GPU Algorithms
References

Suresh Venkatasubramanian

August 25, 2012

1 Notes

This is a list of references I used for my lectures on GPU algorithms at the MADALGO 2012 Summer School on Algorithms for Modern Parallel and Distributed Models. This is not a comprehensive list of papers in the area, or even a fleshed out survey: the field is far too large and active for any survey to be either complete or timely. I chose papers that exemplified certain aspects of algorithms that I wanted to highlight, and my overall focus was on the algorithm primitives, rather than the problems being solved. The papers cited here have numerous references pointing to related work on GPU computing. For a frequently updated clearinghouse of information on doing high performance computing with GPUs, the reader is encouraged to visit `http://hgpu.org` where they will find most of the papers listed here as well as many more.

In addition to the specific paper references listed in the next section, I drew inspiration from Mary Hall's course on GPU programming at the University of Utah (`http://www.cs.utah.edu/~mhall/cs6963s11/`). Michael Garland visited Utah a few months ago and gave an amazing talk on GPU algorithms for sparse irregular problems, and was kind enough to share his slides and other materials: you should visit his page at `http://mgarland.org/`. Mark Harris's guide to parallel reductions is an excellently done presentation (`http://developer.download.nvidia.com/compute/cuda/1.1-Beta/x86_website/projects/reduction/doc/reduction.pdf`).

References

[1] Ádám Moravánsky. Dense matrix algebra on the gpu. `http://www.shaderx2.com/shaderx.PDF`, 2003.

[2] P. Agarwal, S. Krishnan, N. Mustafa, and S. Venkatasubramanian. Streaming geometric optimization using graphics hardware. *Algorithms-ESA 2003*, pages 544--555, 2003.

[3] ATI/Animusic. Pipe dream. `http://developer.amd.com/ARCHIVE/LEGACYDEMOS/pages/ATIRadeon9700Real-TimeDemos.aspx`.

[4] J. E.-S. Ayal Stein, Eran Geva. CudaHull: Fast parallel 3D convex hull on the GPU. *Computers & Graphics*, 36(4):265--271, 2012.

[5] J. Backus. Can programming be liberated from the von Neumann style ? a functional style and its algebra of programs. ACM Turing Award Lecture, 1977.

[6] N. Bell and M. Garland. Efficient sparse matrix-vector multiplication on cuda. *NVIDIA Corporation, NVIDIA Technical Report NVR-2008-004*, 2008.

[7] I. Buck, T. Foley, D. Horn, J. Sugerman, K. Fatahalian, M. Houston, and P. Hanrahan. Brook for gpus: stream computing on graphics hardware. In *ACM Transactions on Graphics (TOG)*, volume 23, pages 777--786. ACM, 2004.

[8] D. Durfee. Strange quark. `http://sqcomic.com/2012/08/19/`, Aug 19 2012.

[9] K. Fatahalian, J. Sugerman, and P. Hanrahan. Understanding the efficiency of gpu algorithms for matrix-matrix multiplication. In *Proceedings of the ACM SIGGRAPH/EUROGRAPHICS conference on Graphics hardware*, pages 133--137. ACM, 2004.

[10] O. Foundation. Opengl shading language. `http://www.opengl.org/documentation/glsl/`.

[11] N. Govindaraju, J. Gray, R. Kumar, and D. Manocha. Gputerasort: high performance graphics co-processor sorting for large database management. In *Proceedings of the 2006 ACM SIGMOD international conference on Management of data*, pages 325--336. ACM, 2006.

[12] A. Grosset, P. Zhu, S. Liu, S. Venkatasubramanian, and M. Hall. Evaluating graph coloring on gpus. In *Proceedings of the 16th ACM symposium on Principles and practice of parallel programming*, pages 297--298. ACM, 2011.

[13] S. Guha, S. Krishnan, K. Munagala, and S. Venkatasubramanian. Application of the two-sided depth test to csg rendering. In *Proceedings of the 2003 symposium on Interactive 3D graphics*, pages 177--180. ACM, 2003.

[14] J. D. Hall, N. A. Carr, and J. C. Hart. Cache and bandwidth aware matrix multiplication on the gpu. Technical Report UIUCDCS-R-2003-2328, UIUC, 2003.

