Algorithm Selection of MPI Collectives Considering System Utilization

Majid Salimi Beni[1], Sascha Hunold[2] and Biagio Cosenza[1]

[1]Department of Computer Science
University of Salerno, Salerno, Italy

[2]Faculty of Informatics, TU
Wien, Vienna, Austria

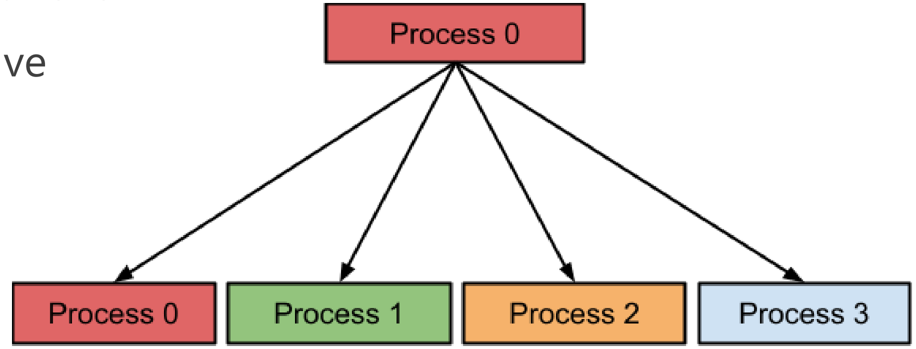
Euro-Par 2023 PhD Symposium
Limassol, Cyprus

Outline

- ❑ MPI Collectives
- ❑ MPI Collective Algorithm Selection
- ❑ Motivation
- ❑ Workload-Aware Algorithm Selection
- ❑ Summary and Future Work

MPI Collectives

- MPI (Message Passing Interface)
  - HPC programming standard
- MPI collectives
  - Time-consuming: Big share of HPC applications' runtime is spent while performing collective communications
  - Efficient implementation
- Collective algorithms
  - Distinct internal characteristics
  - Communication costs and scalability attributes
  - **Collective Algorithm Selection**



Process 0

Process 0

Process 1

Process 2

Process 3

MPI Collective Algorithm Selection

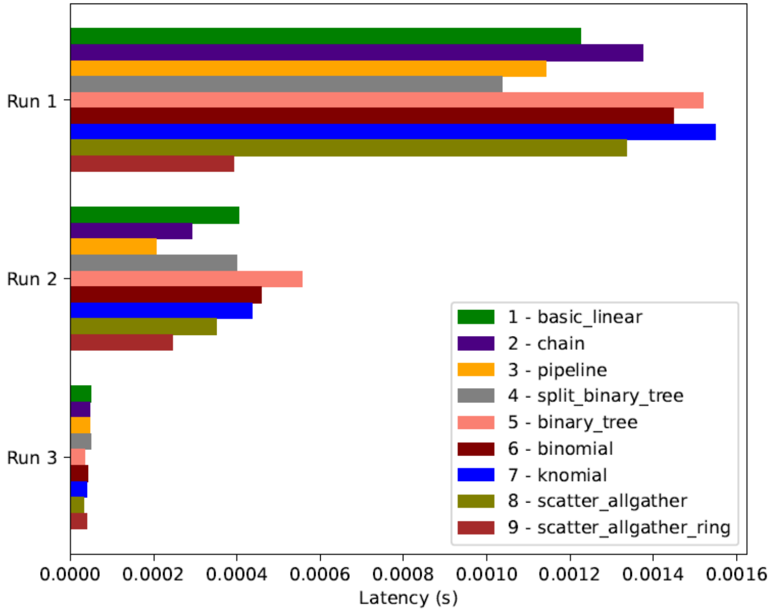
- ❑ Efficient algorithm selection
  - ❑ Optimal performance
  - ❑ Scalability
  - ❑ Communication overhead
  - ❑ Resource utilization
- ❑ Related works' considered parameters
  - ❑ Message size
  - ❑ Process count
  - ❑ Network topology
  - ❑ Available hardware resources

- ❑ Related works' selection approaches
  - ❑ Online/Offline
  - ❑ Machine Learning
  - ❑ Modelling-based

