Scaling Properties of the M3D Code From CDX to ITER

Jin Chen
M3D Group

APS-CEMM, Philadelphia
Oct 29, 2006

Motivation: WHY

To optimize the code and prepare for petascale calculations.

M3D time can be broken down to 3 major parts:

scales up to thousands of processors

	characteristics	
Physics	Do loops	
Linear Solver	matrix invert	**gmres/ilu** → **cg/hypre**
Data copy	cross processor communications	

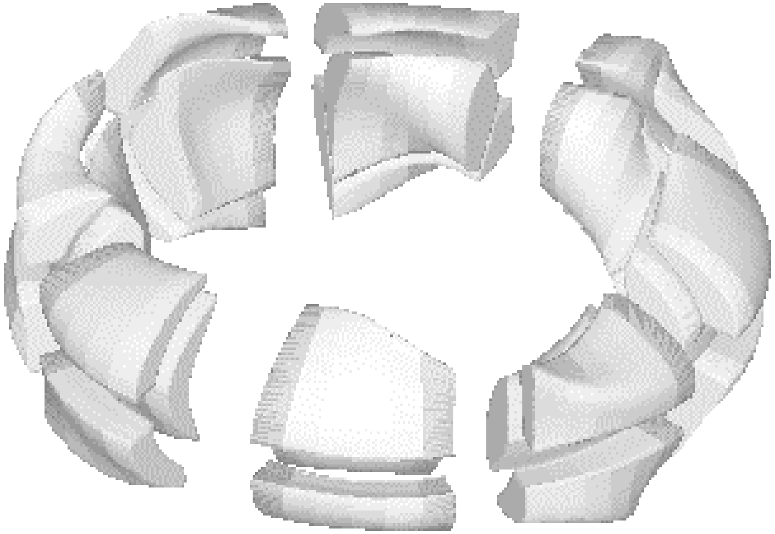
Their efficiencies are critical for optimization on petascale computers.

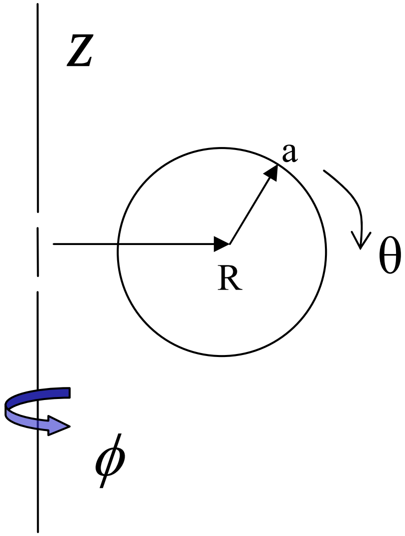
An Example: Total M3D Time = 726 sec

Physics	= 240	
Linear Solver	= 274	(optimized, otherwise >80%.)
Data copy	= 206	

Outline: HOW

1. 3D (r,θ,φ) strong scaling
2. 3D (r,θ,φ) weak scaling
3. 1D (φ) weak scaling
4. 2D (r,θ) weak scaling

