Domain Decomposition in the GPU Era

Domain Decomposition and DDHMC refresher

We decompose space time into hypercuboidal blocks of size L^4. The block coordinate is (integer division),

$$b_i = x_i/L,$$

and the intra block coordinate is,

$$l_i = x_i|L,$$

while we assign to each block a parity,

$$p = (\sum_i b_i)|2.$$

We then define two domains Ω and $\bar{\Omega}$ as the set of points within blocks of parity zero and parity one respectively. Their *exterior* boundaries haloes are ∂_Ω and $\partial_{\bar{\Omega}}$ such that,

$$\partial_\Omega \cap \Omega = \emptyset,$$

and

$$\partial_{\bar{\Omega}} \cap \bar{\Omega} = \emptyset,$$

respectively.

The Dirac operator, with an appropriate non-lexicographic ordering may then be written as

$$D = \begin{pmatrix} D_\Omega & D_\partial \\ D_{\bar{\partial}} & D_{\bar{\Omega}} \end{pmatrix}.$$

- Take the view that the domain will be the ***whole node***
- **Schur decompose and take determinant:**

$$\begin{pmatrix} D_\Omega & D_\partial \\ D_{\bar\partial} & D_{\bar\Omega} \end{pmatrix} = \begin{pmatrix} 1 & D_\partial D_{\bar\Omega}^{-1} \\ 0 & 1 \end{pmatrix} \begin{pmatrix} D_\Omega - D_\partial D_{\bar\Omega}^{-1} D_{\bar\partial} & 0 \\ 0 & D_{\bar\Omega} \end{pmatrix} \begin{pmatrix} 1 & 0 \\ D_{\bar\Omega}^{-1} D_{\bar\partial} & 1 \end{pmatrix}.$$

$$\det D = \det D_\Omega \det D_{\bar\Omega} \det \left\{ 1 - D_\Omega^{-1} D_\partial D_{\bar\Omega}^{-1} D_{\bar\partial} \right\},$$

- **Update only links not crossing between nodes**
- **Two factors on small timestep, boundary determinant on coarse timestep**

$$R = 1 - \mathbb{P}_{\bar\partial} D_\Omega^{-1} D_\partial D_{\bar\Omega}^{-1} D_{\bar\partial}.$$

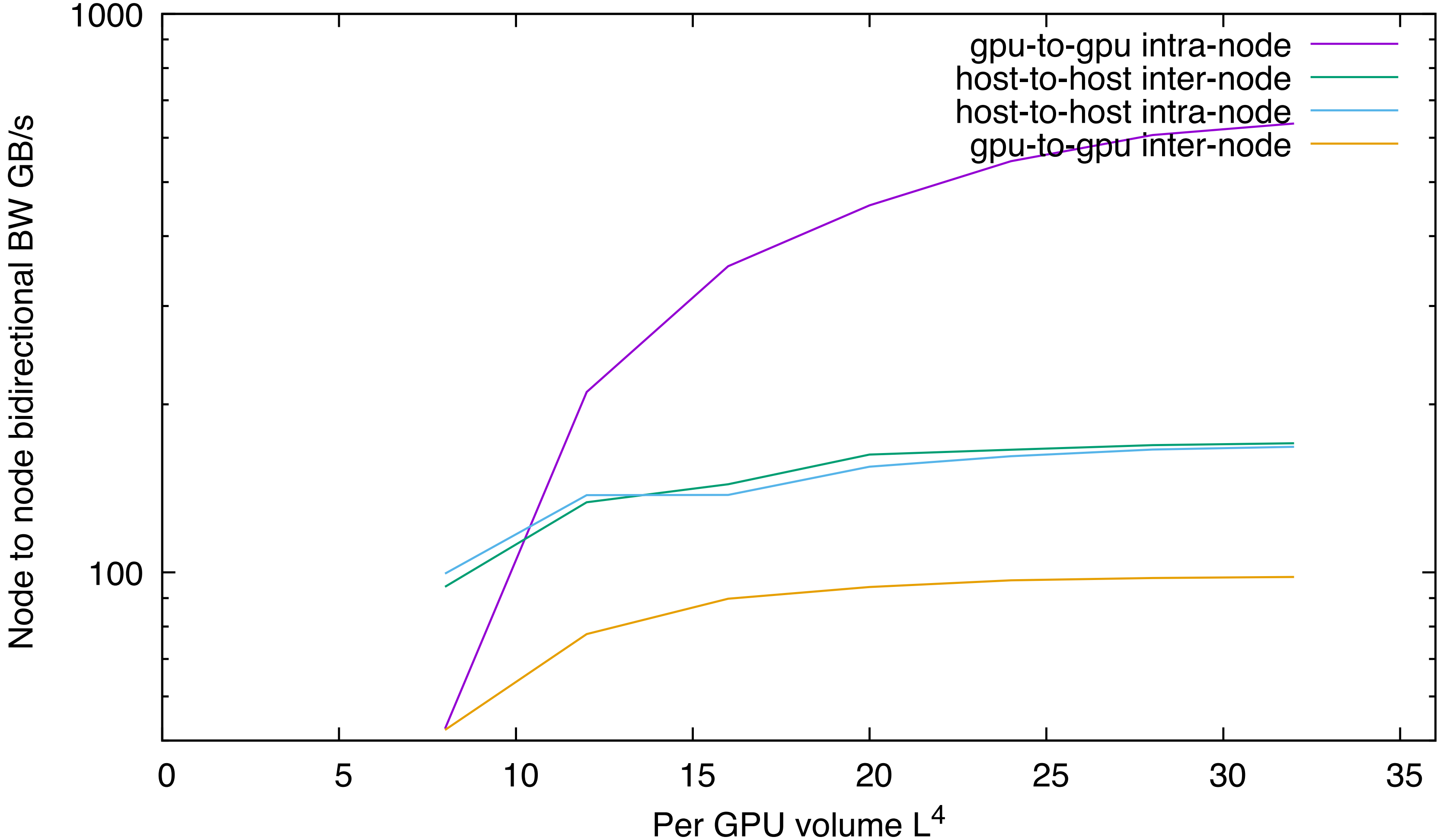
- **Force term suppressible in distance between active links and the boundary pseudofermion**

