# Performance Analysis and Optimization of Parallel I/O in a large scale groundwater application on Petascale Architectures

Vamsi Sripathi

Committee

- Dr. (Kumar) Mahinthakumar
- Dr. Xiaosong Ma
- Dr. Richard Mills
- Dr. Frank Mueller

Computer Science
NC STATE UNIVERSITY

# Outline

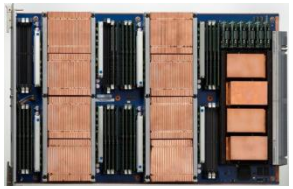
- Motivation and Objectives
- Petascale Systems
  - Cray XT5 and IBM BlueGene/P
- PFLOTRAN
- Analysis & Optimization of Parallel I/O
- Overall Performance Analysis
- Conclusions

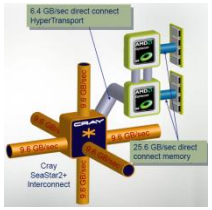
# Motivation and Objectives

- Current petascale architectures are composed of complex sub-systems
- Large scale computational models are being developed to study important scientific phenomenon
- Important to understand performance implications of petascale architectures on real world scientific applications
- Identify scalability challenges and performance bottlenecks
- Design optimization strategies to mitigate scalability issues and improve application performance

# Cray XT5 (JaguarPF)

- World's fastest supercomputer
- 224,256 processor cores
- Peak performance of
  2.3 PetaFlop/sec






**Chip**
AMD Opteron
2.6 GHz 6 cores
62.4 GF

**Compute node**
2 Chips
12 cores
124.8 GF
16 GB DDR2

**Blade**
4 Compute nodes
48 cores
499.2 GF
64 GB DDR2

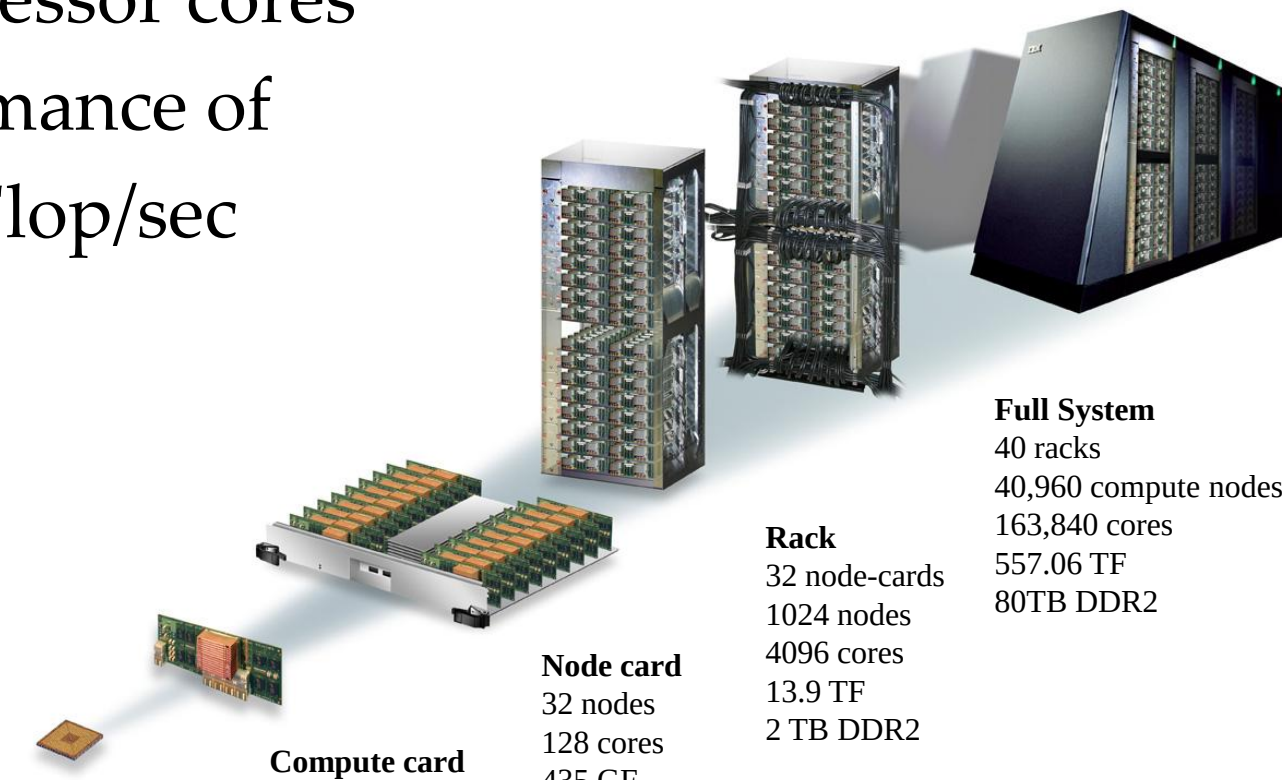
**Full System**
4,672 blades in 200 cabinets
18,688 compute nodes
224,256 cores
2.3 PF
300 TB DDR2



Computer Science
NC STATE UNIVERSITY

# IBM BlueGene/P (Intrepid)

- World's 7th fastest machine
- 163,840 processor cores
- Peak performance of 557.06 TeraFlop/sec



**Chip**
PowerPC 450
850 MHz 4 cores
13.6 GF

**Compute card**
1 node
4 cores
13.6 GF
2GB DDR2

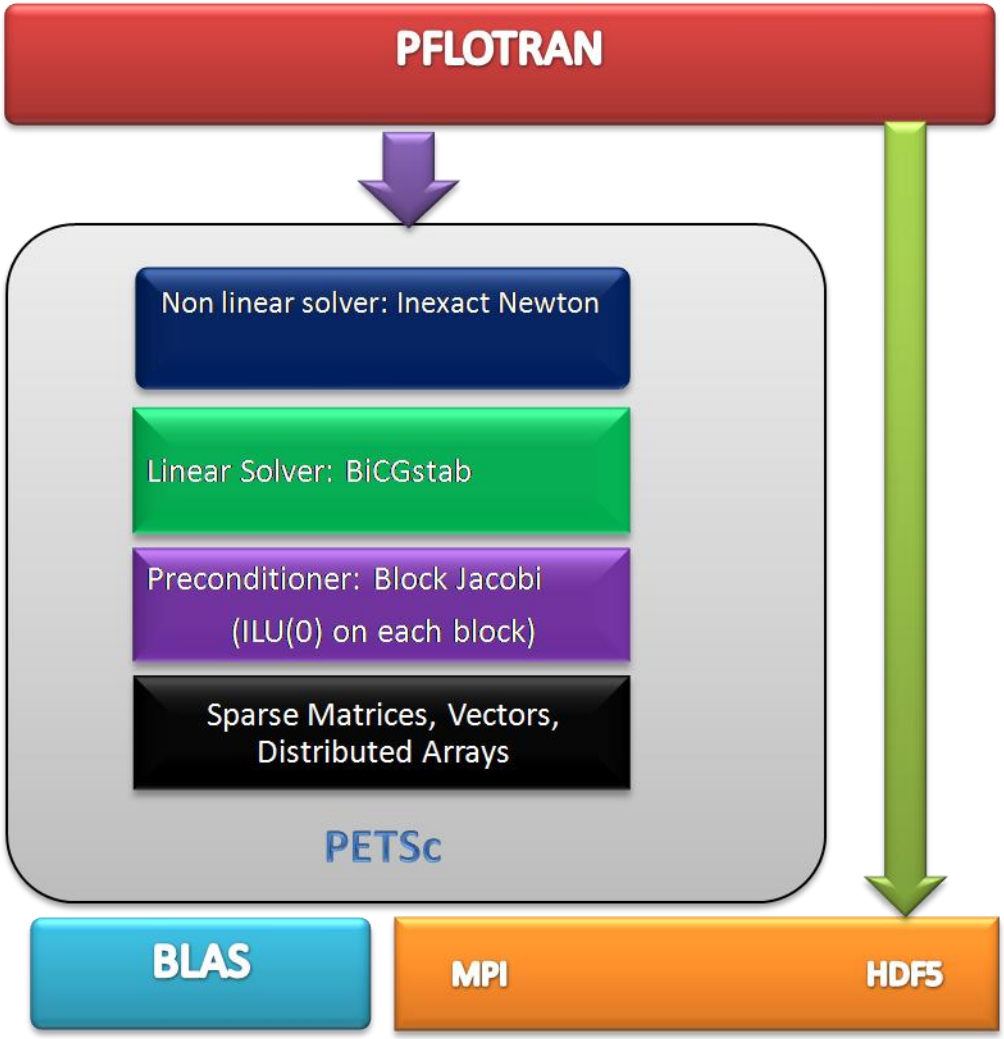
**Node card**
32 nodes
128 cores
435 GF
64GB DDR2

**Rack**
32 node-cards
1024 nodes
4096 cores
13.9 TF
2 TB DDR2

**Full System**
40 racks
40,960 compute nodes
163,840 cores
557.06 TF
80TB DDR2

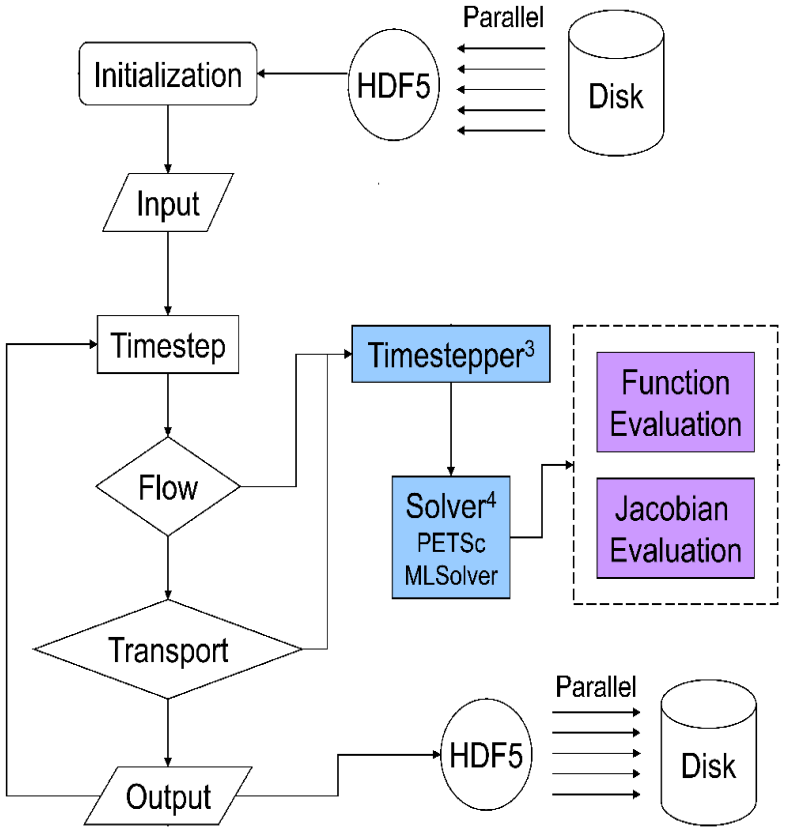
# PFLOTRAN: Architecture

- Large scale groundwater simulation code.
- Models multi-phase subsurface flow and multi component reactive transport in 3D porous media.
- Written in Fortran 90, uses PETSc and parallel HDF5.



# PFLOTRAN: Program Flow

- Parallelization using 3D domain decomposition
- Finite-volume: 7-point stencil
- Each processor is assigned a sub-domain of the problem
- Major stages
  - Initialization stage
  - Flow stage
  - Transport stage
  - Output stage

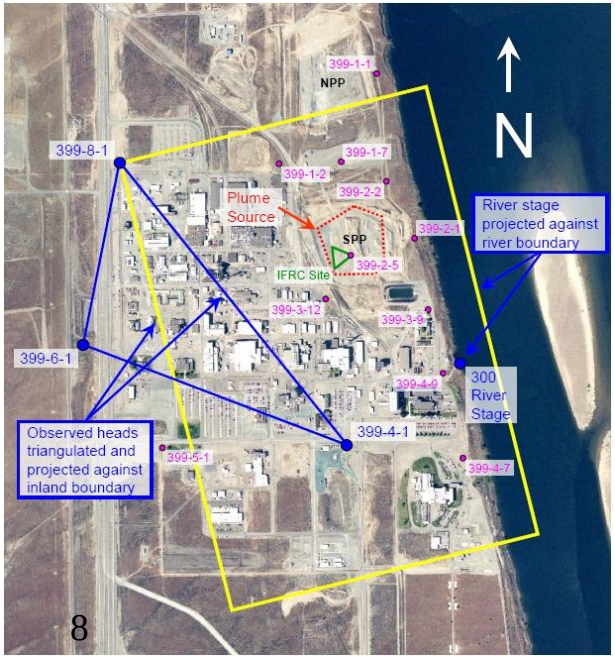


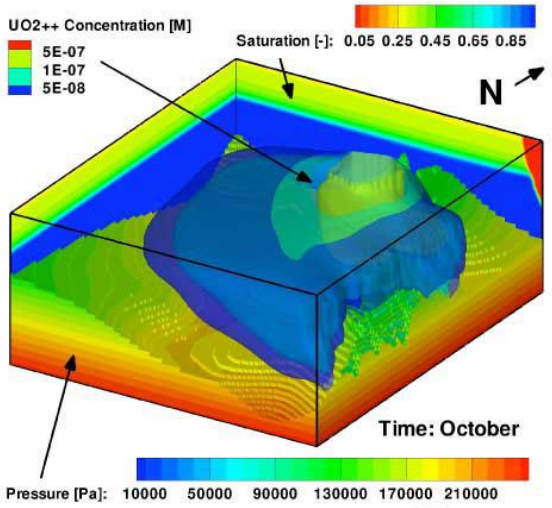



Computer Science
NC STATE UNIVERSITY

# Benchmark problems

- 1 billion DoF: 850*1000*80 cell coupled Flow and Transport problem. 68 million DoF for flow solve. 15 chemical components amounting to total of ~1 billion DoF for transport solve.
- 270 million DoF: 1350*2500*80 cell Flow-only version of the problem.

