DEM-1: A Particle Simulation Machine for Efficient Short-range Interaction Computations

Ryo Takata                                    Kenji Kise, Hiroki Honda, Toshitsugu Yuba
Graduate School of Information Systems        Graduate School of Information Systems
University of Electro-communications          University of Electro-communications
1-5-1, Chofigaoka, Chofu, Tokyo, Japan       1-5-1, Chofigaoka, Chofu, Tokyo, Japan
Image Technology Laboratory Corp.             {kis, honda, yuba}@yuba.is.uec.ac.jp
3-36-19, Tamagawa, Chofu, Tokyo, Japan       takata@yuba.is.uec.ac.jp

Abstract

This paper describes the architecture of high performance, particle simulation machine, DEM-1 for short-range particle interaction computations. All existing particle simulation machines have specialized pipelines to calculate long-range particle interactions effectively. However, their ability to perform particle simulations efficiently diminishes with short-range interactions. The communication cost component of particle simulations will play a significant role in performance when the computation cost becomes $O(N)$. DEM-1’s three dimensional torus high-speed network reduces the communication cost while 2048 local processors perform the time integration. The elimination of pipeline bubbles in DEM-1 is achieved by specially designed cut-off judgment units. Each specialized pipeline consists of dedicated data path supported by position vector prefetch dual ported memory. The performance of DEM-1 will be presented with very large-scale Embedded Atom Method (EAM) molecular dynamics simulations.

1. Introduction

Particle simulation has been widely used in various fields such as astrophysics, molecular dynamics, gas-solid flows, and computer graphics. These simulations can be executed in parallel, because most of the computation time is spent on the calculation of interactions that can be evaluated simultaneously. The machine GRAPE[1] has been developed to reduce the execution time of large-scale astrophysical simulations. The machines MD GRAPE[2], MD One[3], MD Engine[4], and MDM[5] have been developed to reduce the execution time of molecular dynamic simulations of protein and other particle simulations. These parallel particle simulation machines have specialized parallel hardware pipeline to calculate the arbitrary central force and hence to increase the performance. In certain particle simulations, short-range cut-off interactions dominate the system behavior. Molecular dynamics simulation of metallic solids with Embedded Atom Method (EAM) falls into this category of computations.

It is difficult to perform these simulations efficiently on the available particle simulation machines because they are not optimized to perform such short-range interactions. Here, we propose a particle simulation machine DEM-1 optimized for such applications. Section 2 describes the background and motivation. Section 3 focuses on the architectural details of DEM-1. Section 4 summarizes the performance of DEM-1. Section 5 summarizes related work.

2. Background and motivation

2.1 Sample application

Embedded Atom Method (EAM) [6] is widely used in molecular dynamics simulations of metallic solids. In EAM, the potential energy is calculated by using the following formula.

$$E_{tot} = \sum_i F_i(\rho_{\text{tot}}) + \frac{1}{2} \sum_{i \neq j} \phi_{ij}(r_{ij})$$  \hspace{1cm} (1)

The force exerted onto atom $i$ is represented by

$$f_i = -\sum_{j \neq i} \left( F_i(\rho_j, \rho_{\text{tot}}) + F_j(\rho_i, \rho_{\text{tot}}) \right) \phi'_{ij}(r_{ij}) r_{ij}$$  \hspace{1cm} (2)

For example, consider the simulation of BCC iron. The cut off radius for such computation is 3.64Å. Typically 4 particles exist inside a cell. So approximately 100 particles have a chance to interact with the target particle in linked-cell-list method. However, only 10-20 particles actually exert force on the target particle.

The simulation parameters of the computation vary according to the dimensions of the target system and the objective of the simulation. Radiation damage simulation
of iron requires more than one million atoms and 10 thousand steps. Creep or crack growth simulations need over 100 million atoms.

In EAM simulations with a particle simulation machine, two potentials \((\rho, \Phi)\) and two forces \((\rho', \Phi')\) are calculated by hardware pipeline, while \(F'(\rho)\) calculation or time step integration such as particle position vector updates are done by the host computer.

