Reconfigurable Molecular Dynamics Simulator

Navid Azizi, Ian Kuon, Aaron Egier, Ahmad Darabiha and Paul Chow
University of Toronto
Why is Molecular Dynamics interesting?

- Simulates interaction of atoms over time
- Many possible applications
  - Biomolecules
- Computationally intensive to handle >1000 atoms
- Large computer clusters used in the past
- Can a reconfigurable simulation system do this better?
What is Molecular Dynamics?

- Simulate using classical Newtonian mechanics
  \[ F = m \, a \]

- Integrate acceleration to get position and velocity changes

- Use a very small timestep \(~ 1\) femtosecond
Molecular Dynamics Background

- Simulation Procedure per timestep
  - Sum force over all interacting atoms
  - Calculate acceleration
  - Integrate acceleration to update atom position and velocity
  - Repeat for all atoms
Background - Forces

- Two types of forces
  - Bonded – O(n)
    - No hardware acceleration required
  - Non Bonded – O(n^2)
    - Needs hardware acceleration
Background – Force Calculation

- Lennard-Jones (LJ) potential models interaction

\[
\phi_{LJ}(r) = 4\varepsilon \left[ \left( \frac{\sigma}{r} \right)^{12} - \left( \frac{\sigma}{r} \right)^6 \right]
\]

- Force on a atom is the gradient of potential
**Background – Simulating Large Volumes**

- Any interesting volume has far too many atoms to simulate.

- Solution – Periodic Boundary Conditions

<table>
<thead>
<tr>
<th>Replicated Box</th>
<th>Replicated Box</th>
<th>Replicated Box</th>
</tr>
</thead>
<tbody>
<tr>
<td>A’</td>
<td>B’</td>
<td></td>
</tr>
<tr>
<td>A’</td>
<td>B’</td>
<td></td>
</tr>
<tr>
<td>A’</td>
<td>B’</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Replicated Box</th>
<th>Box being simulated</th>
<th>Replicated Box</th>
</tr>
</thead>
<tbody>
<tr>
<td>A’</td>
<td>B’</td>
<td>A’</td>
</tr>
<tr>
<td>A’</td>
<td>B’</td>
<td>A’</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Replicated Box</th>
<th>Replicated Box</th>
<th>Replicated Box</th>
</tr>
</thead>
<tbody>
<tr>
<td>A’</td>
<td>B’</td>
<td>A’</td>
</tr>
<tr>
<td>A’</td>
<td>B’</td>
<td>A’</td>
</tr>
<tr>
<td>A’</td>
<td>B’</td>
<td>A’</td>
</tr>
</tbody>
</table>
Architectural Design – System Overview

- PairGen
- Force Computer
- Sun Workstation
- Particle Memory
- Acceleration Update
- Verlet Update
- Slope Memory
- Function Value Memory
- System Control

Reconfigurable Molecular Dynamics Simulator
Architectural Design – Force Computer

- $r^2$ from PG used for function lookup

- Interpolate to obtain a more accurate force magnitude
**Architectural Design – Force Computer**

- $r^2$ is larger than 18-bits
  - Look up table has a 18-bit address

---

**Graph:**

- **Y-axis:** Pseudo-Acceleration
- **X-axis:** Separation$^2$
- **Data Points:**
  - $0.00E+00$ at $5E-19$
  - $1.00E+09$ at $1E-18$
  - $2.00E+09$ at $1.5E-18$
  - $3.00E+09$ at $2E-18$

---

Reconfigurable Molecular Dynamics Simulator
Architectural Design – Force Computer

Address to lookup tables

Residual for interpolation

r^2

high bits

middle bits

low bits

FF...FF
Precision and Scaling Factors

- Architecture uses integer operations to reduce complexity

- **Precision**: number of bits used to represent a value
- **Scaling Factor**: the weight of the least significant bit of the value
Calculating the Precision and Scaling Factors

- Calculations made with atoms at varying distances

- Scaling Factor = the minimum value
- Precision = \( \log_2 \) of the difference between minimum and maximum

