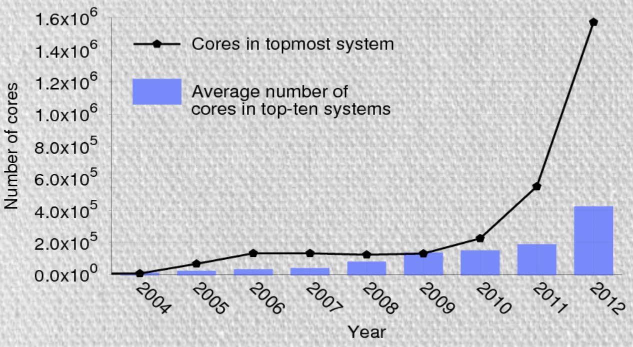
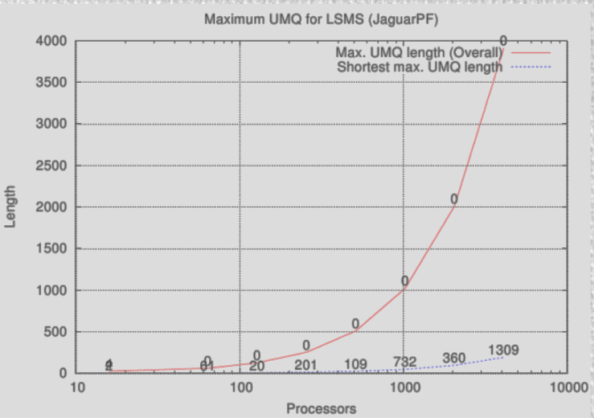
SIGNIFICANTLY REDUCING MPI INTERCOMMUNICATION LATENCY AND POWER OVERHEAD IN BOTH EMBEDDED AND HPC SYSTEMS