### 2.2 Performance bottleneck

The performance of MD One particle simulation machine\[3\] was analyzed with a model;

\[
T_{\text{step}} = T_{\text{pre}} + T_{\text{w}} + T_{\text{calc}} + T_{\text{r}} + T_{\text{other}} + T_{\text{move}} + T_{\text{movc}} + T_{\text{other}} (3)
\]

where \(T_{\text{step}}\) is overall time required for a step, \(T_{\text{pre}}\) is initialization time of MD One, \(T_{\text{w}}\) is transfer time of position vectors and coefficients, \(T_{\text{calc}}\) is pipeline calculation time (including target particle position vector write time), \(T_{\text{r}}\) is result read time, \(T_{\text{other}}\) is other calculation time of bond interactions or center of mass velocity correction, etc., \(T_{\text{move}}\) is position vectors, velocities, etc. update time, and \(T_{\text{movc}}\) is other overheads such as logging in, respectively.

All existing particle simulation machines assume that the calculation time dominates the total performance.

\[
T_{\text{calc}} >> T_{\text{pre}} + T_{\text{w}} + T_{\text{r}} + T_{\text{other}} + T_{\text{move}} + T_{\text{movc}} (4)
\]

This may be true for the simulations with interactions whose calculation costs are \(O(N^2)\) or \(O(N^{1.5})\). Here \(N\) is the number of particles in the system. In the case of Lennard-Jones interactions the calculation cost \(O(N)\) is significant. Therefore, it is possible to discuss them on the same basis. However, it is important to note that even with \(O(N^2)\) computations, the effective performance can rapidly decrease with the increase of number of boards. In many molecular dynamics applications such as EAM, the calculation cost of interactions is \(O(N)\) and the effect of other terms cannot be neglected. Most of these costs can be attributed to: (1) communication time between host computer and hardware pipelines, (2) execution time of calculations related to the portion of simulations other than the interaction calculation.

In addition to these bottlenecks, most of the existing particle simulation machines have inefficient pipeline for computations related to particle pairs encountered in linked-cell-list method particle search. The pipeline requires approximately 13 times many operations than software calculations (see Figure 1 and Appendix A). This reduces the effective performance of existing particle simulation machines by one order. The original GRAPE was targeted to astrophysical simulations where gravitational forces are calculated with no cut off. It appears that the GRAPE family members inherit its architecture with no performance improvement. MD-Engine \[4\] has facility to utilize generated pair list without software overhead, but its performance improvement factor did not exceed 2. Thus, it is possible to achieve another performance improvement of 6.5.

### 3. Architecture of DEM-1

#### 3.1 Design policy

DEM-1 consists of communicating microelements; each integrates 4 MIPS microprocessors, 4 hardware pipelines and link interface along with external memory. Eight elements are connected together to create a higher-level structure. Thus, three levels of integrations are defined in DEM-1 system configurations. These higher-level elements always have faster interconnection speed than the lower level elements. DEM-1 has three-dimensional torus network on its top level and the simulation space can be mapped on to it spatially. The processors, that are located adjacent to the hardware pipelines, work in parallel to accelerate \(O(N)\) tasks such as time integration. In addition to that, existence of local processors makes the local time step integration possible and hence resulting in great communication cost reduction.

DEM-1 is equipped with specialized pipelines to calculate short-range interactions of particles. Even though the calculation cost is only \(O(N)\), each potential or force calculations requires as many as 30 arithmetic operations. More than 90% of the total computation time consists of such computations. Therefore, it is possible to accelerate it by parallel hardware pipelines, as it is with existing particle simulation machines. Naturally, care should be taken not to block operations by external and/or internal bus bottleneck when designing the pipeline.

DEM-1 integrates dual ported memory and pre-fetches position vectors (and coefficients) of the interacting particles. To eliminate the pipeline bubble, we introduce multiple cut off judgement units before its main pipeline.

