Parallelization and Performance of 3D Ultrasound Imaging Beamforming Algorithms on Modern Clusters

Fan Zhang, Angelos Bilas, Kostas N. Plataniotis
Dept. of Electrical and Computer Engineering, University of Toronto

Amar Dhanantwari, Robert Abiprojo, Stergios Stergiopoulos
Defence R&D Canada, Toronto
Medical Imaging

• Medical imaging is important for improving health care
  – Helps timely and accurate diagnosis

• Various methods for diagnostic imaging
  – Diagnostic X-rays, Nuclear medicine, Computed Tomography (CT), Magnetic Resonance Imaging (MRI), Ultrasound

• Ultrasound has many advantages over other methods
  – Wide applicability: Tissue, internal organs, blood, etc.
  – Safe: No harmful radiation, no drugs, dyes, or chemicals
  – Convenient: Real time processing, portable
3D Ultrasound System

• Today there are 2D and 3D ultrasound systems
• 3D has advantages over 2D
  – Full 3D representation of the object
  – Accurate focus on the location of interest
  – Interactive viewing
• Rapid growth of 3D ultrasound imaging market
  – In 1999, ultrasound was 1/3 of the 3D medical imaging market
  – In 2006, it will have ½ of market ($1 billion)
Structure of 3D Ultrasound Systems

3D Object to be scanned

Ultrasound Probing

Input Time Series

Processing

Reconstructed 3D Volume

Sub-sector of the area of interest

array of sensors

Back-end computing architecture

Display
Examples of Current 2D & 3D Ultrasound

Real Picture

2D Image

3D Image
Current 3D Ultrasound Systems

• Problem
  – Poor image resolution
  – Slow probe scanning

• Solution
  – Better technology allows use of larger sensor arrays
  – Improved probing techniques
  – Advanced signal processing algorithms

• Increased requirement for back-end processing with large I/O throughput and CPU processing
Previous and Current Solutions

• Specialized processing hardware
  – High design costs
  – Long design cycles
  – Not much flexibility in modifying system parameters

• Recently commodity processor and interconnect technologies have improved dramatically
Our Goal

- Evaluate if a cluster of commodity components can serve as a back-end for 3D ultrasound processing
  - Low design cycle, cost, and increased flexibility

- Evaluate the characteristics of ultrasound processing on such architectures and the effects of various system parameters on compute time
Outline

• Motivation & Goal ✓
• Ultrasound processing algorithm
• Parallel implementation
• Results
• Conclusions
3D Ultrasound Processing Algorithm
Parallel Algorithm

- Decompose input data to each processor by each row of sensors
  - Maximize parallelization and minimize communication between processors
  - Can be decomposed by each element of the sensors, however increases data traffic among processors
- We use MPI as the communication abstraction
Implementation Platform

<table>
<thead>
<tr>
<th>Processors</th>
<th>Intel Pentium III 800 MHz</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cache</td>
<td>32K (L1), 512K (L2)</td>
</tr>
<tr>
<td>Memory</td>
<td>512MB SDRAM</td>
</tr>
<tr>
<td>OS</td>
<td>Redhat Kernel 2.2.16-3</td>
</tr>
<tr>
<td>PCI buses</td>
<td>32 bits, 33 MHz</td>
</tr>
<tr>
<td>NIC</td>
<td>Myricom M3M-PCI64B</td>
</tr>
<tr>
<td></td>
<td>3Com 100M Ethernet</td>
</tr>
<tr>
<td>Communication Library</td>
<td>MPICH-Score 4.0</td>
</tr>
</tbody>
</table>
System Parameters

- Isolate the most important system parameters
- Examine their impact on system performance
- Verify that each parameter is independent
- Define a base case parameter set
- Vary each parameter individually by keeping all other parameters constant

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Values Examined</th>
</tr>
</thead>
<tbody>
<tr>
<td>No. of sensors</td>
<td>32X32, 24X24, 16X16, 8X8</td>
</tr>
<tr>
<td>FFT Size (samples)</td>
<td>512, 1024, 2048, 4096</td>
</tr>
<tr>
<td>Filter bandwidth (MHz)</td>
<td>0.5, 1.0, 1.5, <strong>2.0</strong>, 2.6, 3.0, 4.0</td>
</tr>
<tr>
<td>Focal zone size (cm)</td>
<td>0.5, <strong>1.0</strong></td>
</tr>
<tr>
<td>No. of beams per sub-sector</td>
<td>16X16, <strong>8X8</strong>, 4X4</td>
</tr>
</tbody>
</table>
Input Data

- The sensor array and acquisition cards are currently being built
- We generate inputs with Field-II ultrasound simulator
- Equivalent to data delivered to each node’s memory by acquisition card
Outline

• Motivation & Goal ✓
• Ultrasound processing algorithm ✓
• Parallel implementation ✓
• Results
• Conclusions
Average Execution Time Breakdown

- Low communication overheads
  - Comm.: 0.22% - 11%
- FFT and steering take most of the time
  - FFT: 20% – 84% of the total time
  - Csteer: 4% – 47% of the total time
  - IFFT: 2% – 47% of the total time
Execution Time vs Number of Processors

- Execution time scales linearly with the number of processors
- Important parameters for performance
  - FFT size
  - Number of sensors
Absolute Performance

• 16-processor system does not provide real-time performance
  – 2 frames/s for the base case
  – 5.5 frames/s for the fastest case

• Advantage of using commodity nodes and interconnects
  – Use new technology with minimal effort

• 8-processor system with 2GHz P4 processors
  – 2 - 3.5 x speedup
  – Average overall speedups: 2.3
  – FFT & IFFT speedups: 1.5 - 1.7
  – Steering speedups: 3.8 - 4.0
  – 9 frames/s for the fastest case (expect 18 frames/s for 16-processor system with new P4 processor)
Performance vs. Image Quality

• Variation of the parameter values affect not only the performance but also the quality of images

• Best performance with “acceptable” quality

• Image quality is subjective and more difficult to quantified between different images

• We use some standard correlations to compare image quality

• Verify human perception correspond to the correlation coefficient
Image Quality

![Graph showing the relationship between correlation coefficient and number of sensors, number of beams, FFT size, focal size, and bandwidth.]

- **No. of Sensors**: 32x32, 24x24, 16x16, 8x8
- **No. Of Beams**: 16, 8, 4
- **FFT Size**: 4096, 2048, 1024, 512
- **Focal Size**: 05, 10
- **Bandwidth**: 4.0, 3.0, 2.6, 2.0, 1.5, 1.0, 0.5
Conclusions

• We propose the use of commodity architectures for back-end processing in Ultrasound systems

• We evaluate the computational behavior of ultrasound algorithms on such architectures
  – FFT & beam steering take most of the processing time (85% ~ 92%)
  – Parallel performance scales with number of processors
  – Close to real-time performance today

• We expect that commodity architectures will play an important role in medical imaging applications
  – Ability to take advantage of latest components with minimum effort
  – Flexibility in modifying algorithmic parameters
Future Work

• Portable ultrasound system
  – Miniaturization of the computing architecture
    • Our computing architecture bulky
  – Various possibilities
    • Reduce size of cluster
    • Explore alternative computing architectures with mixed DSP/special/general purpose components

• Explore ultrasound algorithm design space
  – Algorithm we use was developed independently of architecture
  – Given that this architecture makes sense, how can one best design the ultrasound algorithms?

• Deploy prototype
  – Expected within 2003