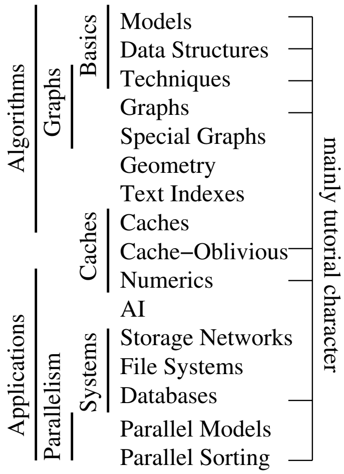
Preface

Algorithms that process large data sets have to take into account that the cost of memory accesses depends on where the accessed data is stored. Traditional algorithm design is based on the von Neumann model which assumes uniform memory access costs. Actual machines increasingly deviate from this model. While waiting for a memory access, modern microprocessors can execute 1000 additions of registers. For hard disk accesses this factor can reach seven orders of magnitude. The 16 chapters of this volume introduce and survey algorithmic techniques used to achieve high performance on memory hierarchies. The focus is on methods that are interesting both from a practical and from a theoretical point of view.

This volume is the result of a *GI-Dagstuhl Research Seminar*. The Gesellschaft für Informatik (GI) has organized such seminars since 1997. They can be described as "self-taught" summer schools where graduate students in cooperation with a few more experienced researchers have an opportunity to acquire knowledge about a current topic of computer science. The seminar was organized as Dagstuhl Seminar 02112 from March 10, 2002 to March 14, 2002 in the International Conference and Research Center for Computer Science at Schloss Dagstuhl.

Chapter 1 gives a more detailed motivation for the importance of algorithm design for memory hierarchies and introduces the models used in this volume. Interestingly, the simplest model variant — two levels of memory with a single processor — is sufficient for most algorithms in this book. Chapters 1–7 represent much of the algorithmic core of external memory algorithms and almost exclusively rely on this simple model. Among these, Chaps. 1–3 lay the foundations by describing techniques used in more specific applications. Rasmus Pagh discusses data structures like search trees, hash tables, and priority queues in Chap. 2. Anil Maheshwari and Norbert Zeh explain generic algorithmic approaches in Chap. 3. Many of these techniques such as time-forward processing, Euler tours, or list ranking can be formulated in terms of graph theoretic concepts. Together with Chaps. 4 and 5 this offers a comprehensive review of external graph algorithms. Irit Katriel and Ulrich Meyer discuss fundamental algorithms for graph traversal, shortest paths, and spanning trees that work for many types of graphs. Since even simple graph problems can be difficult to solve in external memory, it



makes sense to look for better algorithms for frequently occurring special types of graphs. Laura Toma and Norbert Zeh present a number of astonishing techniques that work well for planar graphs and graphs with bounded tree width.

In Chap. 6 Christian Breimann and Jan Vahrenhold give a comprehensive overview of algorithms and data structures handling geometric objects like points and lines — an area that is at least as rich as graph algorithms. A third area of again quite different algorithmic techniques are string problems discussed by Juha Kärkäinen and Srinivasa Rao in Chap. 7.

Chapters 8–10 then turn to more detailed models with particular emphasis on the complications introduced by hardware caches. Beyond this common motivation, these chapters are quite diverse. Naila Rahman uses sorting as an example for these issues in Chap. 8 and puts particular emphasis on the often neglected issue of TLB misses. Piyush Kumar introduces *cache-oblivious algorithms* in Chap. 9 that promise to grasp multilevel hierarchies within a very simple model. Markus Kowarschik and Christian Weiß give a practical introduction into cache-efficient programs using numerical algorithms as an example. Numerical applications are particularly important because they allow significant instruction-level parallelism so that slow memory accesses can dramatically slow down processing.

Stefan Edelkamp introduces an application area of very different character in Chap. 11. In artificial intelligence, search programs have to handle huge state spaces that require sophisticated techniques for representing and traversing them.

Chapters 12–14 give a system-oriented view of advanced memory hierarchies. On the lowest level we have storage networks connecting a large number of inhomogeneous disks. Kay Salzwedel discusses this area with particular emphasis on the aspect of inhomogeneity. File systems give a more abstract view of these devices on the operating system level. Florin Isaila explains the organization of modern file systems in Chap. 13. An even higher level view is offered by relational database systems. Josep Larriba-Pey explains their organization in Chap. 14. Both in file systems and databases, basic algorithmic techniques like sorting and search trees turn out to be relevant.

Finally, Chaps. 15 and 16 give a glimpse on memory hierarchies with multiple processors. Massimo Coppola and Martin Schmollinger introduce abstract and concrete programming models like BSP and MPI in Chap. 15. Dani Jimenez, Josep-L. Larriba, and Juan J. Navarro present a concrete case study of sorting algorithms on shared memory machines in Chap. 16. He studies programming techniques that avoid pitfalls like true and false sharing of cache contents.

Most chapters in this volume have partly tutorial character and are partly more dense overviews. At a minimum Chaps. 1, 2, 3, 4, 9, 10, 14, and 16 are tutorial chapters suitable for beginning graduate-level students. They are sufficiently self-contained to be used for the core of a course on external memory algorithms. Augmented with the other chapters and additional papers it should be possible to shape various advanced courses. Chapters 1–3 lay the basis for the remaining chapters that are largely independent.

We are indebted to many people and institutions. We name a few in alphabetical order. Ulrik Brandes helped with sources from a tutorial volume on graph drawing that was our model in several aspects. The International Conference and Research Center for Computer Science in Dagstuhl provided its affordable conference facilities and its unique atmosphere. Springer-Verlag, and in particular Alfred Hofmann, made it possible to smoothly publish the volume in the LNCS series. Kurt Mehlhorn's group at MPI Informatik provided funding for several (also external) participants. Dorothea Wagner came up with the idea for the seminar and advised us in many ways. This volume was also partially supported by the Future and Emerging Technologies programme of the EU under contract number IST-1999-14186 (ALCOM-FT).

January 2003

Ulrich Meyer
Peter Sanders
Jop Sibeyn

http://www.springer.com/978-3-540-00883-5

Algorithms for Memory Hierarchies
Advanced Lectures
Meyer, U.; Sanders, P.; Sibeyn, J. (Eds.)
2003, XV, 429 p., Softcover
ISBN: 978-3-540-00883-5