Evaluation of UPC on the Cray X1E

Cray User Group 2006, Lugano, Switzerland
May 10, 2006

Richard F. Barrett

Oak Ridge National Laboratory
Oak Ridge, TN 37831

http://www.csm.ornl.gov/ft
http://www.nccs.gov

Yiyi Yao and Tarek El-Ghazawi

The George Washington University
And

http://www.csm.ornl.gov/ft
http://www.nccs.gov

CUG 2005

"Evaluation of UPC on the Cray X1"

Tarek A. El-Ghazawi, François Cantonnet, Yiyi Yao, and Jeffrey Vetter

Since then,

1. X1@ORNL => X1E
2. NPB-UPC: pragmas inserted.

X1E at ORNL: Phoenix

- 1024 Multi-streaming vector processors (MSP)
- Each MSP
  - 4 Single Streaming Processors (SSP)
  - 4 scalar processors (400 MHz)
  - Memory bw is roughly half cache bw.
  - 2 MB cache
  - 18 GFLOP peak (~18.5 TFLOPS)

- 4 MSPs form a node
  - 8 GB of shared memory.
  - Inter-node load/store across network. 56 cabinets

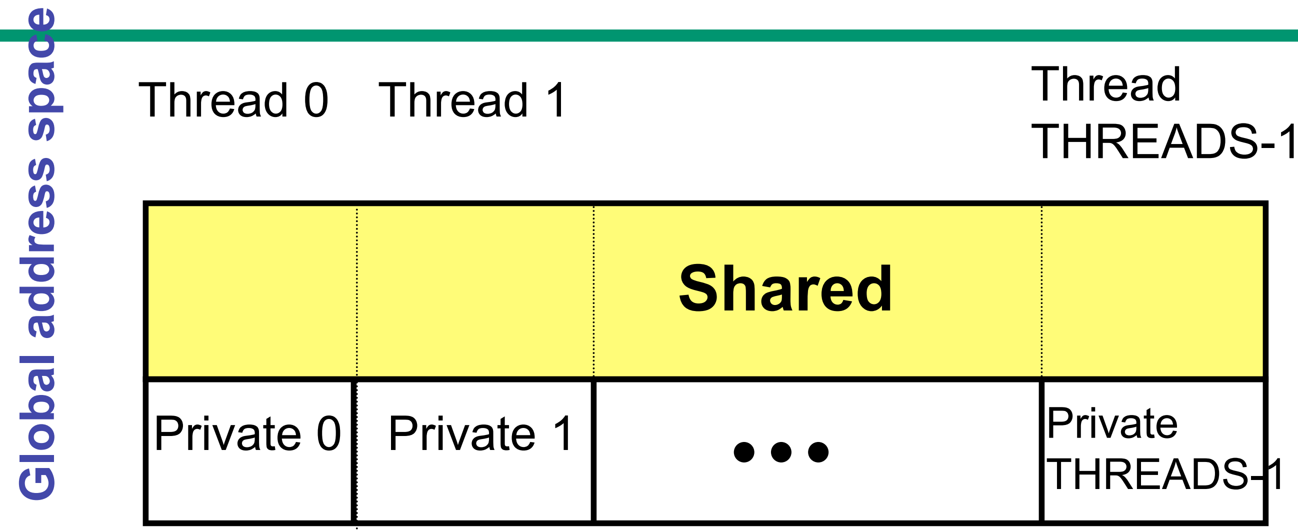
Memory Latency

Memory location	*Relative access time*
D-cache	1X
E-cache	2X
Local (node) memory	7X
Remote (off node) memory	10X-32X

A Brief History of UPC

- Extension of ISO C
- Partitioned global address space language. (PGAS)
  - DSM programming model
  - SPMD execution model
- May 1999: Initial specification.
  - Tech report: *Carlson, Draper, Culler, Yellick, Brooks, and Warren.*
- May 2000: First UPC Consortium Meeting.
- Feb 2001: Spec v1.0
- Dec 2003: UPC Collectives spec.
- July 2004: UPC I/O spec.
- May 2005: v1.2
- Sept 2005: First PGAS meeting in Minneapolis.

