S7391—Turbocharging VMD Molecular Visualizations with State-of-the-Art Rendering and VR Technologies

John E. Stone
Theoretical and Computational Biophysics Group
Beckman Institute for Advanced Science and Technology
University of Illinois at Urbana-Champaign

http://www.ks.uiuc.edu/Research/gpu/
S7391, GPU Technology Conference
14:00-14:50, Room 212B, San Jose Convention Center,
San Jose, CA, Tuesday May 9th, 2017
VMD – “Visual Molecular Dynamics”

- Visualization and analysis of:
  - Molecular dynamics simulations
  - Lattice cell simulations
  - Quantum chemistry calculations
  - Cryo-EM densities, volumetric data
  - Sequence information
- User extensible scripting and plugins
- http://www.ks.uiuc.edu/Research/vmd/
10 Years of GPU Computing in VMD

- Has stood the test of time
- Modeling, Visualization, Rendering, and Analysis

Blast from the past:
CUDA starting with version 0.7 !!!
Quad core Intel QX6700, three NVIDIA GeForce 8800GTX GPUs, RHEL4 Linux

Goal: A Computational Microscope
Study the molecular machines in living cells

Ribosome: target for antibiotics

Poliovirus
VMD Petascale Visualization and Analysis

• Analyze/visualize large trajectories too large to transfer off-site:
  – User-defined parallel analysis operations, data types
  – Parallel rendering, movie making

• Supports GPU-accelerated Cray XK7 nodes for both visualization and analysis:
  – GPU accelerated trajectory analysis w/ CUDA
  – OpenGL and GPU ray tracing for visualization and movie rendering

• Parallel I/O rates up to 275 GB/sec on 8192 Cray XE6 nodes – can read in 231 TB in 15 minutes!

Parallel VMD currently available on:

ORNL Titan, NCSA Blue Waters, Indiana Big Red II, CSCS Piz Daint, and similar systems

NCSA Blue Waters Hybrid Cray XE6 / XK7
22,640 XE6 dual-Opteron CPU nodes
4,224 XK7 nodes w/ Telsa K20X GPUs
Interactive Remote Visualization and Analysis

• Enabled by hardware H.264/H.265 video encode/decode
• Enable visualization and analyses not possible with conventional workstations
• Access data located anywhere in the world
  – Same VMD session available to any device
• Linux prototype in-development using NVIDIA Video Codec SDK, easy-to-use NvPipe wrapper library
NVIDIA Video CODEC SDK and NvPipe

- GPUs (Kepler-on) include NVENC and NVDEC video codec acceleration hardware
- Independent of GPU compute hardware
- Hardware-accelerated codecs can overlap with interactive rendering, and computation
- NvPipe provides an easy to use API for interactive video streaming, abstracting many low level codec details, ideal for basic remote visualization implementations: https://github.com/NVIDIA/NvPipe
NvPipe
https://github.com/NVIDIA/NvPipe

- Simplified API for producing a basic encoder/decoder system.
- **Roughly 100 lines of code for basic encode/decode “Hello World” loops with minimal error handling logic**
- Encode/decode ends up being simpler than your networking code 😊
- Encode loop structure:
  - User selects encoder type, e.g. NVPIPE_H264_NV, and target encoder bitrate parameter
  - User provides uncompressed RGB or RGBA image buffer, image dimensions, and size of the output memory buffer
  - NvPipe compresses the frame using the NVENC hardware encoder, and returns the number of bytes of output written to the output buffer
- Symmetric decode loop structure:
  - Provide decoder with compressed buffer, buffer size in bytes, and image dimensions as input
  - Decoder produces uncompressed output image
- Optionally supports FFMPEG back-ends (but I haven’t tried those yet)
**In-progress: Vulkan-based rasterization path for VMD**