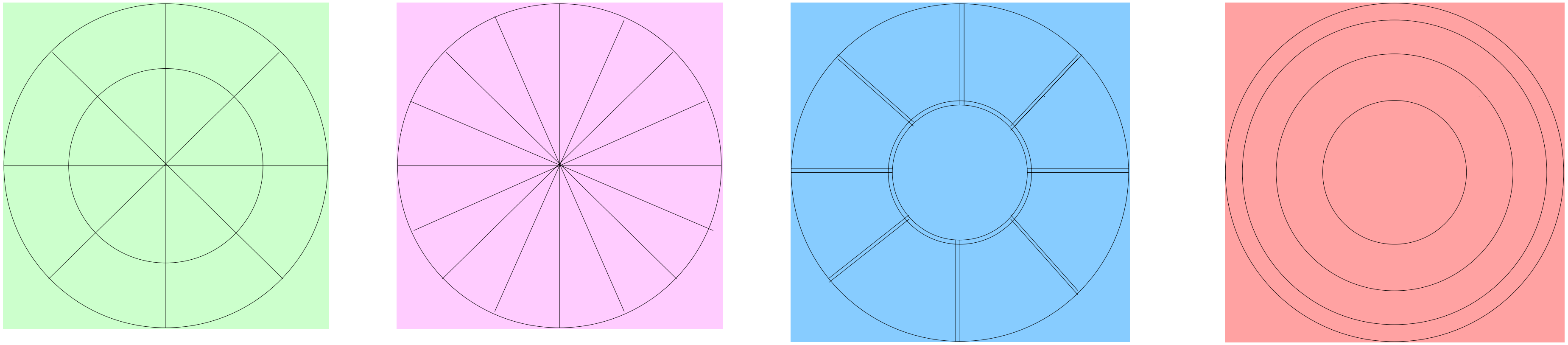




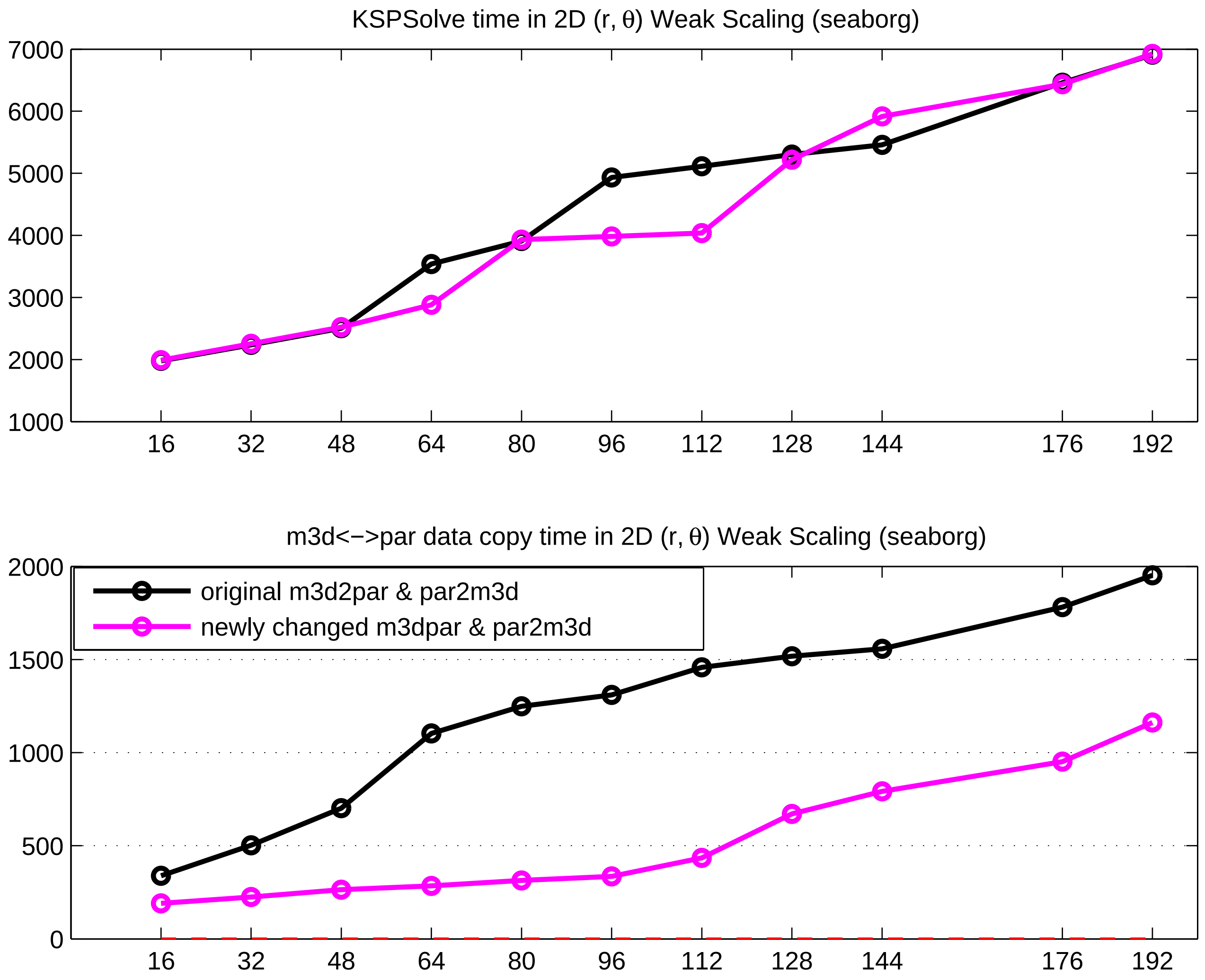
Strategy to improve data copy

a) Reduce toroidal ghost changes
b) Reduce poloidal ghost changes:

2 times faster on seaborg



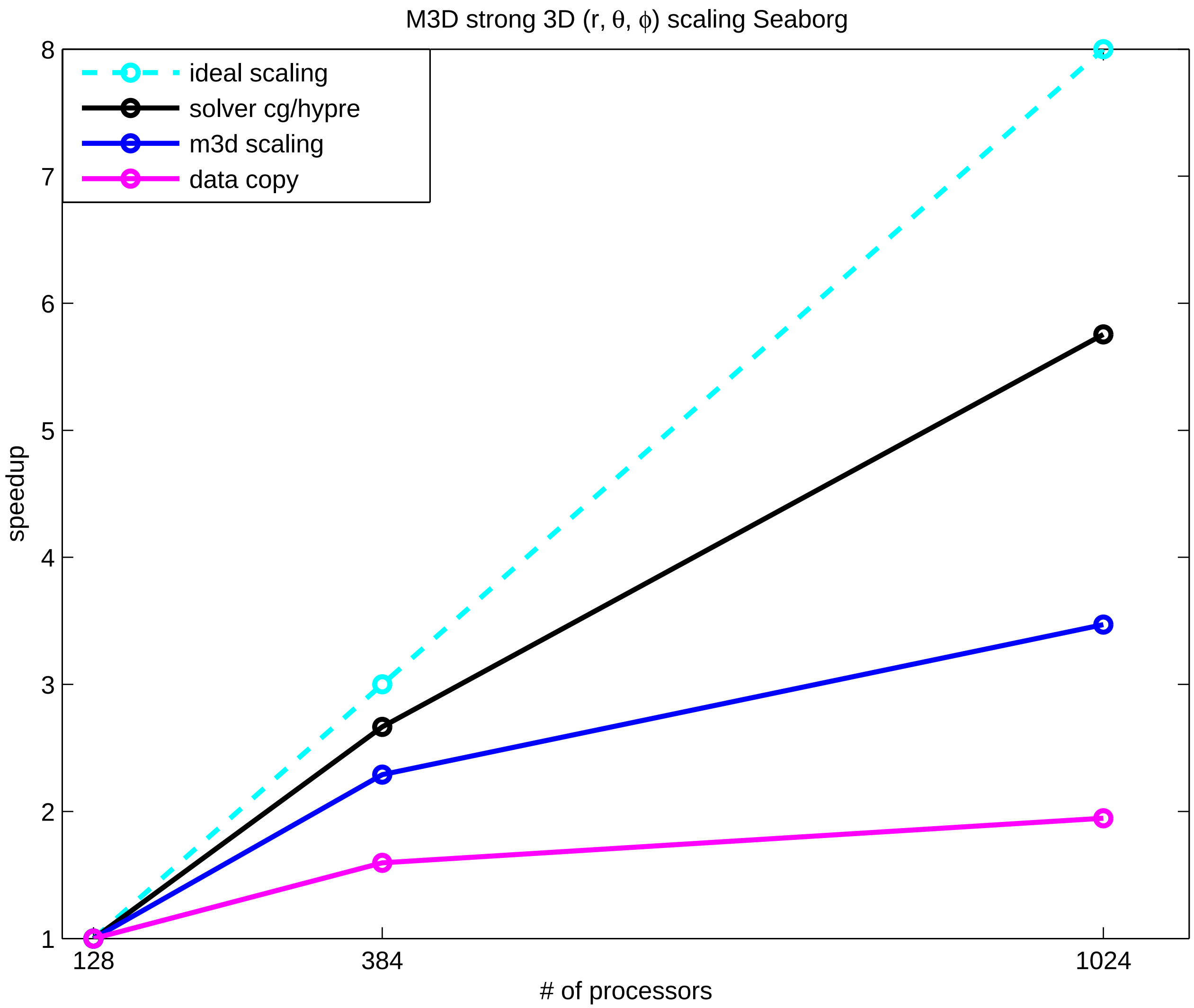
Strategy to improve data copy – II



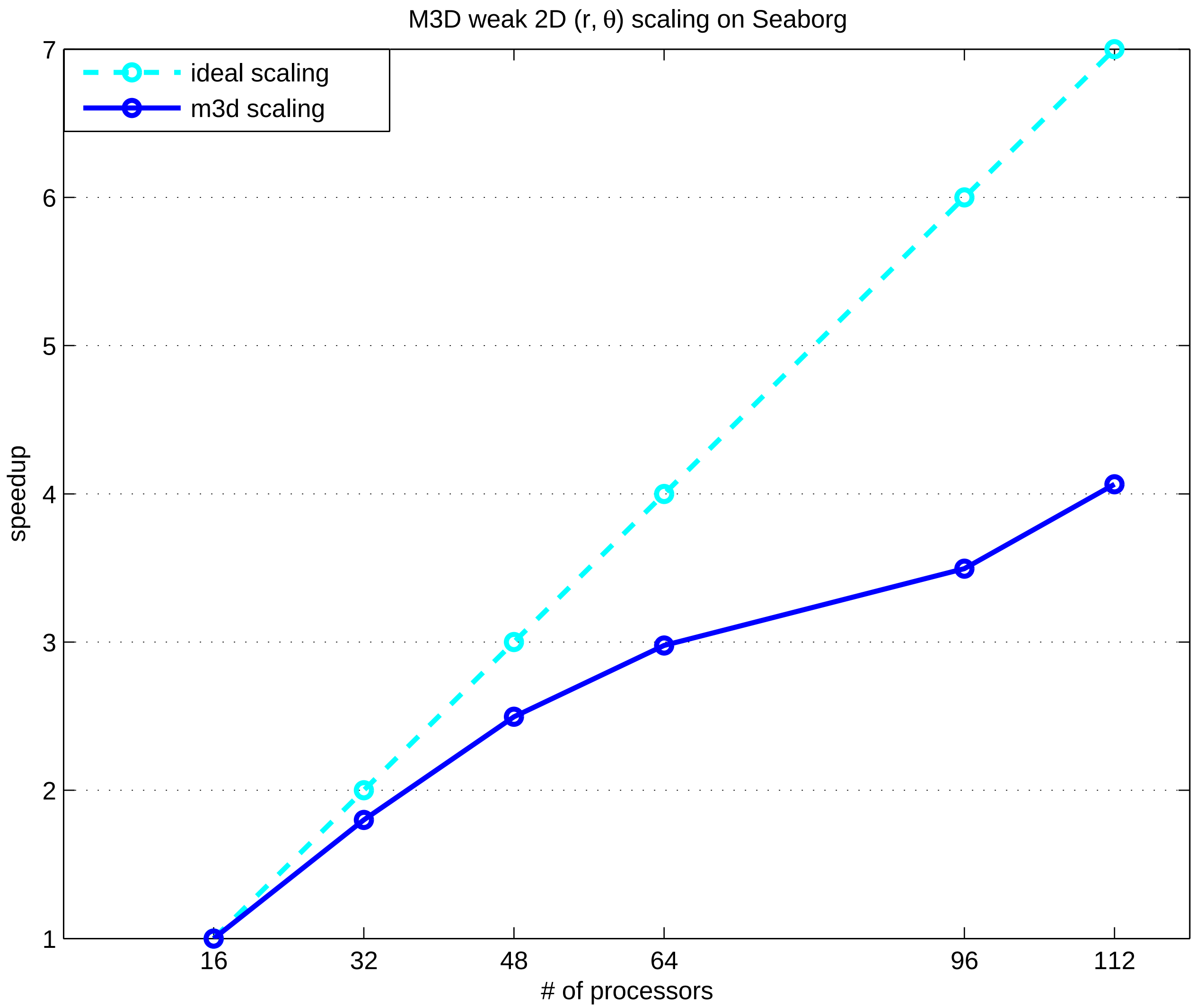
Seaborg: NERSC IBM SP RS/6000.

