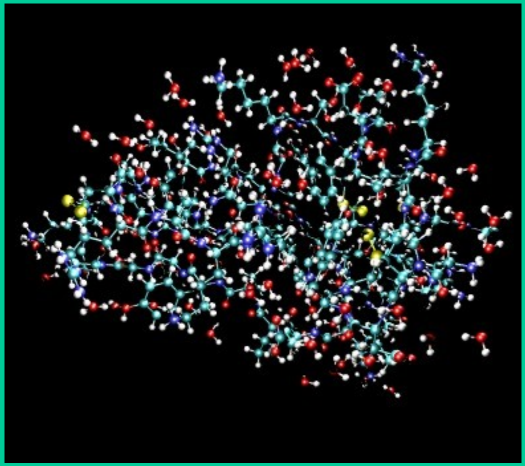
3D FFT with 2D decomposition

Roland Schulz

April 30th 2008

Atomic-Detail Computer Simulation

Model System



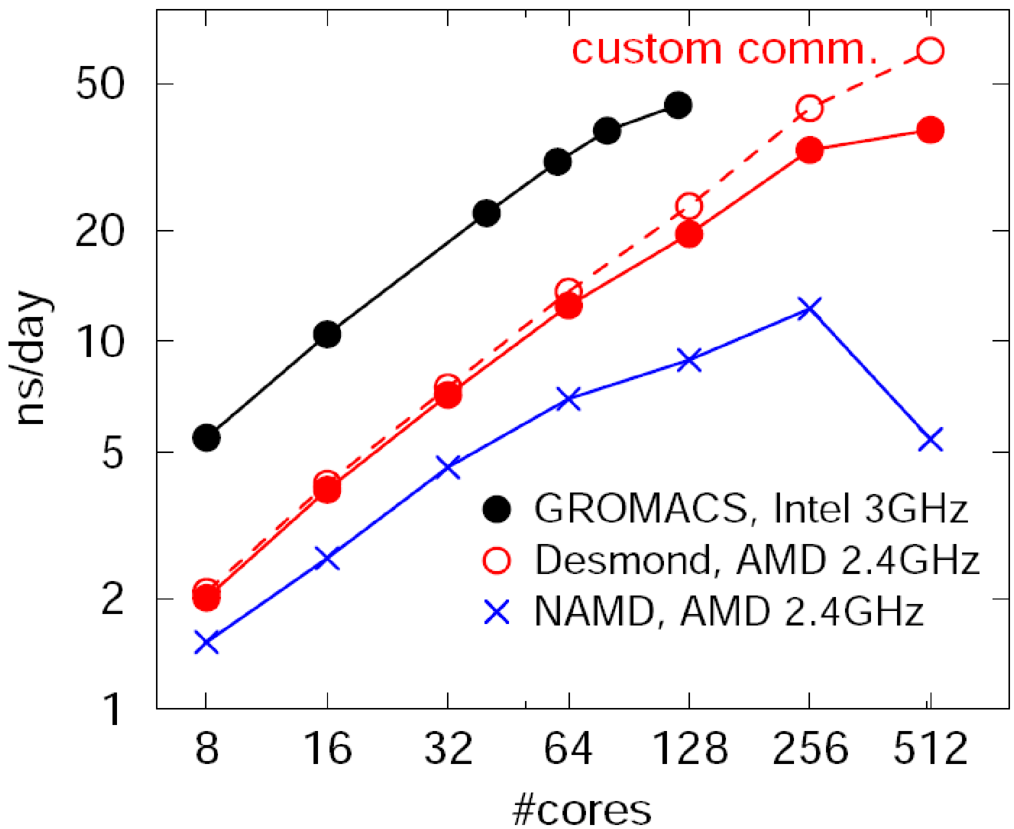
Molecular Mechanics Potential

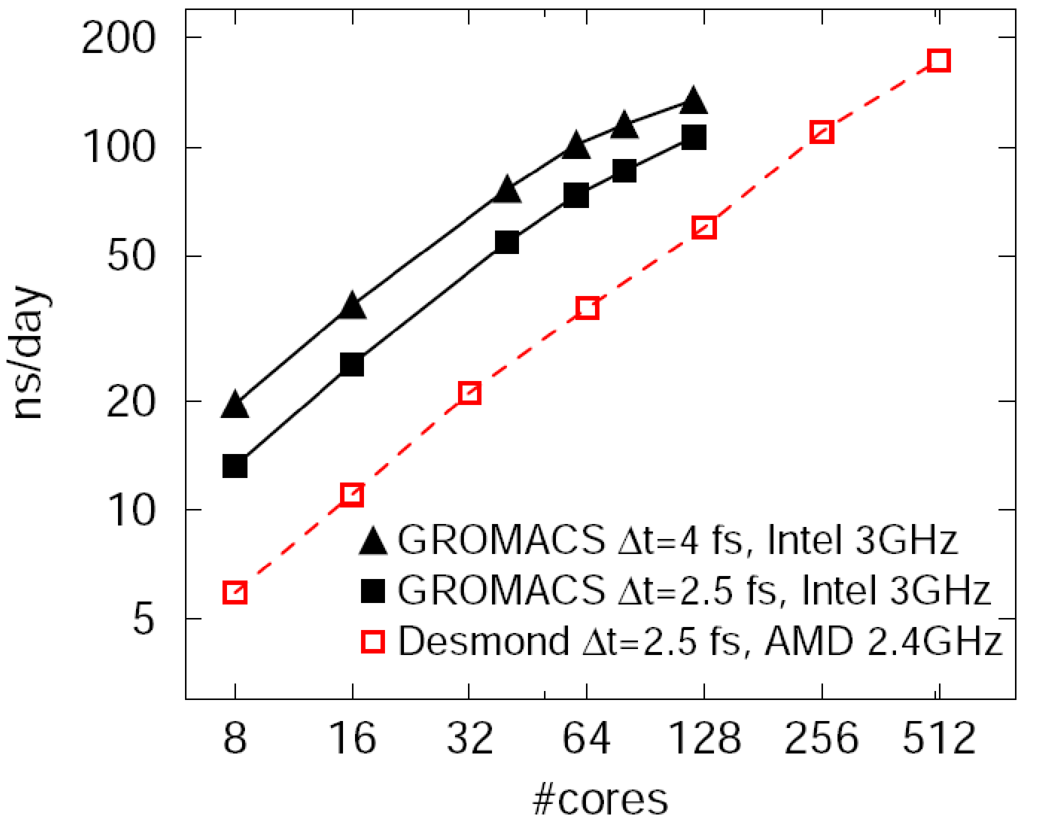
$$V(r^N) = \sum_{\text{bonds}} \frac{1}{2} k_b (l - l_0)^2 + \sum_{\text{angles}} \frac{1}{2} k_a (\theta - \theta_0)^2$$

$$+ \sum_{j=1}^{N-1} \sum_{i=j+1}^{N} \left\{ 4\epsilon_{i,j} \left[\left(\frac{\sigma_{ij}}{r_{ij}} \right)^{12} - \left(\frac{\sigma_{ij}}{r_{ij}} \right)^{6} \right] + \frac{q_i q_j}{4\pi\epsilon_0 r_{ij}} \right\}$$