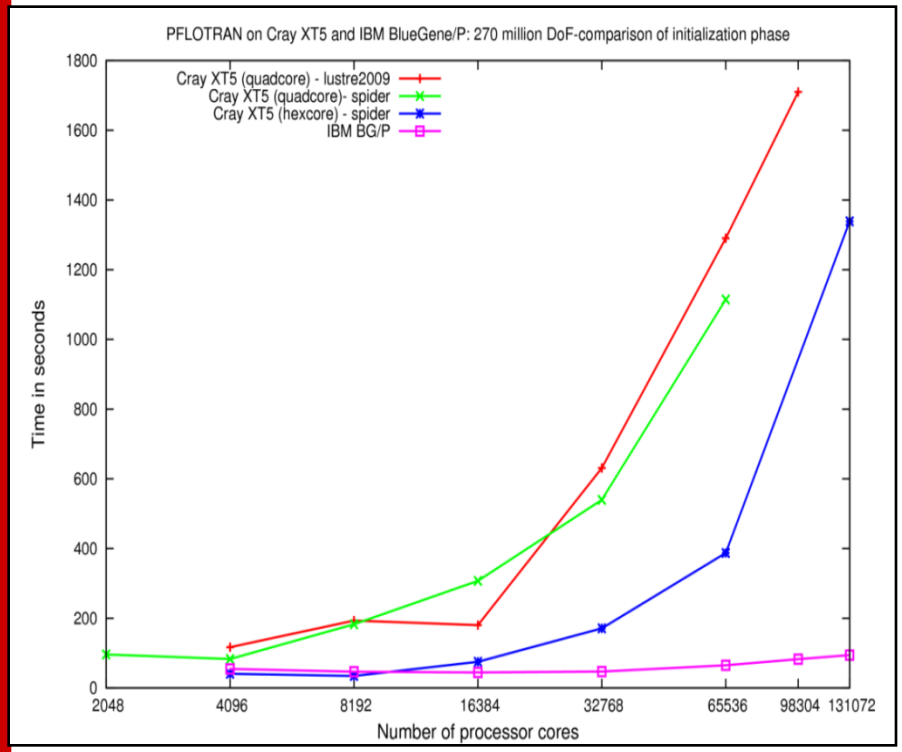


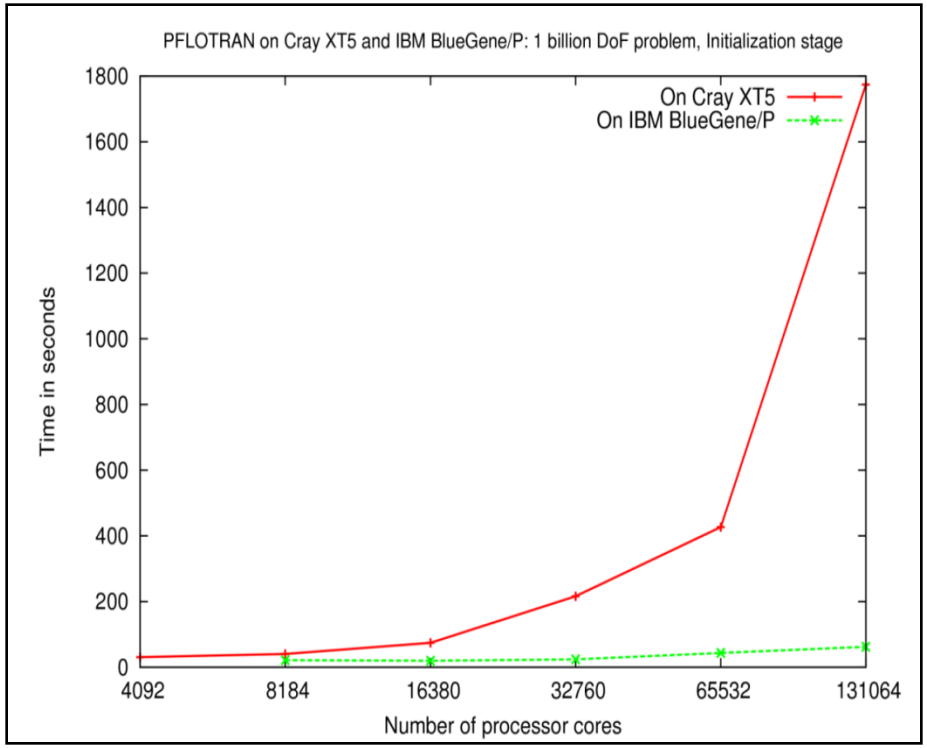






Computer Science
NC STATE UNIVERSITY

# Initialization stage: Scaling



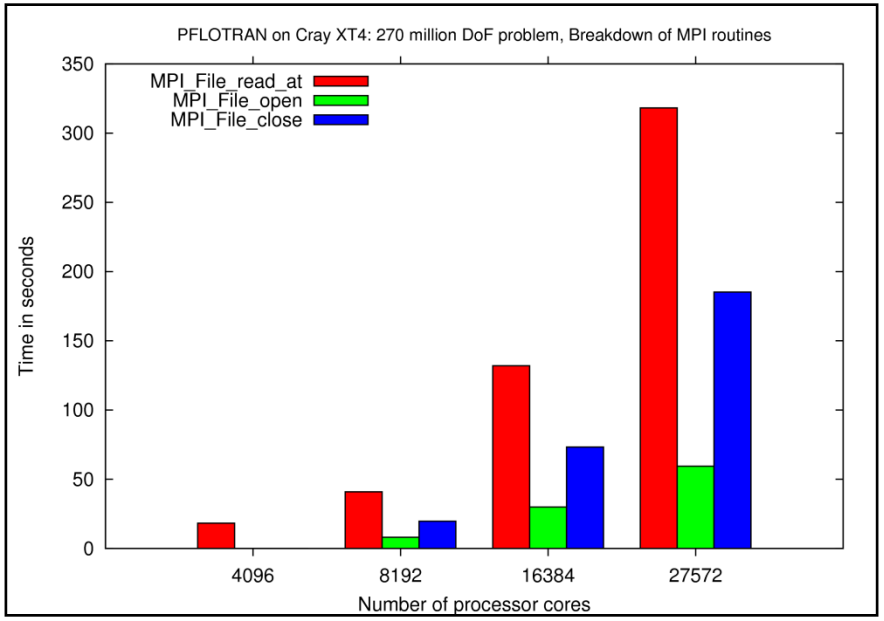

- Dominated by HDF5 Read I/O.
- Upgraded configurations of Lustre gave slightly better performance.
- But, overall BG/P outperforms XT5 by a huge margin.



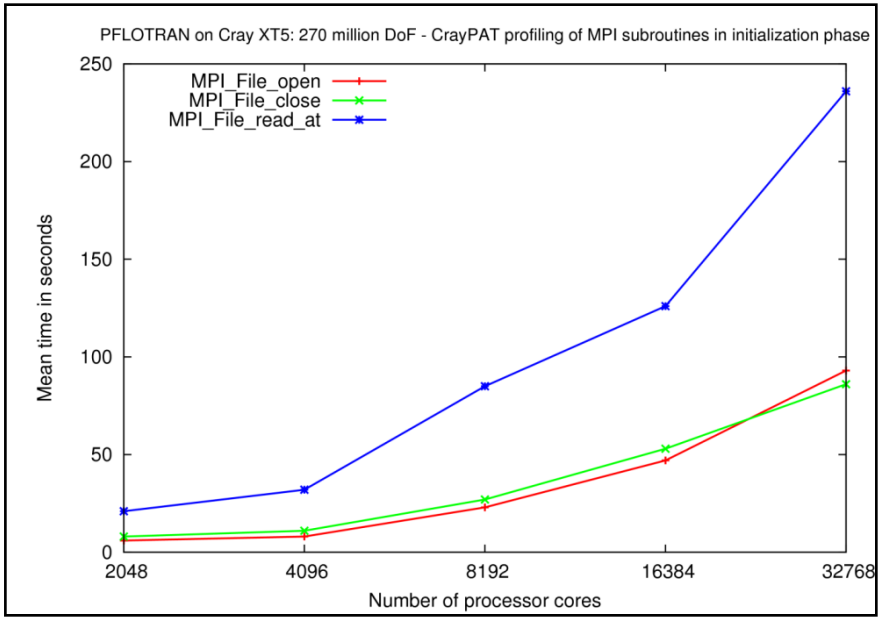
Computer Science
NC STATE UNIVERSITY

# Initialization stage: Analysis

Cray XT4

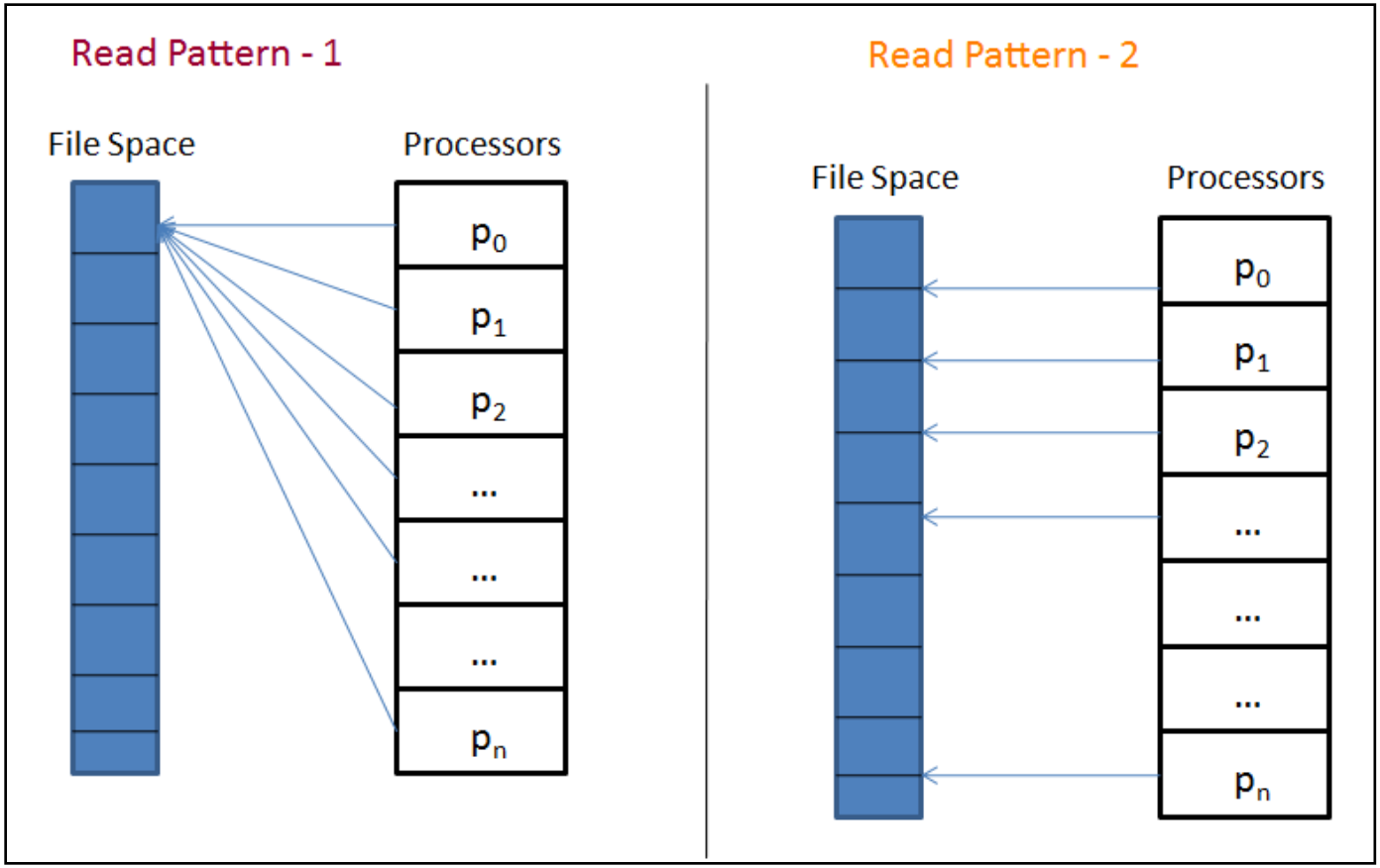



Cray XT5




- All process participate in parallel read
- File open/close expensive at scale
- Individual read operations not efficient

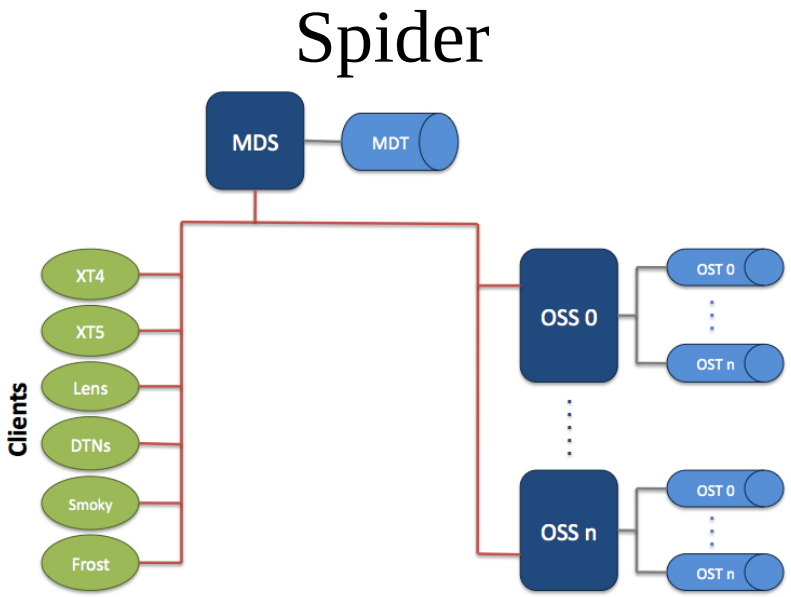
# Access Patterns: Default Method




- Pattern-1 called 18 times and Pattern-2 called 2 times

# I/O Performance bottlenecks

- Object based parallel file system
- 3 major components
  - Object Storage Servers (OSS)
  - Meta Data Servers (MDS)
  - File system clients

