Distributed FFTs for integrated algorithms

Tze Meng Low
Elliott Binder

ECP EXASCALE COMPUTING PROJECT

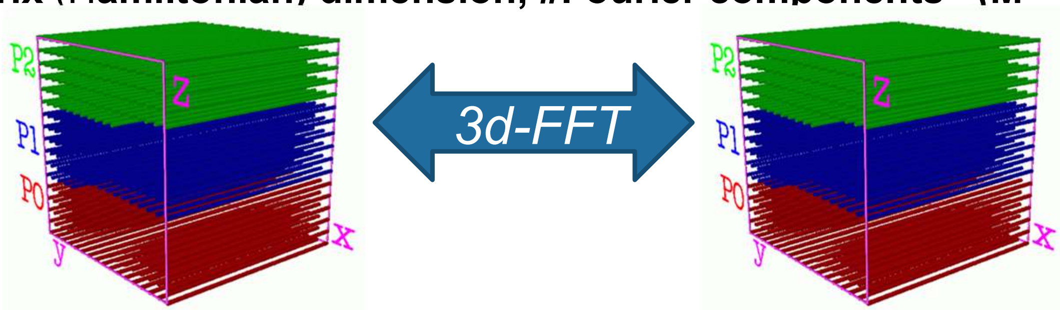
Planewave Pseudopotential Method in DFT

$$\{-\frac{1}{2}\nabla^2 + \int \frac{\rho(r')}{|r-r'|}dr' + \sum_I \frac{Z}{|r-R_I|} + V_{XC}(\rho(r))\}\psi_j(r) = E_j\psi_j(r)$$

Computational Task (CG solver)	Scaling
Orthogonalization	MN^2 (α N^3)
Subspace diagonalization	N^3
3d FFTs (most communications)	NMlogM (α N^2)
Nonlocal pseudopotential	MN^2 (N^2 real space)

N: number of eigenpairs (bands, states, electrons) required (lowest in spectrum) (N is in the range of 100s to 1000s)

M: matrix (Hamiltonian) dimension, #Fourier components (M ~ 256^3 to 1024^3)

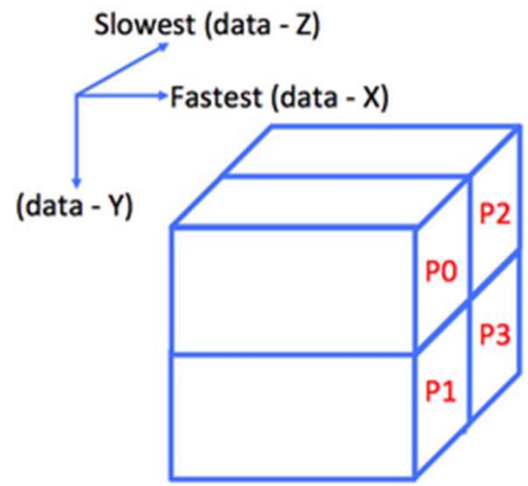


Basic 3DFFT

- 3 functions to invoke distributed 3DFFT. Similar to all many distributed 3DFFT framework

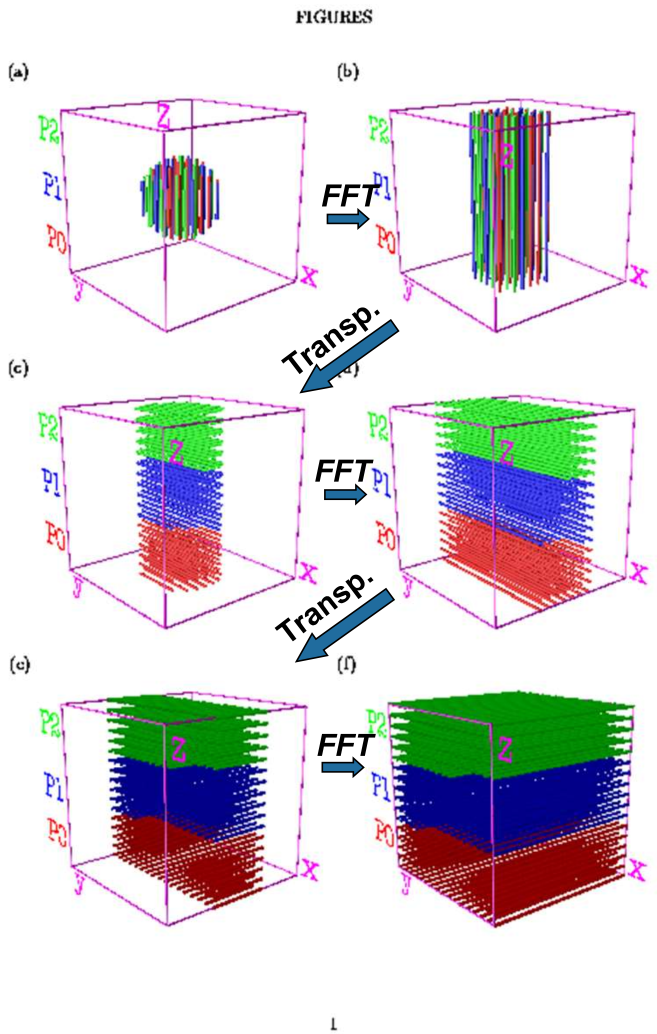
```
fftx_plan  plan = fftx_plan_distributed(r, c, M, N, K);

fftx_execute(plan, (double*)device_out_buffer,
                   (double*)device_in_buffer, FFTX_FORWARD);

fftx_plan_destroy(plan);
```