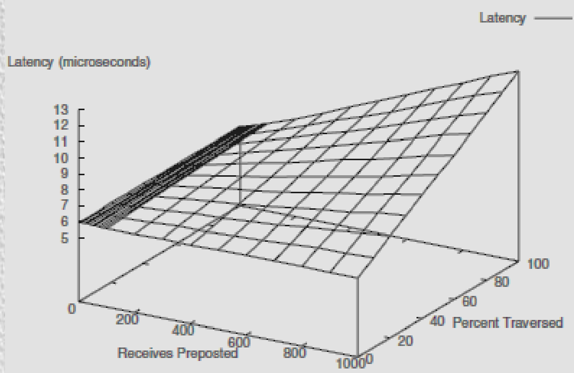
Pavlos M. Mattheakis

Ioannis Papaefstathiou

Motivation

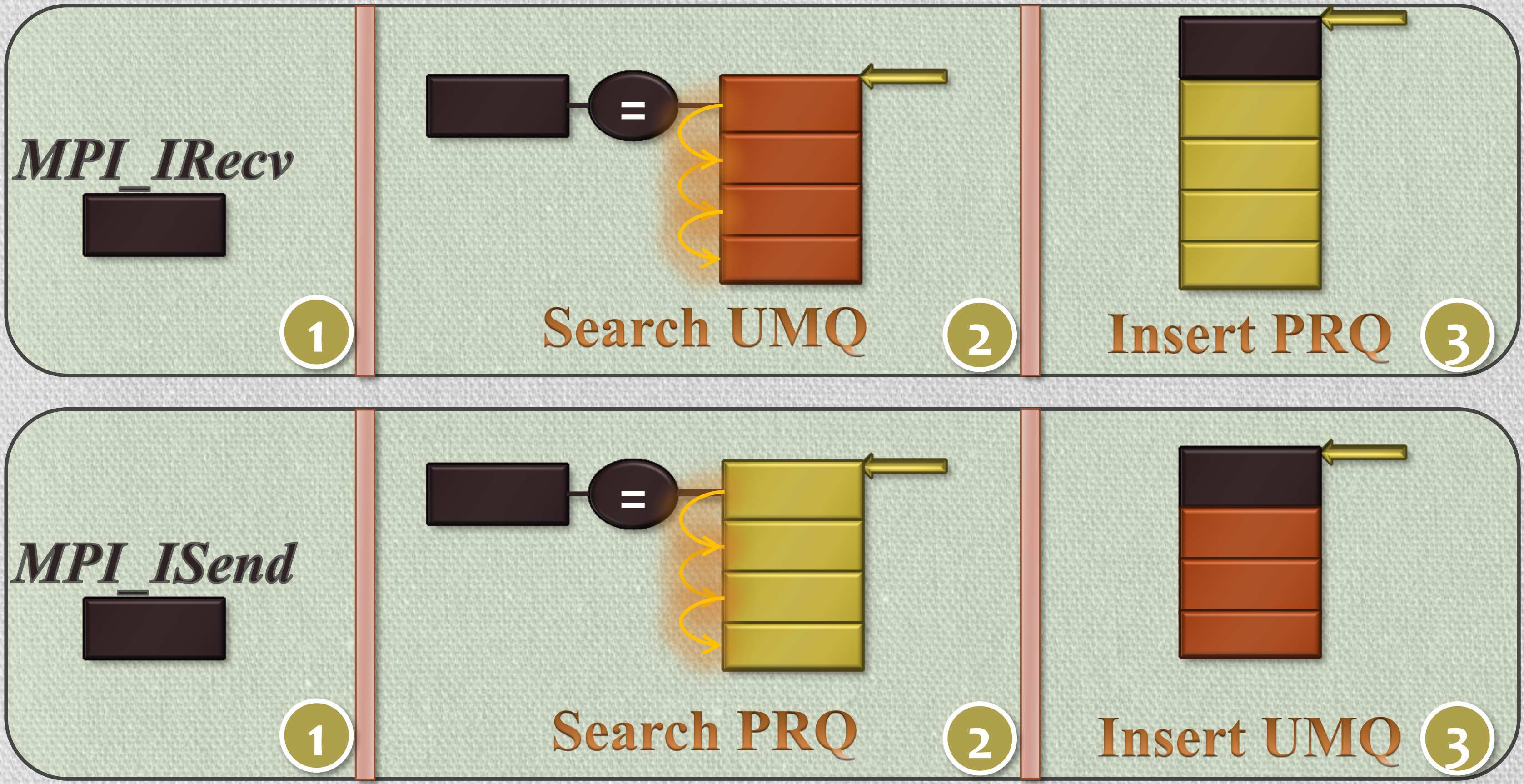






- **Number Of Nodes in HPC Systems Increases Exponentially**
  - *IBM 2012*
- **Asynchronous MPI Messages increase Linearly with the Nodes.**
  - *Rainer Keller et al. : "Characteristics of the Unexpected Message Queue of MPI Applications“, EuroMPI 2010*
- **MPI Protocol Implementations do not scale**
  - *Keith D. Underwood et al. "The Impact of MPI Queue Usage on Message Latency", ICPP 2004*

Background: MPI Asynchronous Commands



Scenario: A node buffers N messages in UMQ. Then executes N receives crossing the whole UMQ each time:

- N(N+1)/2 MPI Message Comparisons

Previous Work

- **Offloaded MPI Stack on the NIC Processor**
  - Myricom Lanai Z8ES, 2009
  - Quadrics Elan4, 2002
- **List Manager Accelerator**
  - Accelerating List Management for MPI, CLUSTER 2005
  - An MPI Implementation for Multiple Processors Across Multiple FPGAs, FPL 2006
- **TCAM-Based Accelerator**
  - A Hardware Acceleration Unit for MPI Queue Processing, *IPDPS* 2005
- **Microcoded Architecture**
  - An architecture to perform NIC based MPI matching, CLUSTER 2007