POWER PLAY

A German physicist and a hedge-fund magnate are competing to push protein simulations into the realm of the millisecond. **Brendan Borrell** finds out what is at stake.


custom comm.
50
20
10
5
2
1
ns/day
8
16
32
64
128
256
512
#cores
GROMACS, Intel 3GHz
Desmond, AMD 2.4GHz
NAMD, AMD 2.4GHz

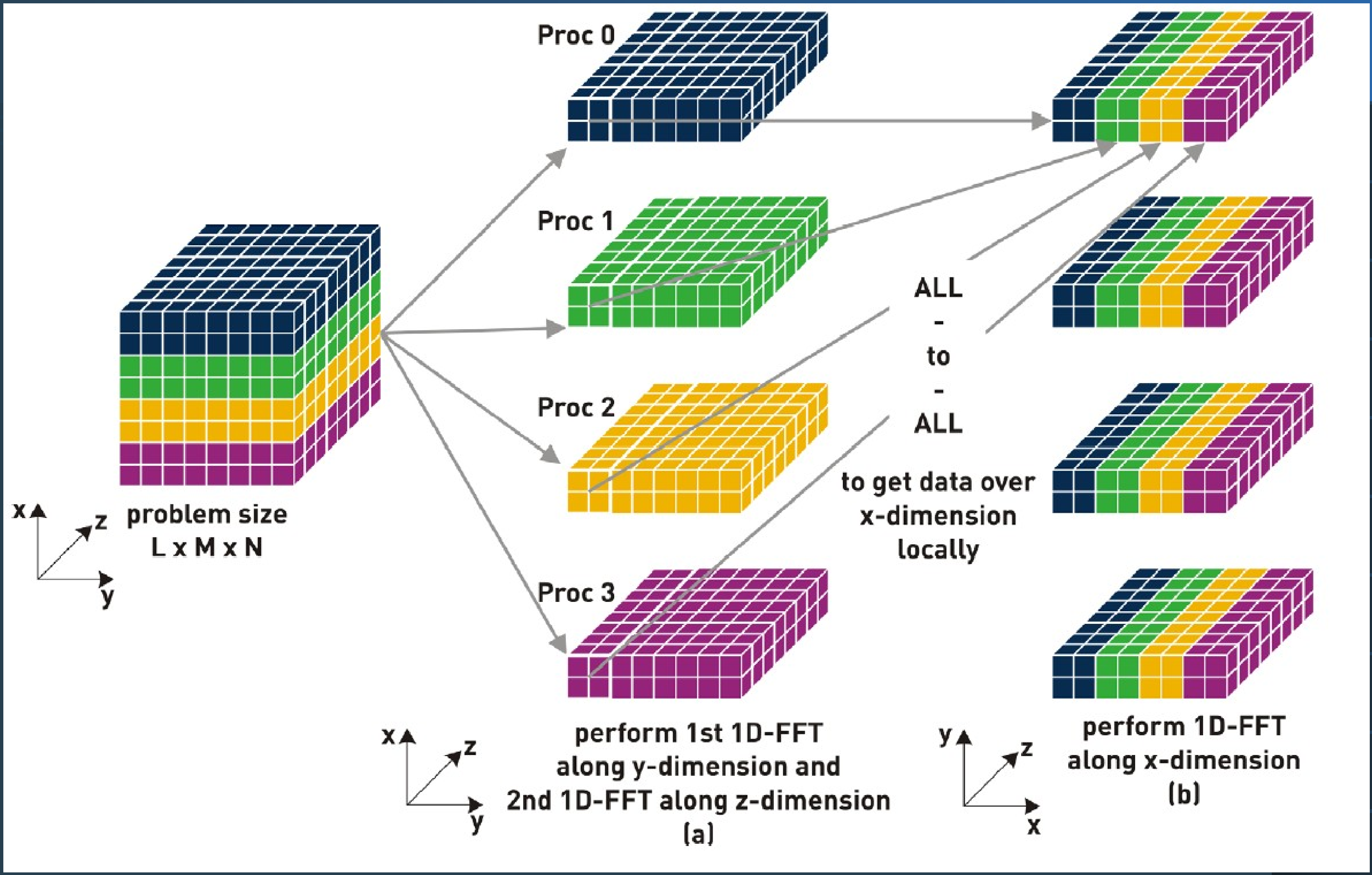

200
100
50
20
10
5
ns/day
8
16
32
64
128
256
512
#cores
GROMACS Δt=4 fs, Intel 3GHz
GROMACS Δt=2.5 fs, Intel 3GHz
Desmond Δt=2.5 fs, AMD 2.4GHz