a distributed memory computer with 6,080 processors.
Each processor has a peak performance of 1.5 GFlops.
The processors are distributed among 380 compute *nodes* with 16 processors per node. Processors on each node have a shared memory pool of between 16 and 64 GBytes

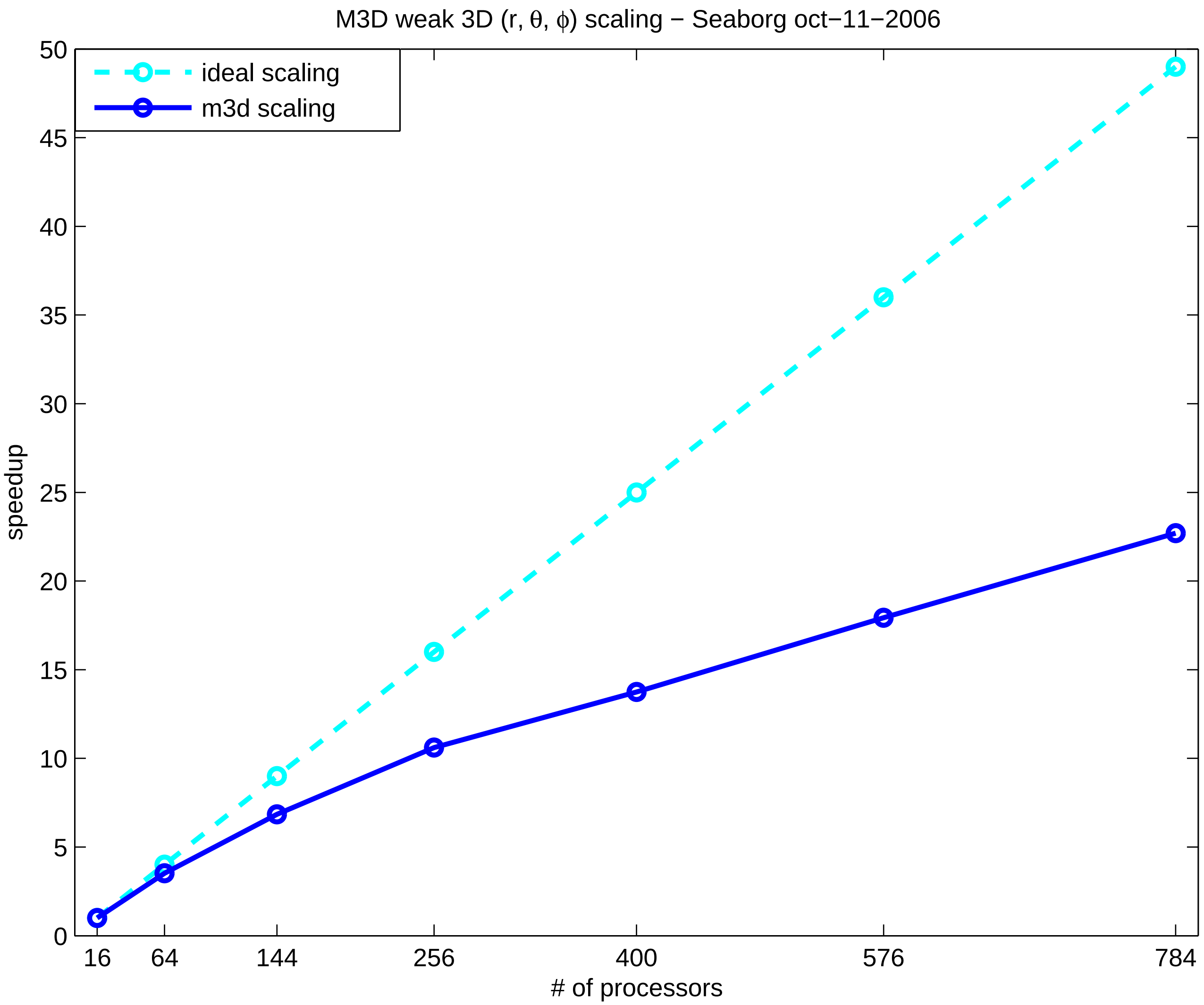
3D (r,θ,φ) strong scaling



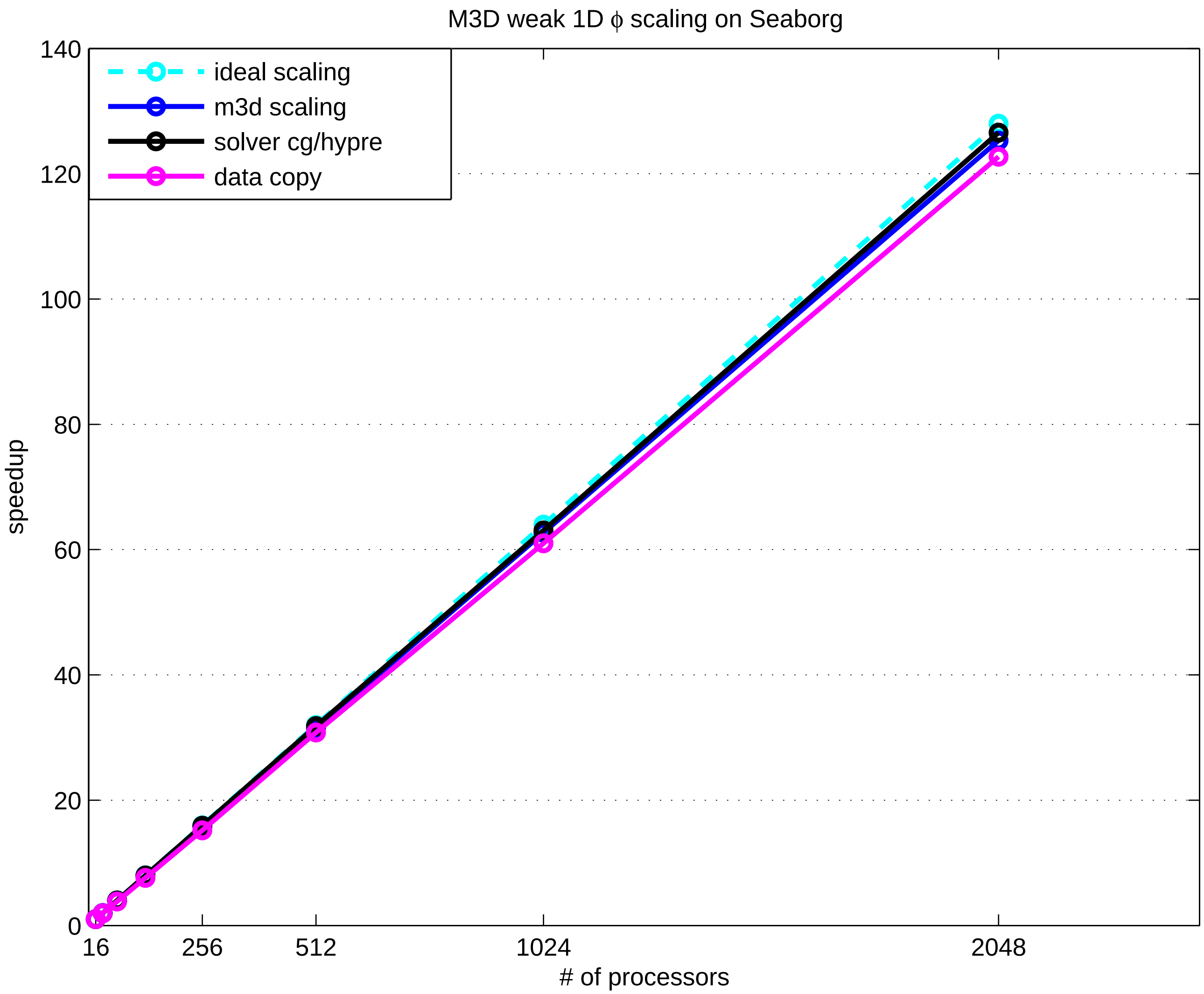
2D (r,θ) weak scaling



3D (r,θ,φ) weak scaling



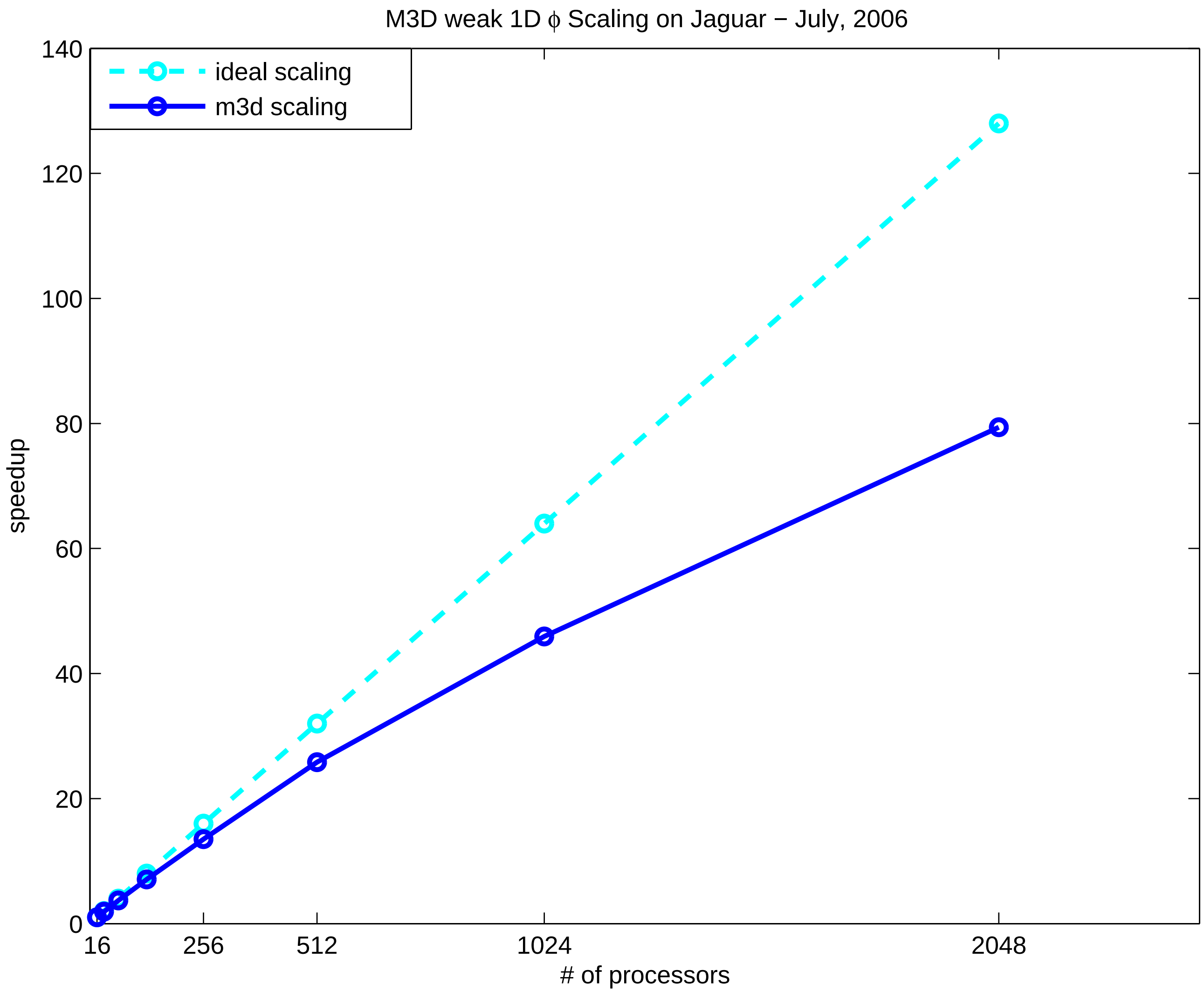
1D (φ) weak scaling



Jaguar: XT3

Compute-node processor count	10,424 cores Note: *2 CPUs now share the memory and interconnect bandwidth of a single CPU before the upgrade*
Compute-node processor size	2.6 GHz dual core
Compute-node memory	4 GB *Maintaining 2 GB per core*
Lustre file system capacity	100 TB
Luster default stripe width	4 OSTs *The stripe size can be changed with the lfs stripesize command*
UNICOS/lc	Upgraded to 1.4.22 *Executables must be recompiled*
Interconnect	Full 3D torus

1D (φ) weak scaling



Problems fixed on Jaguar

- Runtime memory limitation
  - Solution: use only 1 processor per node

    yod –SN m3dp_fsymm_opt.x …
- Code crashes when the number of processor increases from 2048 to 3076 or 4096:

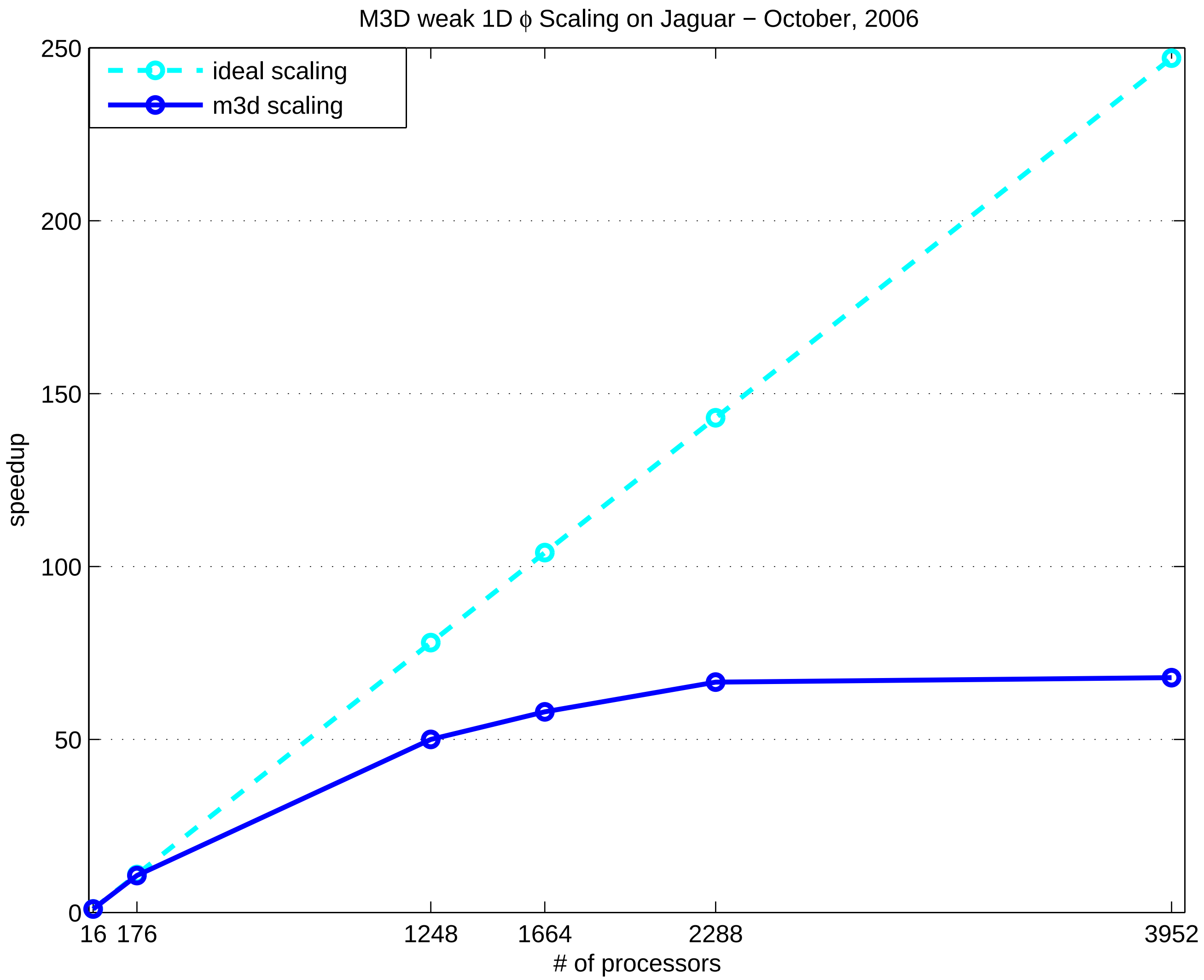
  module load gmalloc

  link –gmalloc as the last library to build m3dp.x
- Wait too long when debugging code
  - We need dedicated time to fix bugs only appeared on large number of processors.
- Fortran static array (stack)

  yod –SN –stack 500M m3dp_fsymm_opt.x …

All the problems were fixed after 9/15/06 upgrades.

1D (φ) weak scaling



BGL at Argonne