- Modern API, reduced dependence on extensions for modern functionality
- Significantly reduced API overheads relative to OpenGL, *some other apps have seen ~2x performance gains vs. OpenGL*
- Shaders (e.g. GLSL) compiled to SPIR-V intermediate code
- Compile-time rather than runtime verification of rendering pipelines
- Integration with windowing system is handled by Vulkan extensions
- Multi-GPU rendering wasn’t part of Vulkan 1.0 spec, but is in development

https://www.khronos.org/vulkan/
VMD 1.9.3 supports EGL for in-situ and parallel rendering on clouds, clusters, and supercomputers

- Eliminate dependency on windowing systems
- Simplified deployment of parallel VMD builds supporting off-screen rendering
- Maintains 100% of VMD OpenGL shaders and rendering features
- Support high-quality vendor-supported commercial OpenGL implementations in HPC systems that were previously limited to Mesa

Poliovirus
OpenGL: GLX vs. EGL

Viz Application
(user)

OpenGL

X server
(root)

Driver

GPU

Viz Application
(user)

OpenGL

EGL

Driver

GPU
Swine Flu A/H1N1 neuraminidase bound to Tamiflu: VMD EGL rendering demonstrating full support for all VMD shaders and OpenGL features, multisample antialiasing, ray cast spheres, 3-D texture mapping, ...
Benefits of EGL Platform Interfaces

- Enumerate and select among available platforms, potentially supporting multiple vendors in the same host/node
  - Allows specific target implementation to be bound, e.g. GPU, CPU-integrated GPU, software rasterizer

- **EGL interfaces make it EASY to bind a GPU to a thread with optimal CPU affinity with respect to NUMA topology, NVLink GPU topology**
  - High-perf. multi-GPU image compositing, video streaming
  - EGL plays nicely with MPI, CUDA/OpenCL, OptiX, NVENC, etc
  - NVIDIA EGL supports multiple GPU indexing schemes, e.g. PCIe ordering
  - Exploit NVLink interconnect topology on IBM OpenPOWER platforms, e.g. “Minsky”, upcoming DOE/ORNL “Summit” system
Example Node NUMA Topology

- CPU 1
  - CPU Bus 25GB/sec
  - QuickPath (QPI)
  - HyperTransport (HT)

- CPU 2

- DRAM

- IOH 1
  - PCIe 3.0 x16
  - PCIe 3.0 x16

- IOH 2
  - PCIe 3.0 x4/x8/x16

- GPU 1
  - PCIe 3.0 x16

- GPU 2
  - PCIe 3.0 x16

- GPU 3
  - PCIe 3.0 x16

- GPU 4
  - PCIe 3.0 x16

- DRAM

- CPU Bus 25GB/sec

- QuickPath (QPI)

- HyperTransport (HT)

- PCIe 3.0 x16

- PCIe 3.0 x16

- PCIe 3.0 x16

- PCIe 3.0 x16

NIH BTRC for Macromolecular Modeling and Bioinformatics
http://www.ks.uiuc.edu/
VMD EGL Performance on Amazon EC2 Cloud

<table>
<thead>
<tr>
<th>MPI Ranks</th>
<th>EC2 “G2.8xlarge” GPU Instances</th>
<th>HIV-1 movie rendering time (sec), (I/O %) 3840x2160 resolution</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>626s (10% I/O)</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>347s (19% I/O)</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>221s (31% I/O)</td>
</tr>
<tr>
<td>8</td>
<td>2</td>
<td>141s (46% I/O)</td>
</tr>
<tr>
<td>16</td>
<td>4</td>
<td>107s (64% I/O)</td>
</tr>
<tr>
<td>32</td>
<td>8</td>
<td>90s (76% I/O)</td>
</tr>
</tbody>
</table>

Performance at 32 nodes reaches ~48 frames per second

64M atom HIV-1 capsid simulation rendered via EGL

Close-up view of HIV-1 hexamer rendered via EGL
EGL Is Supported Now!