DEM-1 will be provided to users in three configurations:

![Figure 1. Linked-cell-list method and cut off sphere (2 dimensional view)](image)
1) A Compact PCI add-on board.
2) A node of up to 8 DEM-1 boards.
3) A cluster consists of up to 8 DEM-1 nodes.

System specifications of DEM-1 are listed in Table 1.

Figure 2 describes these three different configurations and their hierarchical connections. By adding one DEM-1 board to a PC, particle simulations can reach with performance over 100Gflops. A DEM-1 node in a 6U system rack arrangement could provide 1Tflops - computing speed by the side of the desk.

The DEM pipelined processor Chip is designed to achieve the goals outlined above. The DEM pipeline is upward compatible with MD-GRAPE chip[2] and can calculate arbitrary central forces. Therefore, DEM-1 should be able to execute most of the particle simulations running on MD-GRAPE or MD One[3] such as molecular dynamics simulations of proteins. A DEM Chip integrates 4 pipelines and 4 processors to have 14.4Gflops effective performance.

3.2 Detailed design

**DEM-Cluster** A DEM-Cluster consists of 8 DEM-nodes connected via 3 RHiNET2-SW[7]. Each node can communicate 3 contacting nodes with approximately 1Gbyte/sec speed in parallel. Additional RHiNET2-SW connects the cluster to external network.

**DEM-Node** Up to 8 DEM-boards are tightly connected to make a DEM-node. The node has maximum of three RHiNET2 NICs to communicate with other computers/storages or DEM-nodes.

**DEM-Board** A DEM-board integrates 8 DEM-chips, particle memories connected to each DEM-chip, a switch-FPGA with 6 channel-link interface transceivers and a 64bit, 66MHz PCI interface. A DEM-board has 115.2Gflops performance and memories that can hold 8,000,000 particles. PCI interface is used to communicate with host CPU in single board configuration. Up to 256Mbytes of DDR FCRAM attached to each DEM-chip is used to store position vectors, velocities, accelerations, derivative of accelerations, coefficients and linked-cell-lists. Six LVDS channel-link interfaces connected to switch FPGA have 133 Mbytes transfer rate to each channel and direction.

**DEM-Chip** Figure 3 shows the block diagram of the DEM-Chip. The DEM-chip integrates four hardware pipelines with dedicated on chip processors and cache memories, memory controller, which handles linked-cell-list addressing, on chip DRAM memory and local link interface in a single LSI. The 8 bytes wide memory bus of DEM-Chip can transfer 16 bytes per main CLK cycle with its DDR FCRAM interface.

**DEM-Pipeline** Block diagram of DEM-pipeline is shown in Figure 4. The design of DEM pipeline is based on the MD GRAPE CHIP[2] with 2 major enhancements: (1) Cut off judgement unit, (2) position vector pre-fetch dual портовой памяти.
3.3 Pipeline operations

Two cut off operation units use reduced accuracy and fixed point arithmetic. The reduced accuracy results in a slight loss in efficiency. Because of this imprecise arithmetic several particles actually lying outside of the cut off sphere may pass through the unit (see "rough cut off sphere" in Figure 1). However, this improves the performance and hardware resource utilization.

The frequency of two cut off judgement units is twice the frequency of the main pipeline. These units eliminate the bubbles in main pipeline and increase the effective performance of the entire pipeline by a factor of four. Hardware resources required for the cut off judgement units are estimated to be less than 10% of the total pipeline. In addition, main pipeline can calculate two interactions of a particle pair alternatively, for the pipeline with two function tables.

Figure 5 shows detailed block diagram of cut off judgement unit. In DEM system, coordinates are converted to 40bit signed fixed point format and then stored in memory. Fixed-point 40bit subtractions are performed to obtain relative distance vector between particles i and j. Obtained relative distance is represented also in 40bit fixed point format but their maximum absolute values are limited to twice of unit cell size, because particles i and j are contained within the neighboring cells. Therefore we can drop certain upper bits without any problems. Also, we drop lower bits for we allow less accuracy in rough cut off judgement.