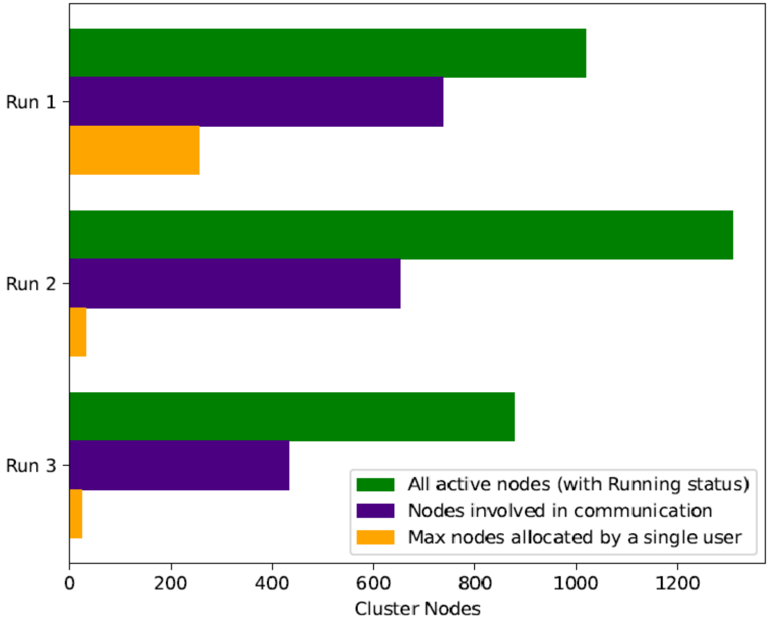
Cluster Utilization and network congestion are not considered in related work!

Motivation

- Large-scale clusters are utilized by many users at the same time
- Collective algorithms may behave differently under heavy network traffic
- Performance Variability
- **Ignoring the cluster utilization** can lead to a non-optimal algorithm selection



Latency of 3 runs of MPI_Bcast algorithms (OMPI) on 512 processes

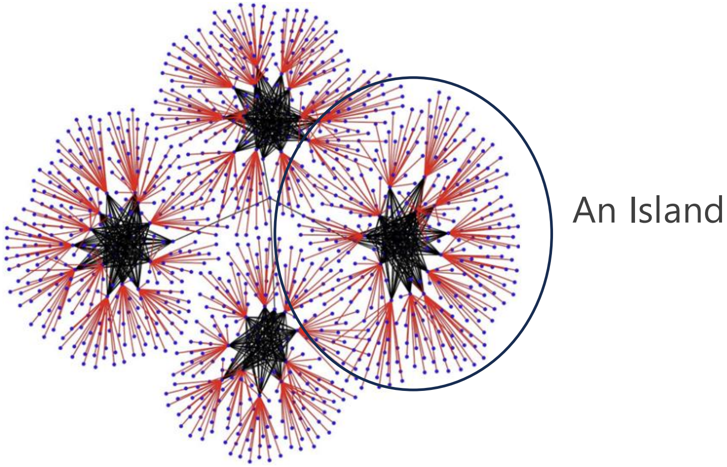


Cluster utilization data for the three runs

Workload-Aware Algorithm Selection

- ❑ Taking cluster utilization into account
  - ❑ When selecting the algorithm
- ❑ Running HP2P[1] benchmark before the collective
  - ❑ Measures the peer-to-peer latency and bandwidth between the pairs

