Simulating Black Holes with CUDA

Adam G.M. Lewis1,2, Abdul H. Mroué2, and Harald P. Pfeiffer2

1Department of Physics, 60 St. George Street, University of Toronto, Toronto, ON M5S 3H8, Canada
2Canadian Institute for Theoretical Astrophysics, 60 St. George Street, University of Toronto, ON M5S 3H8, Canada

This decade will see the first detections of gravitational waves: ripples in spacetime produced strongly by collisions of dense objects like black holes. It will then be possible to study such events through careful comparison of their gravitational radiation against predictions generated through simulations. These simulations are computationally very expensive, requiring 10,000’s of FLOPS per grid point per time step. Using NVIDIA’s CUDA framework, we have developed techniques to automatically port our black hole code to GPUs. We have also manually optimized certain key routines, which have sped up by 10-40 times in response.

A new astronomy

Contemporary physics describes gravity using the general theory of relativity. Mass is thought to bend space and time much as a billiard ball would bend a stretched rubber sheet. These bends curve the masses towards one another. As a direct consequence, orbiting masses create ripples in spacetime which spiral outwards as “gravitational radiation”. The first experiment sensitive enough to detect these, Advanced LIGO (aLIGO), should be fully operational within the next 3-5 years.

GPU porting strategy

Streamline memory synchronization

Automatic CUDA translation

Hand-code key routines

• Extend DataMesh class to copy between host and device only when data is requested.
• MemoryHandler retains memory on device, providing temporary memory without cudaMemcpy/cudaFree calls.
• "CodeWriter" outputs equivalent CUDA kernel for each expression template.
• Runtime dominated by 10 feasibility-hand-coded processes.

Spectral Einstein Code (SpEC)

We simulate gravity using Spectral Einstein Code (SpEC) [5]. Taking as input the curvature of space according to some observer at an early time, SpEC makes discrete steps in that observer’s time coordinate and evolves the slices according to Einstein’s equations.

• FirstDeriv: Compute first derivative of a tensor w.r.t. given full set of coordinates.
• MultiplySpatialComponentsByJacobian: Perform co-ordinate transformation on spatial part of a given tensor.
• GeneralizedHarmonicEquations: Solve Einstein equations at a time step.

Preliminary results

Overall 1.63x speedup obtained for simple test cases.

Hand-coded ports have yielded more dramatic results:

• FirstDeriv:
• MultiplySpatialComponentsByJacobian:
• GeneralizedHarmonicEquations:

References and Image Credits

1. NASA, JPL. Caltech.
2. NASA/Dana Berry, Sky Works Digital.
3. K. Thorne (Caltech) and T. Carnahan (NASA GSFC).