UPC Memory Model



- A pointer-to-shared can reference all locations in the shared space, but there is data-thread affinity

UPC Execution Model

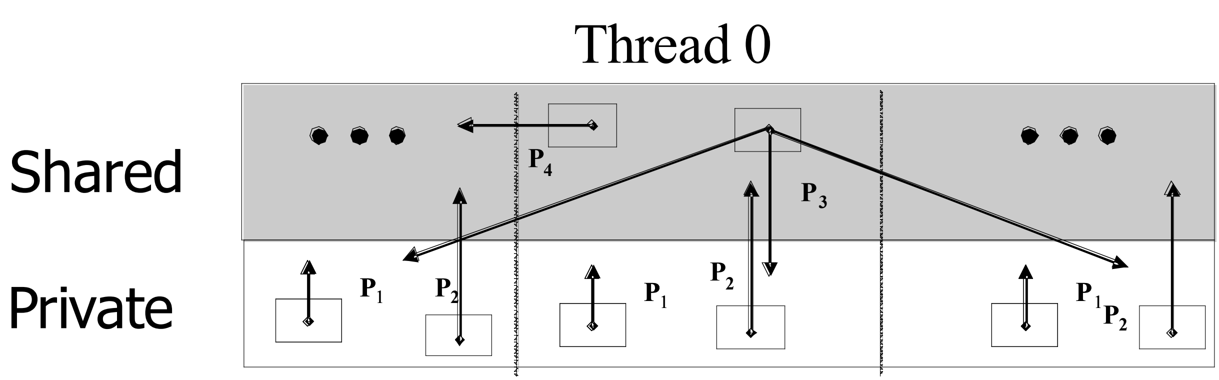
- A number of threads working independently in a SPMD fashion
  - MYTHREAD specifies thread index (0..THREADS-1)
  - Number of threads specified at compile-time or run-time
- Synchronization when needed:
  - Barriers
  - Locks
  - Memory consistency control

UPC Pointers

- How to declare them?

```
int *p1;              /* private pointer pointing locally                */
shared int *p2;       /* private ptr pointing into the shared space      */
shared int  *shared p3; /* shared ptr pointing into the shared space */
```

- You may find many using "shared pointer" to mean a pointer pointing to a shared object, e.g. equivalent to p2 but could be p3 as well.



Memory Consistency Models

- Has to do with ordering of shared operations, and when a change of a shared object by a thread becomes visible to others
  - Relaxed consistency: shared operations can be reordered by the compiler / runtime system
  - Strict consistency: enforces sequential ordering of shared operations. (No operation on shared can begin before the previous ones are done, and changes become visible immediately)

Memory Consistency Models

- User specifies the memory model through:
  - declarations
  - pragmas for a particular statement or sequence of statements
  - use of barriers, and global operations
- Programmers responsible for using correct consistency model

The NAS Parallel Benchmarks (NPB)

- Evolved from NASA applications (CFD)
- Strong scaling.
- Now MPI, OpenMP, HPF, Co-array Fortran, UPC, Java, Grid,…

http://www.nas.nasa.gov/Software/NPB/

NPB problem sizes (Classes)

```
if      (class == 'S')
    problem_size = 12;  dt = "0.015";   niter = 100;
else if (class == 'W')
    problem_size = 36;  dt = "0.0015";  niter = 400;
else if (class == 'A')
    problem_size = 64;  dt = "0.0015";  niter = 400;
else if (class == 'B')
    problem_size = 102; dt = "0.001";   niter = 400;
else if (class == 'C')
    problem_size = 162; dt = "0.00067"; niter = 400;
```

UPC version of NPB

- Modified from NPB2.4-MPI and NPB2.4-OMP
  - *F. Cantonnet, Y. Yao, and T. El-Ghazawi*

•*upc_forall, upc_barrier, upc_notify_wait, upc_lock, upc_memput, upc_memget, upc_reduce_sum*

•*Relaxed mode*

•*Future plans: Reduce reliance on global barrier using fence, organization, etc.*

Conjugate Gradient (CG)

- Computes an approximation to the *smallest eigenvalue* of an spd matrix.
- Unstructured grid computations requiring *irregular long-range communications*.