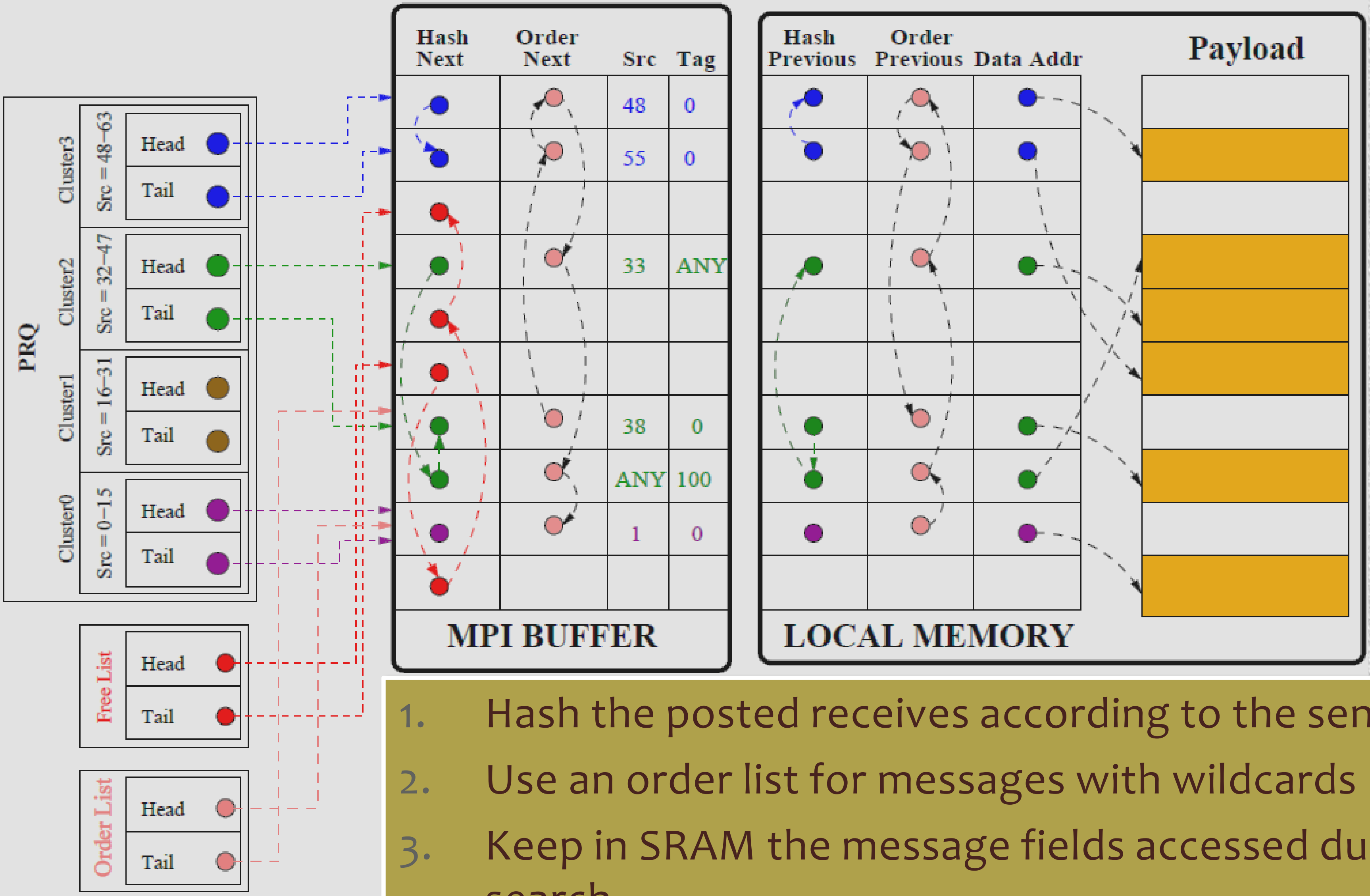
Contributions

- Novel Scheme for Processing Asynchronous MPI Messages