[15] M. Harris, G. Blelloch, B. Maggs, N. Govindaraju, B. Lloyd, W. Wang, M. Lin, D. Manocha, P. Smolarkiewicz, L. Margolin, et al. Optimizing parallel reduction in cuda. *Proc. of ACM SIGMOD, 21*, 13:104--110, 2007.

[16] K. Hoff III, J. Keyser, M. Lin, D. Manocha, and T. Culver. Fast computation of generalized voronoi diagrams using graphics hardware. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, pages 277--286. ACM Press/Addison-Wesley Publishing Co., 1999.

[17] D. Kirk, W. Wen-mei, and W. Hwu. *Programming massively parallel processors: a hands-on approach*. Morgan Kaufmann, 2010.

[18] H. T. Kung and C. E. Leiserson. Systolic arrays (for VLSI). *Sparse Matrix Proc., SIAM*, pages 256--282, 1978.

[19] N. Leischner, V. Osipov, and P. Sanders. Gpu sample sort. In *Parallel & Distributed Processing (IPDPS), 2010 IEEE International Symposium on*, pages 1--10. Ieee, 2010.

[20] W. Mark, R. Glanville, K. Akeley, and M. Kilgard. Cg: A system for programming graphics hardware in a c-like language. In *ACM Transactions on Graphics (TOG)*, volume 22, pages 896--907. ACM, 2003.

[21] J. Markoff. From PlayStation to Supercomputer for $50,000. `http://www.nytimes.com/2003/05/26/business/technology-from-playstation-to-supercomputer-for-50000.html`, May 2003. New York Times.

[22] M. McCool and S. Du Toit. *Metaprogramming GPUs with Sh*. AK Peters Wellesley, 2004.

[23] D. Merrill, M. Garland, and A. Grimshaw. Scalable gpu graph traversal. In *Proceedings of the 17th ACM SIGPLAN symposium on Principles and Practice of Parallel Programming*, PPoPP '12, pages 117--128, New York, NY, USA, 2012. ACM.

[24] D. Merrill and A. Grimshaw. Revisiting sorting for gpgpu stream architectures. In *Proceedings of the 19th international conference on Parallel architectures and compilation techniques*, pages 545--546. ACM, 2010.

[25] Microsoft. Hlsl. `http://msdn.microsoft.com/en-us/library/windows/desktop/bb509561(v=vs.85).aspx`.

[26] J. Nickolls, I. Buck, M. Garland, and K. Skadron. Scalable parallel programming with cuda. *Queue*, 6(2):40--53, 2008.

[27] NVIDIA. Parallel programming and computing platform | CUDA. `http://www.nvidia.com/object/cuda_home_new.html`.

[28] T. Purcell, C. Donner, M. Cammarano, H. Jensen, and P. Hanrahan. Photon mapping on programmable graphics hardware. In *Proceedings of the ACM SIGGRAPH/EUROGRAPHICS conference on Graphics hardware*, pages 41--50. Eurographics Association, 2003.

[29] G. Rong and T. Tan. Jump flooding in gpu with applications to voronoi diagram and distance transform. In *Proceedings of the 2006 symposium on Interactive 3D graphics and games*, pages 109--116. ACM, 2006.

[30] S. Sengupta, M. Harris, Y. Zhang, and J. Owens. Scan primitives for gpu computing. In *Proceedings of the 22nd ACM SIGGRAPH/EUROGRAPHICS symposium on Graphics hardware*, pages 97--106. Eurographics Association, 2007.

[31] D. Shreiner. *OpenGL programming guide: the official guide to learning OpenGL, versions 3.0 and 3.1*, volume 1. Addison-Wesley Professional, 2010.

[32] D. Srikanth, K. Kothapalli, R. Govindarajulu, and P. Narayanan. Parallelizing two dimensional convex hull on nvidia gpu and cell be. In *International Conference on High Performance Computing (HiPC)*, pages 1--5, 2009.

[33] S. Venkatasubramanian. The graphics card as a stream computer. In *SIGMOD-DIMACS Workshop on Management and Processing of Data Streams*, 2003.

[34] Wikipedia. Flynn's taxonomy. `http://en.wikipedia.org/wiki/Flynn's_taxonomy`, August 2012.

[35] M. Zechner and M. Granitzer. Accelerating k-means on the graphics processor via cuda. In *Intensive Applications and Services, 2009. INTENSIVE'09. First International Conference on*, pages 7--15. IEEE, 2009.