An Application Based MPI Message Throughput Benchmark

Brian Barrett & K. Scott Hemmert

{bwbarre,kshemme}@sandia.gov

Sandia National Laboratories

Cluster 2009

September 3, 2009

Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

Interconnects Are Important

- DOE labs have long history of highly scalable distributed memory applications
- Application scalability influenced by:
  - Application developer / algorithms / etc.
  - System noise
  - Network hardware and software stack
- Network performance for motivating applications:
  - Latency: some
  - Bandwidth: many
  - Message rate: maybe tomorrow
- Procurements seem to drive network development
  - Benchmarks drive procurement requirements

Benchmarks as Research Drivers

- Latency (~10% variation)
  - Half round-trip pretty hard to measure wrong
- Bandwidth (~2x variation)
  - Buffer reuse?
  - Buffer contents?
  - Interaction with memory allocator?
  - One message or aggregate?
- Message rate (~10x variation)
  - Receive queue length?
  - Cache interactions?
  - Expected or unexpected messages?

National Nuclear Security Administration

Sandia National Laboratories

Application Challenges

- Ghost cell updates on today's code bulk messages at end of iteration
- Memory copies to pack / unpack buffers
- Network bandwidth improvements slowing compared to processor / memory bandwidth

- Motivates applications to move away from large messages after copies
  - Message rates more important
  - Communication patterns different from message rate benchmarks
- Generalize applications into 4 communication patterns

National Nuclear Security Administration

Sandia National Laboratories

Single-Direction Communication

- Mimics existing benchmarks
- Not application representative, but useful as baseline
- Added “cache invalidation” between iterations

```
for (i = 0 ; i < niters ; ++i) {
  nreqs = 0;

  cache_invalidate();
  synchronize();
  start = timer();
  if (rank < size / 2) {
    for (k = 0 ; k < nmsgs ; ++k) {
      MPI_Isend(send_buf + (nbytes * k),
      nbytes, MPI_CHAR, rank + (size / 2), tag,
      comm, &reqs[nreqs++]);
    }
  } else {
    for (k = 0 ; k < nmsgs ; ++k) {
      MPI_Irecv(recv_buf + (nbytes * k),
      nbytes, MPI_CHAR, rank - (size / 2), tag,
      comm, &reqs[nreqs++]);
    }
  }
  MPI_Waitall(nreqs, reqs,
              MPI_STATUSES_IGNORE);
  total += (timer() - start);
}
```

National Nuclear Security Administration

Sandia National Laboratories

Pair-based Communication

- Each process communicates with multiple peers, both sending and receiving
- Likely to generate (potentially large) unexpected message count
- Only `nmsgs` posted receives at a time

```
for (i = 0 ; i < niters ; ++i) {
  cache_invalidate();
  MPI_Barrier(MPI_COMM_WORLD);
  start = timer();
  for (j = 0 ; j < npeers ; ++j) {
    nreqs = 0;
    for (k = 0 ; k < nmsgs ; ++k) {
      offset = nbytes * (k + j * nmsgs);
      MPI_Irecv(recv_buf + offset,
              nbytes, MPI_CHAR,
              recv_peers[j], tag,
              MPI_COMM_WORLD, &reqs[nreqs++]);
    }
    for (k = 0 ; k < nmsgs ; ++k) {
      offset = nbytes * (k + j * nmsgs);
      MPI_Isend(send_buf + offset,
              nbytes, MPI_CHAR,
              send_peers[npeers - j - 1], tag,
              MPI_COMM_WORLD, &reqs[nreqs++]);
    }
    MPI_Waitall(nreqs, reqs,
              MPI_STATUSES_IGNORE);
  }
  total += (timer() - start);
}
```