- Profiled with real world applications

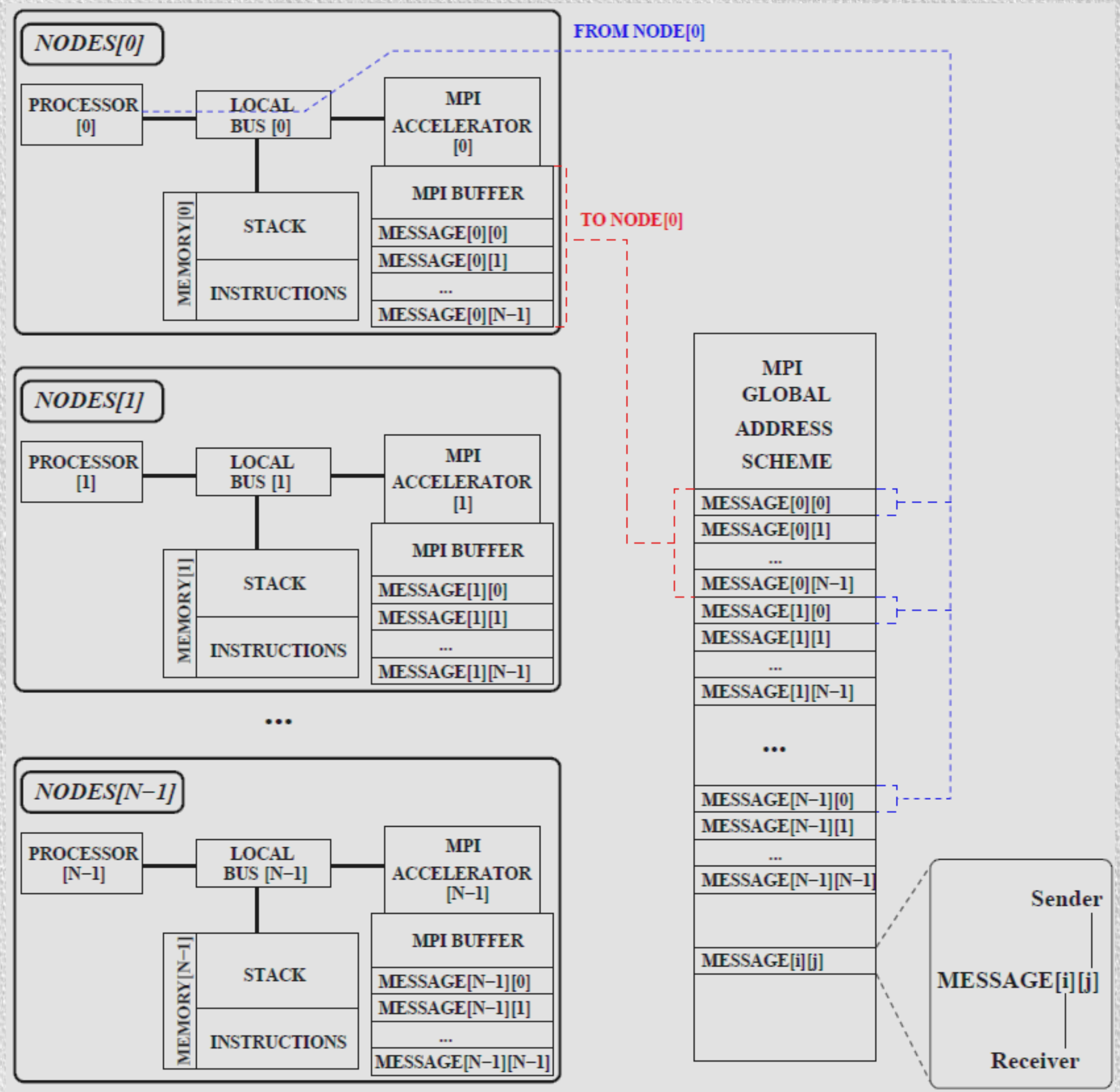
- Implementation in a state-of-the-art CMOS technology