In current design, we extract 8 bits signed integer to calculate rough squared distance between particle ij. The extract point depends on the simulation conditions (i.e. system size and unit cell size), so that it is made possible to adjust extract point parameter (POINT) by software. Cut off radius, RCUTSQ, used in the final comparator stage is also programmed by software. The programmer is expected to take account of the errors introduced by the imprecise arithmetic when he determines RCUTSQ.

Pipeline operation timing is shown in figure 6. Note that two cut off judgement units run in double frequency. In the example shown in Figure 6, cut off judgement unit #2 finds two neighboring particle pairs (r_{11}, r_{12}) and put them into cut off FIFO #2. Then main pipeline reads relative distance vector r_{11} form cut off FIFO #2 to calculate r and \phi sequentially. Pipeline controller selects
accumulator\( (2, \rho) \) and accumulator\( (2, \phi) \) respectively to accumulate these results. Next, particle pair \( l_2 \) is calculated in the same manner. During main pipeline calculation, two cut off judgement pipelines are busy checking for next candidate particle pairs but failed to find any. Cut off FIFOs become empty and main pipeline executes a bubble cycle, which does not affect any accumulators. Then cut off judgement unit \#1 and \#2 find particle pairs \( r_{10} \) and \( r_{10} \) to be inside cut off sphere. Main pipeline resumes effective cycle that updates accumulator\( (1, \rho) \). Cut off judgement unit halts when it detects that the corresponding cut off FIFO becomes full.

Unlike other particle simulation machines, DEM-1 has dedicated data flow path to each pipeline. This feature eliminates pipeline overhead introduced by pipeline data sharing and increases effective performance. This "data sharing overhead" will be discussed in section 5 in detail.

DEM chip have eight 16Kbytes position vector pre-fetch dual ported memories. Usually, data have been written to these 8 DPMs by broadcast. In others words, they have copies of each other. However, their reads can occur independently. Position vectors and coefficients are used many times repeatedly during the pipeline calculation. Thus it is possible to pre-fetch them via relatively slower external bus without reducing the pipeline performance. External bus bandwidth required to keep internal pipelines running busy depends on the size of the dual ported memory and pre-fetch strategy. Table 2 and figure 7 show some typical cases of pre-fetch strategies. The memory controller of DEM-Chip pre-fetches particle along \( z \) axis. In case\#1, a single column of cells and surrounding cells are read into pre-fetch dual ported memory. The case requires 21.5 bytes/clock transfer rate, which is higher than 16 bytes/clock peak transfer rate of DEM-Chip external memory bus. On the other hand, only 5.4 bytes/clock transfer rate is sufficient in case\#2 in which 4x4 columns of cells are calculated at a time. The number of xy planes contained in 16K bytes is 5.7. Note that the minimum number of xy planes is 4 considering overlapped pre-fetch during pipeline calculations. Number of average particles, which are contained in a cell, may increase depending on the interactions we adopt. We can adjust "object cell size" (Lcell) parameter to get optimum transfer rate (Nbpc) and number of xy planes in DPM (Nzpl). Case\#3 shown in Table 3, in which number of particles within a cell (Npc) equals to 8, describes such a situation.

4. Performance

4.1 EAM simulations on DEM-1

DEM-1 executes EAM simulation as the following.

(1) Load initial coordinates and velocities to every node.
(2) Communication interface units exchange neighboring cells between chips, boards and nodes.
(3) DEM pipelines calculate \( \rho \) and \( \phi \)