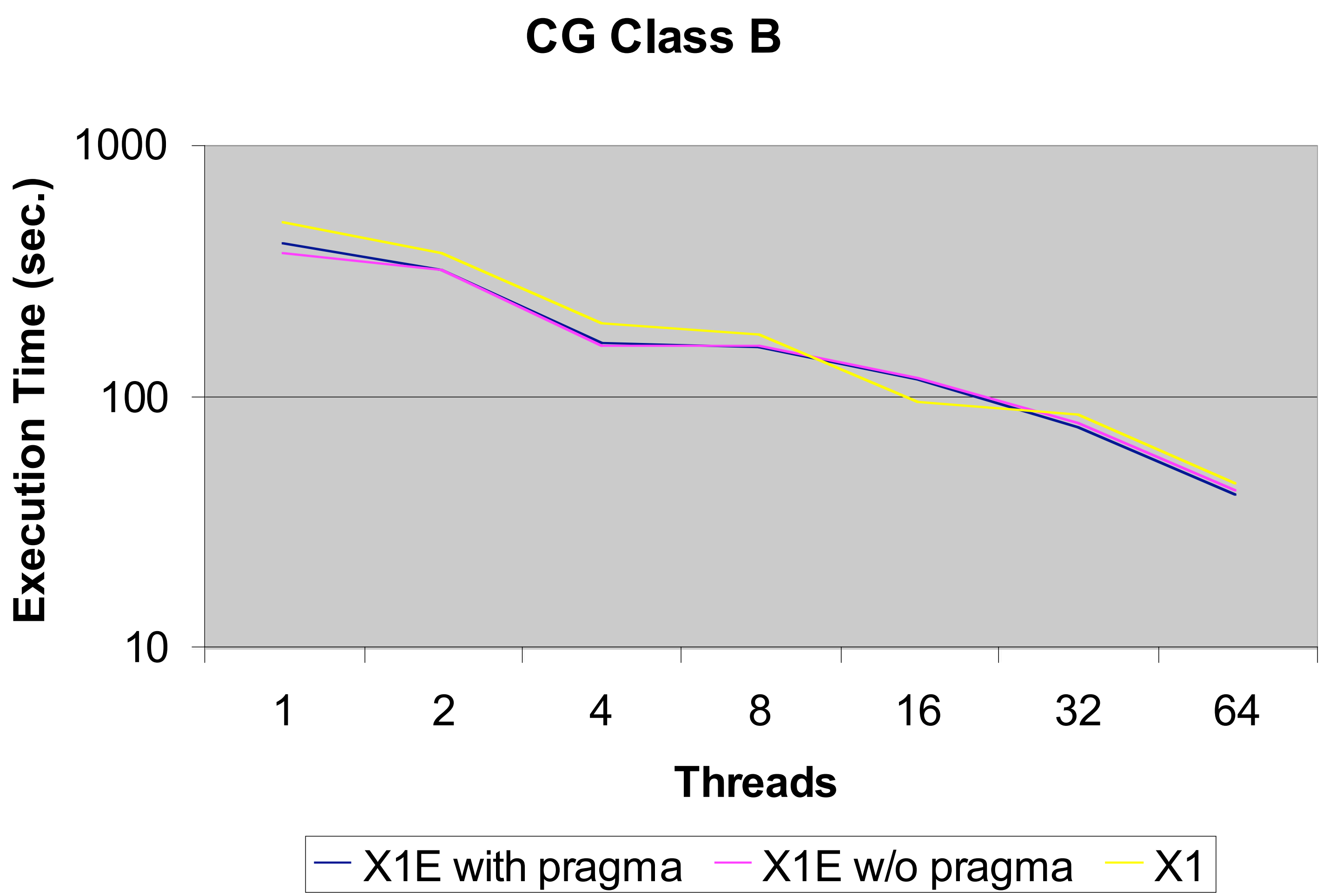
Loopmarking listing file (MG)
Before Pragmas

```
1155.  1 2 r-------<        for ( i1 = d1; i1 <= mm1-1; i1++)
1156.  1 2 r               {
1157.  1 2 r                 u((2*i3-d3-1), (2*i2-d2-1), (2*i1-d1-1)) =
1158.  1 2 r                   u((2*i3-d3-1), (2*i2-d2-1), (2*i1-d1-1))
1159.  1 2 r                   +z((i3-1), (i2-1), (i1-1));
1160.  1 2 r------->          }
```