Novel Scheme



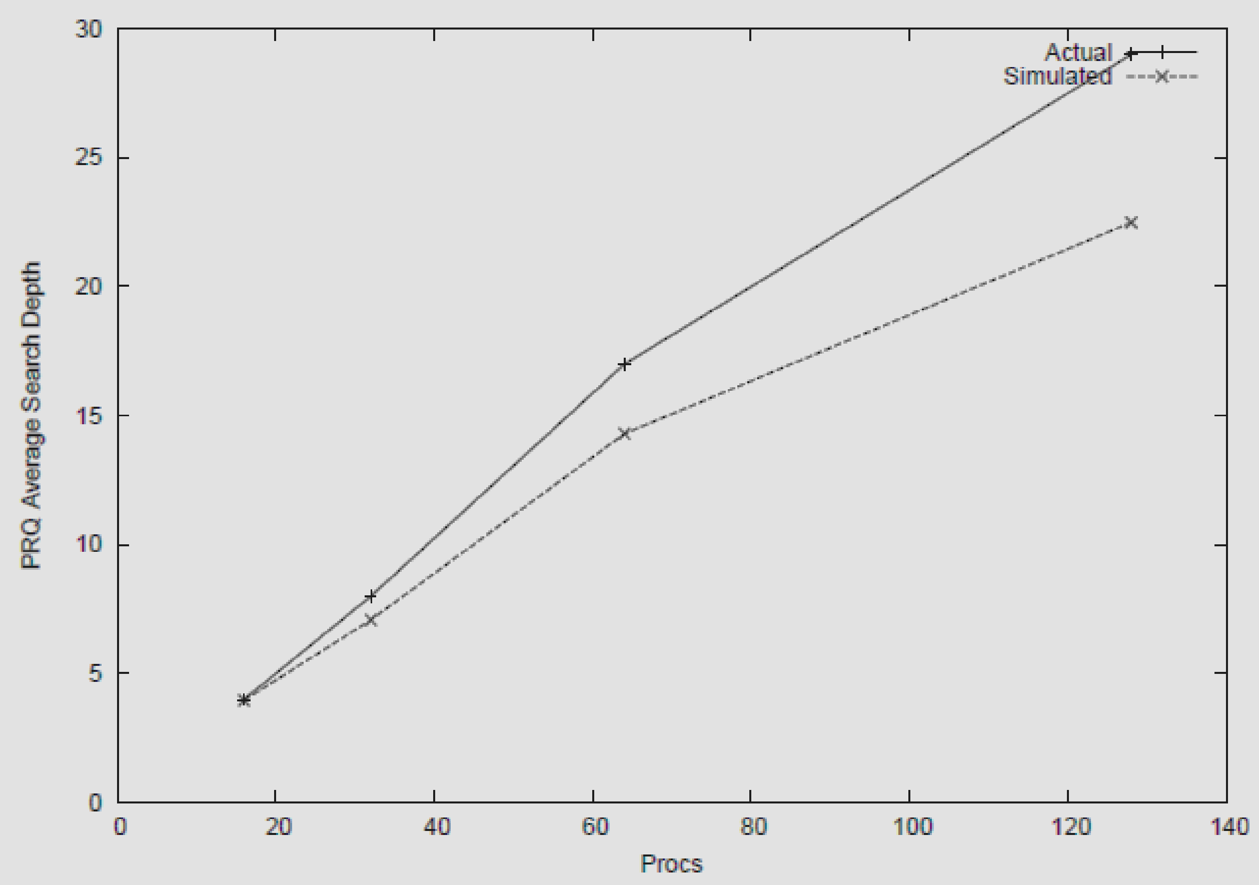
1. Hash the posted receives according to the sender
2. Use an order list for messages with wildcards
3. Keep in SRAM the message fields accessed during search