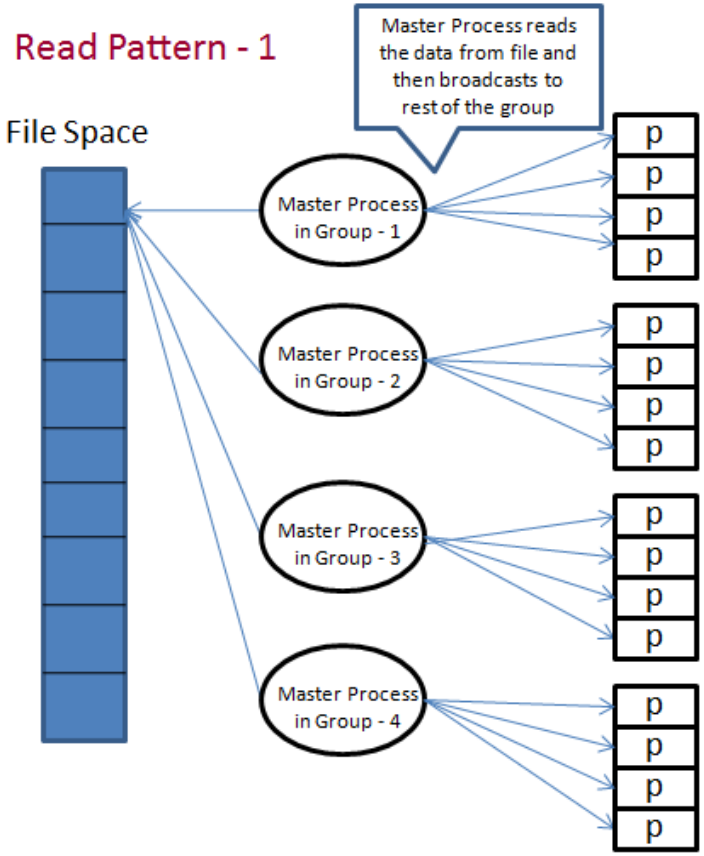


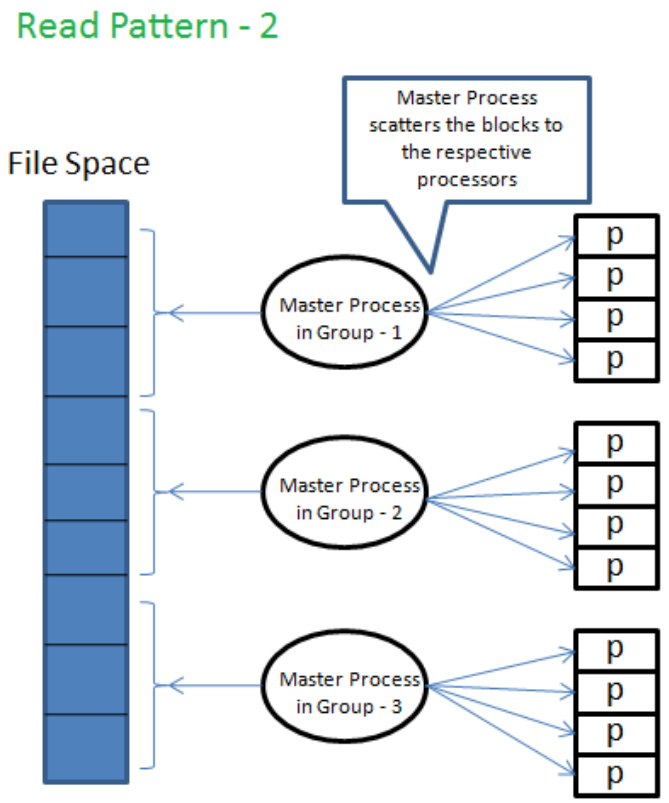

- All running jobs need to poll the MDS for file access
- Bottleneck at OST's with higher processor count

# Performance Optimization

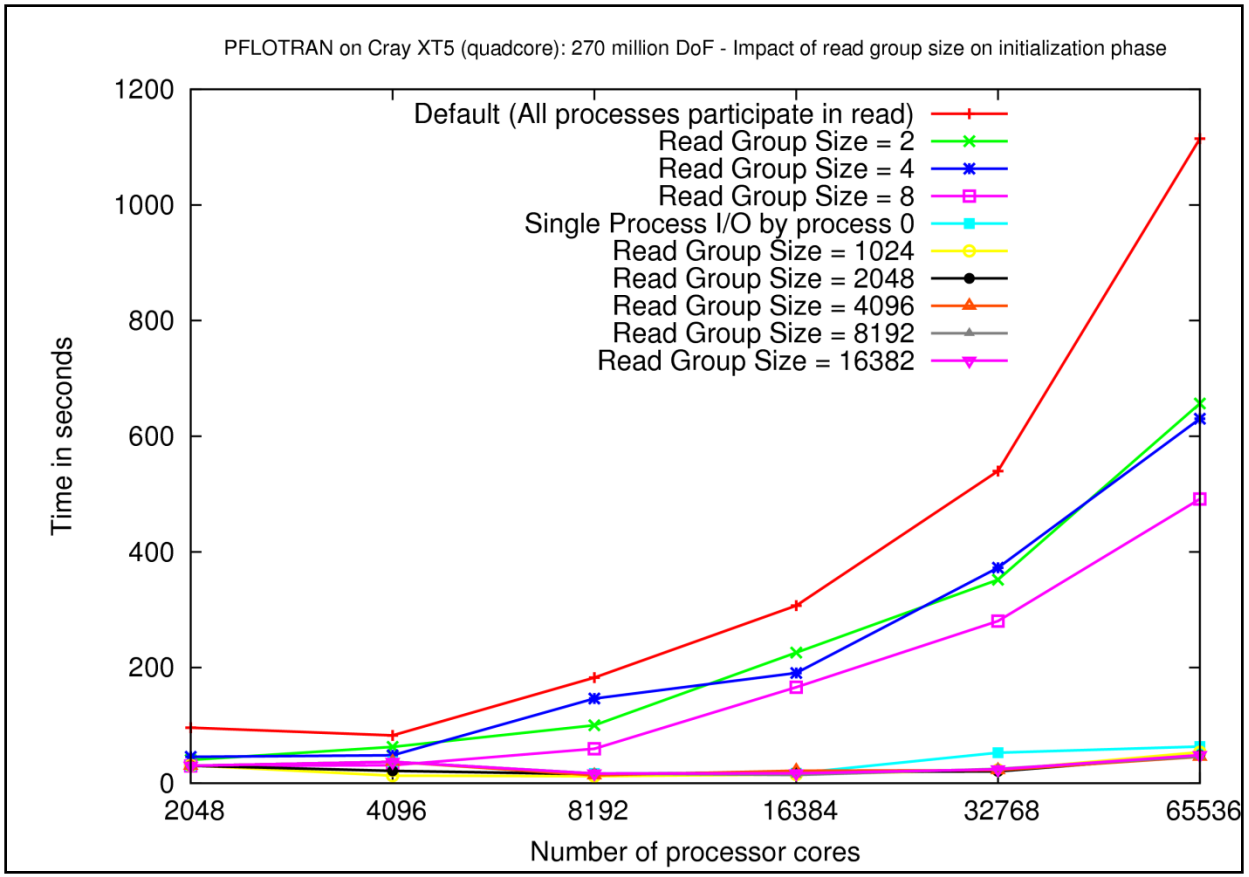
- Implement two-phase I/O approach
  - Split the MPI global communicator into multiple sub communicators.
  - The root process in each sub-communicator is responsible for performing the I/O operations for the entire group.
  - Communication phase: Root gathers start indices and length
  - I/O phase: Root perform reads and scatters data to group

# Our customized Access Patterns



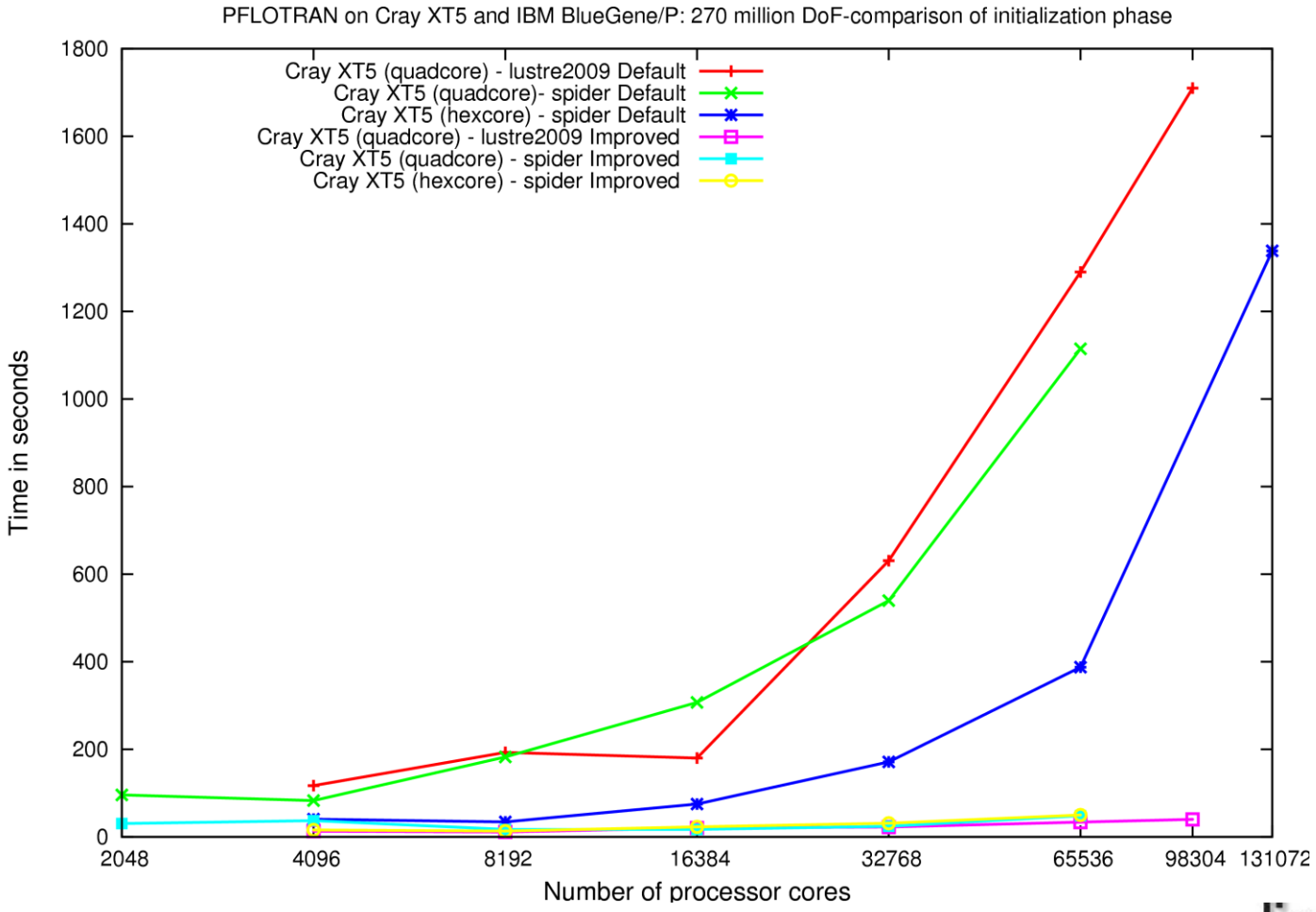



# Impact of group size

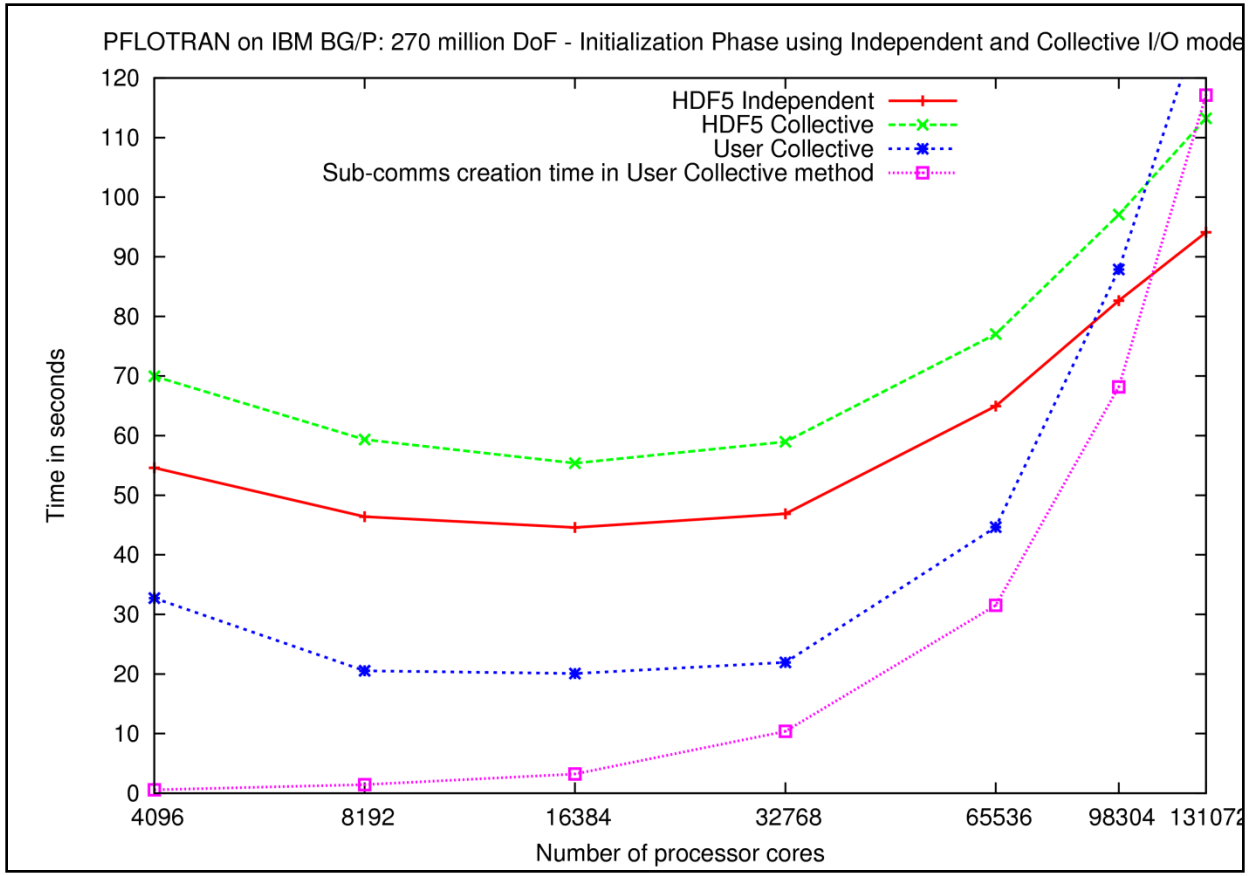



- Large group size implies fewer readers and better performance.
- 25X improvement over default method