- Assumptions
  - Data is evenly distributed in blocks of pencils
  - All temporary data and communication buffers will be created as part of the plan
  - Requires at least 2 x (MNK / (rc)) additional space on the GPU for intermediate results



3D Data cube on 2x2 Grid

Requirements for Integrated Planewave algorithm



- Many zero elements in data cube
- Exploiting zero elements reduces computation and communication

Communications

- **Ratio of Cube to Cylinder Volume is π/16 = 0.196**
- **5.1 factor less data communicated compared to full (library) 3D FFT**

FLOPS

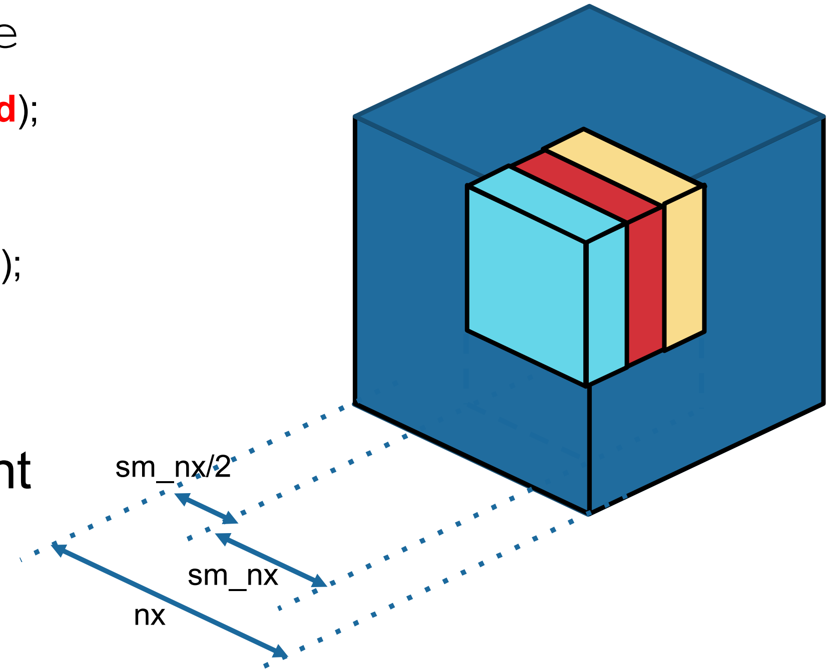
- **Three sets of 1D FFTS (π/16)N^2, (1/2)N^2, N^2**
- **Full 3D FFT 3N^2 sets of 1D FFTS**
- **1.70 factor lower FLOP count compared to full (library) 3D FFT**

Embedded 3DFFT

- Similar interface with is_embedded set to `true`

```
fftx_plan  plan = fftx_plan_distributed(r, c, M, N, K, is_embedded);

fftx_execute(plan, (double*)device_out_buffer,
                   (double*)device_in_buffer, FFTX_FORWARD);

fftx_plan_destroy(plan);
```

- Exploit locations of non-zeros to reduce amount of data communicated
- Padding performed for 2nd and 3rd stage performed as part of the communication
- Same assumptions as the basic 3DFFT



Backup Slides with code

```
fftx_plan  plan = fftx_plan_distributed(r, c, M, N, K, batch, is_embedded);

DEVICE_SYNCHRONIZE();
MPI_Barrier(MPI_COMM_WORLD);

for (int t = 0; t < 1; t++) {

  double start_time = MPI_Wtime();

  fftx_execute(plan, (double*)out_buffer, (double*)in_buffer, (is_forward ? FFTX_FORWARD: FFTX_BACKWARD));

  double end_time = MPI_Wtime();

  double min_time    = min_diff(start_time, end_time, MPI_COMM_WORLD);
  double max_time    = max_diff(start_time, end_time, MPI_COMM_WORLD);

  DEVICE_MEM_COPY(fftx_out, out_buffer, (Mo*No*Ko/p) * sizeof(complex<double>)*batch, MEM_COPY_DEVICE_TO_HOST);
  DEVICE_SYNCHRONIZE();

}
```

Execute with command

srun -A CSC304_crusher -N 4 -n $(($P * $P)) --gpus-per-node 8 --gpu-bind=closest -t 00:01:00 ./testmddft3d.x

EXASCALE COMPUTING PROJECT

Performance on Crusher