- Cloud+Workstations with most recent NVIDIA drivers
- VMD on HPC systems w/ latest Tesla P100 GPUs:
  - Cray XC50, CSCS Piz Daint, driver 375.39
  - IBM OpenPOWER, drivers 375.66 and later support both GLX and EGL
Info) VMD for OPENPOWER, version 1.9.3 (April 27, 2017)
Info) http://www.ks.uiuc.edu/Research/vmd/

[...]

Info) Multithreading available, 160 CPUs detected.
Info) Free system memory: 501GB (98%)
Info) Creating CUDA device pool and initializing hardware...
Info) Detected 4 available CUDA accelerators:
Info) [0] Tesla P100-SXM2-16GB 56 SM_6.0 @ 1.48 GHz, 16GB RAM, AE3, ZCP

[...]

Info) EGL: node[0] bound to display[0], 4 displays total
Info) EGL version 1.4
Info) OpenGL Pbuffer size: 4096x2400
Info) OpenGL renderer: Tesla P100-SXM2-16GB/PCIe
Info) Features: STENCIL MSAA(4) MDE CVA MTX NPOT PP PS GLSL(OVFGS)
Info) Full GLSL rendering mode is available.
Info) Textures: 2-D (32768x32768), 3-D (16384x16384x16384), Multitexture (4)
Info) Created EGL OpenGL Pbuffer for off-screen rendering
VMD “QuickSurf” Representation, Ray Tracing

All-atom HIV capsid simulations w/ up to 64M atoms on Blue Waters
Lighting Comparison, STMV Capsid

Two lights, no shadows

Two lights, hard shadows, 1 shadow ray per light

Ambient occlusion + two lights, 144 AO rays/hit
VMD w/ OptiX 4.1

- Interactive RT on laptops, desktops, and cloud
- Large-scale parallel rendering: in situ or post hoc visualization tasks
- Remote RT on NVIDIA VCA clusters
- Stereoscopic panoramic and full-dome projections
- Omnidirectional VR for YouTube, VR HMDs
- **Top-end Pascal Tesla GPUs roughly 2x faster than Kepler**
- GPU memory sharing via NVLink on Quadro GP100, Tesla P100


VMD/OptiX GPU Ray Tracing of all-atom Chromatophore w/ lipids.
**VMD TachyonL-OptiX Interactive RT w/ OptiX Progressive RT API**

- Scene Graph
  - Scene Graph
  - TrBvh
  - RT Acceleration Structure

- RT Progressive Subframe
  - rtContextLaunchProgressive2D()
  - Check for User Interface Inputs, Update OptiX Variables
  - rtBufferGetProgressiveUpdateReady()
  - rtContextStopProgressive()

- Draw Output
  - Framebuffer

Biomedical Technology Research Center for Macromolecular Modeling and Bioinformatics
Beckman Institute, University of Illinois at Urbana-Champaign - www.ks.uiuc.edu
Preparation, Visualization, Analysis of All-Atom Cell-Scale Simulations

- Interactive rasterization w/ OpenGL/EGL now, Vulkan in future releases of VMD
- Interactive ray tracing on CPUs and GPUs
- Support for large host memory (TB), up to 2 billion atoms per “molecule” now
- Parallel analysis, visualization w/ MPI


Proto-Cell Rendered with VMD+OptiX

- 113M particles
- 1,397 copies of 14 different membrane proteins
- Preparing for simulations on pre-exascale computers
Interactive Ray Tracing of Cells

- High resolution cellular tomograms, **billions of voxels**
- Even isosurface or lattice site graphical representations involve ~100M geometric primitives
- 24GB Quadro M6000s used for interactive RT of cellular tomograms of this size
- **Latest Quadro GP100 GPUs** benefit from OptiX 4.1 support for NVLink and distribution of scene data across multiple GPUs

VMD Molecule Instancing: In-Progress Development

• VMD 1.9.4 supports instancing of graphical representations associated with molecules
• Will exploit VBO caching in OpenGL to eliminate host-GPU geometry transfers
• OptiX instancing of geometry buffers to eliminate GPU memory consumption for instances
In-Progress VMD VR Development, Demos