$$\delta R^{-1} = \mathbb{P}_{\bar\partial} D^{-1} \delta D D^{-1} D_{\bar\partial}.$$

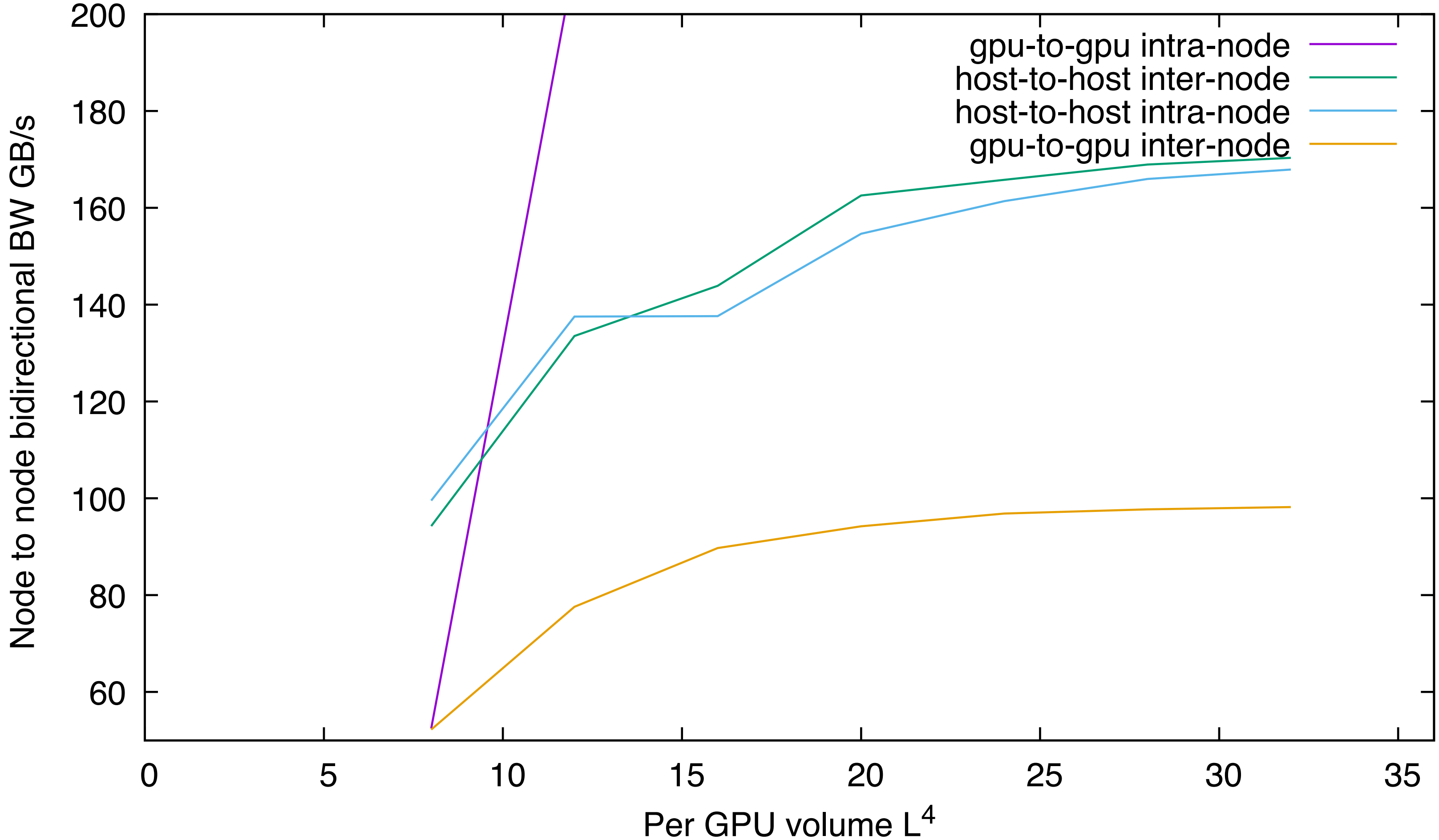
Why adopt DDHMC?