Loopmark listing file (MG)
Pragmas

```
1155.  m 2         #pragma _CRI concurrent
1156.  m 2         #pragma _CRI ivdep
1157.  m 2 MV---<      for ( i1 = d1; i1 <= mm1-1; i1++)
1158.  m 2 MV          {
1159.  m 2 MV            u((2*i3-d3-1), (2*i2-d2-1), (2*i1-d1-1)) =
1160.  m 2 MV              u((2*i3-d3-1), (2*i2-d2-1), (2*i1-d1-1))
1161.  m 2 MV               +z((i3-1), (i2-1), (i1-1));
1162.  m 2 MV--->       }
```

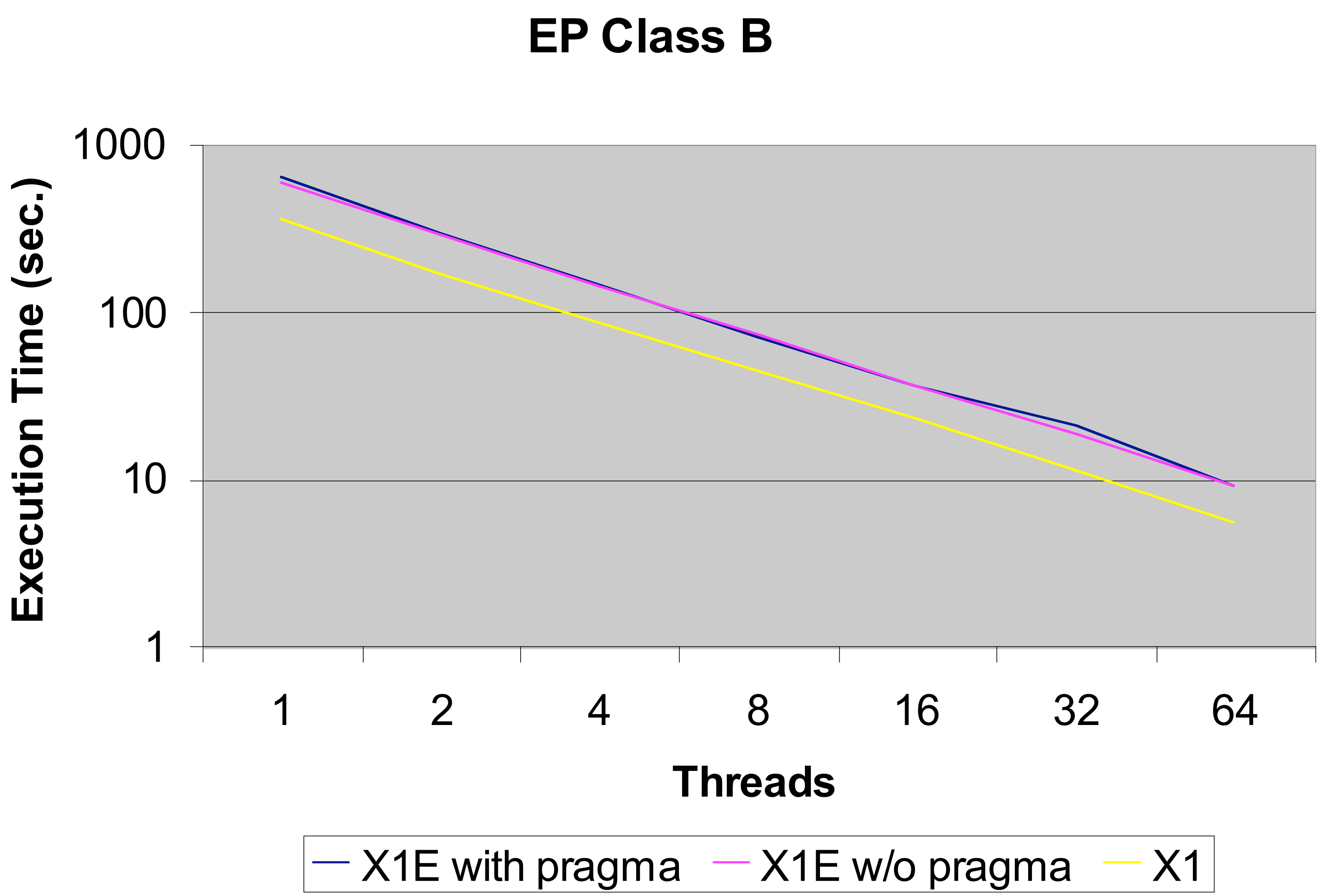
CG Performance



Monte Carlo "Pleasantly" Parallel (EP)

- *Compute independent Gaussian deviates with mu = 0 and sigma^2 = 1.*
- Only communication is a summation of ten values at the end of execution.