# Improved performance on XT5

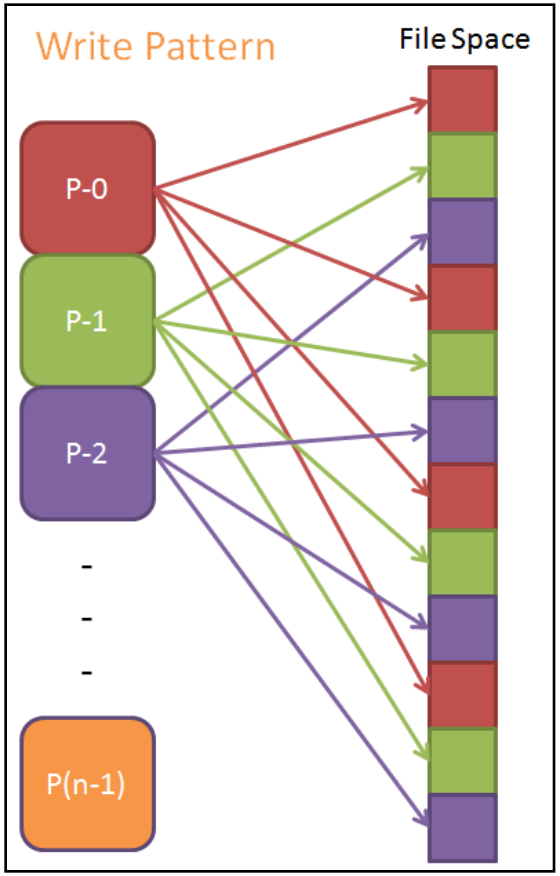


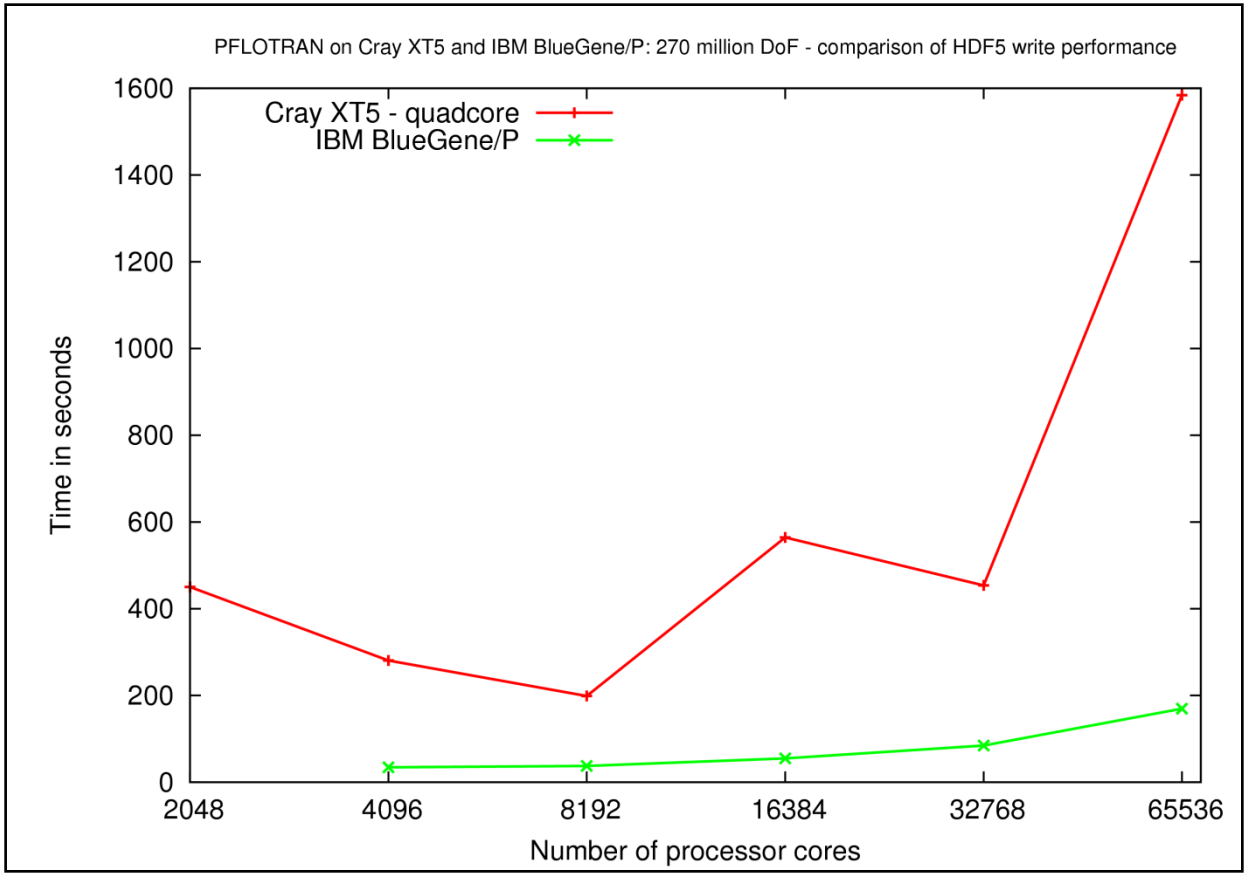
# Improved performance on BG/P



# Output Stage: Scaling

Default

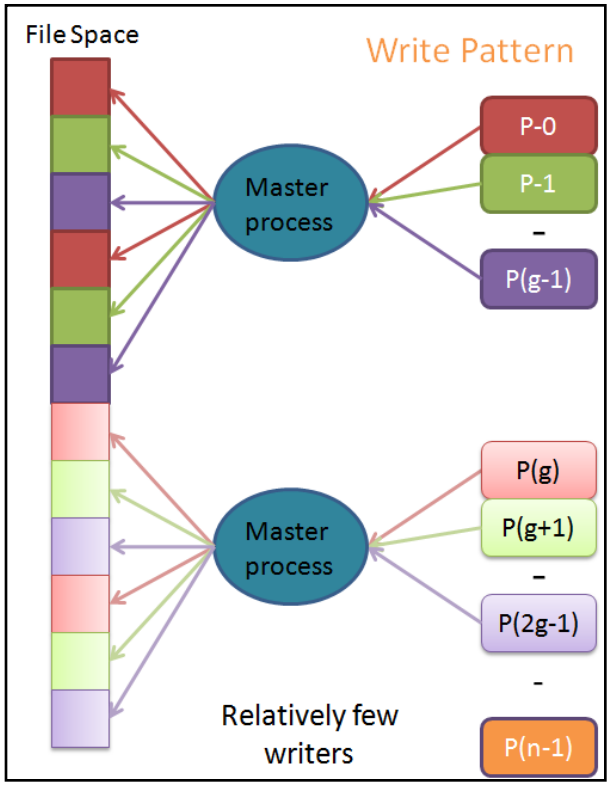


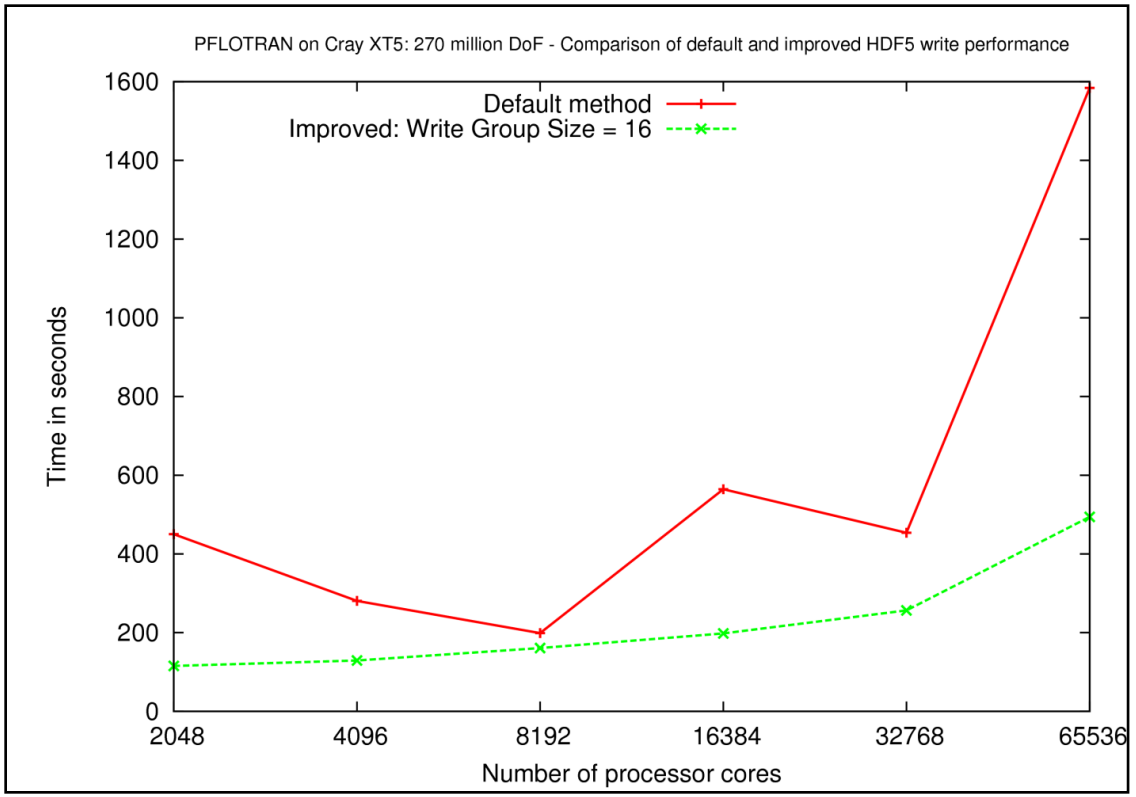




- Default uses HDF5 Collective I/O mode

# Write performance improvement

Our modified approach



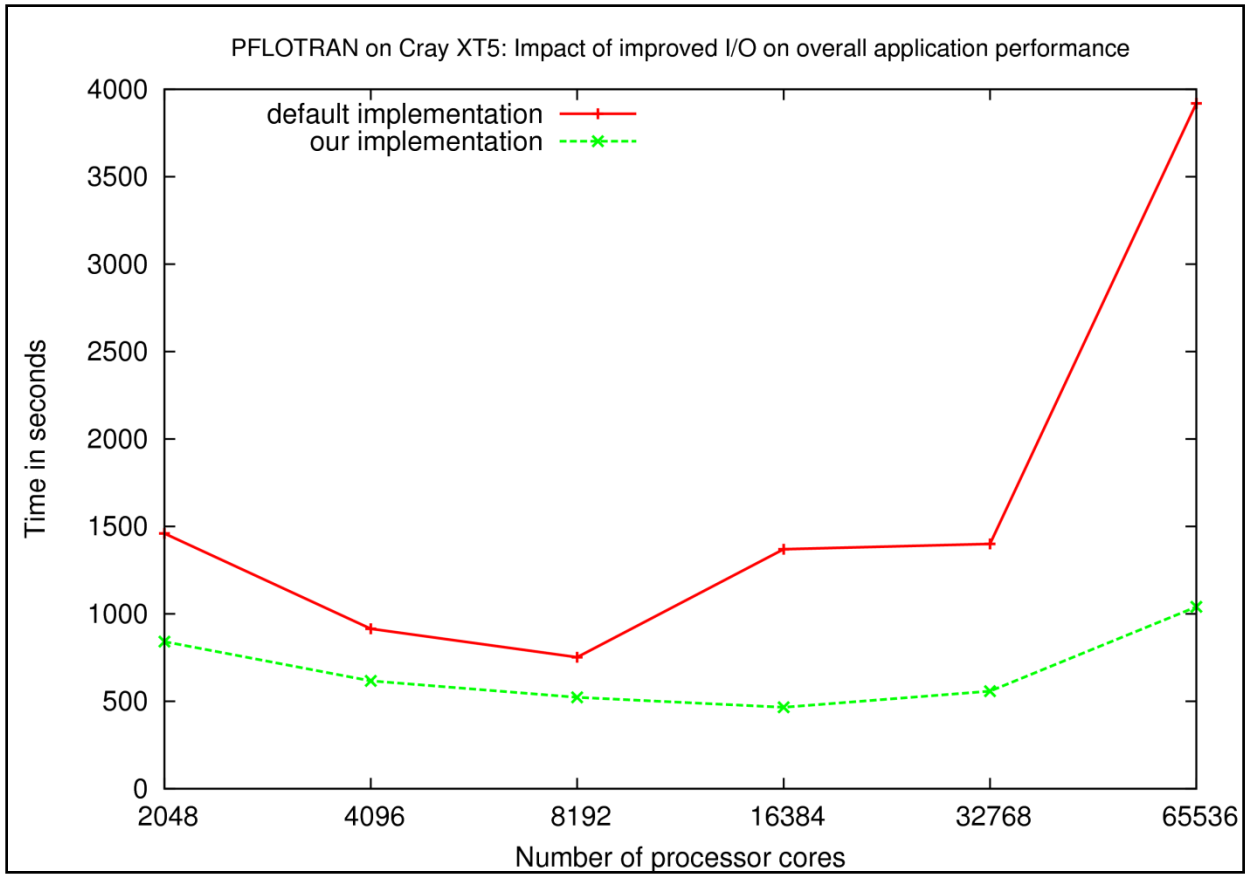




- 3X improvement over default method



Computer Science
NC STATE UNIVERSITY

# Overall Improvement on XT5



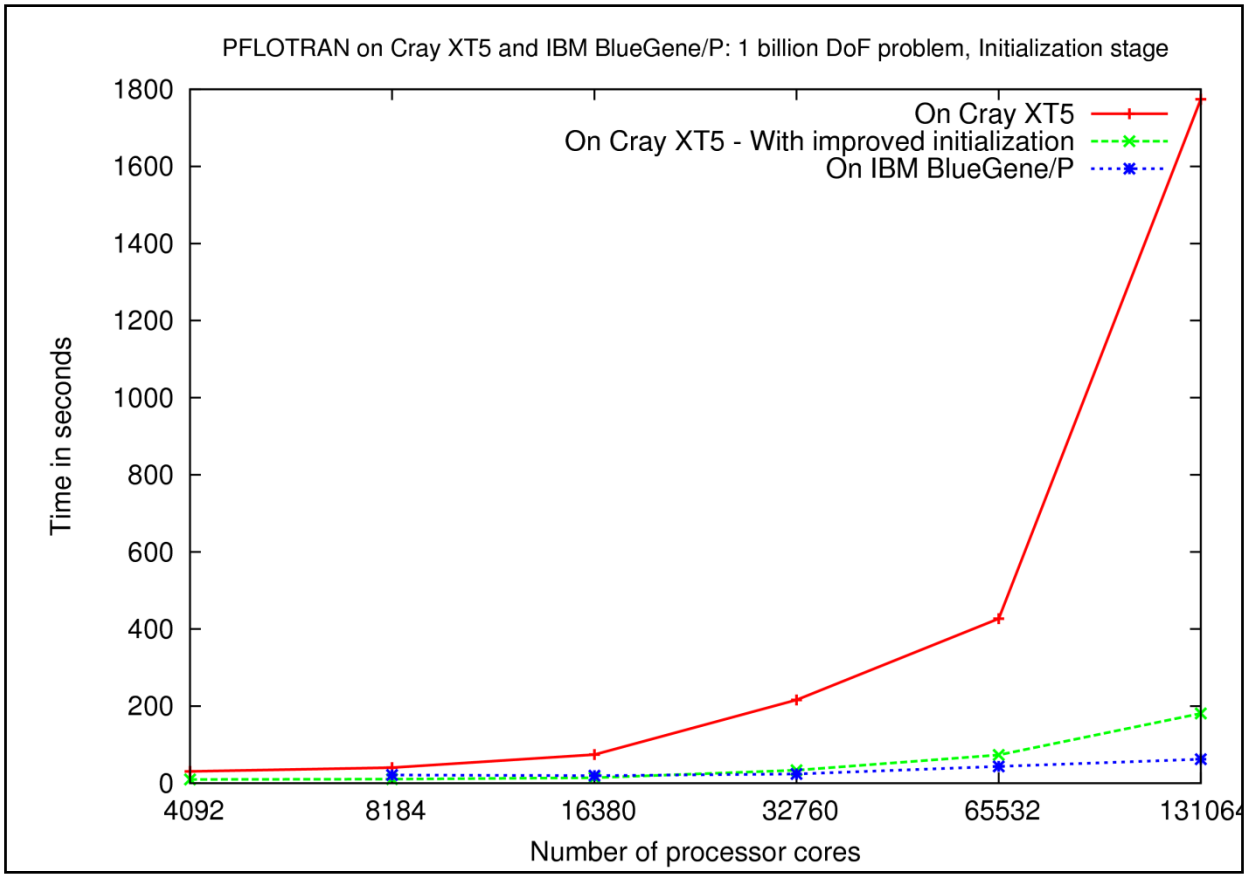

- 5X improvement for entire application



Computer Science
NC STATE UNIVERSITY

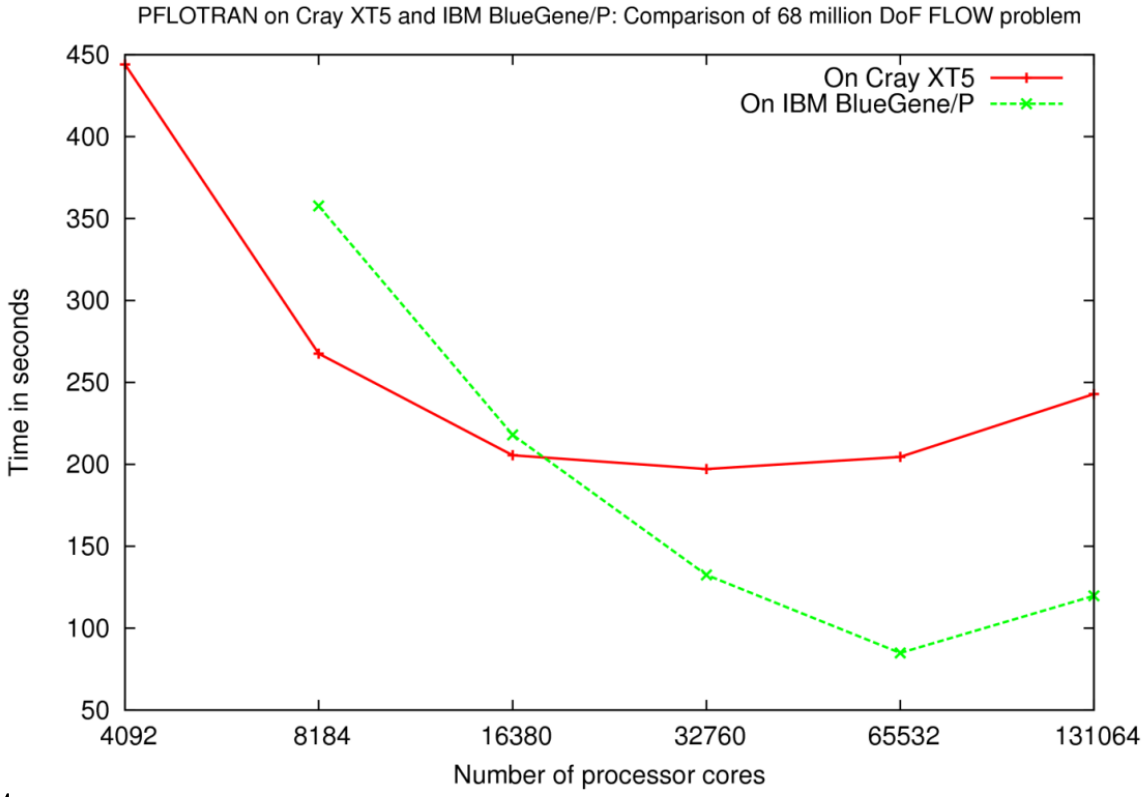
# Overall Performance Analysis

# Initialization stage



# Flow stage

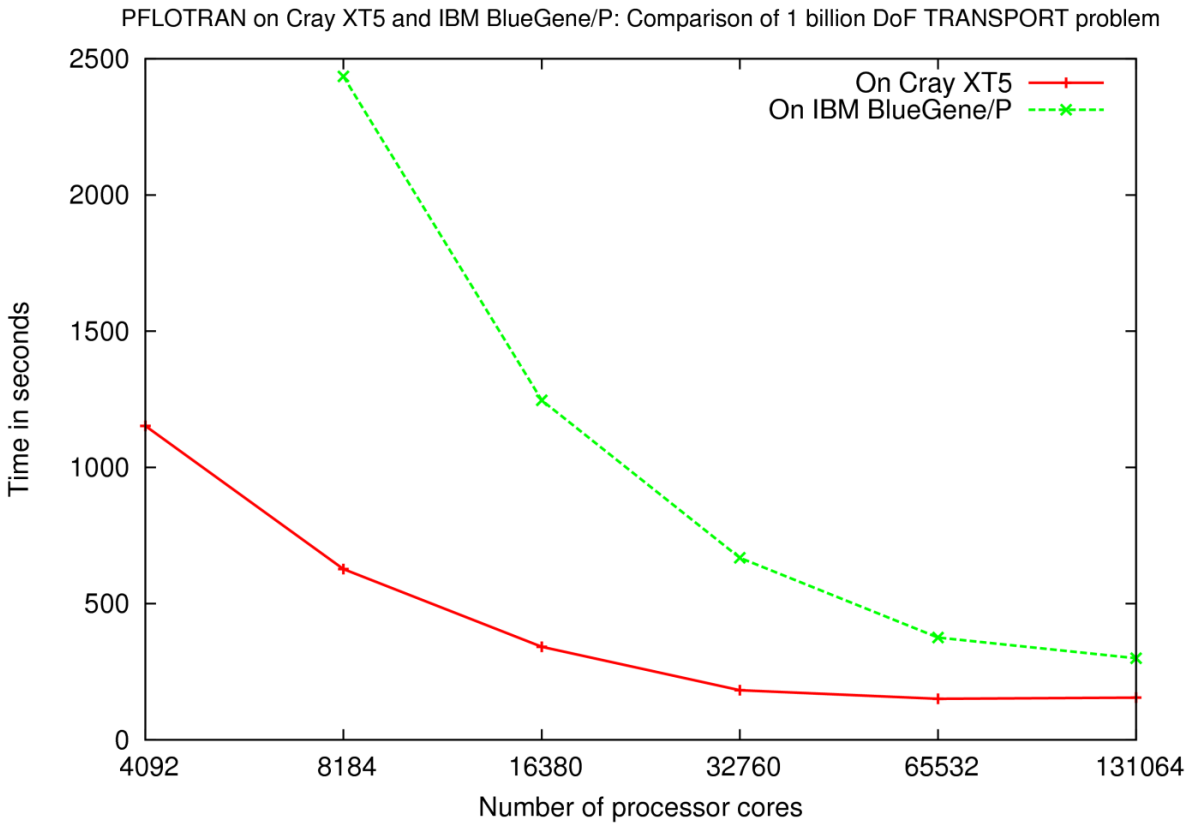
- Large number of linear iterations (31,167 – 35,667)
- Increase in iteration count with processor count because of preconditioner.