Berk Hess

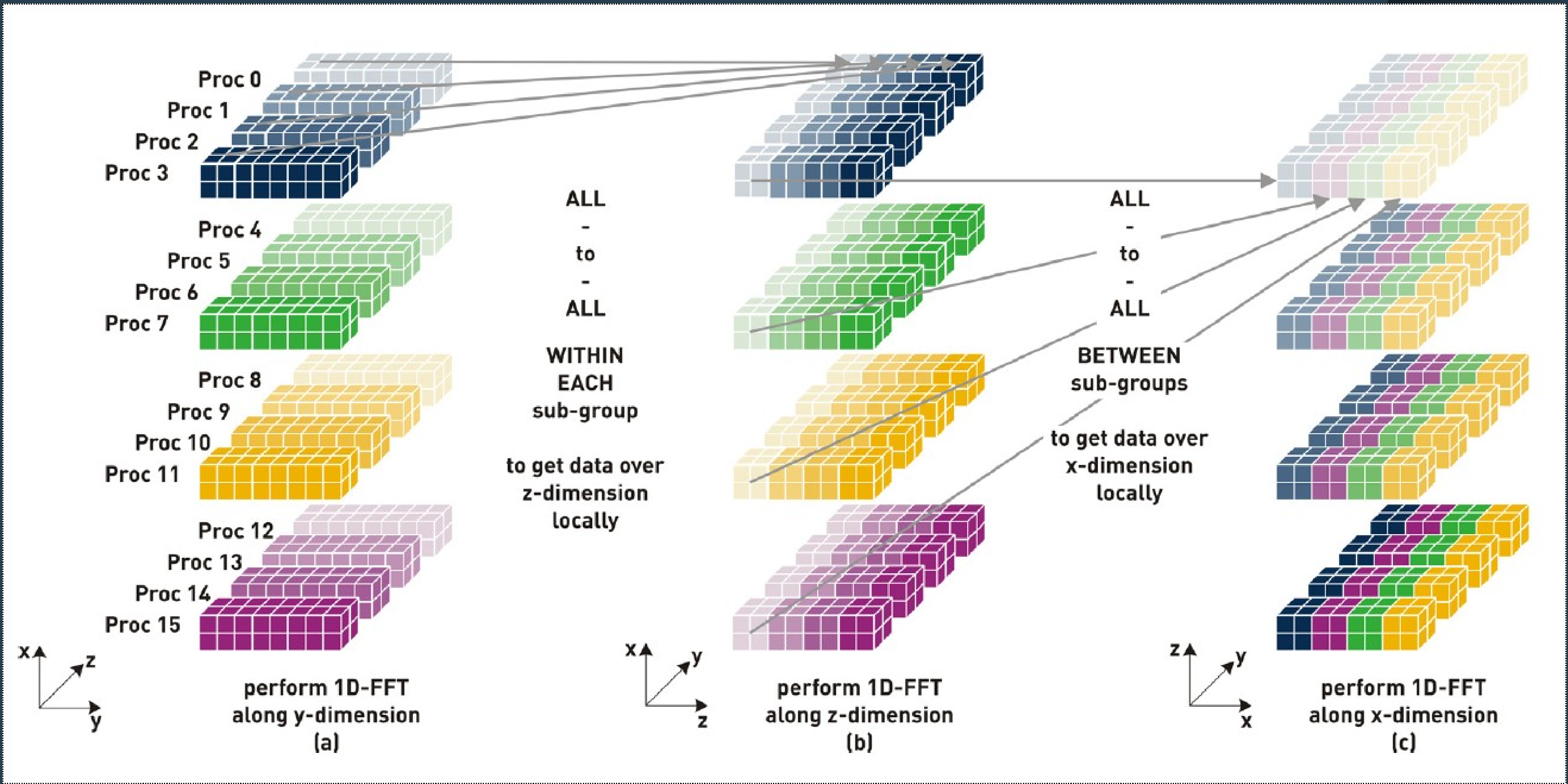
FFT allows O(N log N) electrostatics

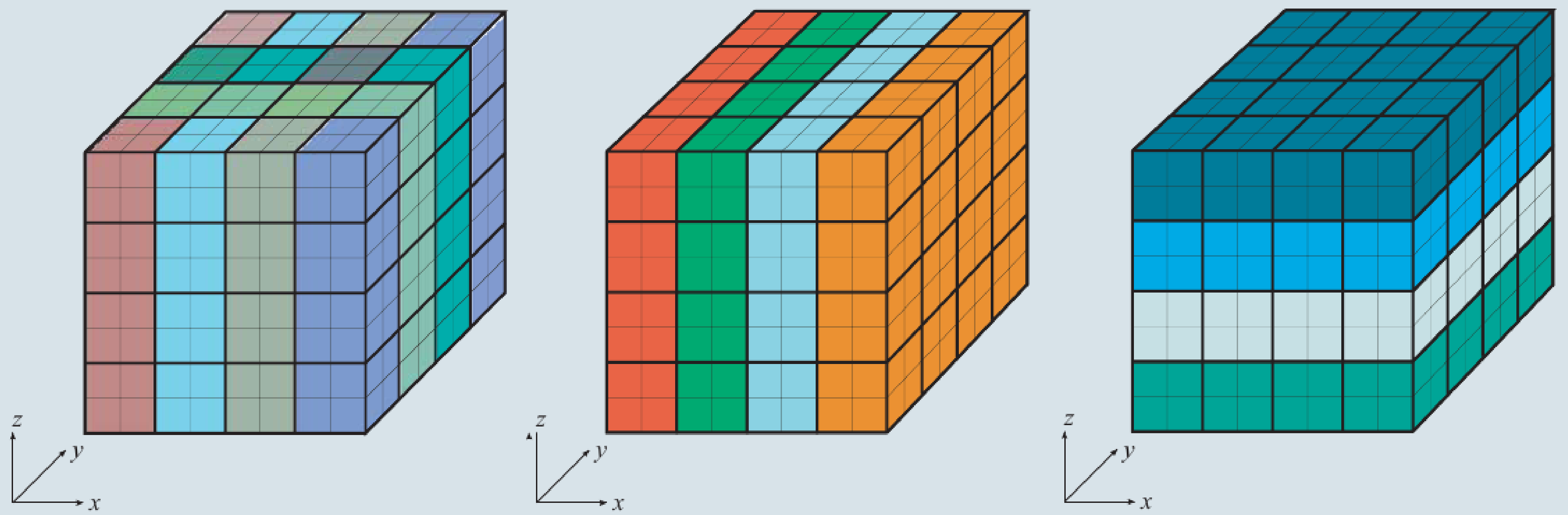
$$X_k = \sum_{n=0}^{N-1} x_n e^{-\frac{2\pi i}{N} nk}$$

$$\begin{aligned}
X_k &= \sum_{m=0}^{\frac{N}{2}-1} x_{2m} e^{-\frac{2\pi i}{N}(2m)k} + \sum_{m=0}^{\frac{N}{2}-1} x_{2m+1} e^{-\frac{2\pi i}{N}(2m+1)k} \\
&= \sum_{m=0}^{M-1} x_{2m} e^{-\frac{2\pi i}{M} mk} + e^{-\frac{2\pi i}{N} k} \sum_{m=0}^{M-1} x_{2m+1} e^{-\frac{2\pi i}{M} mk} \\
&= \begin{cases} E_k + e^{-\frac{2\pi i}{N} k} O_k & \text{if } k < M \\ E_{k-M} - e^{-\frac{2\pi i}{N}(k-M)} O_{k-M} & \text{if } k \geq M \end{cases}
\end{aligned}$$



- Heike Jagode


Proc 0
Proc 1
Proc 2
Proc 3
Proc 4
Proc 5
Proc 6
Proc 7
Proc 8
Proc 9
Proc 10
Proc 11
Proc 12
Proc 13
Proc 14
Proc 15
ALL - to - ALL
WITHIN EACH sub-group
to get data over z-dimension locally
ALL - to - ALL
BETWEEN sub-groups
to get data over x-dimension locally
x z y
perform 1D-FFT along y-dimension
(a)
x y z
perform 1D-FFT along z-dimension
(b)
z y x
perform 1D-FFT along x-dimension
(c)



Communication along *z*-axis Communication in *yz*-plane Communication in *xy*-plane