- Rational differs from previous use by CLS (who had 6^4 domains).
- GPU systems will have (and do have) substantial caches.
    - Ratio between multinode and single node performance will grow
- Can imagine 32^4 data points per GPU and 32x64^3 per node
    - Percentage of active links is (31/32)^4 = 88%
- Better sampling efficiency
- Imagine O(60TF/s) single node is possible. Already expect 40TF/s on A100/80 with 8 GPU's
- If 1/32 of data comes from off node can generate enormous network requirement
    - 6x higher then fastest current system

$$60\mathrm{TFlop/s} \times 0.65B/F/32 = 1200GB/s.$$



1000
100
Node to node bidirectional BW GB/s
0
5
10
15
20
25
30
35
Per GPU volume L^4
gpu-to-gpu intra-node
host-to-host inter-node
host-to-host intra-node
gpu-to-gpu inter-node



Node to node bidirectional BW GB/s
Per GPU volume L^4
gpu-to-gpu intra-node
host-to-host inter-node
host-to-host intra-node
gpu-to-gpu inter-node
200
180
160
140
120
100
80
60
0
5
10
15
20
25
30
35

Booster performance

- **After a lot of effort working around MPI issues**

Atos Sequana - 16 nodes, 2x2x2x2, comms in 4D

- **2x AMD Rome**
- **4x A100**
- **4x HDR-200**

```
================================================================================
Grid : Message : 532.798612 s :  Per Node Summary table Ls=12
================================================================================
Grid : Message : 532.798619 s :  L                Wilson          DWF4
Grid : Message : 532.798622 s : 8                 10427.4         193879.8
Grid : Message : 532.798626 s : 12                77312.0         1110694.8
Grid : Message : 532.798630 s : 16                228321.2        2066010.2
Grid : Message : 532.798685 s : 24                1603918.3       4094931.1
Grid : Message : 532.798689 s : 32                2384139.2       5768859.2
================================================================================
```

Volume per GPU - will weak scale well

- **4.8x faster than Summit per node !**
- **Expect another 25% if GDR improves**

Conclusions

Projection of gains:

- **Summit : 6.9TF/s vs 1.2 TF/s**
  - **5.7x gain**
- **Booster : 9.5TF/s vs 5.7 TF/s**
  - **1.6x Gain**
- **Hypothetical 60TF/s node with Booster network:**
  - **10x Gain**

My Conclusion: it is imperative to develop DDHMC for multiGPU systems

May afford 10x acceleration of HMC

Valence analysis is accelerated by trivial parallelism + deflation, HMC IS NOT