QCD on Earth Simulator


Center for Computational Physics
Institute of Physics
University of Tsukuba

High Energy Accelerator Research Organization (KEK)
Physics Motivation

Nf=2+1 full QCD Simulation

by CP-PACS/JLQCD joint collaboration (ref.[1])

Program: PHMC (Polynomial Hybrid Monte Carlo)

<table>
<thead>
<tr>
<th>Name Place</th>
<th>GPU</th>
<th>CPU</th>
<th>Memory</th>
<th>Total</th>
<th>Nodes</th>
<th>GFLOPS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hitachi SR8000/F1KEK</td>
<td></td>
<td></td>
<td></td>
<td>1200</td>
<td>64</td>
<td>1200</td>
</tr>
<tr>
<td>CP-PACS CCP,Tsukuba</td>
<td>0.3</td>
<td></td>
<td>2048</td>
<td>614</td>
<td>2048</td>
<td></td>
</tr>
<tr>
<td>Hitachi SR8000/G1CCP,Tsukuba</td>
<td>14.4</td>
<td></td>
<td></td>
<td>172.8</td>
<td>12</td>
<td>172.8</td>
</tr>
<tr>
<td>Fujitsu VPP5000SIPC,Tsukuba</td>
<td>9.6</td>
<td></td>
<td>768</td>
<td>24</td>
<td>80</td>
<td>768</td>
</tr>
<tr>
<td>Earth Simulator ES Center</td>
<td>64</td>
<td></td>
<td>640</td>
<td>40960</td>
<td>64</td>
<td>40960</td>
</tr>
</tbody>
</table>

#node for QCD is a rough estimate
ES QCD Project

Study of the Standard Model of Elementary Particles on the Lattice with the Earth Simulator

leader: A. Ukawa @ CCP, Tsukuba

approved by Japan Lattice Field Theory Forum

approved by the ES Center as one of "Projects for using the Earth Simulator" (Epoch Making Simulation)
ES Machine Overview

# Processing Nodes (PN) 640 crossbar switches
bandwidth: 12.3 GB/s bi-directional

Each PN has 8 vector-type arithmetic processors (AP) (shared memory) peak 8x8GFlops = 64 GFlops

total peak speed 640x64 GFlops = 40.96 TFlops

Software we use
fortran-90
micro-tasking by hand
parallelization
for 8AP in a node
MPI for inter-node comm.

Interconnected Network (fullcrossbar, 12.3GB/s x 2)
Coding Style

Test of MULT (Wilson-Dirac matrix multiplication) on 1 AP(CPU) (performance def. by 1296 flops/site)

A: naive with list vectors
B: developed at CP-PACS
   vectorized in t direction
C: tuned for CP-PACS
D: tuned for SR8000
   same principle as for B
E: tuned for VPP500
   see next page
F: tuned for VPP5000
   conditional branch

Performance ranges 10% -- 60% (L=32), 75% (L=48)
PHMC developed for SR8000 @ KEK, low performance on ES

linearized in z-t plane, then subdivided into four

long vector without list vectors

few redundant calculations at boundaries
Target Lattice Size

Performance depends on lattice size and #nodes we use.

a) 20x20x20xNt on 5x2 nodes = 4x10x20xNt per node
b) 24x24x24xNt on 3x4 nodes = 8x 6x24xNt per node
c) 32x32x32xNt on 4x4 nodes = 8x 8x32xNt per node

\[ Nt \approx 2 \, Ns \]

legends in following figures:
ftrace (analyzer) including redundant calculations
theor. only effective part (1296flops/site)
original:
- calculate contribution from 8 dirs. in one large loop, then sum-up

Intending to overlay calculations and communications, we have rewritten:
- t,z direction first, then sum-up, y,x direction next, then sum-up
- array structure is also different

Performance drops by about 15% due to the revision
Performance drops when vector length just exceeds multiples of 256 (=size of vector registers)
MULT micro-tasking Parallelization

Cost for micro-tasking parallelization is 3-4 point
Boundary copy is heavy
  4 point for 1AP   7 point for 8AP
gather, then MPI isend, MPI_irecv, then scatter to construct long messages
all nodes connected by crossbar switches => throughput is (almost) the same if volume/shape per node is the same

MPI performance is consistent with a report from the ES center
Comm. performance drops by 20% due to buffer copy
BREAKDOWN OF OVERHEADS (1)