- M. Eleftheriou

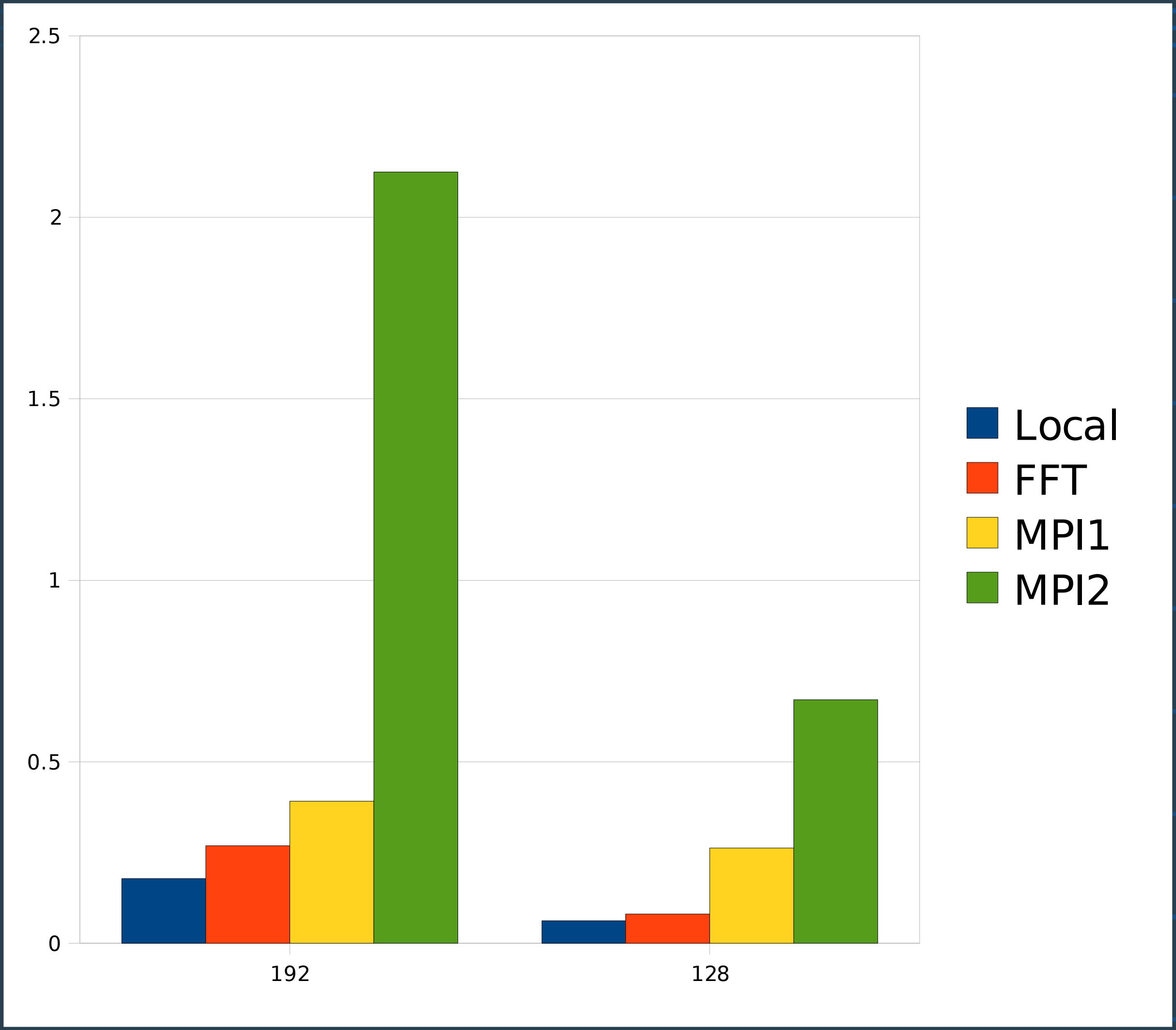
Implementation

- Local without MPI

1. each processor has only one input number, squared processor grid
   $Nx = Ny = 1$
2. any size of data, squared processor grid
   $Px = Py$, $Nx = Ny$, $Nx \bmod Px = 0$
3. any size of data, any processor grid able to evenly divide data
   $Nx \bmod Px = 0$, $Ny \bmod Py = 0$