VMD VR ray tracing:
Google Cardboard [1]
Demo w/ Indiana U., SC’15 [2]

Prototype of VR user interaction with VMD models in room-scale VR with NVIDIA @ SC’16


Goal: Intuitive interactive viz. in crowded molecular complexes

Results from 64 M atom, 1 μs sim!
Immersive Viz. w/ VMD

- VMD began as a CAVE app (1993)
- Use of immersive viz by molecular scientists limited due to cost, complexity, lack of local availability, convenience
- Commoditization of HMDs excellent opportunity to overcome cost/availability
- This leaves many challenges still to solve:
  - Incorporate support for remote visualization
  - UIs, multi-user collaboration/interaction
  - Rendering perf for large molecular systems
  - Accomodating limitations idiosyncracies of commercial HMDs

VMD running in a CAVE w/ VR Juggler
HMD Ray Tracing Challenges

- HMDs require high frame rates (90Hz or more) and minimum latency between IMU sensor reads and presentation on the display.
- Multi-GPU workstations fast enough to direct-drive HMDs at required frame rates for simple scenes with direct lighting, hard shadows.
- Advanced RT effects such as AO lighting, depth of field require much larger sample counts, impractical for direct-driving HMDs.
- Remote viz. required for many HPC problems due to large data.
- Remote viz. latencies too high for direct-drive of HMD.
- Our two-phase approach: moderate-FPS remote RT combined with local high-FPS view-dependent HMD reprojection w/ OpenGL.
VMD TachyonL-OptiX: Multi-GPU on NVIDIA VCA Cluster

Scene Data Replicated, Image Space + Sample Space Parallel Decomposition onto GPUs

VCA 0: 8 K6000 GPUs

VCA N: 8 K6000 GPUs
**Biomedical Technology Research Center for Macromolecular Modeling and Bioinformatics**

**Progressive Ray Tracing Engine**
Ray tracing loop runs continuously in new thread

Decodes H.264 video stream from remote VCA GPU cluster

**HMD Display Loop**
HMD loop runs in main VMD application thread at max OpenGL draw rate

View-dependent stereo reprojection for current HMD head pose

HMD distortion correction

**Camera + Scene**

**Omnistereo Image Stream**

**HMD Video**

**15Mbps Internet Link**

**Remote VCA GPU Cluster**
Ray tracing runs continuously, streams H.264 video to VMD client

**VMD**

**HMD**
Stereoscopic Panorama Ray Tracing w/ OptiX

• Render 360° images and movies for VR headsets such as Oculus Rift, Google Cardboard
• Ray trace panoramic stereo spheremaps or cubemaps for very high-frame-rate display via OpenGL texturing onto simple geometry
• Stereo requires spherical camera projections poorly suited to rasterization
• Benefits from OptiX multi-GPU rendering and load balancing, remote visualization
A) Monoscopic circular projection. Eye at center of projection (COP).

B) Left eye stereo circular projection. Eye offset from COP by half of interocular distance.

C) Stereo eye separation smoothly decreased to zero at zenith and nadir points on the polar axis to prevent incorrect stereo when HMD sees the poles.
# Remote Omnidirectional Stereoscopic RT Performance @ 3072x1536 w/ 2-subframes

