Optimization of collective communications in HeteroMPI

Alexey Lastovetsky Maureen O'Flynn Vladimir Rychkov
{Alexey.Lastovetsky, Maureen.OFlynn, Vladimir.Rychkov}@ucd.ie

Heterogeneous Computing Laboratory
School of Computer Science and Informatics, University College Dublin,
Belfield, Dublin 4, Ireland
http://hcl.ucd.ie

The 14th European PVM/MPI Users Group conference
October 1, 2007, Paris, France

Optimization of MPI collective communication operations

- Tweaking the hardware/software settings
  - TCP, kernel, compiler, ...
    - Authority!

Optimization of MPI collective communication operations

- Tweaking the hardware/software settings
  - TCP, kernel, compiler, ...
    - Authority!
- Optimized algorithms
  - Low-level modification of MPI implementation
    - Knowledge of implementation details!
  - High-level model-based optimization
    - Take into account the behavior of the native communication operations and implement the optimized ones on top of MPI

General approach

- Combined algorithms for collective operations
  - The best algorithm for each given number of processors/message size

General approach

- Combined algorithms for collective operations
  - The best algorithm for each given number of processors/message size
  - The algorithms are implemented with help of low-level and/or point-to-point operations
    - Portable if implemented on top of MPI

General approach

- Combined algorithms for collective operations
  - The best algorithm for each given number of processors/message size
  - The algorithms are implemented with help of low-level and/or point-to-point operations
    - Portable if implemented on top of MPI
  - The performance of algorithms depends on the combination of hardware/software/MPI
    - Automatically tuned collective operations upon installation and after the platform has been changed

Algorithm design

- Experiment-based (Vadhiyar et al.):
  - Measurement of the performance of the algorithms
  - Reducing the number of measurements
  - A lot of measurements with all processors involved
  - Applicable to any parallel platform

Algorithm design

- Experiment-based (Vadhiyar et al.):
  - Measurement of the performance of the algorithms
  - Reducing the number of measurements
  - A lot of measurements with all processors involved
  - Applicable to any parallel platform
- Model-based (Thakur et al., Pjesivac-Grbovic et al.):
  - Estimation of the parameters of the point-to-point model
  - Prediction of the performace of the algorithms
  - A small number of measurements:
    fast communication experiments between any two processors
  - Applied to homogeneous clusters
  - Inaccurate for the clusters based on a switched network, for HNOC

Optimization for homogeneous/heterogeneous clusters

- Model-based
  - More accurate heterogeneous communication performance model
  - A relatively small number of measurements:
    fast communication experiments between all pairs and triplets of processors,
    gather/scatter experiments for different message sizes
  - Applicable to homogeneous/heterogeneous clusters

Optimization for homogeneous/heterogeneous clusters

- Model-based
  - More accurate heterogeneous communication performance model
  - A relatively small number of measurements:
    fast communication experiments between all pairs and triplets of processors,
    gather/scatter experiments for different message sizes
  - Applicable to homogeneous/heterogeneous clusters
- Portable
  - No low-level MPI-implementation-specific operations
  - Message segmentation and sequence of calls to the native MPI operations
  - Part of HeteroMPI (portable extension of MPI for HNOC)

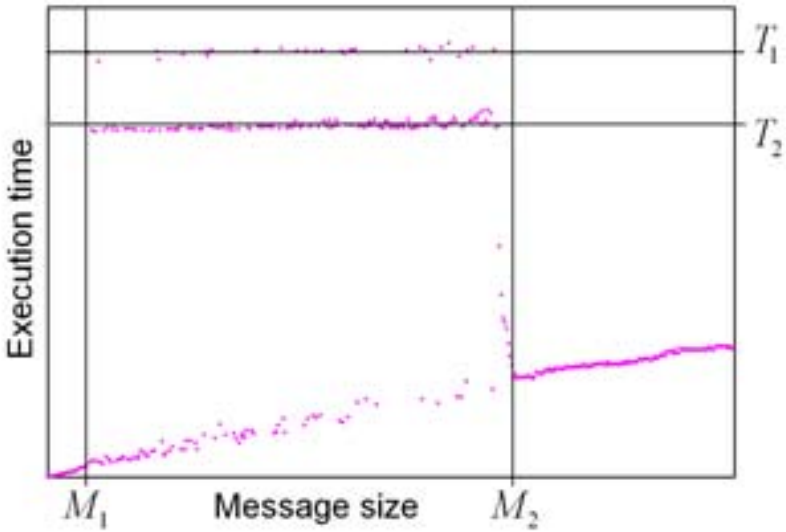
Optimization for homogeneous/heterogeneous clusters