Pre-posted Communication

- Mimic applications which pre-post receives at end of communication phase
- Guaranteed there are no unexpected receives
- Large number of posted receives

```
start = timer();
for (j = 0 ; j < npeers ; ++j) {
  for (k = 0 ; k < nmsgs ; ++k) {
    offset = nbytes * (k + j * nmsgs);
    MPI_Irecv(recv_buf + offset
            nbytes, MPI_CHAR,
            recv_peers[j], tag,
            MPI_COMM_WORLD, &reqs[nreqs++]);
  }
}
total += (timer() - start);

for (i = 0 ; i < niters - 1 ; ++i) {
  cache_invalidate();
  MPI_Barrier(MPI_COMM_WORLD);

  start = timer();
  for (j = 0 ; j < npeers ; ++j) {
    for (k = 0 ; k < nmsgs ; ++k) {
      offset = nbytes * (k + j * nmsgs);
      MPI_Isend(send_buf + offset
              nbytes, MPI_CHAR,
              send_peers[npeers - j - 1], tag,
              MPI_COMM_WORLD, &reqs[nreqs++]);
    }
  }
  MPI_Waitall(nreqs, reqs,
            MPI_STATUSES_IGNORE);
  nreqs = 0;
  for (j = 0 ; j < npeers ; ++j) {
    for (k = 0 ; k < nmsgs ; ++k) {
      offset = nbytes * (k + j * nmsgs);
      MPI_Irecv(recv_buf + offset
              nbytes, MPI_CHAR,
              recv_peers[j], tag,
              MPI_COMM_WORLD, &reqs[nreqs++]);
    }
  }
  total += (timer() - start);
}

start = timer();
for (j = 0 ; j < npeers ; ++j) {
  for (k = 0 ; k < nmsgs ; ++k) {
    offset = nbytes * (k + j * nmsgs);
    MPI_Isend(send_buf + offset,
            nbytes, MPI_CHAR,
            send_peers[npeers - j - 1], tag,
            MPI_COMM_WORLD, &reqs[nreqs++]);
  }
}
MPI_Waitall(nreqs, reqs, MPI_STATUSES_IGNORE);
total += (timer() - start);
```

National Nuclear Security Administration

Sandia National Laboratories

All-start Communication

- Similar to pre-posted communication test, but no pre-posted guarantee
- Simple communication pattern, similar to those used today
- Unlikely to be useful in future

```
for (i = 0 ; i < niters ; ++i) {
  cache_invalidate();
  MPI_Barrier(MPI_COMM_WORLD);

  start = timer();
  nreqs = 0;
  for (j = 0 ; j < npeers ; ++j) {
    for (k = 0 ; k < nmsgs ; ++k) {
      offset = nbytes * (k + j * nmsgs);
      MPI_Irecv(recv_buf + offset,
                nbytes, MPI_CHAR,
                recv_peers[j], tag,
                MPI_COMM_WORLD, &reqs[nreqs++]);
    }
    for (k = 0 ; k < nmsgs ; ++k) {
      offset = nbytes * (k + j * nmsgs);
      MPI_Isend(send_buf + offset,
                nbytes, MPI_CHAR,
                send_peers[npeers - j - 1], tag,
                MPI_COMM_WORLD, &reqs[nreqs++]);
    }
  }
  MPI_Waitall(nreqs, reqs, MPI_STATUSES_IGNORE);
  total += (timer() - start);
}
```

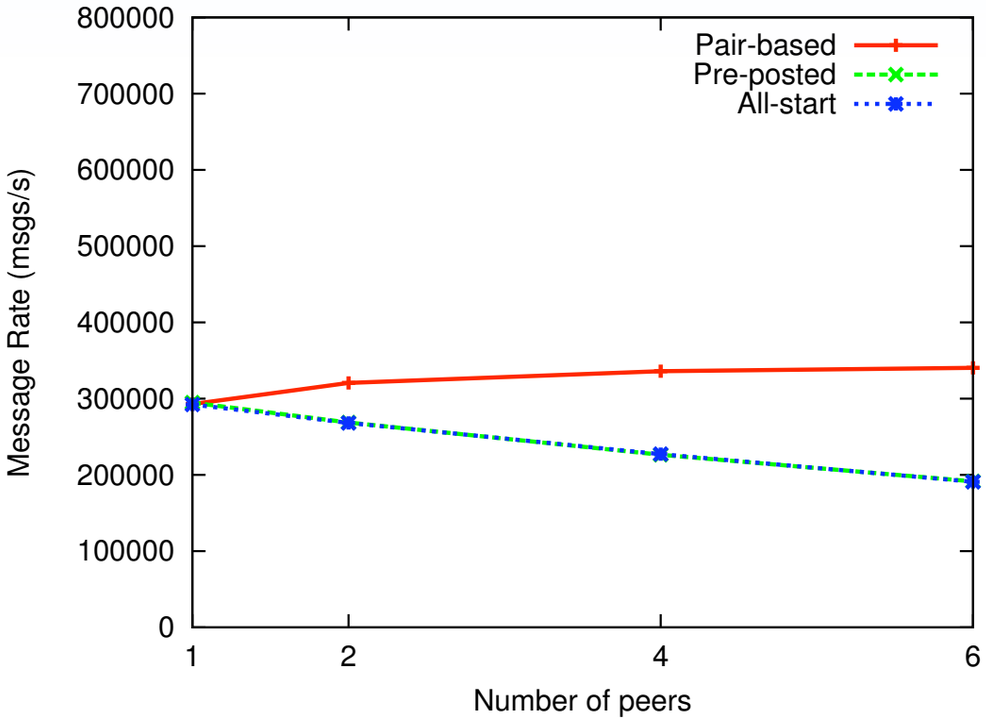
National Nuclear Security Administration