- Each iteration performs 4 MPI_Allreduces
- Computation/Comm. Ratio is low for flow at higher processor count.




- BG/P crosses XT5 after 16k.

# Transport stage

- Computation intensive because of reaction functions.
- Similar scaling pattern but different single node performance.

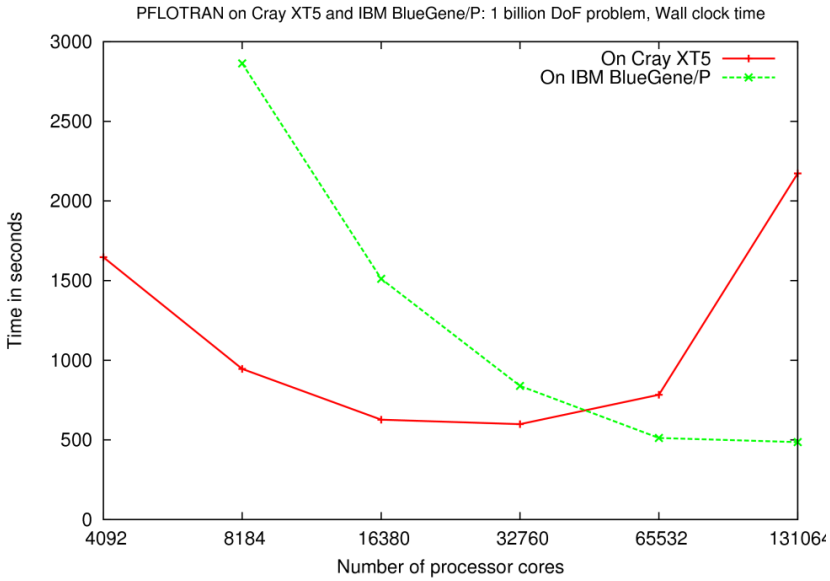




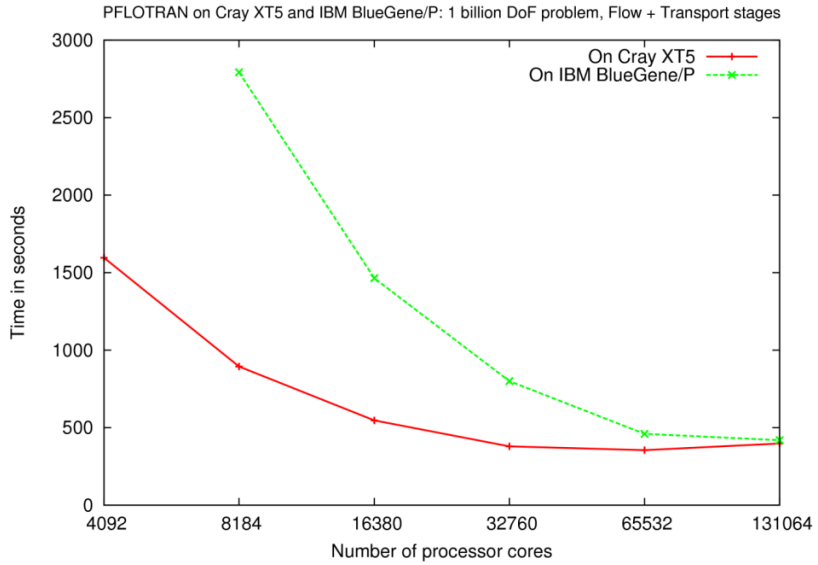

Computer Science
NC STATE UNIVERSITY

# Overall scalability

## Wall Clock Time




## Flow and Transport Stages






Computer Science
NC STATE UNIVERSITY

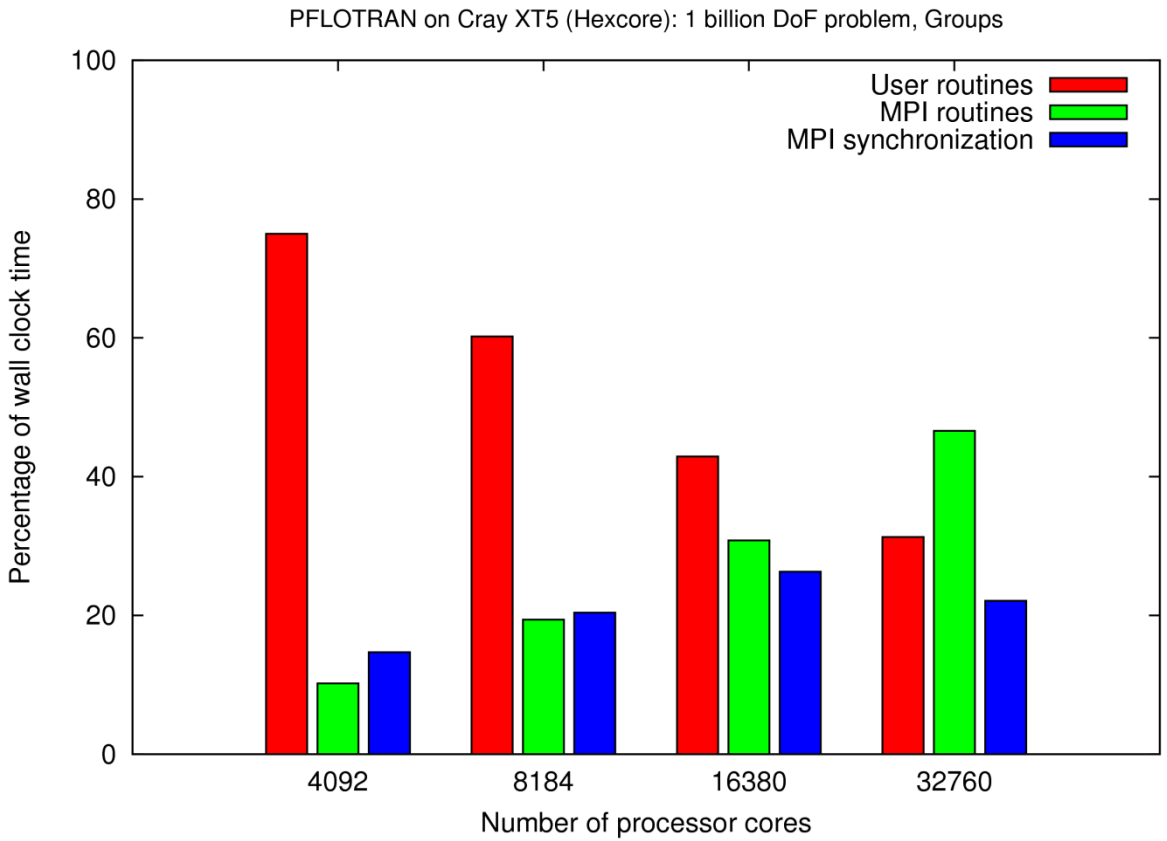
# Performance Analysis Tools

- CrayPAT
  - Binary instrumentation
  - Captures MPI synchronization time
  - Apprentice2 GUI
- Tuning and Analysis Utilities (TAU)
  - Source level instrumentation
  - Paraprof GUI
- Selective instrumentation of PFLOTRAN & PETSc routines to minimize profiling overhead

