Improvement of the Bandwidth of Cross-Site MPI Communication Using Optical Fiber

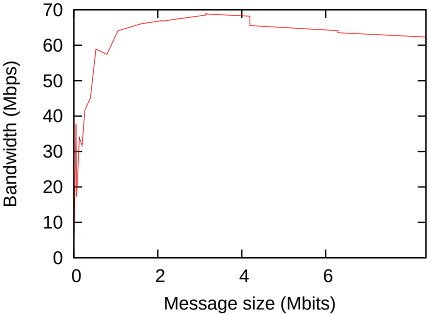
Kiril Dichev Alexey Lastovetsky Vladimir Rychkov
Kiril.Dichev@ucdconnect.ie, Alexey.Lastovetsky@ucd.ie, Vladimir.Rychkov@ucd.ie

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

Grid'5000 Cross-Site Benchmarks

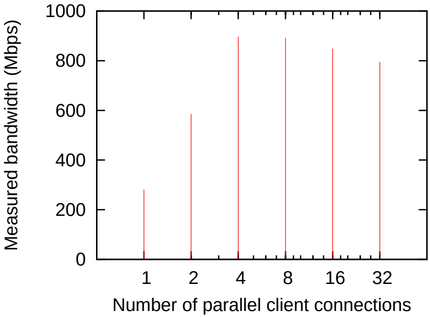
MPI

- NetPIPE with MPI shows low peak bandwidth across sites
- Example: Toulouse-Bordeaux - 70 Mbps



Grid'5000 Cross-Site Benchmarks
TCP

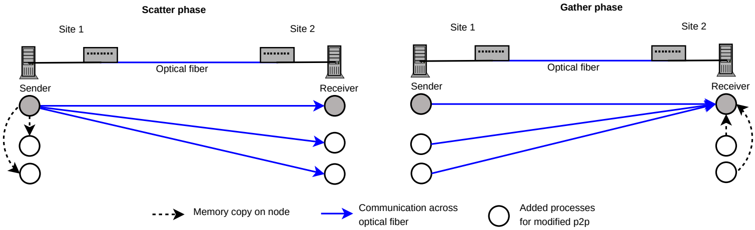
- Standard tests with iperf suggest using multiple TCP clients in parallel improves bandwidth (coming close to 1 Gbps)



Idea

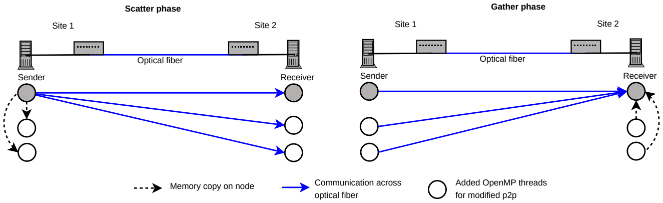
We change MPI point-to-point communication:

- The pattern resembles a two phase scatter-gather
- In the scatter phase, the p2p sender scatters equal message fragments among a number of participants
- In the gather phase, the p2p receiver gathers the pieces
- The scatter/gather is a linear sequence of non-blocking p2p calls



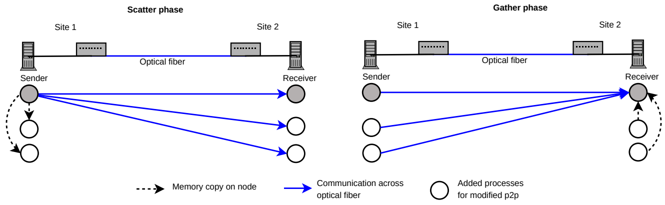
OpenMP implementation

- Multi-threading: A number of OpenMP threads run the p2p calls on the message fragments
- Easy implementation, but zero effect
- MPI libraries either:
  - Don't support MPI_THREAD_MULTIPLE or
  - Don't parallelize send operation for different threads (critical section)

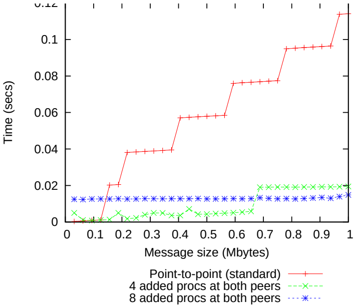


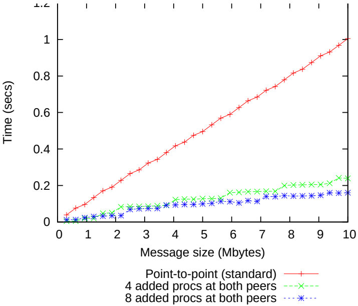
Process spawning MPI implementation

- Spawn extra MPI processes at sender/receiver node at initialization
- Involve them only in p2p communication across sites
- Synchronization required for each p2p communication



Results





- All messages larger than 200 KB were transferred faster
- Standard p2p had throughput of around 80 Mbps
- The throughput with 8 extra processes per sender/receiver was around 500 Mbps
- Significant increase in throughput (nearly 6 times), also observed for other sites

Thank You!