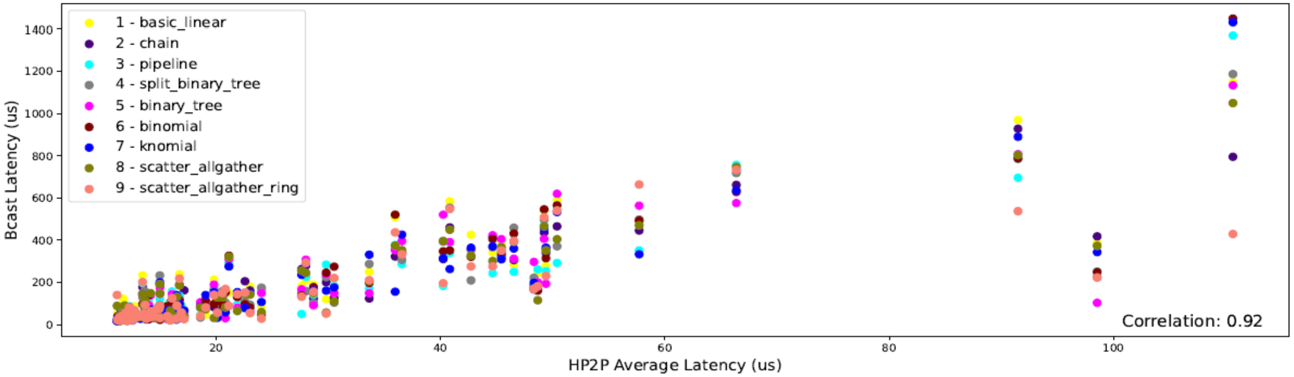
* Nodes are allocated randomly on different islands of the cluster



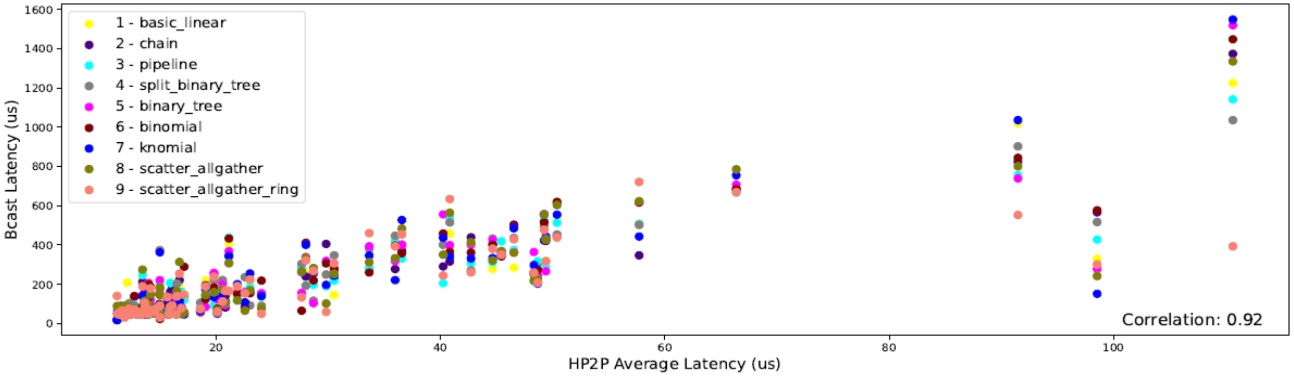
[1] https://github.com/cea-hpc/hp2p

Workload-Aware Algorithm Selection

- 100 series of runs executed on different days and hours of the days



(a) 100 B



(b) 10 KB

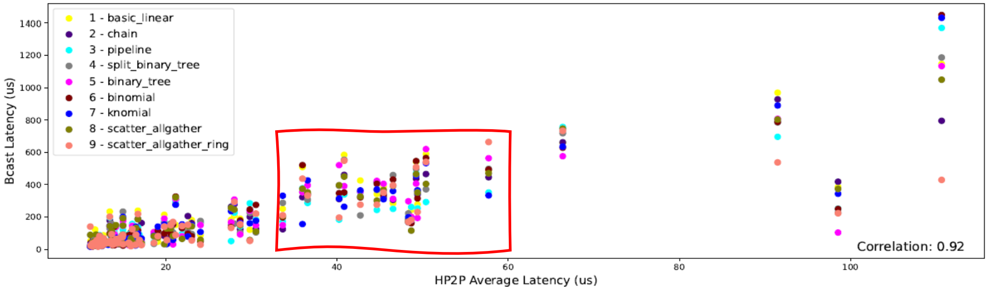
The correlation between latencies of HP2P and Bcast – Sorted based on HP2P latency