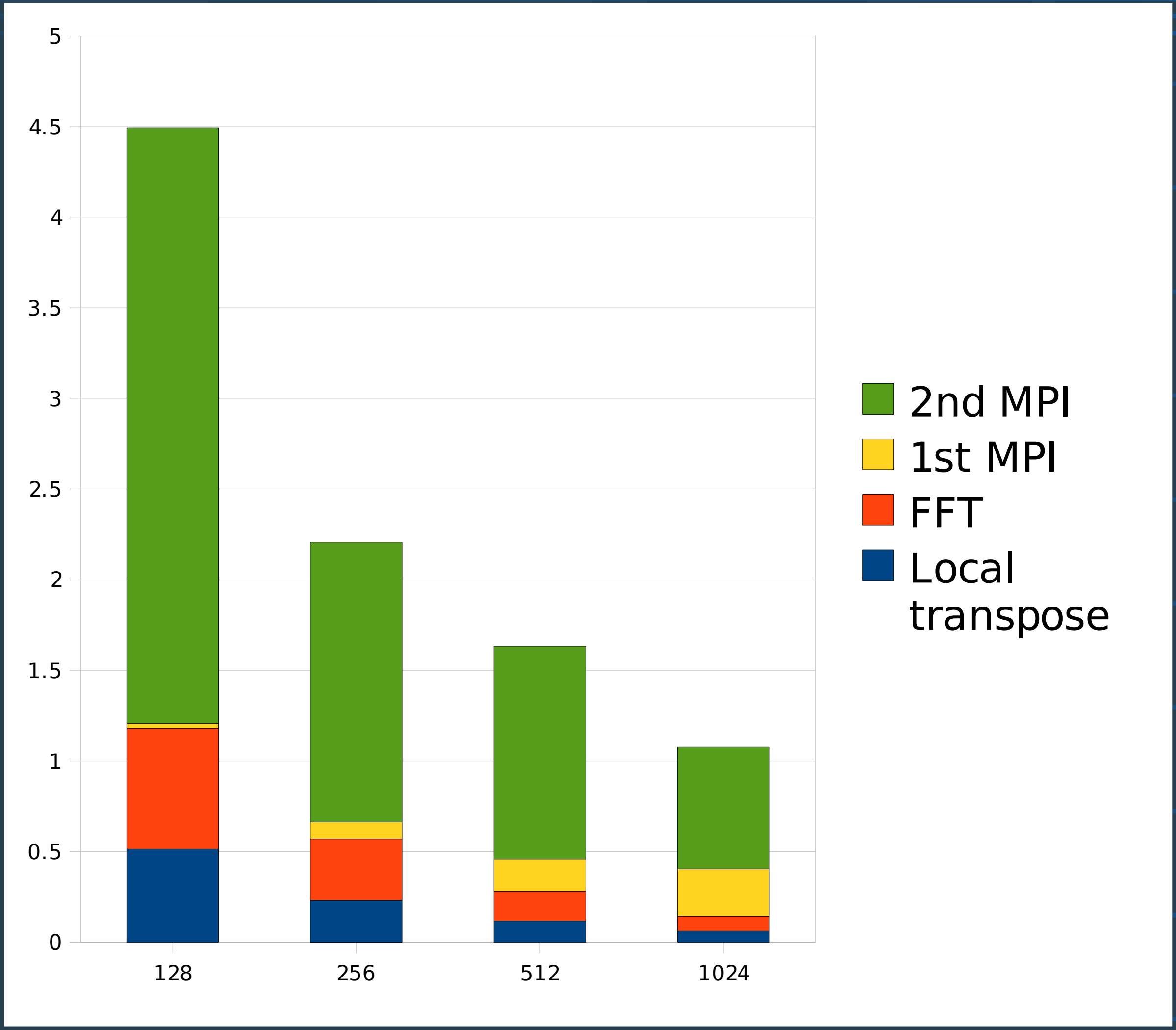
- Loop reordered
- FFTW measure

FFT Benchmark



- N=192: 1152 CPUs, N=128: 1024 CPUs

FFT Strong Scaling



- N=128

Jaguar, MPICH, AllToAll/AllToAllV

AllToAllV	2	4	8	16	32	64	128
0	***5.4***	***6.6***	***9.6***	***15.7***	***25.9***	***46.3***	***22832.0***
1	1.1	1.1	***1.6***	***2.6***	***4.2***	***5.7***	***8.9***
2	1.1	1.1	***1.6***	***2.6***	***3.4***	***5.7***	***9.0***
4	1.1	1.1	***1.6***	***2.1***	***3.5***	***5.5***	***8.9***
8	1.1	1.1	1.3	***2.1***	***3.4***	***5.5***	***8.7***
16	1.1	1.1	1.3	***2.1***	***3.4***	***5.5***	***8.3***
32	1.1	1.2	***1.7***	***2.6***	***4.2***	***6.5***	***9.4***
64	1.1	1.1	***1.7***	***2.5***	***4.0***	***5.6***	***7.2***
128	1.1	1.1	***1.6***	***2.5***	***3.4***	***4.7***	***5.5***
256	1.1	1.1	***1.6***	***2.1***	***2.9***	***3.6***	***7.3***
512	1.1	1.1	1.3	***1.8***	***2.1***	***3.2***	***10.7***
1024	1.1	1.1	1.1	1.2	1.2	***2.2***	***2.0***
2048	1.1	1.1	1.1	1.2	1.1	***2.4***	***2.7***
4096	1.1	1.1	1.1	1.1	***1.9***	***2.4***	***7.0***
8192	1.1	1.1	1.1	1.2	***2.3***	***1.9***	***3.1***
16384	1.0	1.3	1.2	***1.8***	***1.8***	1.4	1.3
32768	1.0	1.3	1.2	1.4	1.5	1.3	1.2
65536	1.0	1.0	1.1	1.2	1.3	1.2	1.2

Jaguar, MPICH/OpenMPI, AllToAll

AlltoAll	2	4	8	16	32	64	128
0	1.2	1.4	1.2	1.2	1.2	1.2	1.4
1	***8.3***	1.1	***2.5***	1.5	***1.6***	1.3	1.3
2	***8.7***	1.1	***2.5***	1.5	1.3	1.3	1.4
4	***8.2***	1.1	***2.4***	1.2	1.3	1.3	1.3
8	***8.6***	1.1	***2.0***	1.2	1.2	1.3	1.3
16	***8.3***	1.1	***2.0***	1.2	1.3	1.3	1.3
32	***9.3***	1.1	***2.0***	1.2	1.2	1.3	1.2
64	***9.1***	1.1	***1.9***	1.2	1.3	1.2	1.1
128	***9.0***	1.1	***1.9***	1.2	1.2	1.2	1.0
256	***8.6***	1.1	***1.8***	***2.6***	***4.1***	***4.9***	***7.0***
512	***8.3***	1.1	***1.6***	***2.5***	***3.0***	***3.9***	***4.3***
1024	***7.0***	1.1	1.0	1.1	1.3	***2.2***	***2.4***
2048	***6.7***	1.2	1.1	1.1	***1.6***	***2.5***	***2.4***
4096	***3.0***	1.0	1.1	1.3	***1.5***	***1.7***	***1.8***
8192	***2.5***	1.0	1.1	1.3	***1.5***	***1.7***	***1.5***
16384	***1.8***	1.1	1.3	1.3	***1.6***	1.4	1.3
32768	1.5	***1.6***	***1.5***	1.4	1.3	1.2	1.0
65536	1.3	1.3	1.2	1.1	1.2	1.1	1.0

Jaguar, MPICH/OpenMPI, AllToAllV

AllToAllV	2	4	8	16	32	64	128
0	***3.8***	***4.2***	***4.8***	***6.1***	***6.7***	***7.5***	***2663.7***
1	***9.4***	1.2	***1.5***	***1.7***	***1.7***	***1.7***	***2.0***
2	***9.5***	1.2	***1.5***	***1.7***	***1.7***	***1.7***	***2.0***
4	***9.5***	1.2	***1.5***	***1.6***	***1.7***	***1.7***	***1.9***
8	***9.3***	1.2	1.5	***1.6***	***1.7***	***1.7***	***2.1***
16	***9.4***	1.2	1.5	***1.6***	***1.7***	***1.7***	***2.1***
32	***10.4***	1.0	1.2	1.3	1.3	1.4	***1.6***
64	***10.1***	1.0	1.2	1.3	1.3	1.4	***1.8***
128	***9.9***	1.0	1.2	1.3	1.3	1.4	***1.8***
256	***9.7***	1.0	1.2	1.3	1.4	***1.5***	1.0
512	***8.7***	1.0	1.2	1.3	1.4	1.5	***2.5***
1024	***7.5***	1.0	1.2	1.3	***2.1***	1.2	1.2
2048	***7.0***	1.1	1.0	1.3	***2.0***	1.1	1.2
4096	***3.3***	1.0	1.1	***1.6***	1.5	1.1	***3.4***
8192	***2.7***	1.0	1.0	***1.6***	1.2	1.1	***2.0***
16384	***2.0***	1.2	1.1	1.2	1.3	1.1	1.0
32768	***1.5***	***1.8***	1.5	***1.7***	1.4	***1.7***	
65536	1.3	1.1	1.3	1.4	1.4	***1.7***	

Jaguar, MPICH, AllToAll/FFTW

AlltoAll – FFTW	2	4	8	16	32	64
4	1.0	**1.5**	1.0	1.1	1.1	1.0
8	1.0	1.0	1.1	1.2	1.1	1.0
16	1.0	1.2	1.2	1.2	1.1	1.0
32	1.0	**1.7**	1.0	1.1	1.1	1.1
64	1.0	**1.7**	1.0	1.1	1.0	1.2
128	1.0	**1.6**	1.0	1.1	1.0	1.2
256	1.0	**1.6**	1.0	1.1	1.0	1.2
512	1.0	**1.5**	1.1	1.1	1.0	1.1
1024	1.0	1.3	**1.4**	**1.8**	**1.6**	**1.4**
2048	1.0	1.2	**1.3**	**1.3**	1.3	1.0
4096	1.0	1.0	1.1	1.0	1.1	**1.4**
8192	1.0	1.0	1.0	1.1	1.1	1.3
16384	1.0	1.1	1.1	1.0	1.0	1.2
32768	1.0	1.0	1.0	1.0	1.0	1.1
65536	1.0	1.1	1.0	1.0	1.0	1.1