# Profiling Groups

- USER, MPI and MPI_SYNC groups

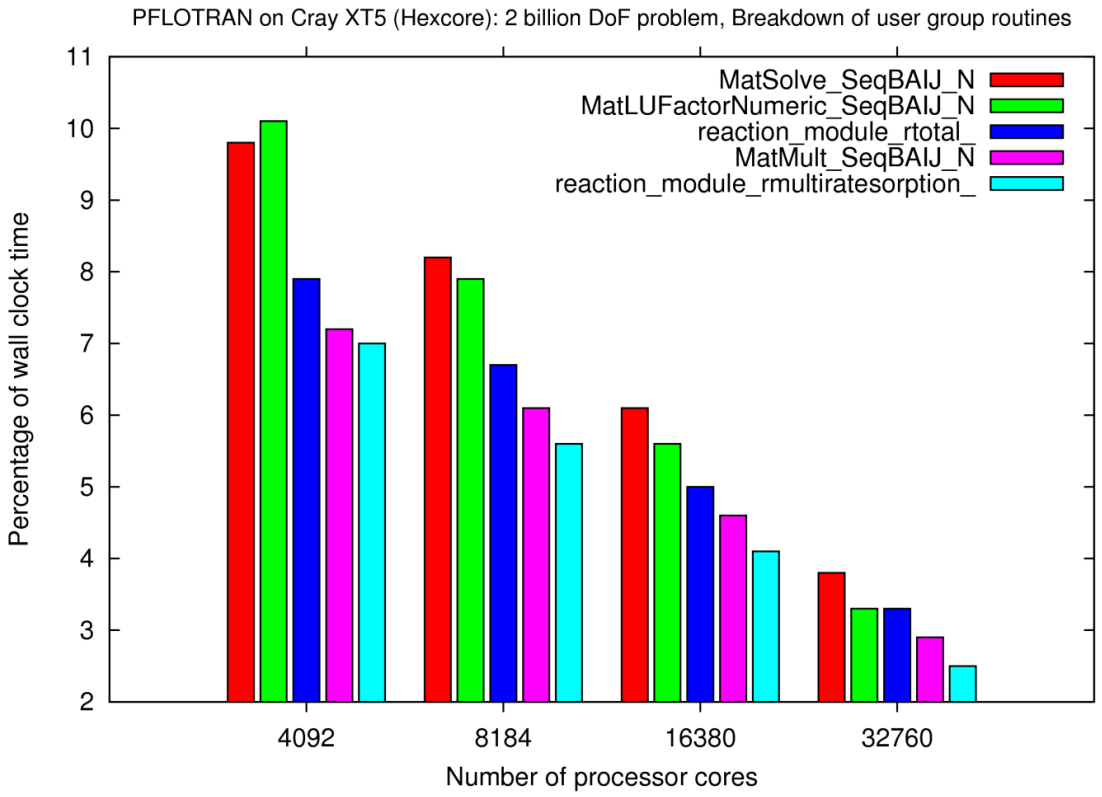
- Increase in MPI & MPI_SYNC group times with scale



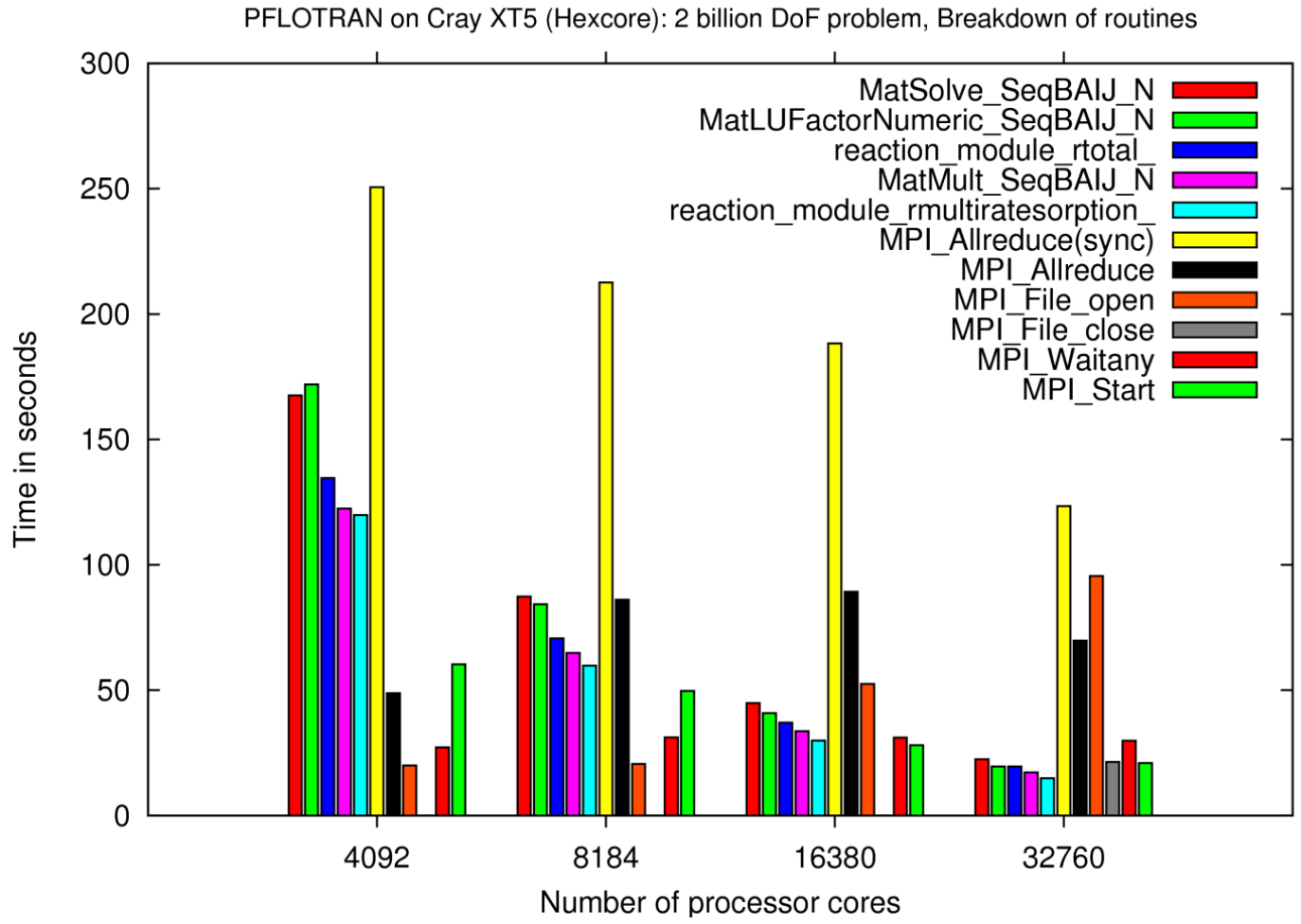
# User Routines

- Dominant routines:
  3 PETSc,
  2 native PFLOTRAN

- All 5 routines belong to Transport stage.



# User & MPI Routines

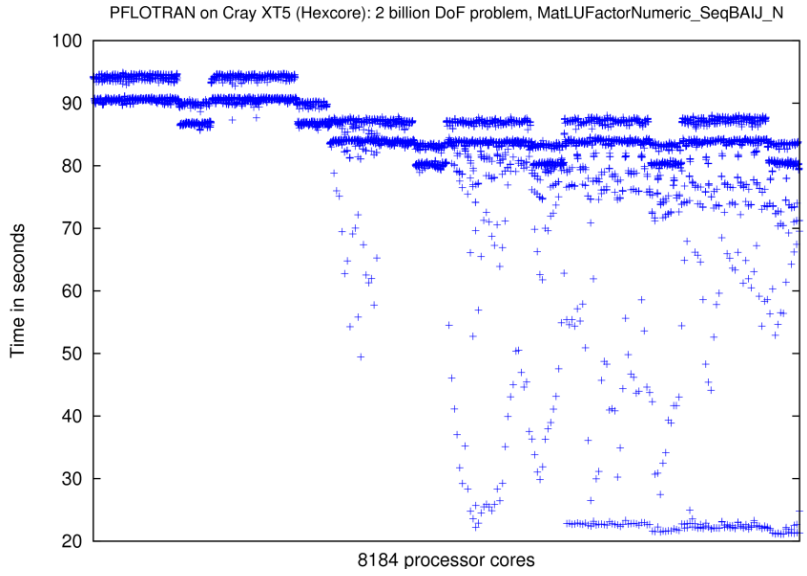


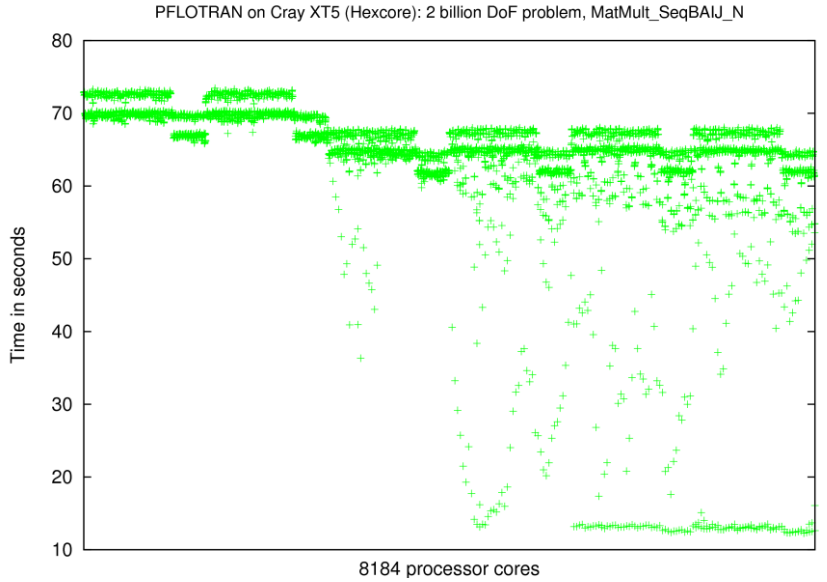

- MPI_Allreduce & File operations are scalability bottlenecks

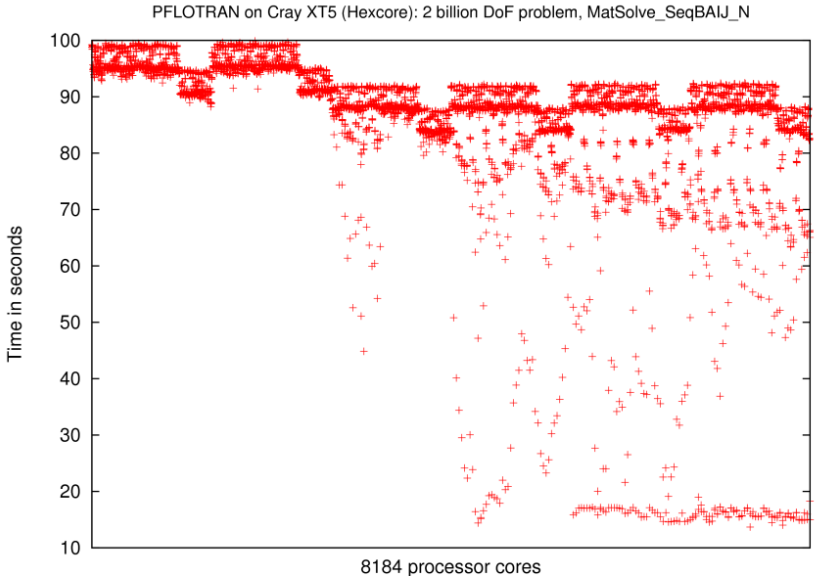
# MPI_Allreduce call sites at 8k cores

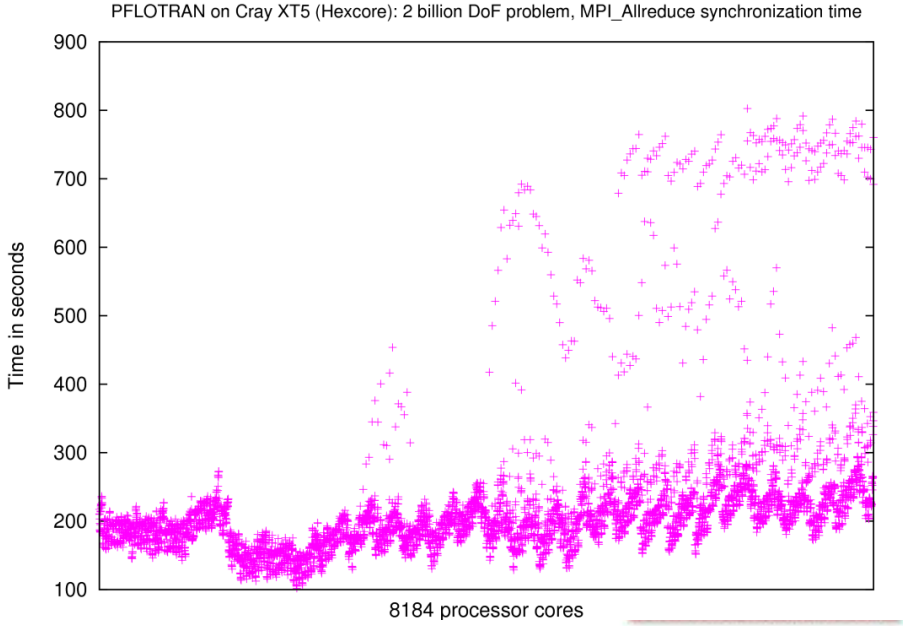
| Message Size | Count | Call site |
|---|---|---|
| 8 bytes | 113,070 | VecDot MPI, VecNorm MPI etc., |
| 16 bytes | 32,725 | VecDotNorm2 |
| > 4KB | 943 | MatZeroRows MPIBAIJ, MatZeroRows - MPIAIJ, MatAssemblyBegin MPIBAIJ, MatAssemblyBegin MPIAIJ |