Sandia National Laboratories

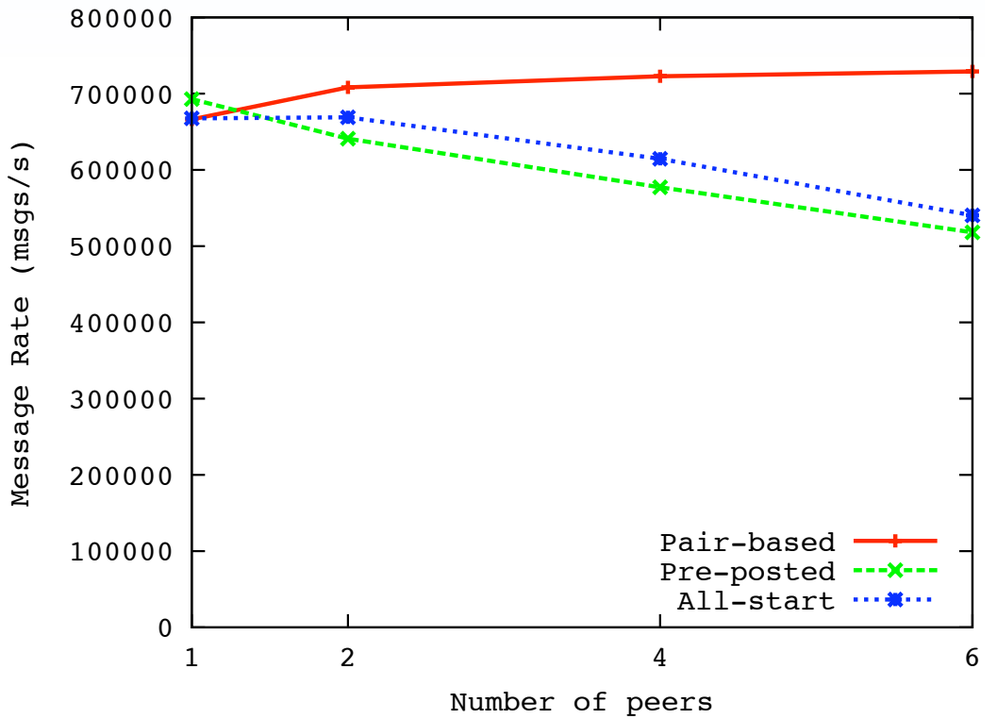
Test Platforms

- **Cray XT**
  - **Sandia Red Storm platform test environment**
  - **2.4 GHz dual-core Opteron processors**
  - **SeaStar with 4 GB/s bidirectional bandwidth**
- **Myrinet**
  - **128 nodes, each with two 2.2 GHz single-core Opterons**
  - **Myrinet 10G NIC (10G-PCIE-8A-C) w/ Myrinet switch**
  - **RHEL 4U7, MX 1.2.7, Open MPI 1.2.8**
- **InfiniBand**
  - **272 nodes, each with four 2.2 GHz quad-core Opterons**
  - **ConnectX HCA w/ Voltaire DDR switch**
  - **RHEL 4 variant, OFED 1.2, Open MPI 1.2.7**