Profiling with real world applications



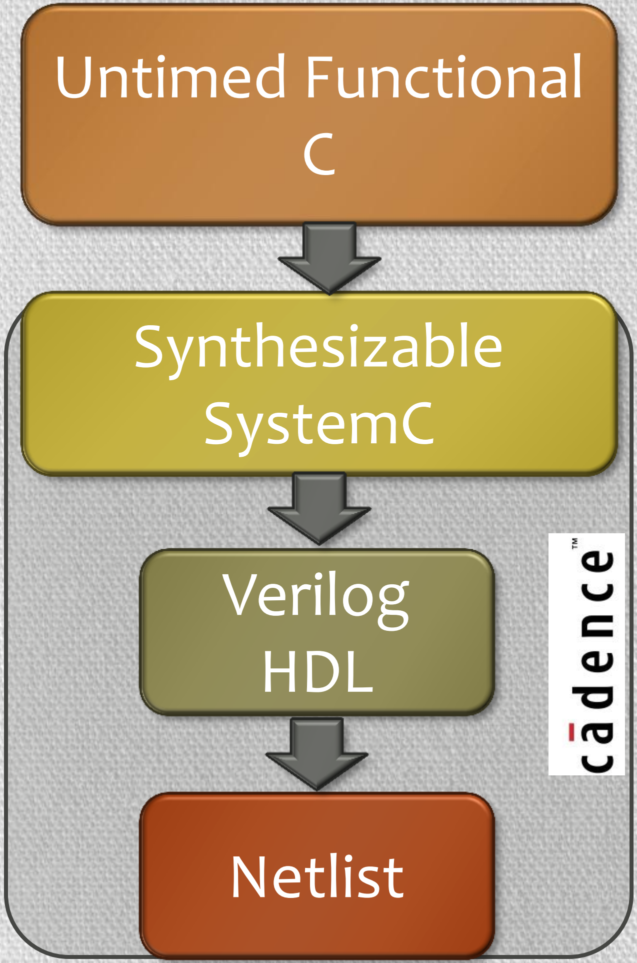
- Imperas OVPsim Software Used
- OR1K Processors Running a lightweight MPI implementation
- MPI Accelerator modeled in C

Simulation Accuracy



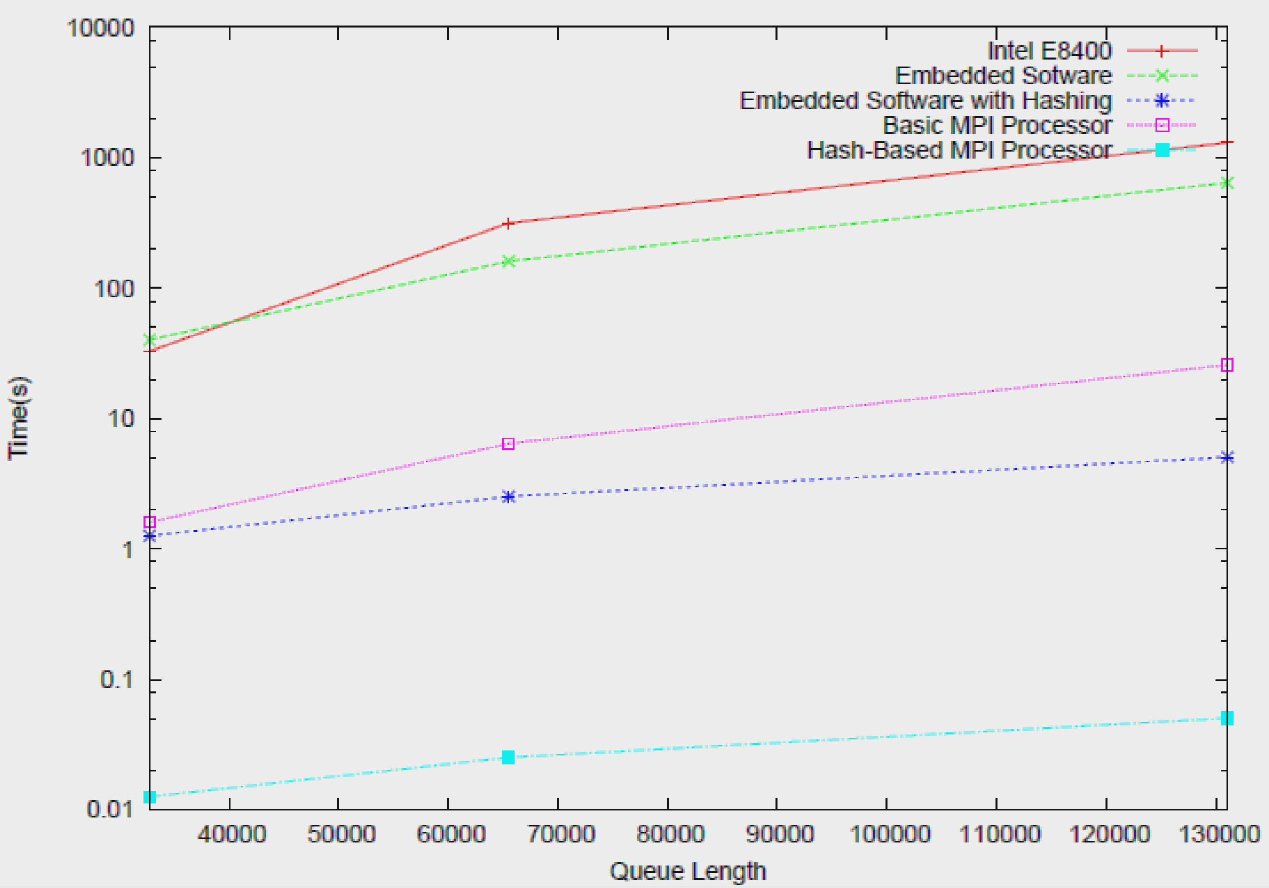
- Average Search Depth of PRQ for the IS benchmark used to evaluate the accuracy of simulation