- Model-based
  - More accurate heterogeneous communication performance model
  - A relatively small number of measurements:
    fast communication experiments between all pairs and triplets of processors,
    gather/scatter experiments for different message sizes
  - Applicable to homogeneous/heterogeneous clusters
- Portable
  - No low-level MPI-implementation-specific operations
  - Message segmentation and sequence of calls to the native MPI operations
  - Part of HeteroMPI (portable extension of MPI for HNOC)
- Automatically tuned
  - HeteroMPI computes the parameters of the model for the particular platform

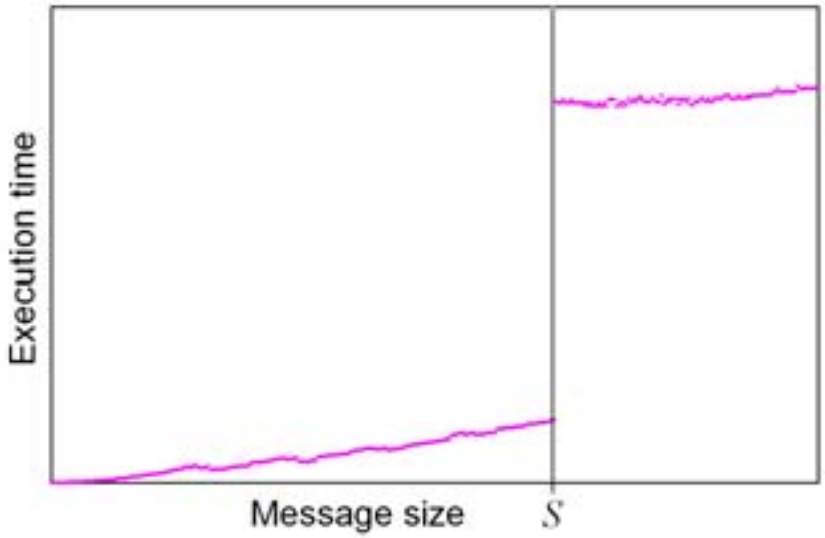
- HeteroMPI is an extension of MPI designed for HNOC:
  - takes into account:
    - heterogeneity of processors
    - network topology
    - computational costs of algorithm
  - works on top of any MPI implementation:
    - includes a small number of additional functions for group management and data partitioning
    - inherits all MPI communication operations
    - **provides the optimized version of collective operations**

Observations:

- many-to-one (gather)



one-to-many (scatter)



Heterogeneous communication performance model:

- predicts the execution time of communication operations
- reflects the observed escalations of the execution time
- applicable to both heterogeneous and homogeneous clusters

Algorithm design

- Split the messages to avoid the escalation of the communication execution time
- Use M_1, M_2 and S thresholds from the model
- Implement by calling the corresponding MPI functions

Pseudo code

```
HMPI_Gather
{
if (M1<=M<=M2) {
  find m such that
     M/m<M1 and M/(m-1)>=M1;
  for (i=0; i<m; i++)
    MPI_Gather(sendbuf + i*M/m, M/m);
else
  MPI_Gather(sendbuf, M);
}
```

```
HMPI_Scatter
{
if (M>=S) {
find m such that
  M/m<S and M/(m-1)>=S;
for (i=0; i<m; i++)
  MPI_Scatter(recvbuf + i*M/m, M/m);
else
  MPI_Scatter(recvbuf, M);
}
```