<table>
<thead>
<tr>
<th>Quantity</th>
<th>Scaling Factor</th>
<th>Precision</th>
</tr>
</thead>
<tbody>
<tr>
<td>Position</td>
<td>( 2^{-64} )</td>
<td>38</td>
</tr>
<tr>
<td>Velocity</td>
<td>( 2^{-15} )</td>
<td>51</td>
</tr>
<tr>
<td>Acceleration</td>
<td>( 2^{-64} )</td>
<td>37</td>
</tr>
</tbody>
</table>
Simulation Environment is Configurable

- Simulation reconfigurability
  - Change precision, scaling factors, number of atoms. forces
  - No wasted hardware
  - No time overhead when precision is reduced

- Entire process is automated
  - One input file controls entire process
    - Hardware
      - C program creates appropriate VHDL
      - Software interface, Software initialization
    - Always match the hardware
Implementation

- Used the Transmogrifier 3
  - 4 interconnected Virtex-E 2000’s
  - 2MB memories connected to each Virtex-E
  - Slow by today’s standards
Verification

- Tested accuracy of implementation
  - Compared TM3 results with software

![Graph showing energy over timestep]

- TM3 -Total Energy
- Software - Total Energy
- TM3 - Kinetic Energy
- Software - Kinetic Energy
- TM3 - Potential Energy
- Software - Potential Energy
System Performance

- For a 8192 atom MD system running on the TM3
  - Frequency: 26 MHz
  - Timestep Duration: 37 sec

- For a 8192 atom software system running on a 2.4GHz Pentium 4
  - Timestep Duration: 10.8 sec

- MD system is 3.4X slower than software
How to improve this?

- Memory
- New FPGA
- Parallelism
Memory Requirements of MD System

- Acceleration Array (8192 atom system uses 0.17 MB)
- Velocity Array (8192 atom system uses 0.17 MB)
- Position Array (8192 atom system uses 0.34 MB)
- Lookup Tables: 2 MB
Improving Performance – Memory Organization

- On TM3 there is only one external SRAM per FPGA
- Single SRAM for all atom information causes large slowdowns
  - Handshaking
  - Serial reads for x, y and z
  - Hardware issues

Better memory system

2.1 seconds/timestep (5X faster than software)
Improving Performance – Clock Speed

- Run on modern FPGA
- All possible improvements for clock speed not explored
- Expect a factor of 4 increase to a 100MHz

Better memory system + Faster Clock Speed

0.51 seconds/timestep (21X faster than software)
Improving Performance – Parallel Architecture

Better memory system + Faster Clock Speed + Parallelize

0.51/n seconds/timestep (21n X faster than software)
Cost, Power Benefits

- **Performance**
  - MD system can deliver a 21X performance benefit over software
  - Assume a conservative 10X performance advantage

- **Cost**
  - Microprocessor motherboard-sized board can fit 4 FPGAs
  - 4 FPGAs (each $200) + board + SRAM + misc. ~ $1500
  - Microprocessor Motherboard + CPU + DRAM ~ $1500
Comparison of MD Simulator and Supercomputers

<table>
<thead>
<tr>
<th>Per Board</th>
<th>1 Pentium + 1 GB DRAM</th>
<th>4 FPGAs + 24 MB SRAM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Performance</td>
<td>1X</td>
<td>40X</td>
</tr>
<tr>
<td>Cost</td>
<td>~$1500</td>
<td>~$1500</td>
</tr>
<tr>
<td>Power</td>
<td>106W</td>
<td>40W</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Metric</th>
<th>Improvement of MD System</th>
</tr>
</thead>
<tbody>
<tr>
<td>Performance/Power</td>
<td>100X Improvement</td>
</tr>
<tr>
<td>Performance/Cost</td>
<td>40X Improvement</td>
</tr>
<tr>
<td>Performance/Space</td>
<td>40X Improvement</td>
</tr>
</tbody>
</table>
Conclusions

- Easily reconfigurable MD System designed
- Molecular dynamics simulation can be done on FPGAs
- Simple enhancements will improve speed
  - Power, Cost and Space savings over software
Future Work

- Improve accuracy
- Target newer FPGA platform
- Support new forces

Acknowledgements

- Funding for the TM3 Project was provided by Micronet and Xilinx