Jaguar, MPICH, AllToAllV/FFTW

FFTW – AlltoAllV	2	4	8	16	32	64	128
4	1.1	***1.3***	***1.7***	***2.5***	***3.8***	***6.6***	***10.9***
8	1.1	***1.3***	***1.5***	***2.5***	***3.7***	***6.5***	***10.2***
16	1.1	1.1	***1.4***	***2.3***	***3.5***	***5.8***	***9.8***
32	1.1	1.3	***1.7***	***3.0***	***4.6***	***7.0***	***9.7***
64	1.1	***1.3***	***1.7***	***2.8***	***3.9***	***5.7***	***7.5***
128	1.1	1.3	***1.6***	***2.6***	***3.5***	***4.9***	***5.8***
256	1.1	1.2	***1.6***	***2.2***	***2.9***	***3.8***	***7.5***
512	1.1	1.2	***1.4***	***1.9***	***2.1***	***3.5***	***14.1***
1024	1.1	1.1	1.2	***1.5***	***1.5***	***3.6***	***2.9***
2048	1.1	1.1	1.2	1.1	***1.3***	***2.7***	***2.6***
4096	1.1	1.0	1.0	1.0	***1.7***	***2.2***	***6.2***
8192	1.1	1.0	1.1	1.2	1.0	***1.6***	***2.9***
16384	1.0	1.1	1.1	1.1	***1.4***	1.2	1.2
32768	1.0	***1.5***	1.1	1.0	1.1	1.0	1.0
65536	1.0	1.0	1.1	1.0	1.0	1.0	1.0

Helics, MPICH, AllToAll/AllToAllV

AllToAllV	2	4	8	16	32	64
4	1.0	1.0	1.2	***2.2***	***3.1***	***4.1***
8	1.1	1.0	1.2	***2.0***	***3.0***	***3.4***
16	1.1	1.0	1.1	***2.0***	***2.4***	***3.4***
32	1.1	1.0	1.1	***1.6***	***2.3***	***3.2***
64	1.1	1.0	1.0	***1.8***	***2.7***	***3.1***
128	1.1	1.0	1.0	***1.8***	***2.1***	***2.1***
256	1.1	1.0	***1.5***	***2.1***	***2.1***	***3.0***
512	1.1	1.2	1.0	1.0	1.0	1.0
1024	1.0	1.0	1.0	1.0	1.0	1.0
2048	1.0	1.0	1.0	1.0	1.0	1.2
4096	1.0	1.1	1.0	1.0	1.0	1.0
8192	1.1	1.1	1.0	1.0	1.0	1.1
16384	1.2	1.1	1.0	1.0	1.0	1.2
32768	1.2	1.0	1.0	1.1	1.0	1.1
65536	1.3	***4.1***	1.1	***1.5***	***1.7***	***1.6***

Helics, MPICH/OpenMPI, AllToAll

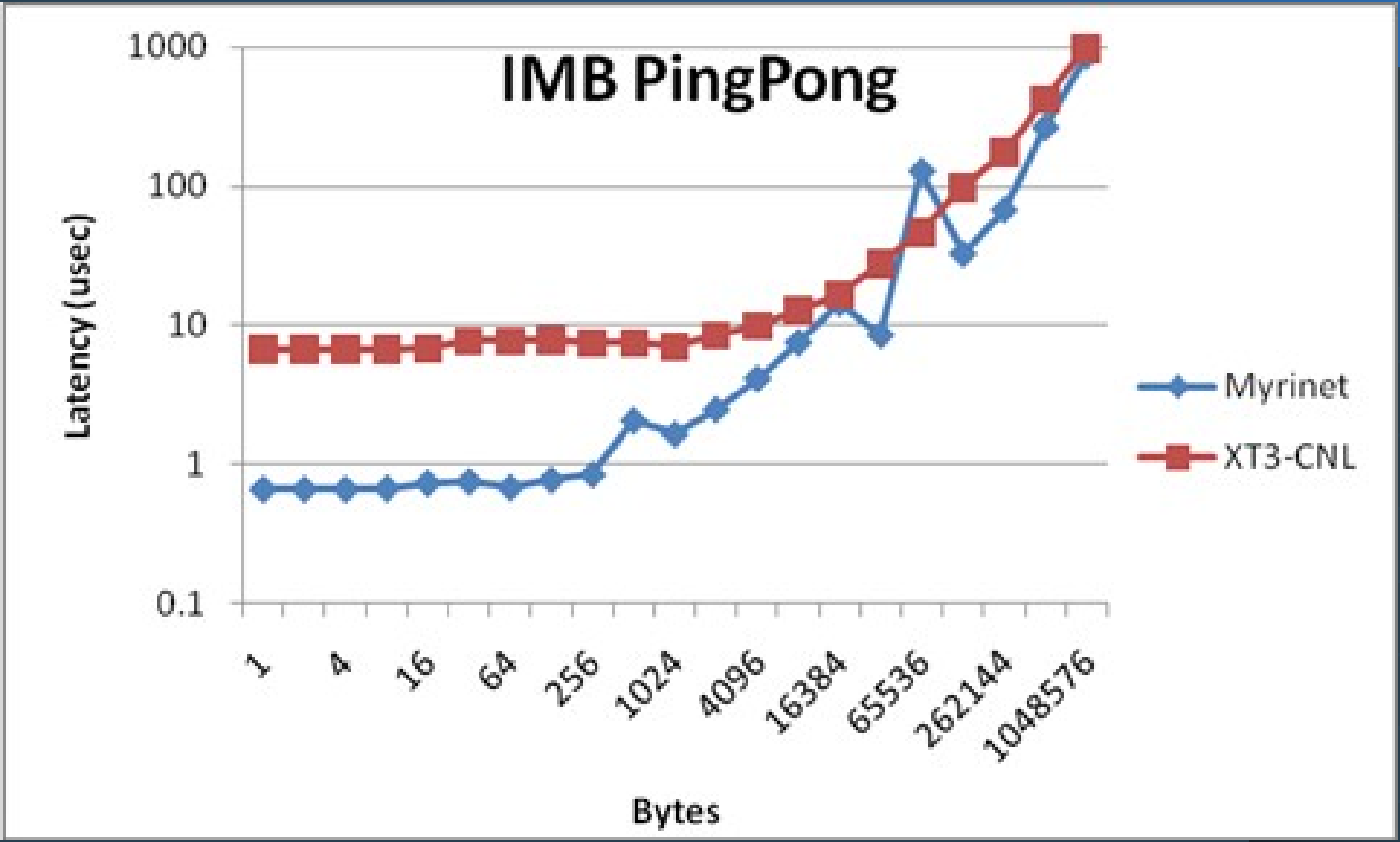
OpenMPI AllToAll	2	4	8	16	32	64
4	1.3	1.0	1.2	1.2	1.4	1.3
8	1.4	1.2	1.2	1.3	1.1	***1.5***
16	1.4	1.2	1.1	1.1	1.3	***1.5***
32	1.4	1.2	1.3	1.2	1.3	1.1
64	1.4	1.2	1.0	1.2	1.1	1.0
128	1.4	1.2	***1.6***	1.2	1.1	1.0
256	1.3	1.2	1.5	***2.0***	***2.1***	***2.9***
512	***1.5***	1.2	1.0	1.0	1.0	1.0
1024	1.3	1.3	***1.6***	***1.6***	***1.6***	***1.6***
2048	1.3	1.2	1.4	1.4	1.4	1.2
4096	1.2	1.3	1.3	1.3	1.2	1.2
8192	1.1	1.1	1.3	1.2	1.0	1.1
16384	1.0	1.1	1.3	1.1	1.0	1.1
32768	***1.7***	1.4	1.4	1.1	1.1	1.1
65536	***1.8***	***2.3***	1.5	1.3	1.2	1.0