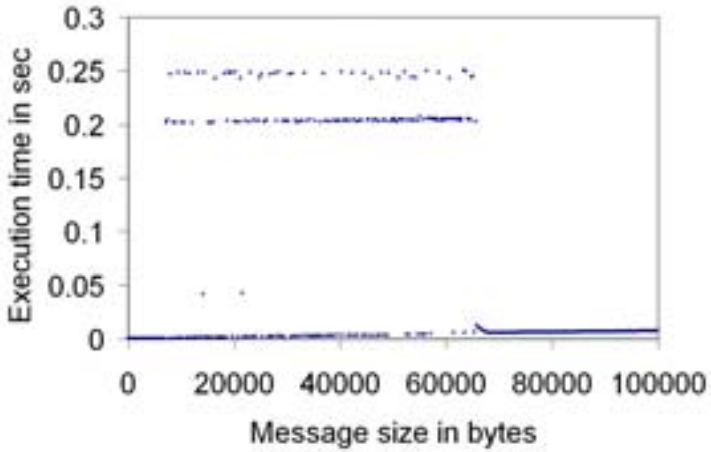
Pseudo code

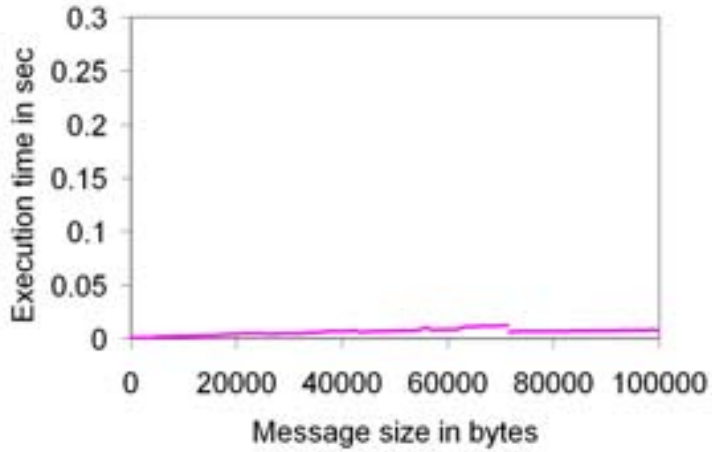
```
HMPI_Gather
{
if (M1<=M<=M2) {
  find m such that
     M/m<M1 and M/(m-1)>=M1;
  for (i=0; i<m; i++) {
    MPI_Barrier(comm);
    MPI_Gather(sendbuf + i*M/m, M/m);
  }
}
else
  MPI_Gather(sendbuf, M);
}
```

```
HMPI_Scatter
{
if (M>=S) {
find m such that
  M/m<S and M/(m-1)>=S;
for (i=0; i<m; i++)
  MPI_Scatter(recvbuf + i*M/m, M/m);
else
  MPI_Scatter(recvbuf, M);
}
```