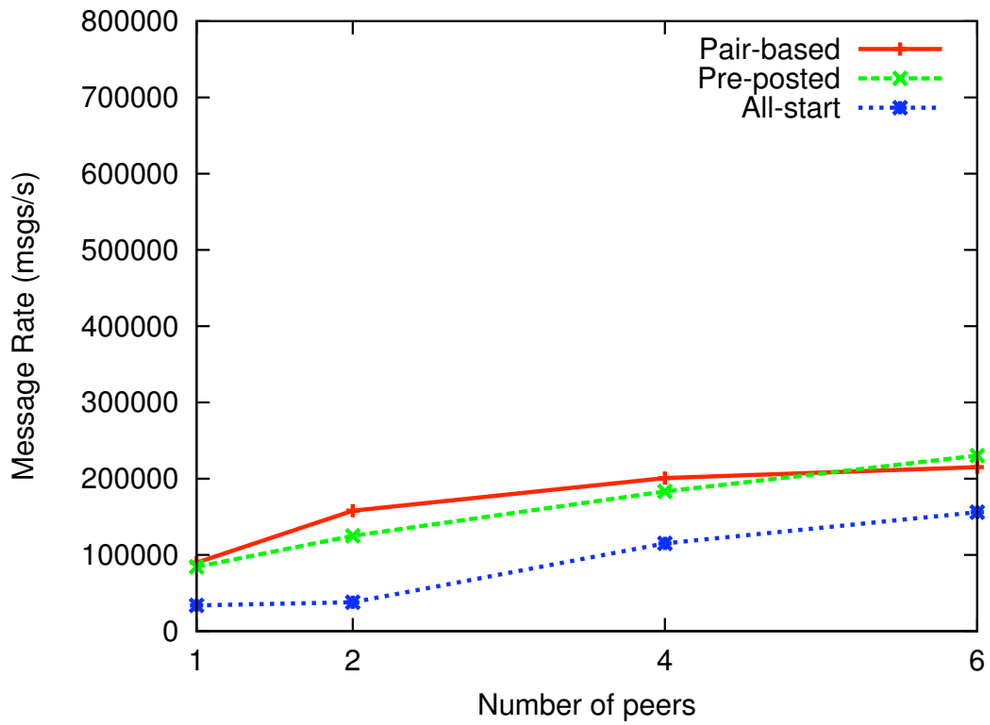
Number of Neighbors Scaling



Cray

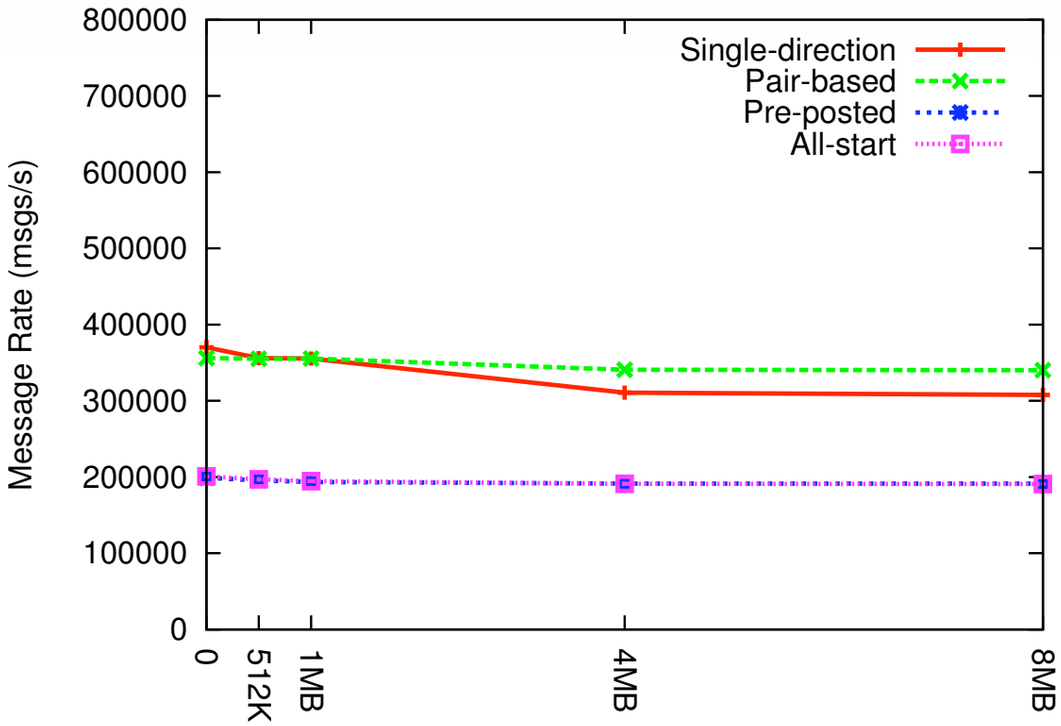


Myrinet

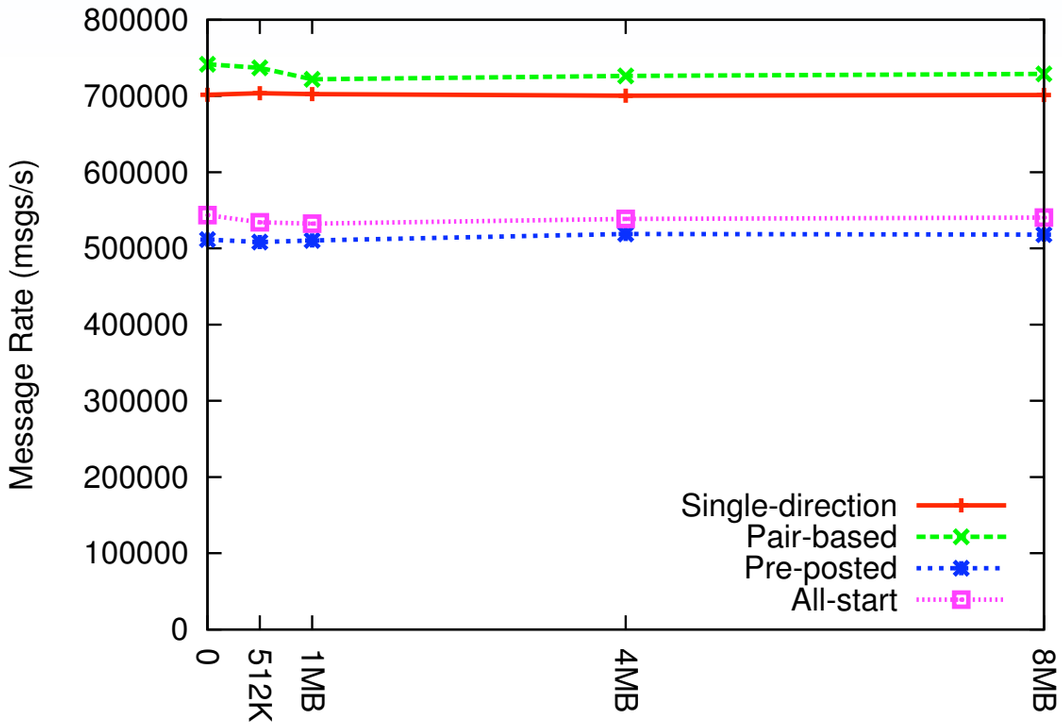


InfiniBand

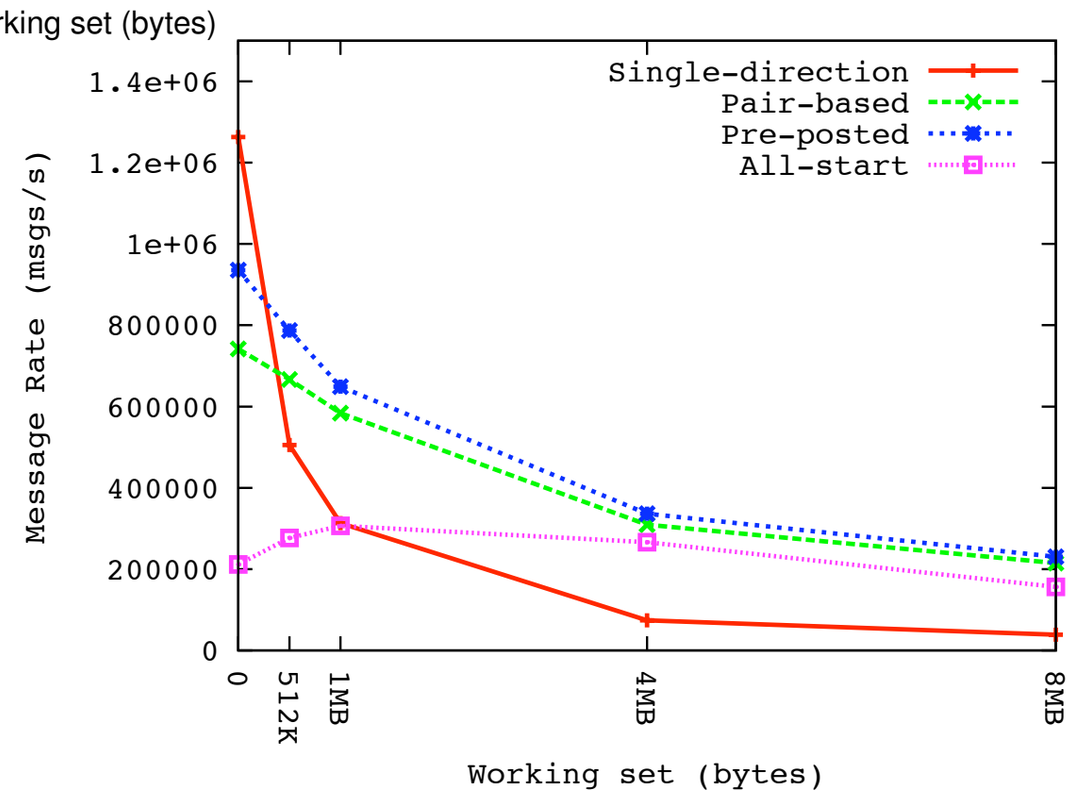