Helics, MPICH, AllToAll/FFTW

FFTW	2	4	8	16	32	64
4	***1.7***	***1.7***	***2.1***	***2.0***	***2.1***	***1.8***
8	***1.7***	***1.7***	***2.2***	***1.8***	***2.1***	***2.0***
16	***1.7***	***1.7***	***1.7***	***1.7***	***1.8***	***1.9***
32	***1.7***	***1.8***	***2.2***	***1.7***	***1.7***	***1.9***
64	***1.7***	1.5	***1.8***	***1.6***	***1.6***	***1.6***
128	***1.7***	1.5	***1.8***	***1.6***	***1.5***	1.5
256	***1.6***	1.2	***1.8***	1.4	1.5	1.2
512	1.5	1.0	***1.7***	***1.9***	***2.2***	***2.2***
1024	1.4	1.1	1.0	1.4	1.5	1.4
2048	1.3	1.0	1.1	***1.5***	1.4	1.5
4096	1.2	1.0	1.2	1.4	1.4	1.3
8192	1.1	1.2	1.0	1.0	1.0	1.0
16384	1.0	1.0	1.0	1.0	1.0	1.0
32768	1.2	***3.5***	1.0	1.1	1.1	1.1
65536	1.3	1.1	1.1	1.4	1.0	1.0

Helics, MPICH, AllToAllV/FFTW

FFTW – AllToAllV	2	4	8	16	32	64
4	***1.8***	***1.6***	***2.9***	***4.5***	***6.7***	***9.6***
8	***1.7***	***1.5***	***2.9***	***3.9***	***6.2***	***6.9***
16	***1.6***	***1.6***	***2.4***	***3.8***	***4.3***	***6.3***
32	***1.6***	***1.6***	***2.6***	***2.7***	***4.0***	***6.0***
64	***1.6***	1.3	***1.9***	***3.0***	***4.5***	***4.8***
128	***1.6***	1.4	***1.9***	***2.8***	***3.0***	***2.9***
256	***1.5***	1.1	***2.7***	***3.2***	***3.1***	***3.7***
512	1.4	1.2	***1.7***	***1.9***	***2.3***	***2.3***
1024	1.4	1.0	1.0	1.4	1.4	***1.5***
2048	1.3	1.0	1.3	***1.5***	1.4	1.3
4096	1.2	1.1	1.3	1.4	1.4	1.4
8192	1.2	1.1	1.0	1.0	1.1	1.1
16384	1.2	1.1	1.0	1.0	1.1	1.2
32768	1.2	***3.4***	1.0	1.1	1.0	1.2
65536	1.0	***4.1***	1.2	1.1	1.3	1.4



- With Sadaf Alam

Helics/Jaguar, MPICH, AllToAll

4	***5.9***	***8.7***	***2.4***	***2.4***	***1.8***	***1.5***
8	***5.7***	***8.6***	***2.9***	***2.2***	***1.8***	1.3
16	***5.7***	***8.2***	***2.6***	***2.1***	1.4	1.2
32	***6.5***	***10.2***	***2.6***	***1.7***	1.4	1.2
64	***6.7***	***10.6***	***2.0***	***1.7***	1.5	1.1
128	***6.3***	***9.8***	***2.0***	***1.7***	1.3	1.1
256	***6.1***	***9.2***	***2.0***	1.5	1.0	1.2
512	***5.0***	***7.7***	***1.6***	1.2	***1.5***	1.4
1024	***4.8***	***6.3***	***2.5***	***1.9***	***1.8***	***1.7***
2048	***4.7***	***5.0***	***1.8***	1.4	1.3	1.1
4096	***3.5***	***3.6***	1.2	1.1	1.3	1.3
8192	***2.5***	***2.0***	1.4	1.1	1.2	1.0
16384	***1.8***	***1.7***	1.0	1.2	1.3	1.1
32768	***2.5***	1.2	1.0	1.2	1.4	1.2
65536	***2.4***	***3.3***	1.1	1.1	1.0	1.0