MCS BGL Configuration

Compute - 1024 dual PowerPC 440 700MHz 512MB nodes

Storage - 14 TB of clusterwide disk (currently using the MCS Parallel Virtual File System (PVFS)) and 3.5TB of home directory filespace.

Network - IBM BlueGene Torus, Global Tree and Global Interrupt

Running Jobs

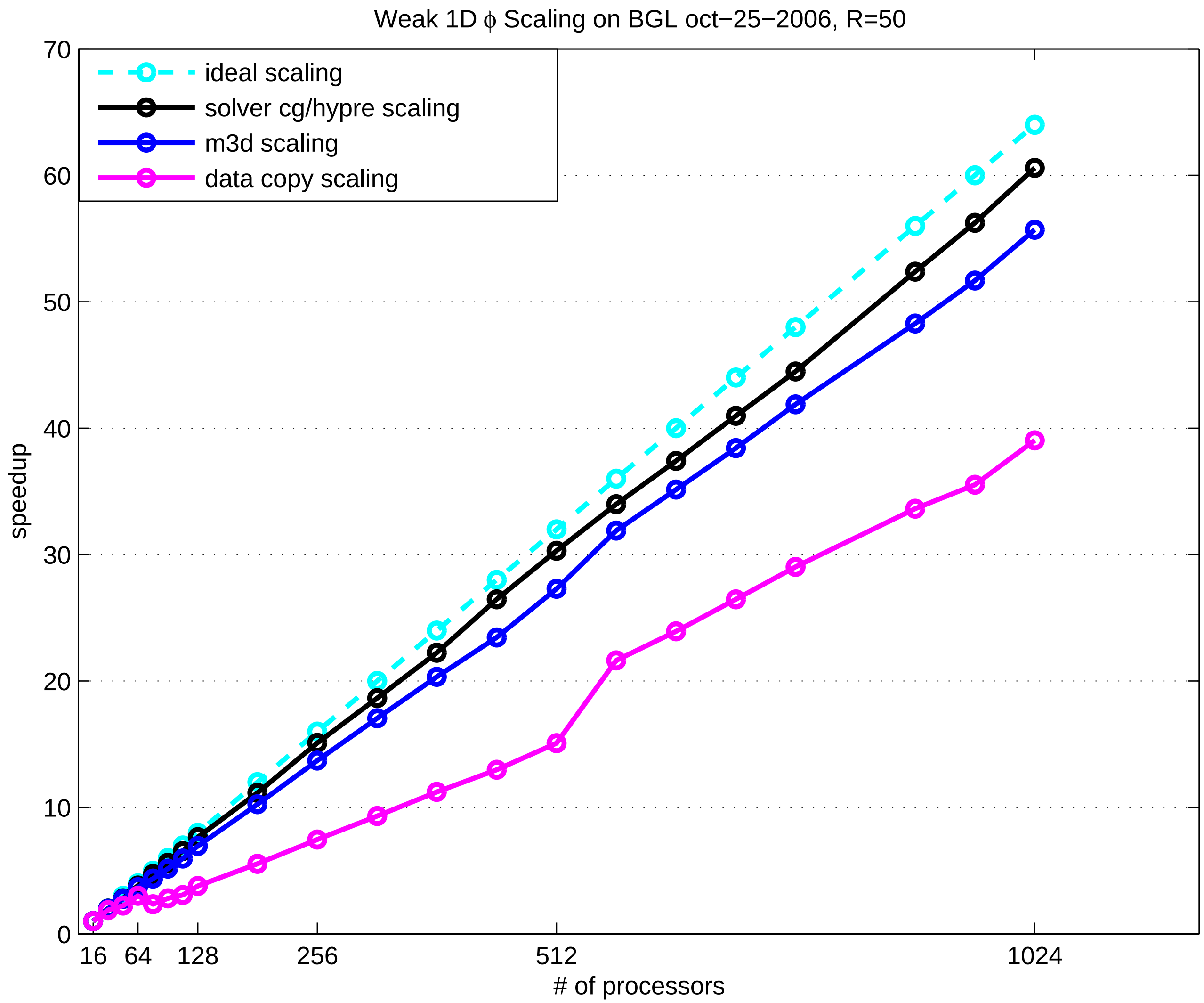
cqsub -t <time> -n <nodecount> -c <#processors> -m <mode> <exe> [arg1,arg2,...]