# Cray XT5 : Load Imbalance



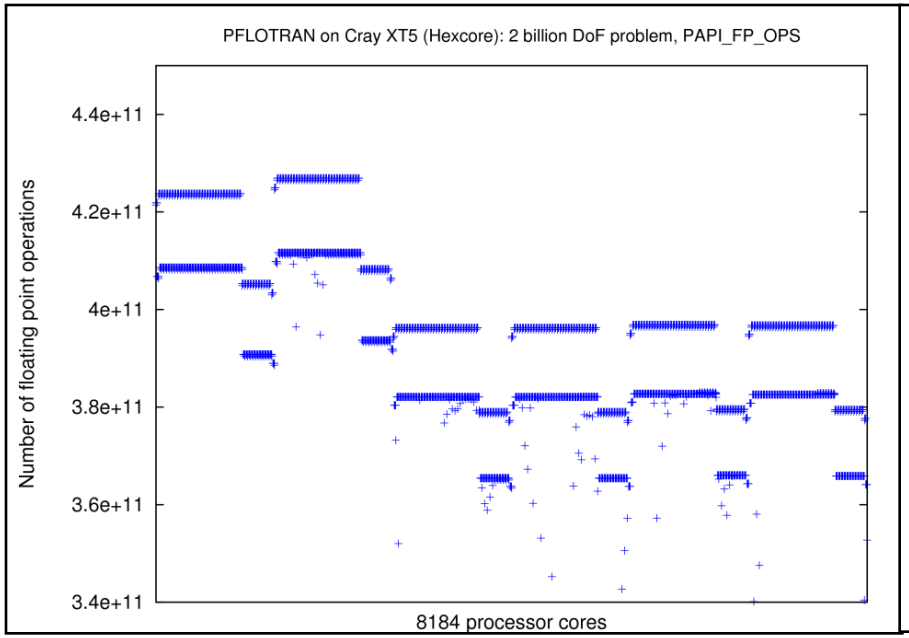



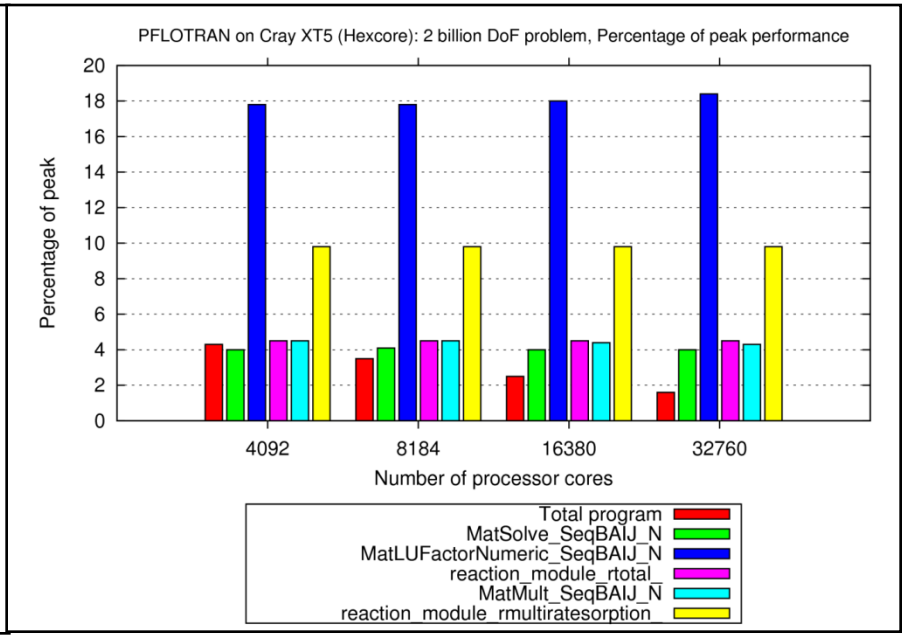



# Cray XT5: Single node

PAPI_FP_OPS for each core at 8k cores



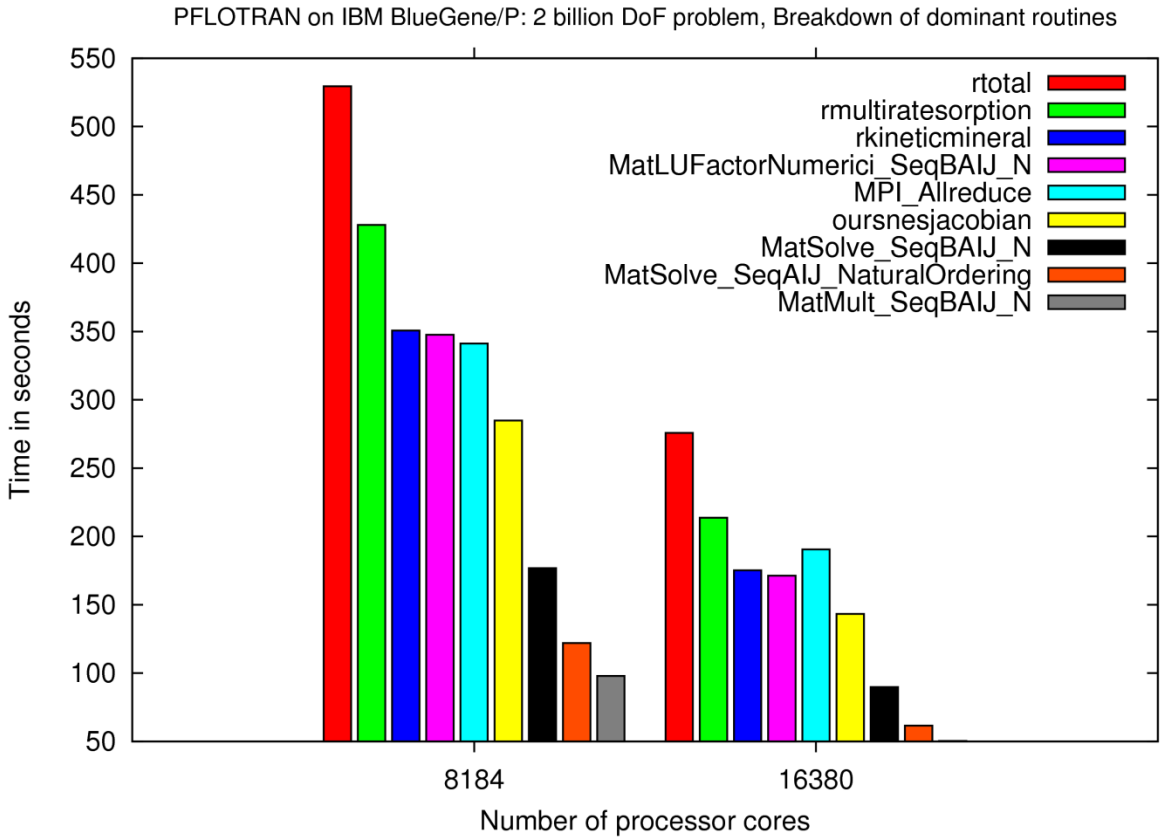

Percentage of Peak Performance





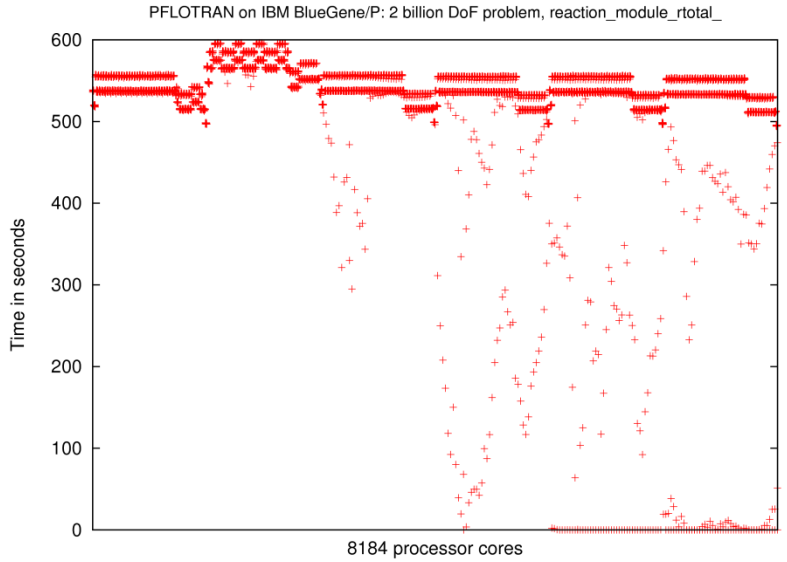

Computer Science
NC STATE UNIVERSITY

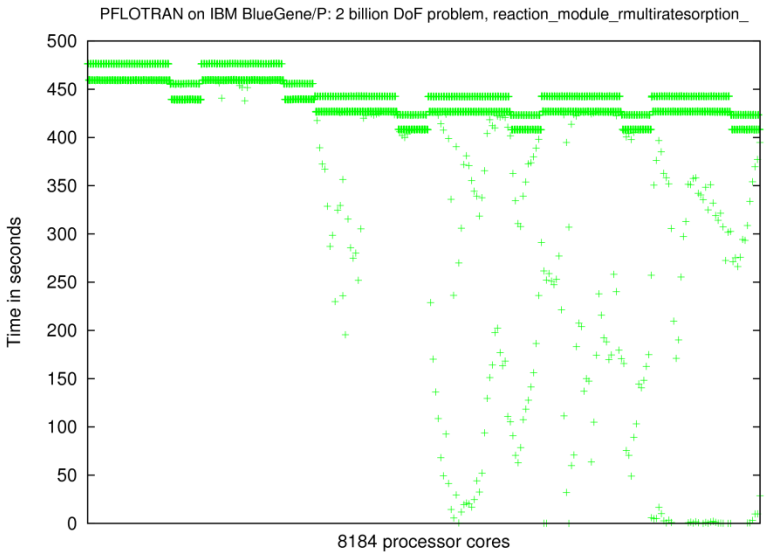
# Dominant routines on IBM BG/P

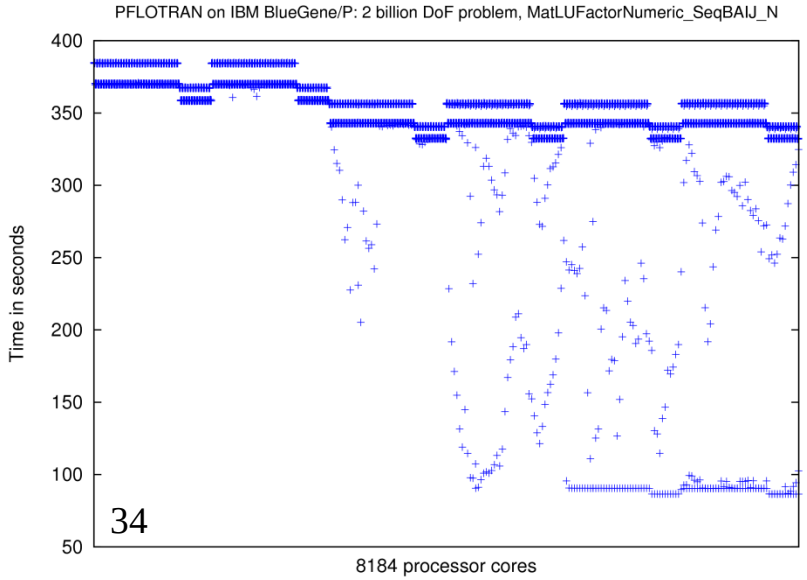


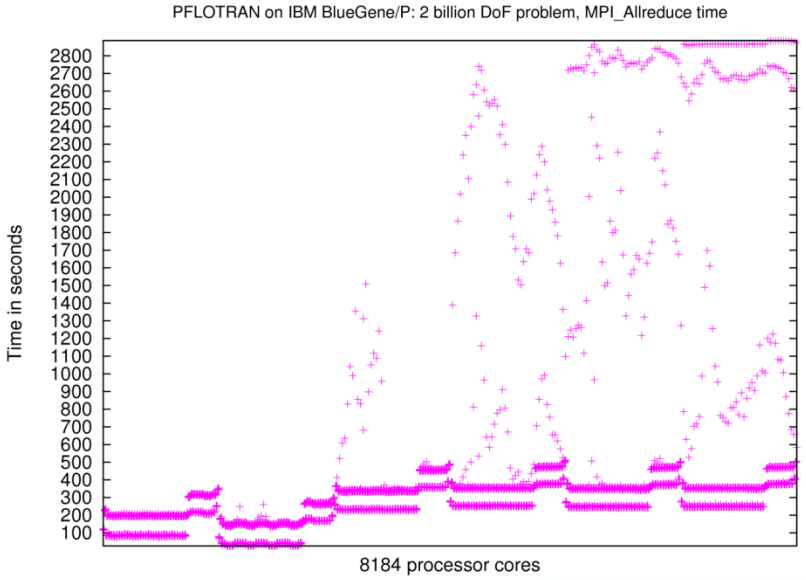

- Dominant routines differ from XT5 because BG/P has slower compute nodes

# IBM BG/P: Load Imbalance

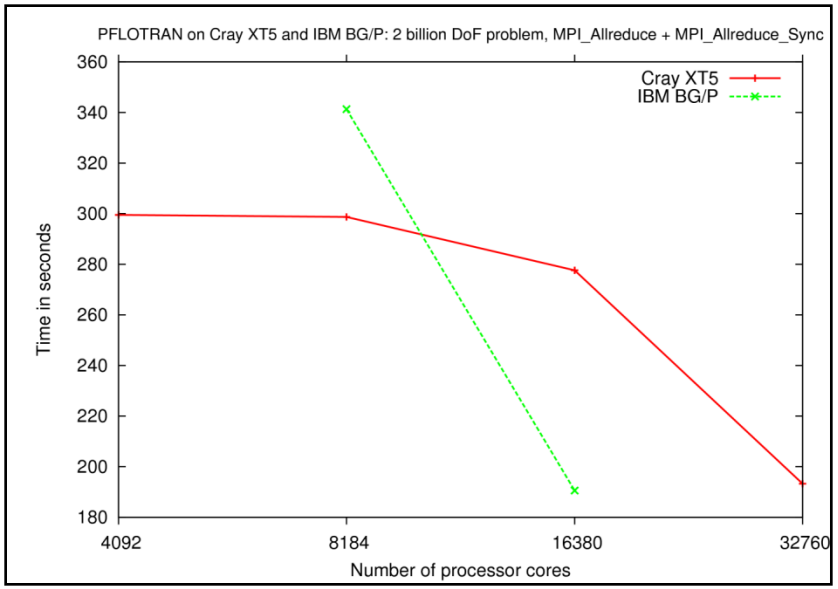


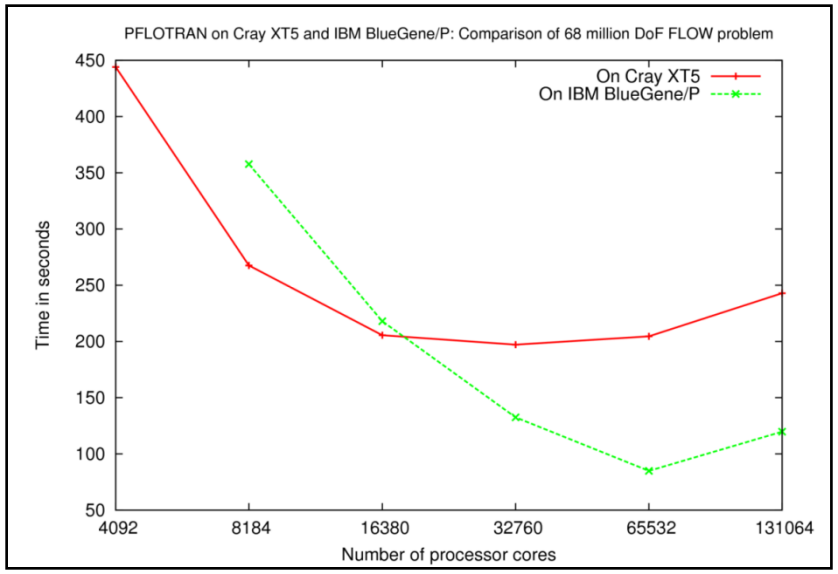



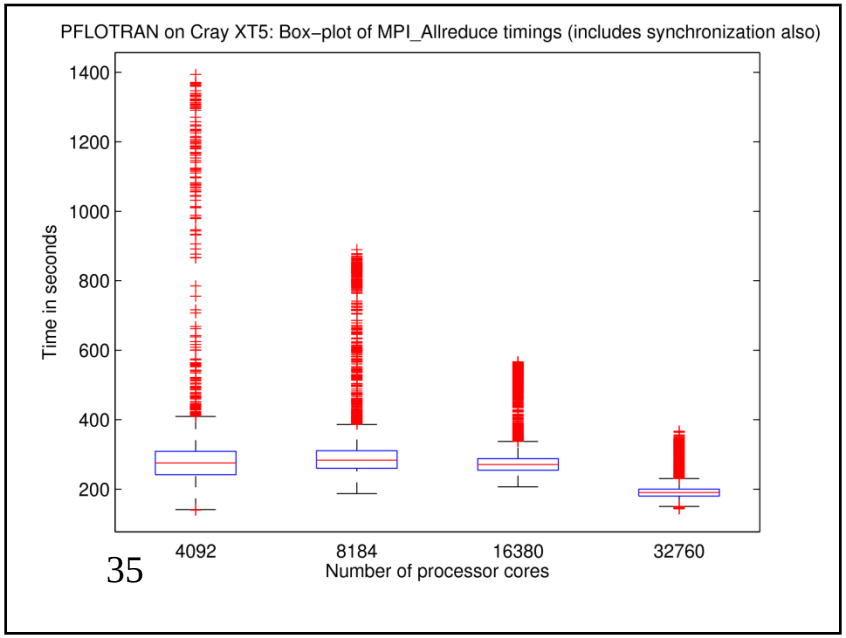



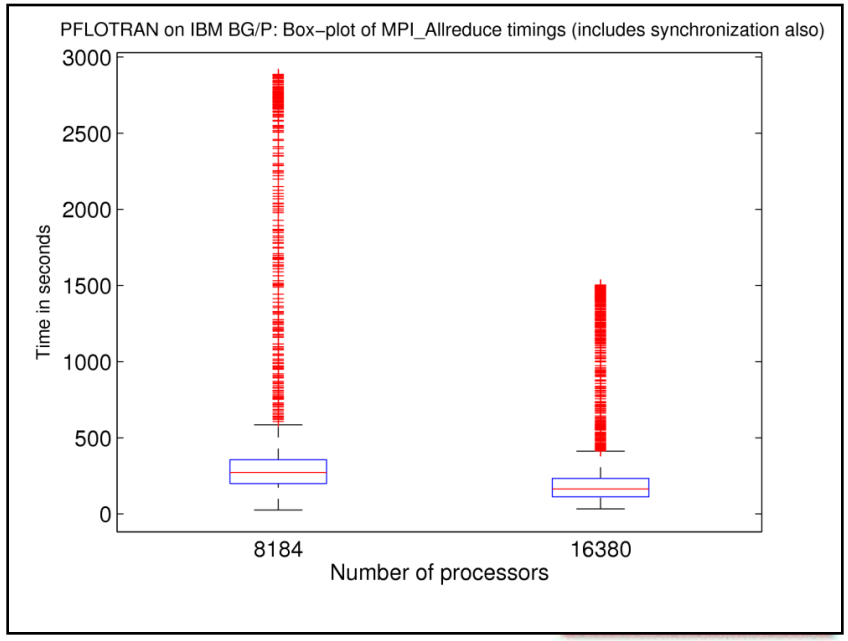



# XT5 and BG/P: MPI_Allreduce



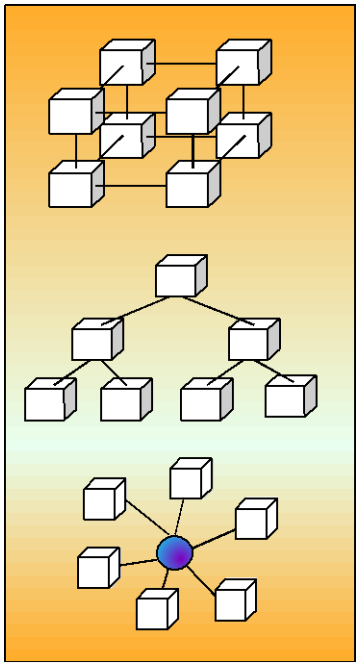



# Custom MPI_Allreduce benchmark

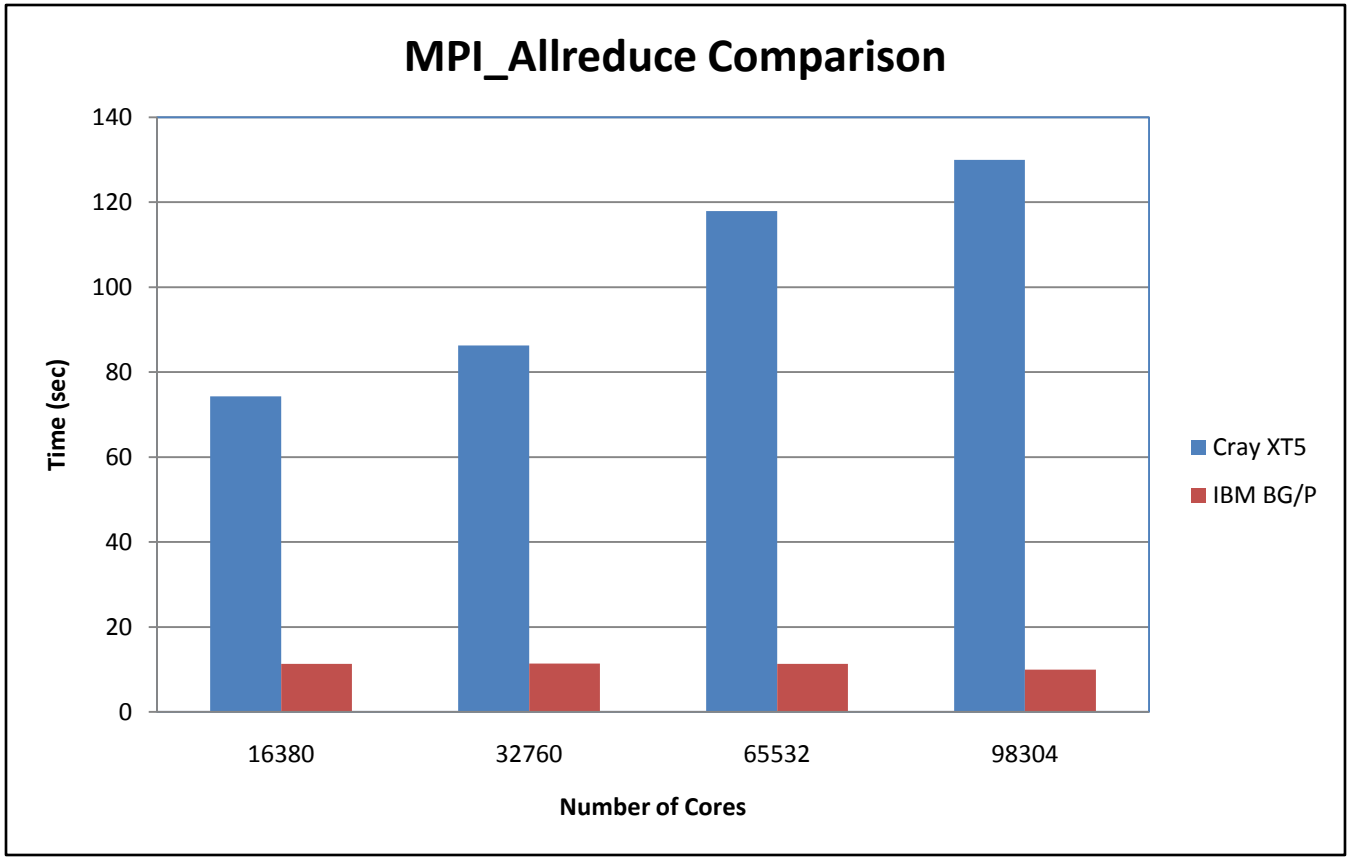
- Vector dot product
- No load imbalance
- 150,000 MPI_Allreduce
- Different communication mechanisms
  - Direct (MPI_Allreduce on Global Comm)
  - Sub-comms (MPI_Reduce on intranode comm, MPI_Allreduce on internode comm, MPI_Bcast - intranode)
  - Asynchronous (MPI_Isend/Irecv within node, MPI_Allreduce on internode comm)
  - Hybrid (OpenMP within node, MPI_Allreduce on global comm)

# XT5 and BG/P: Communication N/W

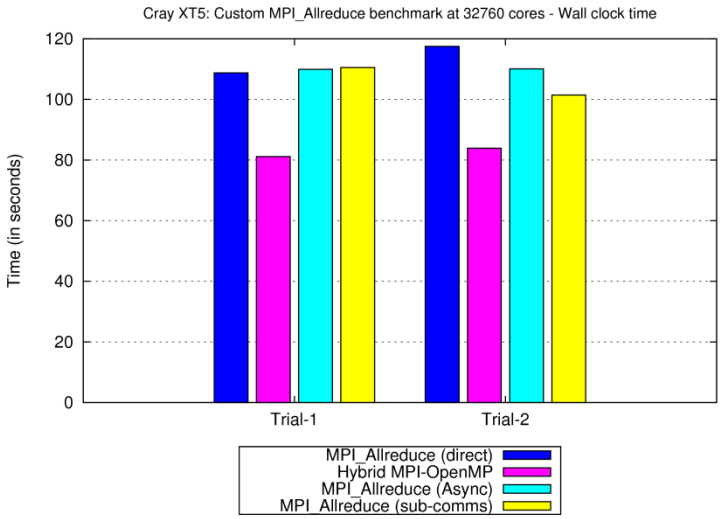
- BG/P
  - 3D Torus
    - Point-to-point communications
    - 5.1 GB/s bidirectional b/w per node
  - Global Collective
    - Collectives, Reductions
    - 5.1 GB/s bidirectional b/w per node
  - Global Barrier
