Lorraine - Saarland Workshop
Algorithms and Engeneering of Large Discrete Structures

November 7, 2003
LORIA, Nancy, France
Room A 208

about 9:30	Coffee reception
10:00	Jens Gustedt, ... : Algorithms for the Grid at INRIA/LORIA
10:30	Peter Sanders, MPII, Saarbrücken, Germany: Parallel and External Algorithms at MPII
11:00	Mathieu Latapy, LIAFA, Paris, and CNRS: Statistical Analysis of Very Large Graphs.
11:30	Ulrich Meyer, MPII, Saarbrücken, Germany: External memory graph traversal
12:00	Jens Gustedt, LORIA and INRIA Lorraine, France: Towards Realistic Implementations of External Memory Algorithms using a Coarse Grained Paradigm
12:30	lunch break
14:00	Philipp Slusallek, Saarland University, Germany: Distributed Realtime Ray-Tracing and Lighting Simulation on Commodity Hardware
14:30	Frédéric Wagner, LORIA and Université Henri Poincaré, Nancy, France: Two Fast and Efficient Message Scheduling Algorithms for Data Redistribution through a Backbone
15:00	Coffee break
15:15	Dariusz Kowalski, MPII, Saarbrücken, Germany: *title to be announced*
15:45	Emmanuel Jeannot, LORIA and Université Henri Poincaré, Nancy, France: Adaptive On-line Compression

Abstracts

Jens Gustedt, ... :
Algorithms for the Grid at INRIA/LORIA

Peter Sanders, MPII, Saarbrücken, Germany:
Parallel and External Algorithms at MPII

I will attempt to give a very short overview of the current work on parallel, distributed and external memory algorithms done in the algorithms group of MPII. From my own work I will talk about collective communication (broadcast, multicast, all-to-all, ...) and about external memory algorithms in theory and practice.

Mathieu Latapy, LIAFA, Paris, and CNRS:
Statistical Analysis of Very Large Graphs.

Very large graphs play a central role in many contexts, like social sciences, computer science, biology, etc. With the growth of the use of computer systems, more and more such graphs become available for use and study. This induces an increasing need of (statistical and algorithmical) tools for their description and analysis. However, the size of these graphs (often several millions of nodes) makes it generally impossible to use classical methods to study them.

Using a representative set of examples, I will present the basics of the statistical description of real-world very large graphs. I will insist on the surprising fact that most of them have several statistical properties in common, whereas other properties puts them into distinct classes. I will then give an overview of the modelling problems induced by this, and of the known solutions. Finally, I will discuss the algorithmical implications of these statistical studies, with a special focus on searching, and on short paths and cliques computation.

Ulrich Meyer, MPII, Saarbrücken, Germany:
External memory graph traversal

Sequential algorithms for graph traversal like breadth-first search (BFS), depth-first search (DFS) and single-source shortest-paths (SSSP) are already fairly well understood. The study of their external-memory counterparts (where the input graph does not fit into the main memory), on the other hand, is still in its infancy. Until recently, efficient external-memory BFS and SSSP algorithms (i.e., algorithms that perform only slightly more than $cO(m/B)$ I/Os, where B is the block size) only existed for either dense graphs (which are quite rare in practice) or special, mostly undirected, graph classes like planar graphs. In particular, for the practically relevant case of general graphs with $m = \mathcal{O}(n)$ edges all previous BFS/SSSP algorithms required $\Omega(n)$ I/Os, thus hardly exploiting the power of block transfers. In this talk we present the first BFS and SSSP algorithms for general sparse undirected graphs that perform $o(n)$ I/Os. These algorithms improve upon the respective previous approaches by factors of up to $\sqrt{B}$.

Jens Gustedt, LORIA and INRIA Lorraine, France:
Towards Realistic Implementations of External Memory Algorithms using a Coarse Grained Paradigm

We present an extension to SSCRAP, our C++ environment for the development of coarse grained algorithms, that allows for easy execution of programs in an external memory setting. Our environment is well suited for regular as well as irregular problems and scales from low end PCs to high end clusters and mainframe technology. It allows running algorithms designed on a high level of abstraction in one of the known coarse grained parallel models without modification in an external memory setting. The first tests presented here in this paper show a very efficient behavior in the context of out-of-core computation (mapping memory to disk files), and even some (marginal) speed up when used to reduced cache misses for in-core computation.

Philipp Slusallek, Saarland University, Germany:
Distributed Realtime Ray-Tracing and Lighting Simulation on Commodity Hardware

It is now possible to perform realtime ray tracing on PCs in software on a single CPU with up to 7 million rays per second for simple scenes. Using distributed computing on a cluster of PCs allows to easily scale to fully interactive performance even for scenes of tens of millions and even billions of polygons including highly complex shading. With new lighting simulation algorithms we can now also fully recompute global illumination at realtime frame rates.

In my talk I will present the OpenRT realtime ray tracing engine and its distributed implementation, discuss how its performance and scalability was achieved, and point out consequences of these recent developments for research and industry. In addition I will briefly address current research activities at Saarland University and discuss a number of open research problems in this field.

Frédéric Wagner, LORIA and Université Henri Poincaré, Nancy, France: Two Fast and Efficient Message Scheduling Algorithms for Data Redistribution through a Backbone

In this work we study the problem of redistributing in parallel data between clusters interconnected by a backbone. This problem is a generalization of the well-known redistribution problem that appears in parallelism. We suppose that at most k communications can be performed at the same time (the value of k depending on the characteristics of the platform). We use the knowledge of the application in order to schedule the messages and perform a control of the congestion by ourselves. Previous results show that this problem is NP-Complete. We propose and study two fast and efficient algorithms for this problem. We prove that these algorithms are 2-approximation algorithms. Simulation results show that both algorithms perform very well compared to the optimal solution. These algorithms have been implemented using MPI. Experimental results show that both algorithms outperform a brute-force TCP based solution, when no scheduling of the messages is performed.

Dariusz Kowalski, MPII, Saarbrücken, Germany: *title to be announced*

Emmanuel Jeannot, LORIA and Université Henri Poincaré, Nancy, France: Adaptive On-line Compression

Quickly transmitting large datasets in the context of distributed computing on wide area networks can be achieved by compressing data before transmission. However, such an approach is not efficient when dealing with high-speed networks. Indeed, the time to compress a large file and to send it is greater than the time to send the uncompressed file. In this paper, we propose an algorithm that allows to overlap communications with compression and to adapt automatically the compression ratio according to the network speed (the slower the network, the more we use efficient and slow compression algorithms). The advantage of such an adaptive algorithm is its generality and that its suitability for a large set of applications.