UNIVERSITÄT SALZBURG

Computer Sciences Workshop

Dissertantenseminar
May 16, 2014

Program

Session A: **Algorithms in Complex Domains**
Chair: *Nikolaus Augsten*
HS T02, 11h00 – 12h30

Information Dissemination in Large Networks
Dominik Kaaser

Efficient Set Similarity Joins
Willi Mann

Straight Skeletons of Monotone Polygons
Peter Palfrader

Session B: **Image Processing and Parallel Computation**
Chair: *Marian Vajteršic*
HS T02, 13h00 – 14h30

Slice Groups for Post-Compression Region of Interest Encryption in SVC
Andreas Unterweger

Biometric recognition of wood logs using digital log end images
Rudolf Schraml

Parallel Nonnegative Tensor Factorization via Newton Iteration on Matrices
Markus Flatz

Session C: **Computational Systems**
Chair: *Christoph Kirsch*
HS T02, 14h45 – 16h15

ACDC-JS: Explorative Benchmarking of JavaScript Memory Management
Martin Aigner

Fast Concurrent Data-Structures Through Explicit Timestamping
Andreas Haas

Development and Validation of an Expert System for Disaster Management
Bukhtiar Mohsin

END

Abstracts

Session A: Algorithms in Complex Domains

Information Dissemination in Large Networks

Dominik Kaaser

Information dissemination is a fundamental problem in parallel and distributed computing. In its simplest variant, known as the broadcasting problem, a message has to be spread among all nodes of a graph. In recent years, several efficient algorithms have been developed to solve the broadcasting problem in the so-called random phone call model. That is, in each step, every node opens a channel to a randomly chosen neighbor for bi-directional communication (Karp et al., FOCS 2000). In our presentation, we will give an overview over our current research in this area.

I. Graph Density has (almost) no Influence on the Performance of Randomized Gossiping

It is known that the performance of randomized broadcasting – *one to all communication* – in complete graphs cannot be achieved in sparse graphs even with best expansion and connectivity properties. Our analysis, however, shows that a similar influence of the graph density does not hold for gossiping – *all to all communication*. Our theoretical results are supported by simulations which show an even better performances than predicted by theory.

II. A Memory Model for Randomized Gossiping

One of the main advantages in randomized gossiping algorithms is their immanent robustness against node failures. We investigate an algorithm in a slightly weaker model in which each node is equipped with a constant amount of memory. This memory can be used to construct a message distribution tree in the graph. However, this approach is highly sensitive towards maleficent nodes. Therefore, we analyze a version in which a set of trees is independently built to regain robustness.

III. On the Influence of Far-Distance Relationships on Rumor Spreading

We enhance the performance of classical randomized push-pull algorithms in random geometric graphs by adding additional long-range edges (Kleinberg, STOC 2000). Each node is connected to a constant number of nodes chosen uniformly at random from the entire graph. We will present a simulation to visualize this procedure.

Efficient Set Similarity Joins
Willi Mann

Set similarity joins identify pairs of similar sets. Two sets are similar if they have many elements in common. Set similarity joins have a large number of applications, for example, they are used to identify pairs of similar objects if the objects can be represented as sets of tokens like strings, trees, and graphs. Another interesting application are search engines, where set similarity joins are used to identify near duplicate webpages in order to remove them from search results and to remove spam pages. Other applications include web mining, document clustering, finding replicated web collections, detecting plagiarism, community mining in a social network site, and many more.

During the past decade, a number of algorithms have been proposed to efficiently compute set similarity joins. Most algorithms are based on the prefix filter principle, and they improve over the plain prefix filter by applying additional filters. This presentation will introduce to the state-of-the-art filtering techniques and compare representative algorithms for set similarity joins.

Straight Skeletons of Monotone Polygons
Peter Palfrader

The straight skeleton $\mathcal{S}(P)$ is a particular type of bisector graph of a polygon P and is defined by a specific wavefront propagation of that polygon. Its edges are the line segments traced out by wavefront vertices during the propagation process, and its vertices witness changes in the topology of the wavefront.

Straight skeletons appear naturally in architecture, where they solve a problem in roof design. They also have applications in terrain generation, design of pop-up cards, or mathematical origami.

The currently best known algorithm for computing straight skeletons of arbitrary simple polygons is due to Eppstein and Erickson (1999) and runs in $\mathcal{O}(n^{17/11+\varepsilon})$ time and space.

We study the characteristics of straight skeletons of monotone polygonal chains, and use them to devise an algorithm for computing straight skeletons and positively weighted straight skeletons of monotone polygons.