Sensitivity to Working Set Size



Cray



Myrinet

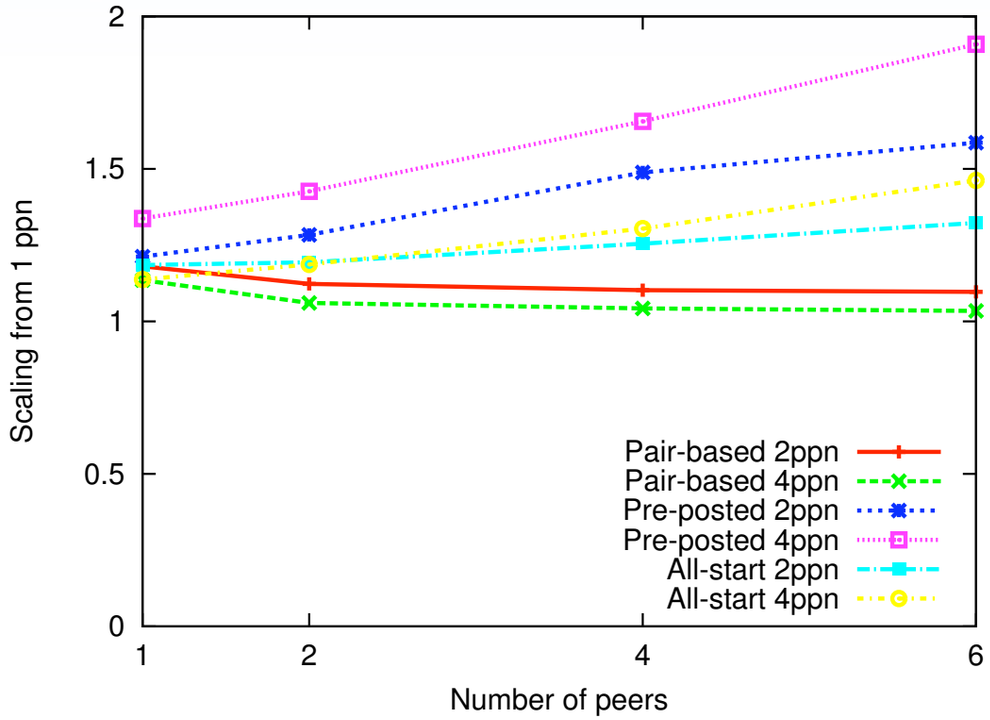


InfiniBand

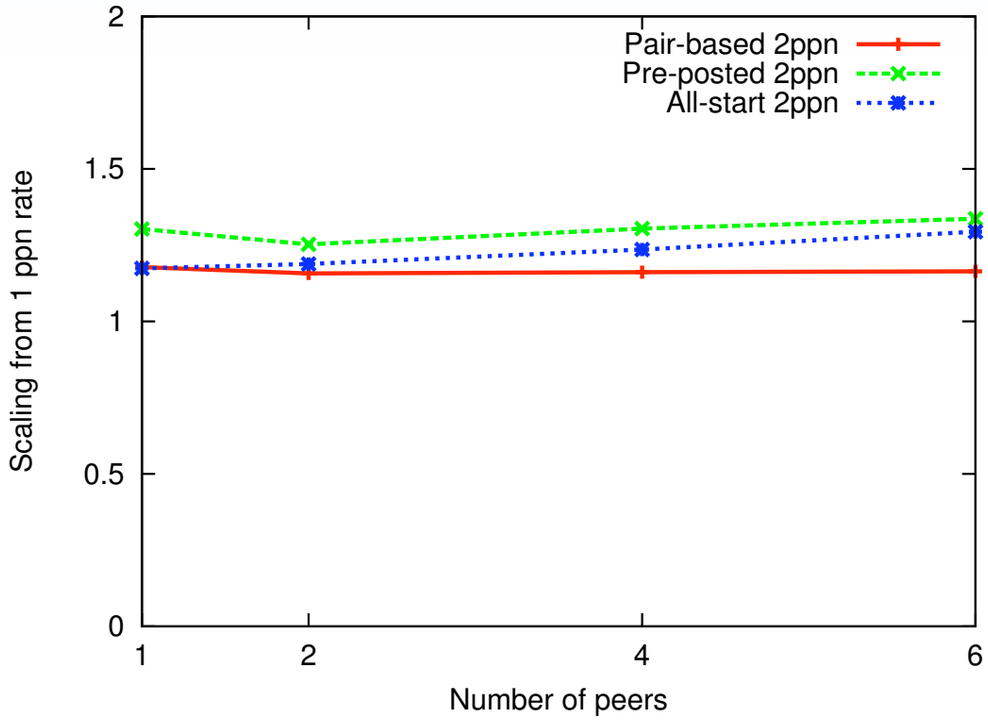
National Nuclear Security Administration

Sandia National Laboratories