<table>
<thead>
<tr>
<th>Number of Interactions</th>
<th>Ni</th>
<th>2</th>
<th>2</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Particles in a Cell</td>
<td>Npc</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>Number of Neighboring Cells</td>
<td>Nnc</td>
<td>3(^2)</td>
<td>27</td>
<td>27</td>
</tr>
<tr>
<td>Number of Neighboring Particles</td>
<td>Nnp</td>
<td>108</td>
<td>108</td>
<td>216</td>
</tr>
<tr>
<td>Number of Interacting Particles</td>
<td>Nip</td>
<td>16.8</td>
<td>16.8</td>
<td>33.5</td>
</tr>
<tr>
<td>Bytes per Particle</td>
<td>Nhpp</td>
<td>20</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td>Object Cell Size</td>
<td>Lcell</td>
<td>1</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>Object Particles *</td>
<td>Npc</td>
<td>Lcell(^2)*Npc</td>
<td>4</td>
<td>64</td>
</tr>
<tr>
<td>Clocks per Particle</td>
<td>Nck</td>
<td>Npc<em>Nnc</em>Nhpq</td>
<td>8.4</td>
<td>8.4</td>
</tr>
<tr>
<td>Clocks to Calculate Object Cells</td>
<td>*</td>
<td>Ncell</td>
<td>Lcell(^2)*2</td>
<td>9</td>
</tr>
<tr>
<td>Particles Required *</td>
<td>Ncell</td>
<td>Ncell*Npc</td>
<td>36</td>
<td>144</td>
</tr>
<tr>
<td>Bytes Required *</td>
<td>Nhpc</td>
<td>Ncell*Nhpq</td>
<td>720</td>
<td>2880</td>
</tr>
<tr>
<td>Size of a DPM</td>
<td>Nhbytes</td>
<td>16384</td>
<td>16384</td>
<td>16384</td>
</tr>
<tr>
<td>Number of XY Planes in DPM</td>
<td>Nhpl</td>
<td>Nhbytes*Nhpq</td>
<td>22.8</td>
<td>8.7</td>
</tr>
</tbody>
</table>

All values with (*) are listed per xy plane.
(4) Embedded processors calculate $F'(\rho)$.
(5) Communication interface units exchange $F'(\rho)$ of neighboring cells between chips, boards and nodes.
(6) DEM pipelines calculate $\rho'$ and $\phi'$.
(7) Embedded processors update coordinates, velocities etc.
(8) Communication interface units exchange moving particles between chips, boards and nodes.
(9) Back to step (2) and repeats until simulation completes. Snap shots may be sent via RHiNET for logging in with certain intervals (may not every step).

### 4.2 Performance estimation

The DEM-1 is capable of solving 512M particle system EAM simulations. Each element, ranging form DEM-pipeline to DEM-node, calculates its spatial region in less than 1sec and the high-speed links can exchange surface particles and coefficients within that computation time.

The performance of DEM-1 depends on the performance of its hardware pipeline. As described in Figure 4, the pipeline executes 4 cut off judgement operations per main clock cycle time, while 0.5 particle pair is fed through the main pipeline, which calculates 2 interactions on the same particle pair within 2 clock cycles ("multiple simultaneous interactions " mode). Therefore, total effective operation of the DEM pipeline is represented by:

$$N_{op} = 4 \times R_{eff\text{ju}} \times N_{\text{cju}} + N_{\text{main}}$$

where $N_{op}$ is the total operations of the pipeline per clock, $R_{eff\text{ju}}$ is efficiency of cut off judgement unit, $N_{\text{cju}}$ is number of operations of cut off judgement unit, $N_{\text{main}}$ is number of operations of the main pipeline, respectively. Efficiency of the cut off judgement unit is limited by the ratio $R_{eff}$ (see Appendix A):

$$4 \times R_{eff\text{ju}} \leq 0.5 \times R_{eff}$$

This leads to $R_{eff\text{ju}} = 0.81$, which denotes that cut off judgement units should idle sometimes. If we assume $N_{\text{cju}} = 9$ and $N_{\text{main}} = 21$ (we omit operations of squared distance calculation in $N_{\text{main}}$ because it has been already counted in $N_{\text{ju}}$), the overall performance is estimated to be $50(= 4 \times 0.81 \times 9 + 21)$. We should count the non-vectorized nature of the hardware pipeline (see also Appendix A). The effective performance of the pipeline is 25 operations per clock. This corresponds to 3.33Gflops with 133MHz clock. The MIPS processor attached to the pipeline is 266Mflops. We expect approximately 14.4G flops performance with four pipelines and four processors.