MULT performance (%) relative to theoretical peak volume/node is fixed

<table>
<thead>
<tr>
<th></th>
<th>(1)</th>
<th>(2)</th>
<th>(3)</th>
<th>(3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>best code, best vlen (ftrace)</td>
<td>78.1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>as above (theor.)</td>
<td>73.9</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>best code (target Nt)</td>
<td>66.2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>code for phmc (1AP)</td>
<td>56.5</td>
<td>52.8</td>
<td>54.8</td>
<td>61.6</td>
</tr>
<tr>
<td>(8AP)</td>
<td>49.6</td>
<td>49.1</td>
<td>52.5</td>
<td>58.9</td>
</tr>
<tr>
<td>with boundary copy</td>
<td>37.7</td>
<td>41.3</td>
<td>46.1</td>
<td>51.9</td>
</tr>
<tr>
<td>with copy and comm.</td>
<td>20.2</td>
<td>25.6</td>
<td>32.9</td>
<td>35.4</td>
</tr>
</tbody>
</table>
| cost of copy&comm., (%)  | 59.2  | 47.9  | 37.3  | 39.9  | (*: relative to total execution time)
Breakdown of Overheads (2)

Volume/cpu is fixed

Inter-node communication is very heavy for our application
Scaling property of MULT

Sizes are fixed to 24x24x24x48

---

#AP  
Sustained Speed (Gflops)
PHMC profile and performance

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Node</td>
<td>Size</td>
<td>Node</td>
<td>Size</td>
</tr>
<tr>
<td>5x2</td>
<td>3x4</td>
<td>4x4</td>
<td>4x4</td>
</tr>
<tr>
<td>copy (%)</td>
<td>36.9</td>
<td>28.6</td>
<td>21.4</td>
</tr>
<tr>
<td>Mult (%)</td>
<td>24.8</td>
<td>29.4</td>
<td>34.8</td>
</tr>
<tr>
<td>Mult of clinv (%)</td>
<td>13.4</td>
<td>18.3</td>
<td>20.1</td>
</tr>
<tr>
<td>Bicgstab (%)</td>
<td>9.8</td>
<td>9.2</td>
<td>8.8</td>
</tr>
<tr>
<td>GFLOPS/cpu</td>
<td>2.01</td>
<td>2.29</td>
<td>2.87</td>
</tr>
<tr>
<td>Efficiency (%)</td>
<td>24.9</td>
<td>28.6</td>
<td>35.9</td>
</tr>
</tbody>
</table>

Measured by profiler

**Multiplication of the inverse clover term (clinv) is next heavy.**
Comparison of PHMC code performance

lattice size: 20x20x20x40

<table>
<thead>
<tr>
<th>Name</th>
<th>Gflops/node</th>
<th>Node #</th>
<th>Node size</th>
<th>Sustained perf.</th>
<th>Sustained perf. (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SR8000/F1</td>
<td>12</td>
<td>32</td>
<td>384</td>
<td>133.8</td>
<td>35</td>
</tr>
<tr>
<td>CP-PACS</td>
<td>0.3</td>
<td>512</td>
<td></td>
<td>153.6</td>
<td>31.0</td>
</tr>
<tr>
<td>SR8000/G1</td>
<td>14.4</td>
<td>8</td>
<td>115.2</td>
<td>50.4</td>
<td>44</td>
</tr>
<tr>
<td>VPP5000</td>
<td>9.6</td>
<td>8</td>
<td>76.8</td>
<td>33.7</td>
<td>44</td>
</tr>
<tr>
<td>Earth-Simulator 64</td>
<td>10</td>
<td>640</td>
<td>200.0</td>
<td>31</td>
<td></td>
</tr>
</tbody>
</table>

8 min/traject. for $m_{ud} \approx m_s \approx m_{s,physical}$ on ES
Conclusions and Future Plan

Arithmetic operation performance strongly depends on coding style

High arithmetic operation performance can be achieved when coded suitably for vector processing

Memory throughput is relatively low for our application

Communication cost is very high for our application

Overlap of arithmetic operations and communications is required. This is in progress
Acknowledgement and References

We thank
ES center for approving our project
K.Itakura (ES Center) for continuous technical support
RCNP (Osaka Univ.) for permission to use SX5 at early stage of this work

References
[1] Light hadron spectrum in three-flavor QCD with O(a)-improved Wilson quark action,
T.Kaneko (CP-PACS/JLQCD Collaboration), this conference (Thursday)
[2] Polynomial hybrid Monte Carlo algorithm for lattice QCD with an odd number of flavors,
[3] Earth Simulator Center Webpage,
http://www.es.jamstec.go.jp/