    - Used for interrupts, barriers
- Cray XT5
  - 3D Torus
  - Both for point-to-point & collectives.
  - 57.6 GB/s bidirectional  b/w per node

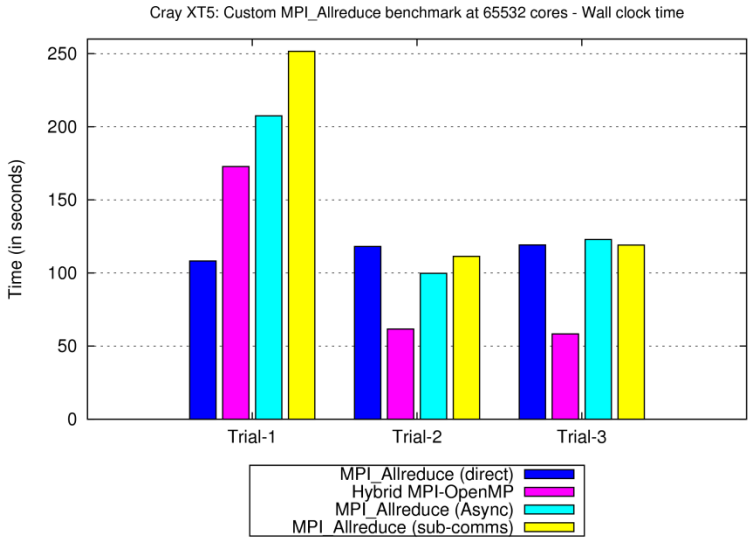


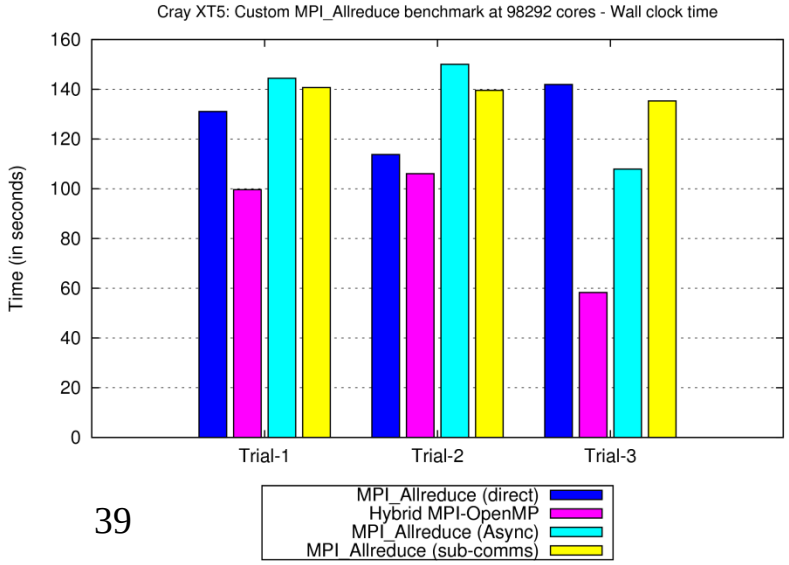
# MPI_Allreduce: XT5 vs. BG/P

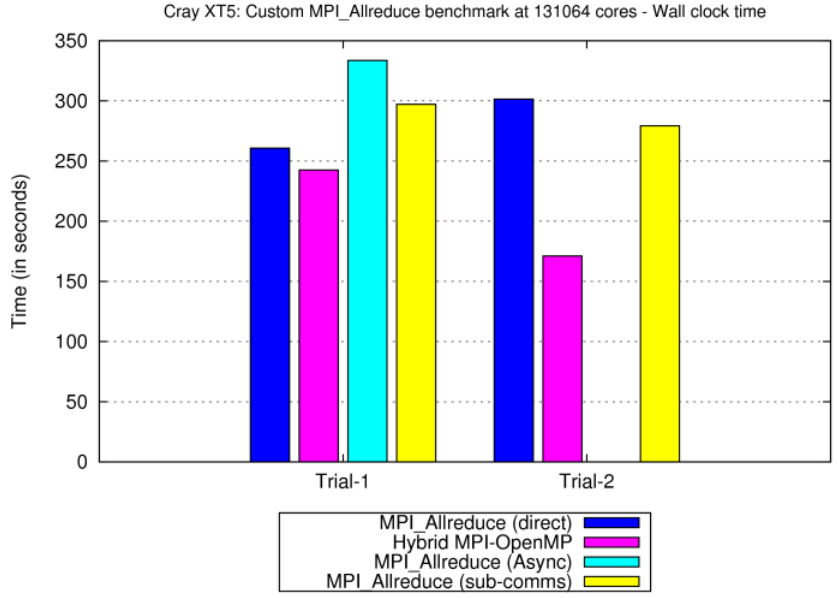


# MPI_Allreduce : Different Methods



# Conclusions

- Architectural implications should be considered during application design
  - On Cray XT5: Leverage computation in order to reduce communication impact
- IBM BG/P is recommended for communication intensive applications
- Cray XT5 is better for computation intensive applications
- Significant system variability on Cray XT5, whereas BG/P delivers stable performance
- MPI_Allreduce has room for improvement on Cray XT5
- Parallel File Systems: Optimal readers/writers required at scale.
  - Two phase I/O approach recommended.

# Acknowledgements


- This work is part of Performance Engineering Research Institute (PERI) and is funded by the U.S. Department of Energy Office of Science, as part of its second Scientific Discovery through Advanced Computing (SciDAC-2) research program.
- This research used resources of the National Center for Computational Sciences at Oak Ridge National Laboratory, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC05-0OR22725.
- This research used resources of the Argonne Leadership Computing Facility at Argonne National Laboratory, which is supported by the Office of Science of the U.S. Department of Energy under contract DE-AC02-06CH11357.




SciDAC
Scientific Discovery through Advanced Computing

Computer Science
NC STATE UNIVERSITY