Estimated pipeline performance is shown in Table 3. A DEM chip has 4 dedicated pipelines to calculate interactions, which run in 133MHz system clock. By filtering out particle pairs laying outside of the cut off sphere with cut off judgement units, the main pipeline works with it maximum efficiency, so that it takes only 67 clocks to calculate 4 interactions required in EAM simulations. We estimate 33msec execution time for 1M particle interaction calculations by a DEM-pipeline.

### Table 3. Estimated pipeline performance

<table>
<thead>
<tr>
<th></th>
<th>symbols</th>
<th>equations</th>
<th>numbers</th>
</tr>
</thead>
<tbody>
<tr>
<td>number of interactions</td>
<td>$N_i$</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>number of particles in a cell</td>
<td>$N_{pc}$</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>number of pipelines</td>
<td>$N_{pipe}$</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>cells per chip</td>
<td>$N_c$</td>
<td>$64^3$</td>
<td>262144</td>
</tr>
<tr>
<td>neighboring cells</td>
<td>$N_n$</td>
<td>27</td>
<td></td>
</tr>
<tr>
<td>number of neighboring particles</td>
<td>$N_{np}$</td>
<td>$N_n^2 \times N_{pc}$</td>
<td>108</td>
</tr>
<tr>
<td>number of interacting particles</td>
<td>$N_{ip}$</td>
<td>$N_{pc}/6.44$</td>
<td>16.8</td>
</tr>
<tr>
<td>clocks per particle</td>
<td>$N_{clk}$</td>
<td>$N_{pc} \times N_{pc}$</td>
<td>67.1</td>
</tr>
<tr>
<td>total clocks</td>
<td>$N_{tc}$</td>
<td>$N_c \times N_{clk} / N_{pipe}$</td>
<td>4396204</td>
</tr>
<tr>
<td>total time (msec)</td>
<td>$T_{t}$</td>
<td>$N_{tc} / 133M \times 1000$</td>
<td>33.1</td>
</tr>
</tbody>
</table>

### Table 4. Estimated communication time

<table>
<thead>
<tr>
<th></th>
<th>system</th>
<th>node</th>
<th>board</th>
<th>chip</th>
</tr>
</thead>
<tbody>
<tr>
<td>particles</td>
<td>512M</td>
<td>64M</td>
<td>8M</td>
<td>1M</td>
</tr>
<tr>
<td>cells</td>
<td>$512^3$</td>
<td>$256^3$</td>
<td>$128^3$</td>
<td>$64^3$</td>
</tr>
<tr>
<td>surface cells</td>
<td>1.5M</td>
<td>384K</td>
<td>96K</td>
<td>24K</td>
</tr>
<tr>
<td>surface particles</td>
<td>6M</td>
<td>1.5M</td>
<td>384K</td>
<td>96K</td>
</tr>
<tr>
<td>transfer bytes</td>
<td>N.A.</td>
<td>120M</td>
<td>30.7M</td>
<td>7.7M</td>
</tr>
<tr>
<td>link speed (bytes/sec)</td>
<td>N.A.</td>
<td>3G</td>
<td>800M</td>
<td>400M</td>
</tr>
<tr>
<td>required time (msec)</td>
<td>N.A.</td>
<td>40</td>
<td>38.4</td>
<td>19.2</td>
</tr>
</tbody>
</table>

4.2 Performance estimation

MD GRAPE[2] is developed at University of Tokyo along with its commercial version MD One[3]. Both of
them have linked-cell-list addressing support hardware, and neighbor list FIFO that stores indices of particles falling inside the cut off sphere. A single host processor controls multiple boards with multiple pipelines, which share common data path within the board.