<table>
<thead>
<tr>
<th>Scene</th>
<th>Per-subframe samples AA : AO (AO per-hit)</th>
<th>RT update rate (FPS)</th>
</tr>
</thead>
<tbody>
<tr>
<td>STMV shadows</td>
<td>1:0</td>
<td>22.2</td>
</tr>
<tr>
<td></td>
<td>2:0</td>
<td>18.1</td>
</tr>
<tr>
<td></td>
<td>4:0</td>
<td>10.3</td>
</tr>
<tr>
<td>STMV Shadows+AO</td>
<td>1:1</td>
<td>18.2</td>
</tr>
<tr>
<td></td>
<td>1:2</td>
<td>16.1</td>
</tr>
<tr>
<td></td>
<td>1:4</td>
<td>12.4</td>
</tr>
<tr>
<td>STMV Shadows+AO+Do F</td>
<td>1:1</td>
<td>16.1</td>
</tr>
<tr>
<td></td>
<td>2:1</td>
<td>11.1</td>
</tr>
<tr>
<td></td>
<td>2:2</td>
<td>8.5</td>
</tr>
<tr>
<td>HIV-1 Shadows</td>
<td>1:0</td>
<td>20.1</td>
</tr>
<tr>
<td></td>
<td>2:0</td>
<td>18.1</td>
</tr>
<tr>
<td></td>
<td>4:0</td>
<td>10.2</td>
</tr>
<tr>
<td>HIV-1 Shadows+AO</td>
<td>1:1</td>
<td>17.4</td>
</tr>
<tr>
<td></td>
<td>1:2</td>
<td>12.2</td>
</tr>
<tr>
<td></td>
<td>1:4</td>
<td>8.1</td>
</tr>
</tbody>
</table>
HMD View-Dependent Reprojection with OpenGL

• Texture map panoramic image onto reprojection geometry that matches the original RT image formation surface (sphere for equirectangular, cube for cube map)

• HMD sees standard perspective frustum view of the textured surface

• Commodity HMD optics require software lens distortion and chromatic aberration correction prior to display, implemented with multi-pass FBO rendering

• Enables low-latency, high-frame-rate redraw as HMD head pose changes (150Hz or more)
VMD can support a variety of HMD lens designs, e.g.
Future Work

• Support for more commodity HMDs as they become generally available
• Support for OSes besides Linux
• Ray tracing engine and optimizations:
  – Multi-node parallel RT and remote viz. on general clusters and supercomputers, e.g. NCSA Blue Waters, ORNL Titan
  – Interactive RT stochastic sampling strategies to improve interactivity
  – Improved omnidirectional cubemap/spheremap sampling approaches
• Tons of work to do on VR user interfaces, multi-user collaborative visualization, ...
“When I was a young man, my goal was to look with mathematical and computational means at the inside of cells, one atom at a time, to decipher how living systems work. That is what I strived for and I never deflected from this goal.” – Klaus Schulten
Acknowledgements

• Theoretical and Computational Biophysics Group, University of Illinois at Urbana-Champaign
• CUDA Center of Excellence, University of Illinois at Urbana-Champaign
• NVIDIA CUDA and OptiX teams
• Funding:
  – DOE INCITE, ORNL Titan: DE-AC05-00OR22725
  – NSF Blue Waters:
    NSF OCI 07-25070, PRAC “The Computational Microscope”,
    ACI-1238993, ACI-1440026
  – NIH support: 9P41GM104601, 5R01GM098243-02
Related Publications
http://www.ks.uiuc.edu/Research/gpu/


Related Publications
http://www.ks.uiuc.edu/Research/gpu/


*Winner of the SC’14 Visualization and Data Analytics Showcase*


- **Unlocking the Full Potential of the Cray XK7 Accelerator.** M. D. Klein and J. E. Stone. Cray Users Group, Lugano Switzerland, May 2014.


Related Publications
http://www.ks.uiuc.edu/Research/gpu/


Related Publications
http://www.ks.uiuc.edu/Research/gpu/


Related Publications
http://www.ks.uiuc.edu/Research/gpu/


Related Publications

http://www.ks.uiuc.edu/Research/gpu/

• **Adapting a message-driven parallel application to GPU-accelerated clusters.**

• **GPU acceleration of cutoff pair potentials for molecular modeling applications.**

• **GPU computing.**

• **Accelerating molecular modeling applications with graphics processors.**

• **Continuous fluorescence microphotolysis and correlation spectroscopy.**