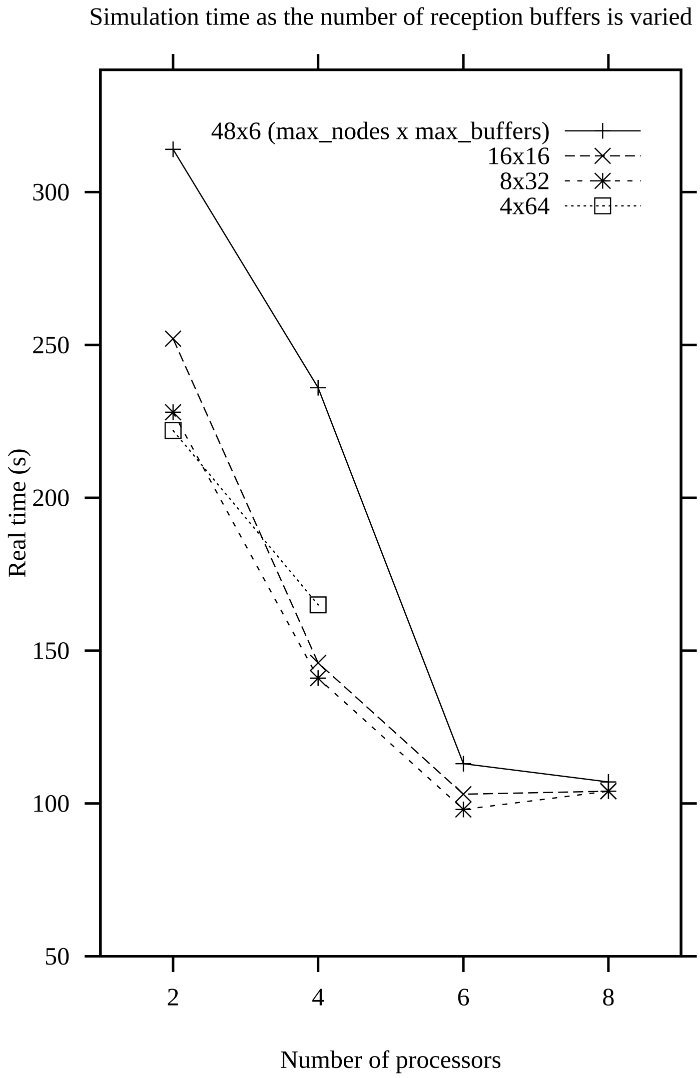
Table 2: Myrinet with MPI/BIP.

np.	*time*	*speedup*	*RM*
1	333	-	0
2	228	1.46	710,905
4	141	2.36	2,304,835
6	98	3.39	2,845,867
8	104	3.20	3,418,919

- With MPI, the speedup results are very interesting and close to what was obtained on a Cray.

Scalability issues

Larger problem size has shown a deadlock problem



- Having more buffers is very beneficial on 2 and 4 processors.

Conclusions

- HPCs represent a serious alternative to expensive parallel computers.
- Fine grain applications like parallel simulation of communication networks can directly benefit from the low latency provided by the new generation of high-speed interconnection networks.
- We found that HPCs can yield very close performance results compared to previous experiments on a Cray T3E with a similar high-level communication software.
- We are currently porting the application on native BIP to fully exploit the hardware asynchronism and to benefit from a lower latency.