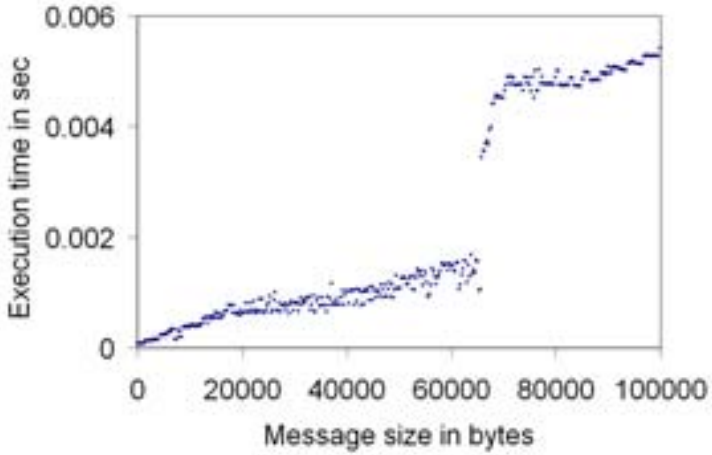
Native | Optimized

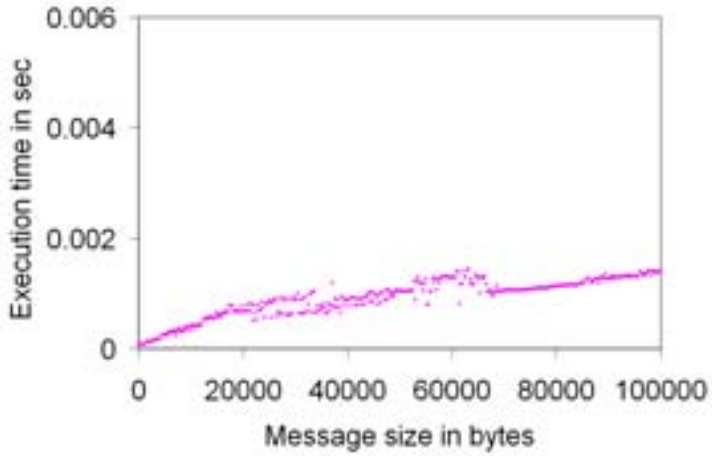
Gather





Scatter

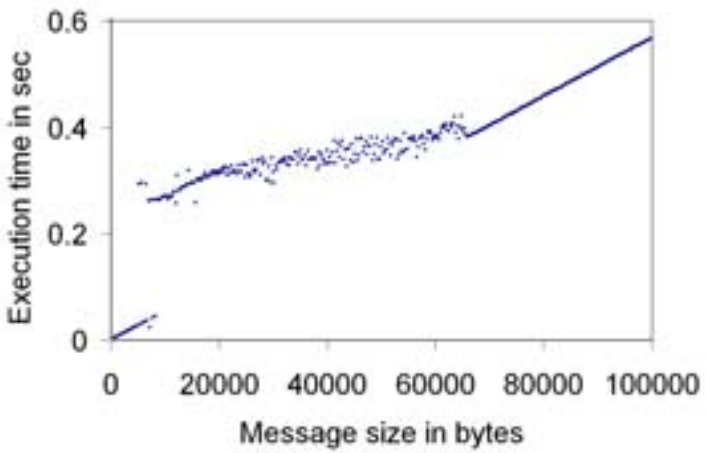


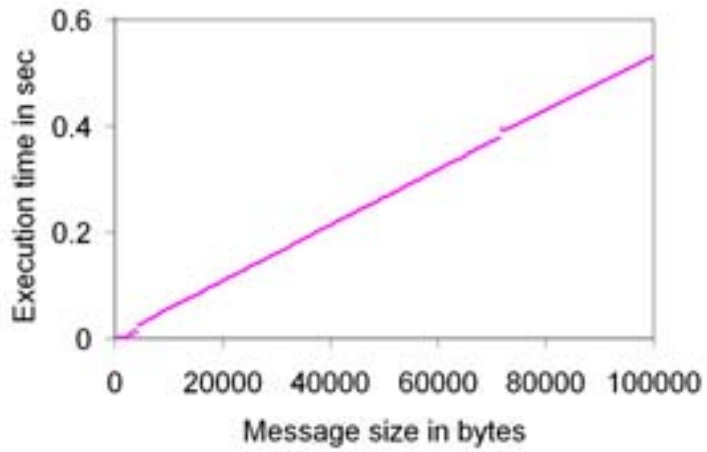


11 x Xeon 2.8/3.4/3.6, 2 x P4 3.2/3.4, 1 x Celeron 2.9, 2 x AMD Opteron 1.8, Gigabit Ethernet, LAM 7.1.3

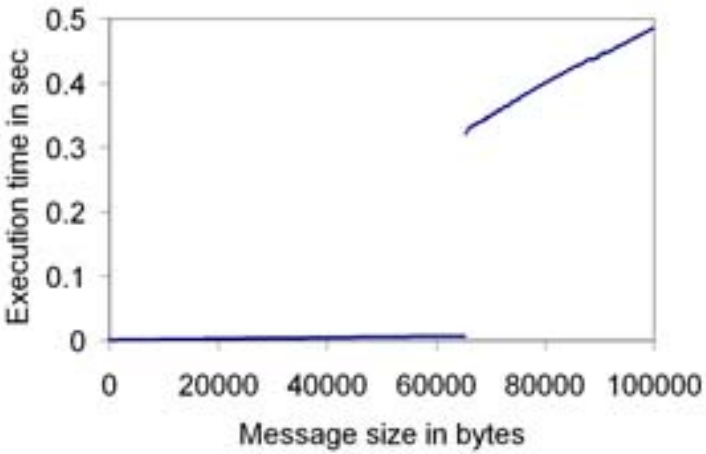
Native | Optimized

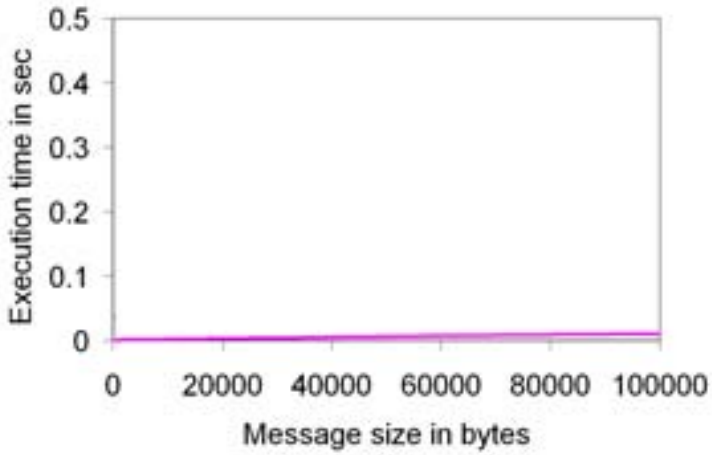
Gather





Scatter





64 x Intel EM64T, Myrinet, Open MPI 1.2.2 over TCP

Optimized HeteroMPI collective communication operations

- outperform the native ones on clusters based on a switched network
- based on the heterogeneous communication performance model
- portable, call the corresponding MPI functions
- automatically tuned, require a relatively small number of measurements

Acknowledgments:

- Science Foundation Ireland
- Innovative Computing Laboratory, University of Tennessee