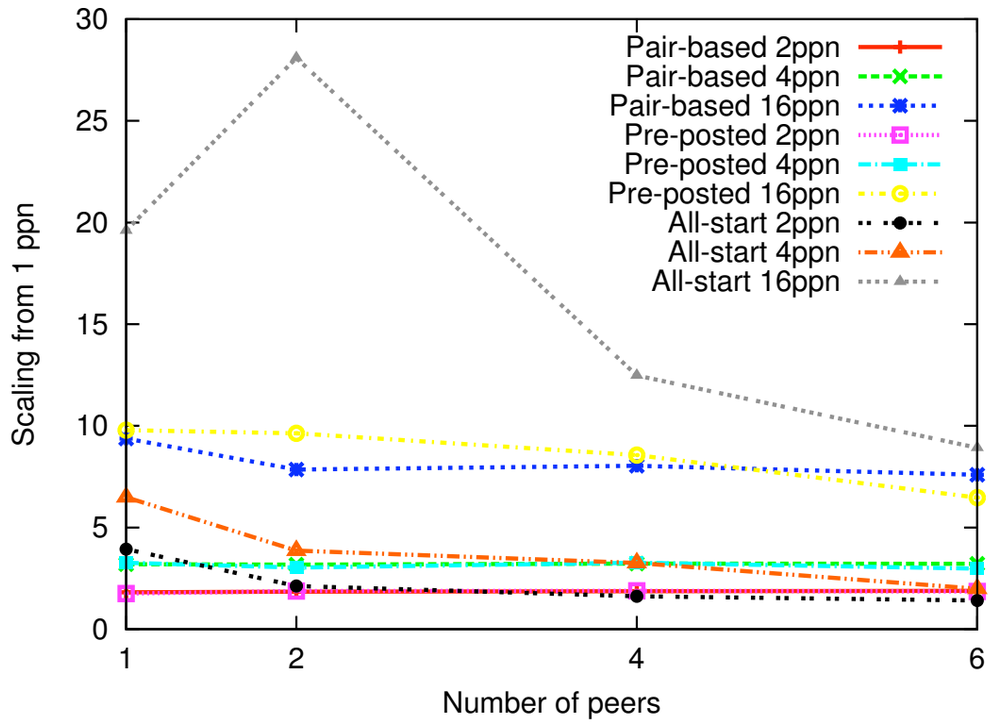
Process-per-node Scaling



Cray



Myrinet



InfiniBand

National Nuclear Security Administration

Sandia National Laboratories

Conclusions

- Benchmarks drive networking innovation
- Current message rate benchmarks provide unrealistic results
- Sandia Message Throughput Benchmark
  - Basis for further analysis
  - Likely to evolve as applications change and grow
  - Freely available

http://www.cs.sandia.gov/smb/