Implementing External Memory Algorithms and Data Structures

(Invited talk)

Lars Arge*

Department of Computer Science
Duke University
Durham, NC 27708
USA

Many modern applications store and process datasets much larger than the main memory of even state-of-the-art high-end machines. In such cases, the Input/Output (or I/O) communication between fast internal memory and slow disks, rather than actual internal computation time, can become a major performance bottleneck. In the last decade, much attention has therefore been focused on the development of theoretically I/O-efficient algorithms and data structures [3, 13].

In this talk we discuss recent efforts at Duke University to investigate the practical merits of theoretically developed I/O-efficient algorithms. We describe the goals and architecture of the TPIE environment for efficient implementation of I/O-efficient algorithms [12, 10, 4], as well as some of the implementation projects conducted using the environment [9, 8, 2, 11, 5, 7, 6, 1], and discuss some of the experiences we have had and lessons we have learned in these projects. We especially discuss the TerraFlow system for efficient flow computation on massive grid-based terrain models, developed in collaboration with environmental researchers [5]. Finally we discuss how the implementation and experimentation work has supported educational efforts.

References

[1] P. K. Agarwal, L. Arge, and S. Govindarajan. CRB-tree: An optimal indexing scheme for 2d aggregate queries. In *Proc. International Conference on Database Theory*, 2003.

[2] P. K. Agarwal, L. Arge, O. Procopiuc, and J. S. Vitter. Bkd-tree: A dynamic scalable kd-tree. Manuscript, 2002.

[3] L. Arge. External memory data structures. In J. Abello, P. M. Pardalos, and M. G. C. Resende, editors, *Handbook of Massive Data Sets*, pages 313–358. Kluwer Academic Publishers, 2002.

[4] L. Arge, R. Barve, D. Hutchinson, O. Procopiuc, L. Toma, D. E. Vengroff, and R. Wickremesinghe. *TPIE User Manual and Reference (edition 082902)*. Duke University, 2002. The manual and software distribution are available on the web at `http://www.cs.duke.edu/TPIE/`.

[5] L. Arge, J. Chase, P. Halpin, L. Toma, D. Urban, J. S. Vitter, and R. Wickremesinghe. Flow computation on massive grid terrains. *GeoInformatica*, 2003. (To appear). Earlier version appeared in *Proc. 10'th ACM International Symposium on Advances in Geographic Information Systems* (ACM-GIS'01).

[6] L. Arge, A. Danner, and S.-H. Teh. I/O-efficient point location using persistent B-trees. In *Proc. Workshop on Algorithm Engineering and Experimentation*, 2003.

[7] L. Arge, K. H. Hinrichs, J. Vahrenhold, and J. S. Vitter. Efficient bulk operations on dynamic R-trees. *Algorithmica*, 33(1):104–128, 2002.

[8] L. Arge, O. Procopiuc, S. Ramaswamy, T. Suel, J. Vahrenhold, and J. S. Vitter. A unified approach for indexed and non-indexed spatial joins. In *Proc. Conference on Extending Database Technology*, pages 413–429, 1999.

[9] L. Arge, O. Procopiuc, S. Ramaswamy, T. Suel, and J. S. Vitter. Scalable sweeping-based spatial join. In *Proc. International Conf. on Very Large Databases*, pages 570–581, 1998.

[10] L. Arge, O. Procopiuc, and J. S. Vitter. Implementing I/O-efficient data structures using TPIE. In *Proc. Annual European Symposium on Algorithms*, pages 88–100, 2002.

[11] L. Arge, L. Toma, and J. S. Vitter. I/O-efficient algorithms for problems on grid-based terrains. In *Proc. Workshop on Algorithm Engineering and Experimentation*, 2000.

[12] D. E. Vengroff. A transparent parallel I/O environment. In *Proc. DAGS Symposium on Parallel Computation*, 1994.

[13] J. S. Vitter. External memory algorithms and data structures: Dealing with MASSIVE data. *ACM Computing Surveys*, 33(2):209–271, 2001.

*Supported in part by the National Science Foundation through ESS grant EIA–9870734, RI grant EIA–9972879, CAREER grant CCR–9984099, ITR grant EIA–0112849, and U.S.-Germany Cooperative Research Program grant INT–0129182. Email: `large@cs.duke.edu`.