MD Engine[4], developed at Fuji Xerox corp., has similar single host processor system architecture but has a special purpose circuit for utilizing the neighbor list. It reduces the calculation cost by 50% under certain conditions.

MDM[5] is a very large-scale and high speed version of MD GRAPE with dedicated wave number space part accelerator WINE-2. MDM has 4 nodes of 6-cpu parallel host computers running MPI. However, its basic architecture is not much different from that of MD GRAPE. Therefore, most of the deficiencies discussed in earlier sections of this paper exist and reduces the performance when it executes particle simulations with short-range interactions.

Both MDM[5] and MD GRAPE[2] have identical memory architecture in which pipelines share the same memory data fed to the pipelines. They also have virtual multiple pipelines at chip level to reduce external bus bandwidth without sacrificing pipeline performance in simulations without cut off. However, if we execute simulations with short-range interactions, their performance is limited by pipeline data sharing. For example, let us consider running an EAM simulation of iron on MD-GRAPE. As we have seen before, typically only 4 particles are contained in a cell. In MD-GRAPE system, 4 chips with 6 virtual multiple pipelines (total 24 pipelines) share a single memory data path. That means, up to 6 cells may be calculated simultaneously. The common pipeline bus feeds particles that fall within union of neighboring cells, which contain approximately 8/3 particles compared to calculations without this overhead. This is described in figure 8. DEM-1’s dedicated data path to each pipeline, supported by position vector pre-fetch dual ported memory, eliminates this overhead.

Together with 6.44 times speed up introduced by cut off judgement units, we can conclude that DEM-1’s pipeline has more than ten times performance advantages over existing pipeline architectures in simulations with short-range interactions.

The features of particle simulation machines are summarized and compared in Table 5 along with DEM-1.

### 6. Conclusion

This paper introduced the design of particle simulation machine (DEM-1) and its performance. DEM-1 can be considered as a collection of computational elements specially designed for particle simulations. DEM chip has 4 on-chip multi processors and 4 hardware pipelines that can handle short-range interactions effectively. DEM-1 has 2048 processors. These processors, located at the bottom of the system hierarchy, are connected together to create 3 dimensional torus network macro structure. They work in parallel with pipelines and execute time step integration locally. The processors conceal the existence of specialized hardware at the lowest level. Therefore, DEM-1 can be treated as a network of high performance processors. Thus, results of recent research on sophisticated load balancing can be applied to DEM-1. It should be noted that, even though DEM-1 is designed to handle simulations with very short-range interaction, it executes any particle simulations with cut off radius efficiently.
We are now designing DEM pipeline prototype on large-scale FPGA (Xilinx XCV2000E). The pipeline will run on 33MHz system clock and expected to deliver 700-800Mflops performances. We are going to determine optimum cut off judgement accuracy, cut off FIFO depth and pre-fetch dual ported memory size using the test bed.

Appendix A: Pipeline performance of existing particle simulation machines.

Linked-cell-list method selects particles in 27 neighboring cells (Figure 1). Particles inside the cut off sphere affect the ith particle that we are focusing on. Usually the cell size is equal to the cut off radius, rcut. One can assume that the number of particles in each area is proportional to its volume, so that the ratio of particles within the 27 adjacent cells and within the cut off sphere can be represented as:

\[
R_{eff} = 27 \times r_c^3 / \left(4\pi r_c^3 / 3\right) \cong 6.44
\]

(A1)

In addition to that, hardwired pipeline cannot take advantage of vectorized linked cell list method, in which \(f_{ji}\) is calculated as \(-f_{ij}\) (Newton's third law) to reduce calculation cost by 1/2. Together, effective performance of the pipeline of existing particle simulation machine is estimated to be 1/13 of its peak performance in the simulations with cut off. Smaller cell size may be used to reduce the ratio, but it will introduce more communication cost and more complexity like empty cells.

Acknowledgements

We thank Dr. Jayantha Herath for his valuable comments on drafts of this paper. We thank Hajime Kondo for his help in designing DEM-1 pipeline.

References