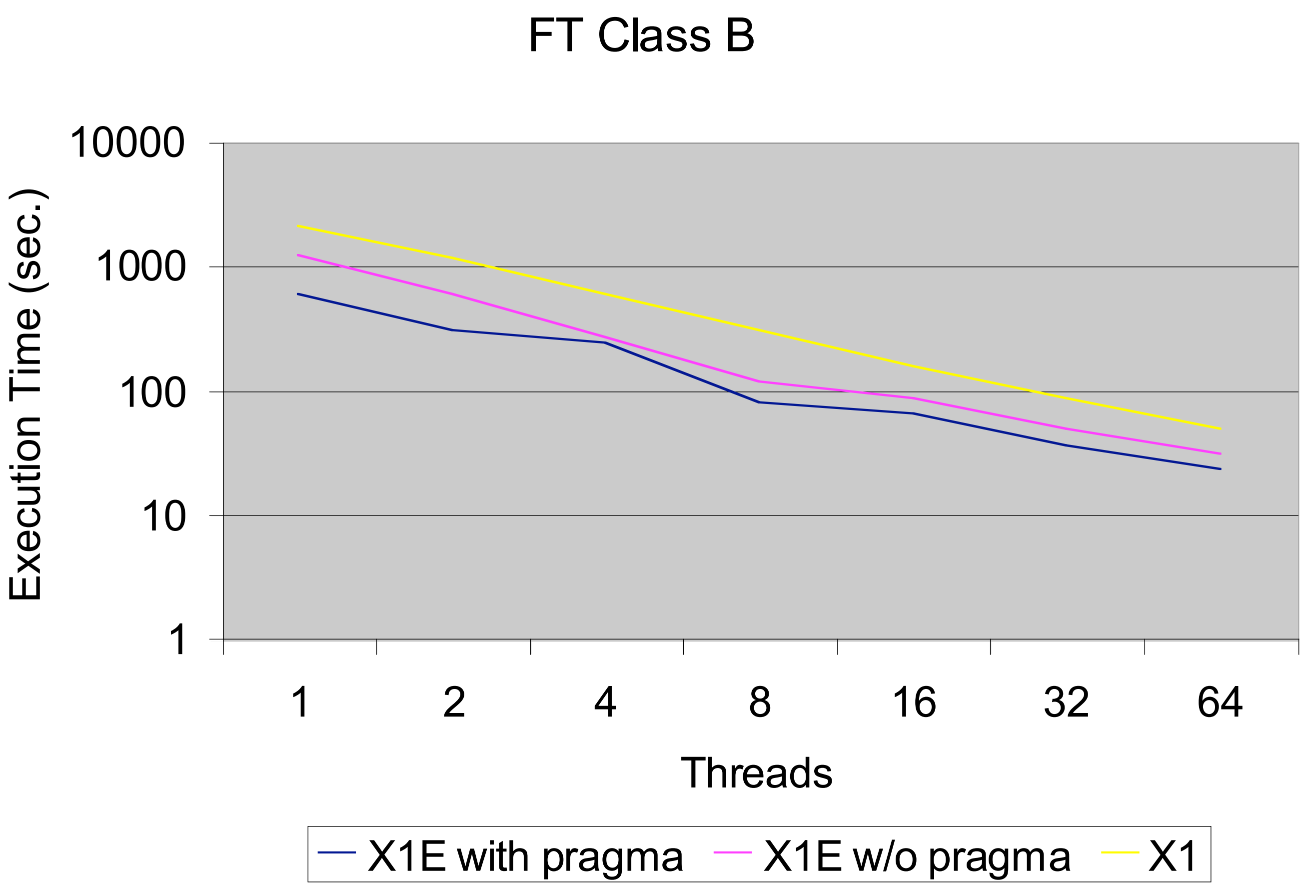
EP Performance



Fourier Transform (FT)

- Solves a 3D partial differential equation using an *FFT-based spectral method*, also requiring long range communication.
- FT performs three 1-D FFT's, one for each dimension.