Our algorithm runs in $\mathcal{O}(n \log n)$ time and $\mathcal{O}(n)$ space, where n denotes the number of vertices of the polygon.

Session B: Image Processing and Parallel Computation

Slice Groups for Post-Compression Region of Interest Encryption in SVC

Andreas Unterweger

In video surveillance, there is the need to disguise people's identities, e.g., by encrypting their faces, also called region of interest encryption. In order to be able to reuse existing surveillance infrastructure, region of interest encryption has to be applied after compression, also referred to as post-compression encryption. However, this introduces artifacts which are caused by dependencies between encrypted and non-encrypted parts of the surveillance video. It is shown that, using a coding tool called slice groups, these artifacts can be reduced at the cost of relatively low bandwidth overhead when using Scalable Video Coding (SVC).

Biometric recognition of wood logs using digital log end images

Rudolf Schraml

Traceability of wood logs is a basic requirement to manage ecological and social issues. State-of-the-art systems propose the usage of RFID (Radio Frequency Identification) to reach traceability from forest site to further processing companies. Another idea relies on biometric recognition of wood logs. Similar as for human fingerprint recognition, we assume that annual ring patterns of log ends can be used to recognize wood logs. In this talk the computation and matching of Gabor-based features from log end images is introduced. Furthermore, first results give information on the feasibility of biometric log recognition.

Parallel Nonnegative Tensor Factorization via Newton Iteration on Matrices

Markus Flatz

Nonnegative Matrix Factorization (NMF) is a technique to approximate a large nonnegative matrix as a product of two nonnegative matrices of significantly smaller size, which are easier to handle and process. Since matrices can be seen as second-order tensors, NMF can be generalized to Nonnegative Tensor Factorization (NTF) for tensors of any order. There are multiple approaches to compute an NTF, one way is to transform the tensor problem into a matrix problem by using matricization. Any NMF algorithm can be used to process such a matricized tensor, including a method based

on Newton iteration. The result matrices can be transformed back into tensor form with appropriate index computations, yielding the sought NTF. In this talk, an approach will be presented to adopt our parallel design of the Newton algorithm for NMF to compute an NTF in parallel for tensors of any order.

Session C: Computational Systems

ACDC-JS: Explorative Benchmarking of JavaScript Memory Management

Martin Aigner

We present ACDC-JS, an open-source JavaScript memory management benchmarking tool. ACDC-JS incorporates a heap model based on real web applications and may be configured to expose virtually any relevant performance characteristics of JavaScript memory management systems. ACDC-JS is based on ACDC, a benchmarking tool for C/C++ that models periodic allocation and deallocation behavior (AC) as well as persistent memory (DC). We identify important characteristics of JavaScript mutator behavior and propose a configurable heap model based on typical distributions of these characteristics as foundation for ACDC-JS. We describe heap analyses of 13 real web applications extending existing work on JavaScript behavior analysis. Our experimental results show that ACDC-JS enables performance benchmarking and debugging of state-of-the-art JavaScript virtual machines such as V8 and SpiderMonkey by exposing key aspects of their memory management performance.

Fast Concurrent Data-Structures Through Explicit Timestamping

Andreas Haas

Concurrent data-structures, such as stacks, queues and deques, often implicitly enforce a total order over ele- ments with their underlying memory layout. However, linearizability only requires that elements are ordered if the inserting methods ran sequentially. We propose a new data-structure design which uses explicit time- stamping to avoid unwanted ordering. Elements can be left unordered by associating them with unordered timestamps if their insert operations ran concurrently. In our approach, more concurrency translates into less ordering, and thus less-contended removal and ultimately higher performance and scalability.

Development and Validation of an Expert System for Disaster Management

Bukhtiar Mohsin

The PLUS Expert System (PES) will be a part of the concept case: "Advance Situational awareness", for the EU FP7-BRIDGE project [1]. PES receives sensor data from the hexacopter and the modelling module. The incoming (raw) data is reduced to its crucial content, processed and analyzed automatically. Through the BRIDGE middleware, this data is passed on to the Master Table where immediate actions can be taken. One important function of the PES is to give expert advice to the first responders. This advice will be based upon international standards, data libraries and SOP?s prepared by various experts in the past, for example the IAEA radioactivity data, and SOPs followed by the fire department. An important task will be to acquire this data and convert it to more useable format. PES is not only about advice to the first responders though. This expert system will also be responsible to process the hexacopter data and present the data in a way that can be very beneficial for over all situational awareness. The idea is to present the sensor data on ergonomic maps as overlays on some pre-existing APIs. These maps will also contain important information like nearby police stations and hospitals. Subsequent important information will be added to these maps as deemed necessary.