Workload-Aware Algorithm Selection

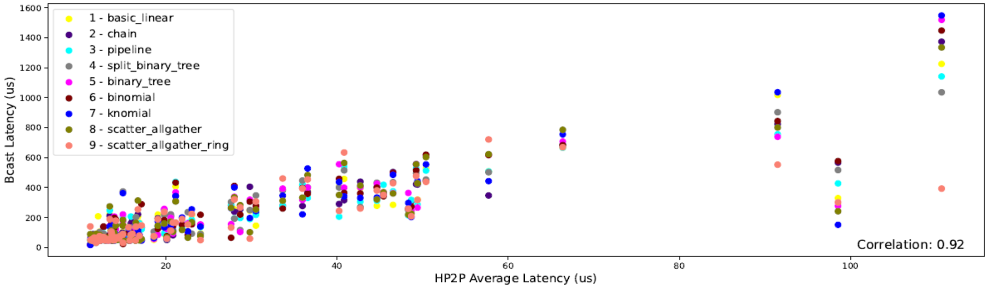
- ❑ Latencies of HP2P and Broadcast are highly correlated
  - ❑ Helps estimate the execution time of the main benchmark

- ❑ Network traffic is impacting algorithms’ performance

- ❑ A good algorithm selection in higher network traffic can highly improve the communication performance



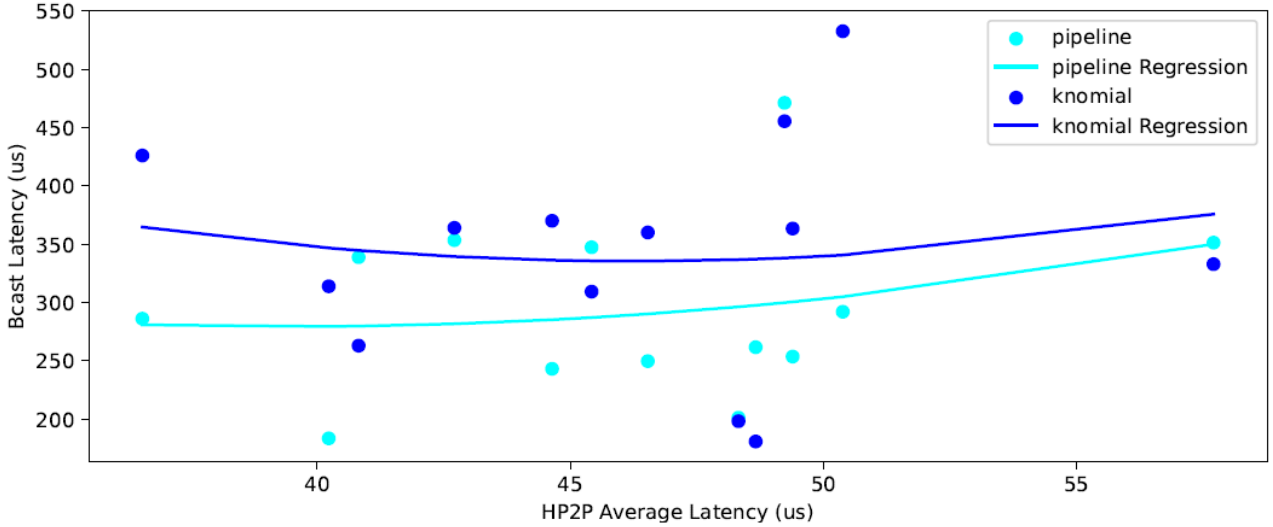
(a) 100 B



(b) 10 KB

Workload-Aware Algorithm Selection

- ❑ Pipeline has shown a higher performance (around 15%) than Knomial
- ❑ For each range of network traffic, different algorithms have diverse behavior



The performance distribution of **Pipeline** and **Knomial** (OMPI Default) between the range of 35 to 60 us.

Summary and Future Work

- □ Workload-aware algorithm selection
  - □ Monitors the network usage
  - □ Chooses the best algorithm
- □ Future Work
  - □ Better characterizing the cluster’s workload
    - □ Collecting data from the job scheduler
    - □ Other microbenchmarks
  - □ Providing more accurate algorithm selector
    - □ Statistical, Regression, Machine Learning
  - □ Automating the selection process

THANK YOU

Algorithm Selection of MPI Collectives
Considering System Utilization

Majid Salimi Beni, Sascha Hunold, Biagio Cosenza

Euro-Par 2023 PhD Symposium
Limassol, Cyprus

✉ Reach me at:
msalimibeni@unisa.it