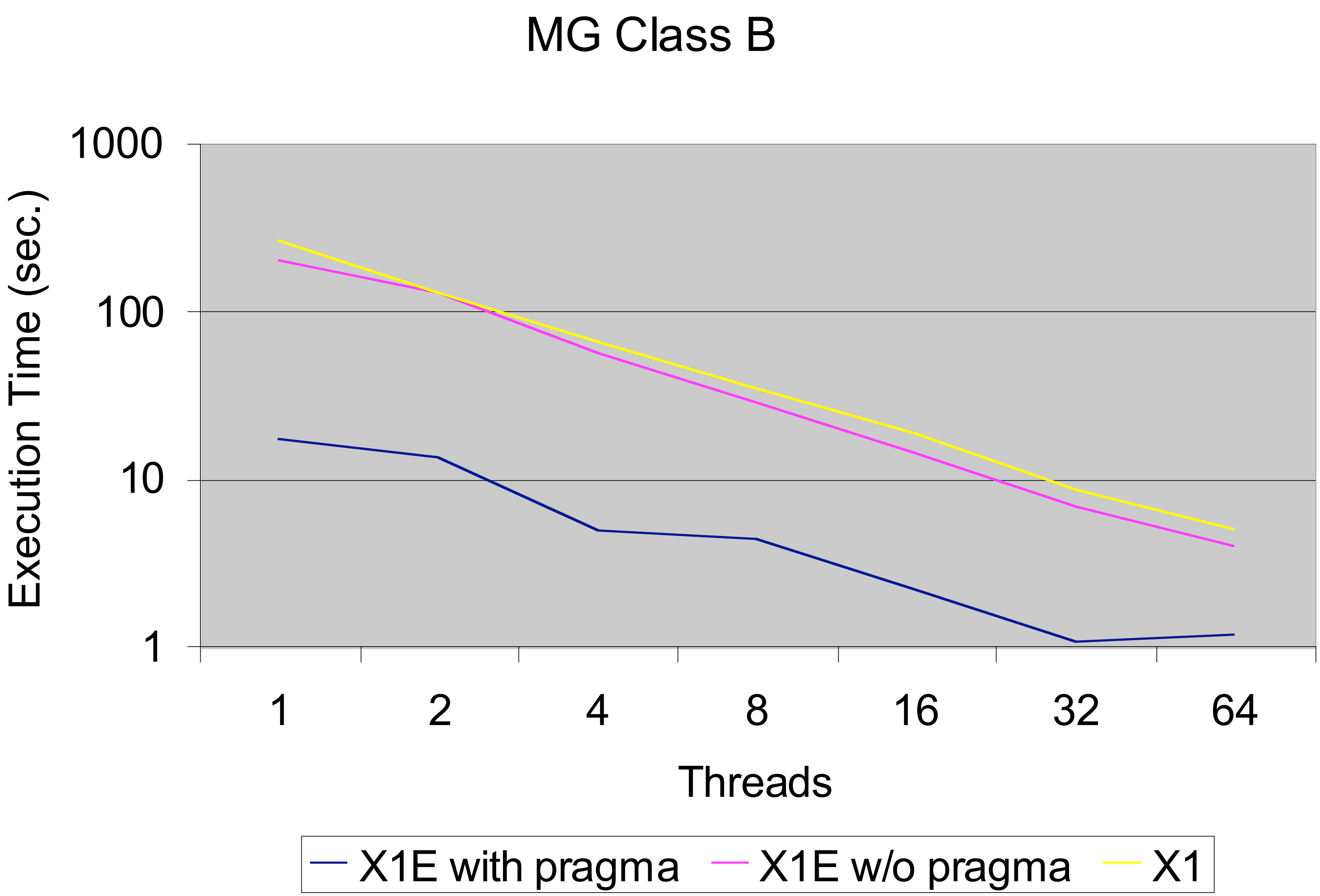
FT Performance



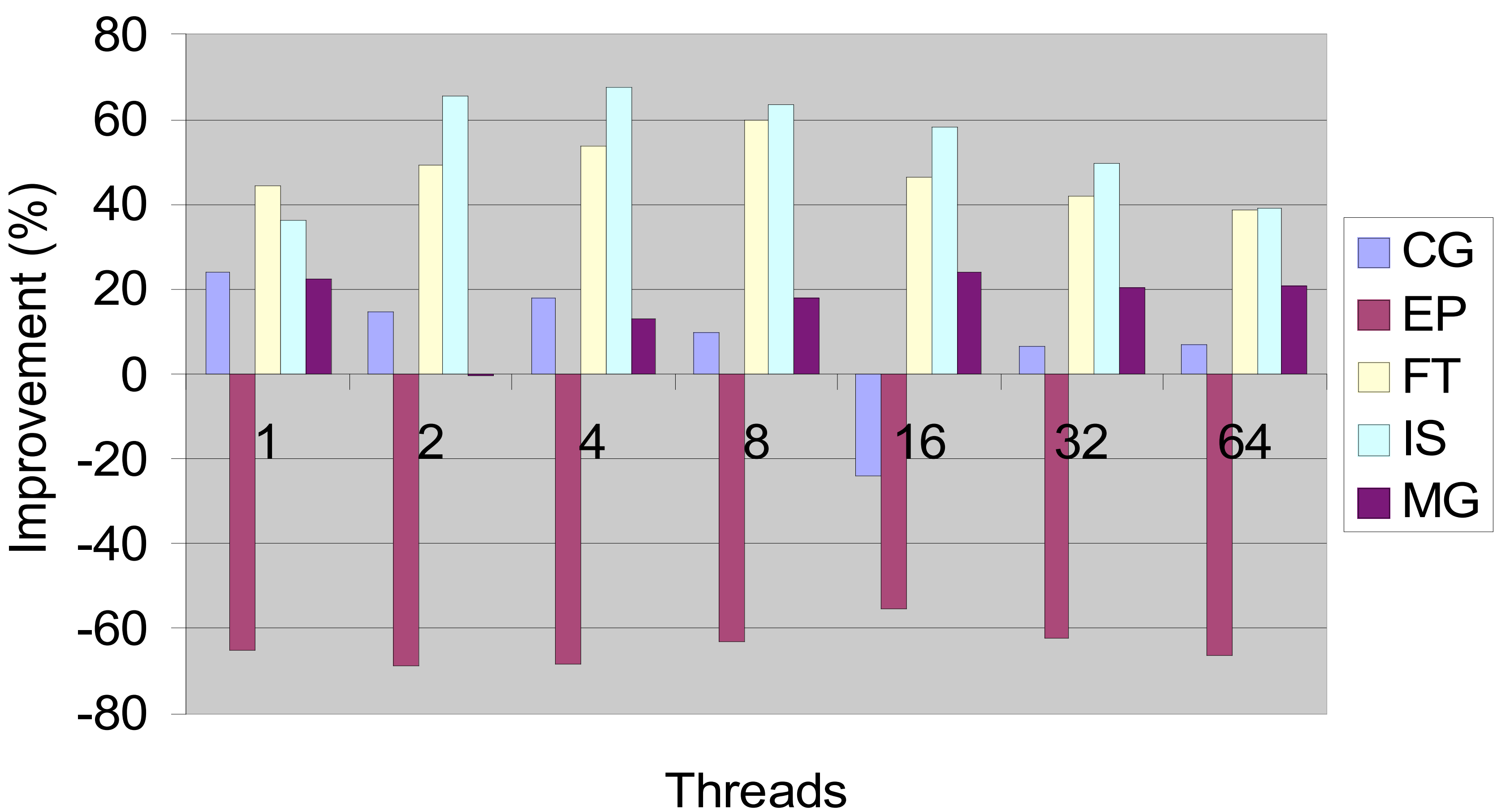
MG: Multi-grid

- uses a V-cycle multi-grid method to compute the solution of the 3-D scalar Poisson equation
- requiring both short and long-range highly structured inter-process communication.

MG Performance



X1E / X1 improvment



Summary

- X1E has a positive effect on performance for some (FT, MG), little effect for others (CG), and a negative effect for one (EP).
- Guided by loopmark listing, careful insertion of pragmas can have a significant effect.

Acknowledgements

- This research was sponsored by the Office of Mathematical, Information, and Computational Sciences, Office of Science, U.S. Department of Energy under Contract No. DE-AC05-00OR22725 with UT-Battelle, LLC. Accordingly, the U.S. Government retains a non-exclusive, royalty-free license to publish or reproduce the published form of this contribution, or allow others to do so, for U.S. Government purposes.

- Cray, esp. Cathy Willis.

PGAS 2006

- October 3-4, Washington, DC (GWU)
- CFP coming soon. (Paper submissions)
- UPC Developers workshop
- CAF Developers workshop