Implementation Flow



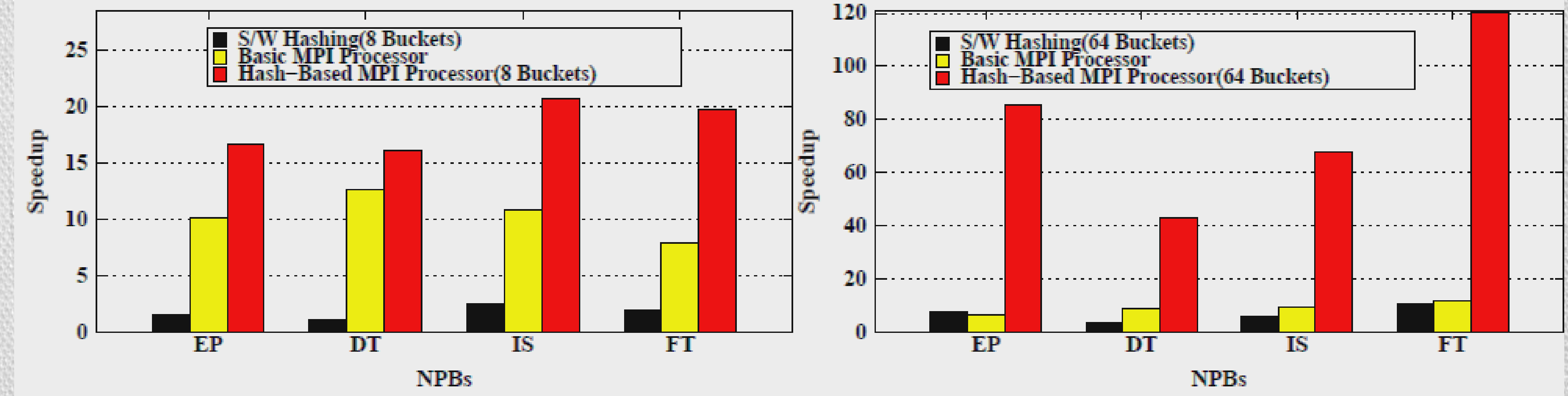
Module	Frequency	Area (um²)	Power (mW)
Accelerator	1GHz	95055	39.08

Results – Brightwell's Benchmark



- Benchmark introduced in "*Implications of application usage characteristics for collective communication offload*", *IJHPC 2006.*
  - *Send N messages from Node 1 to Node 0 with tags [0, N-1]. Post N receives at Node 0 with tags [0, N-1].*
  - *Measure the time needed to unload the UMQ at Node 0.*

Results – NASPB Suite



(a) 32 Processors

(b) 256 Processors

- *Queue processing speedup for MPI processor with hashing scales with the number of processors*
  - *Requires scaling the number of buckets as well*

Conclusions and Future Work

- ***Novel scheme for reducing the processing time in MPI queues***
  - Simulated in a Multiprocessor Environment
  - Synthesized in a state-of-the-art CMOS technology
  - Profiled on real world applications

- ***Future Directions***
  - Further measurements
    - Not only Benchmark-Style Applications
  - Integration of more MPI tasks

- ***Special Thanks to***
    - ***Dimitrios S. Nikolopoulos, Professor - Queen’s University of Belfast***
    - ***Nikos Taburatzis, Msc Student - Technical University of Crete***

Questions?