<time> is in minutes (required)
<nodecount> is the number of nodes
<#processors> number of procs
<mode> is one of 'co' or 'vn'
<exe> is the full path name to the mpi executable **[arg1,arg2,...]**

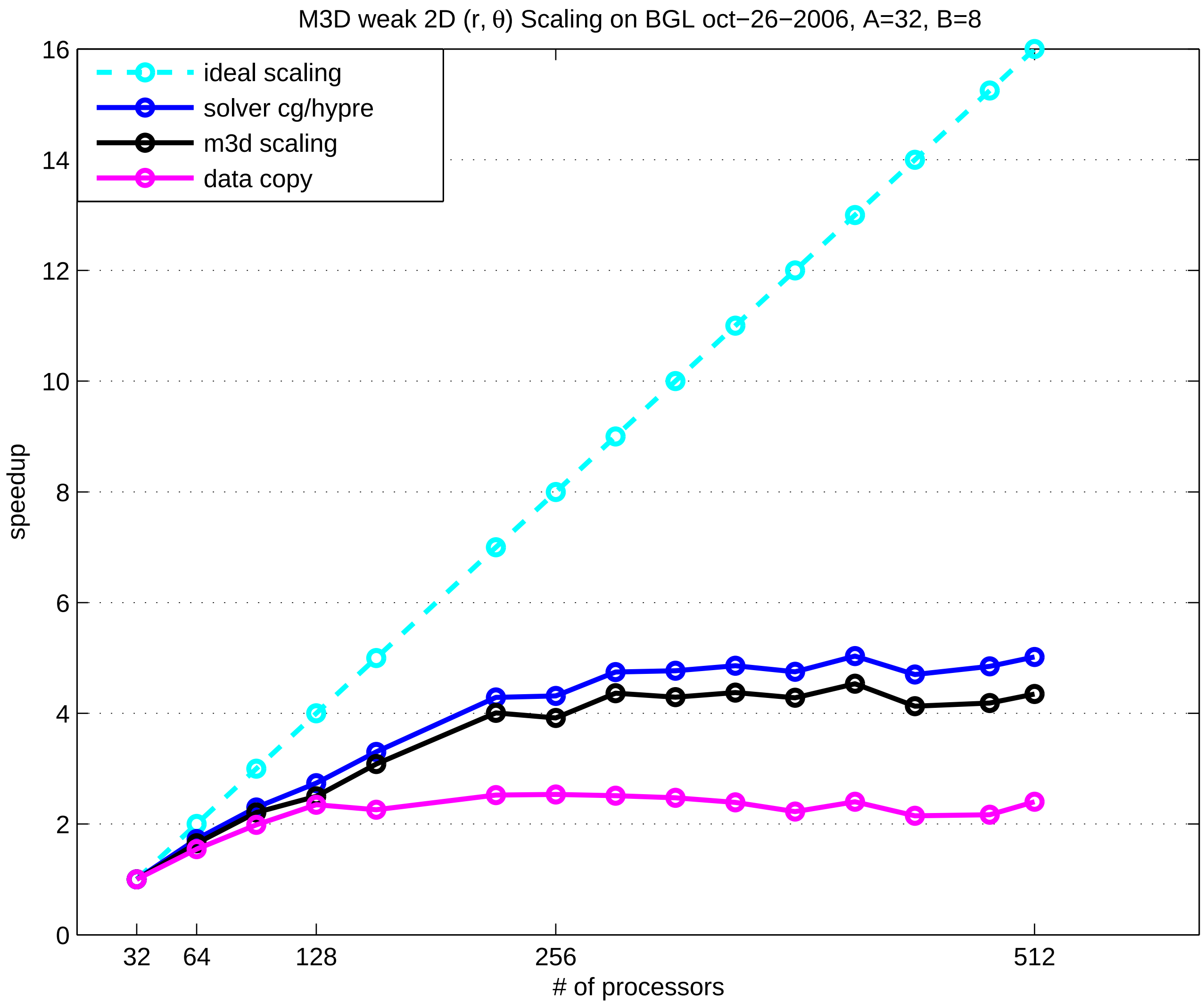
using a partition size smaller than 512, the code cannot use the full Torus network. This will most likely cause the performance to be very poor.

Desired Usage	Partition Size	# of processors
Development, Scaling	32	64
Development, Scaling	64	128
Development, Scaling	128	256
Development, Scaling	256	512
No Development	512	1024
No Development	1024	2048

1D (φ) weak scaling



2D (r,θ) weak scaling



What else ?

- Solver optimized
- Data copy optimized
- Physics code, nothing we can do so far

What else?