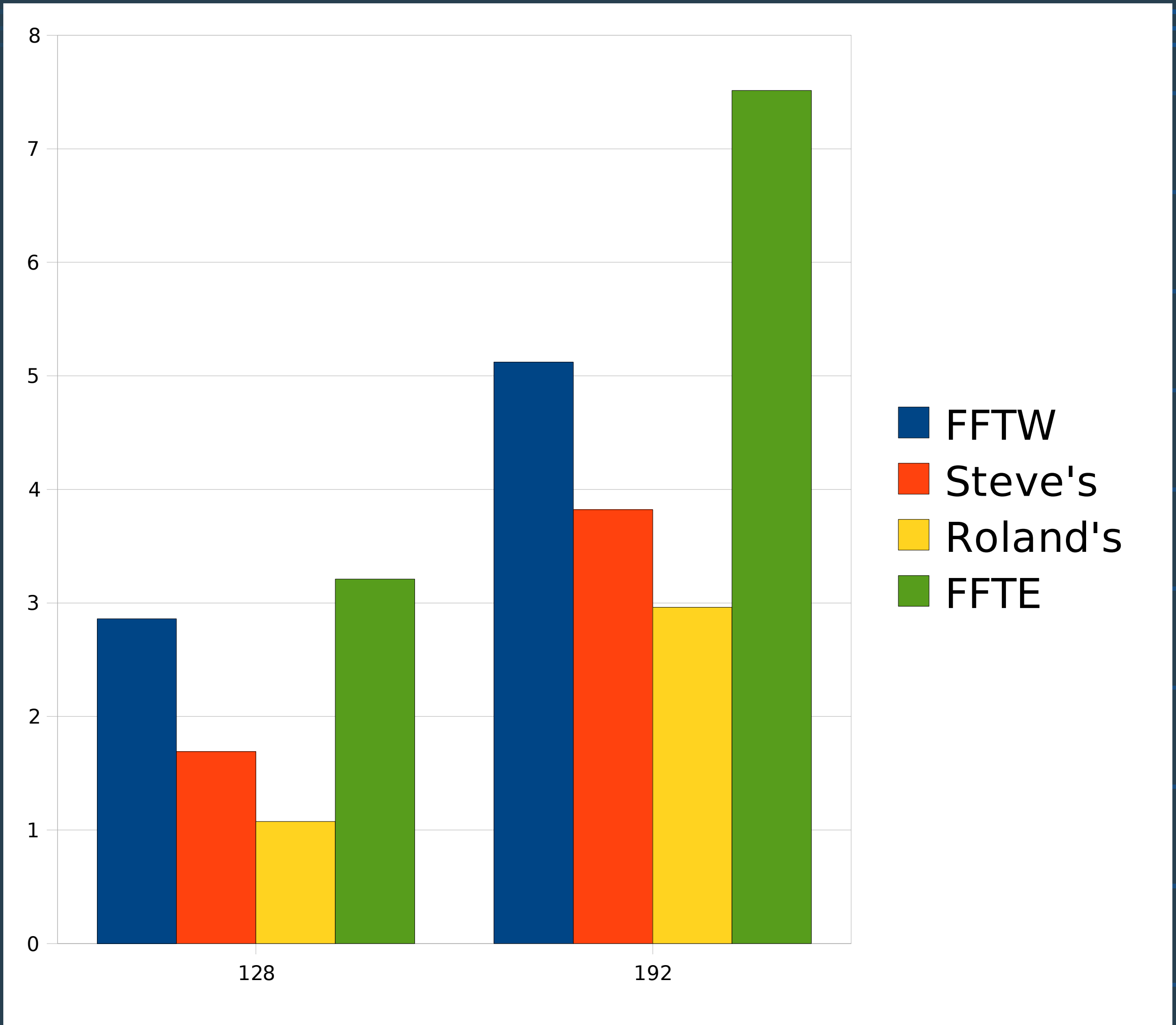
Helics/Jaguar, MPICH, FFTW

FFTW helics-jagaur	2	4	8	16	32	64
4	***9.8***	***9.6***	***4.9***	***4.4***	***3.5***	***2.8***
8	***9.7***	***14.4***	***5.6***	***3.4***	***3.2***	***2.6***
16	***9.3***	***11.6***	***3.8***	***3.2***	***2.4***	***2.4***
32	***10.7***	***10.8***	***5.6***	***2.5***	***2.3***	***2.5***
64	***11.0***	***9.4***	***3.7***	***2.4***	***2.4***	***2.1***
128	***10.3***	***9.2***	***3.7***	***2.5***	***1.9***	***1.6***
256	***9.9***	***7.2***	***3.6***	***1.9***	***1.4***	***1.7***
512	***7.2***	***5.3***	***2.6***	***1.4***	***1.4***	***1.7***
1024	***6.3***	***5.3***	***1.9***	***1.5***	***1.6***	***1.7***
2048	***5.6***	***4.3***	***1.6***	***1.6***	***1.4***	***1.6***
4096	***4.1***	***3.6***	1.3	1.2	1.2	***1.4***
8192	***2.7***	***2.5***	***1.4***	1.2	1.1	1.3
16384	***1.9***	***1.9***	1.2	1.2	1.2	1.1
32768	***2.0***	***3.0***	1.0	***1.4***	1.3	1.0
65536	***1.9***	***3.3***	1.1	***1.3***	1.1	1.1

Transpose optimization

- Usage of FFTW transpose
- Rectangular CPU grid

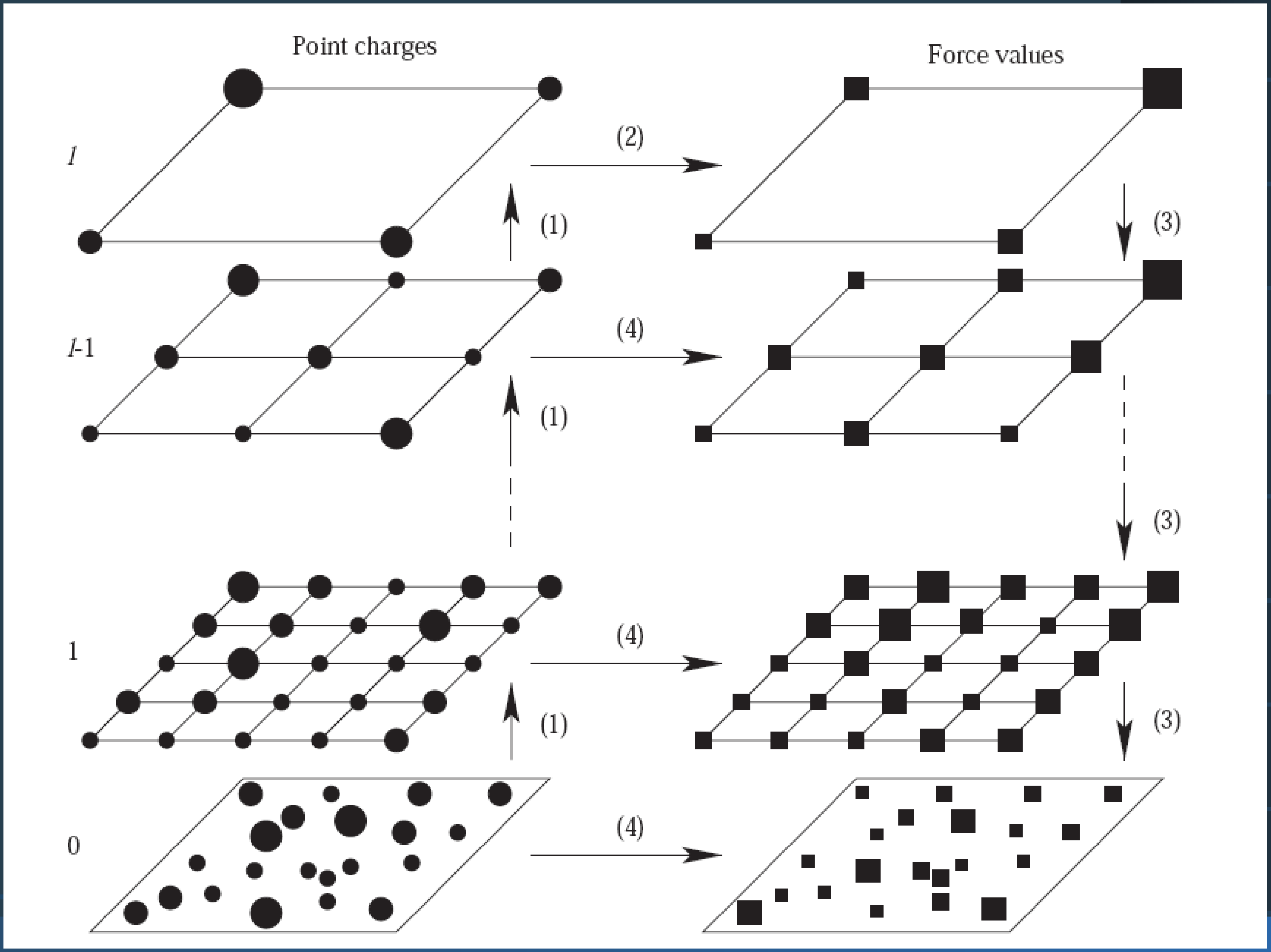
FFT Comparison



- N=192: 1152 CPUs, N=128: 1024 CPUs

Future

- FFTW transpose enables
  - Unevenly dividable data size
  - Real data
- Linear scaling with Multigrid



Summary

- 1d decomposition: algorithm limited
- 2d decomposition: network limited
- amount of data crucial
- MPI transpose not highly optimized
  - AllToAllV >3x slower
  - FFTW (auto-tuning) faster

Thanks

- All lecturer
- Heike; Innovative Computing Lab.
- Sadaf; National C. for Comp. Sciences
- Berk; Max Plank Institute Mainz, DE
- Matteo Frigo and Steven Johnson; MIT
- You all for listening