Diplomarbeitspräsentation

FAKULTÄT FÜR !NFORMATIK

Faculty of Informatics

Towards Visual Exploration of Parallel Programs using a Domain-specific Language

Masterstudium:
Visual Computing

Tobias Klein

Technische Universität Wien
Institut für Computergraphik und Algorithmen
Arbeitsbereich: Computergraphik
Betreuer: Ao.Univ.-Prof. Dr. M. Eduard Gröller
Mitwirkung: Dipl.-Ing. Dr.techn. Peter Rautek

MOTIVATION

Researchers frequently encounter programming challenges that could be solved efficiently in a parallel manner but they often possess only **limited experience** in parallel programming. Additionally, achieving efficient parallel code is a **complex optimization problem**, and a time-consuming, error-prone task.

The execution of parallel programs is often considered as a **black box**, where only input and output is known, but only little is displayed about the execution of the code itself. However, this is crucial for the understanding, correctness and especially the performance of the implementation.

The optimization of parallel programs is always **multi-objective**. For instance, optimizing the code for memory-throughput may influence the instruction-throughput and compromise the overall performance. This is the reason for programmers having to deal with complex code, errors, and performance guess work.

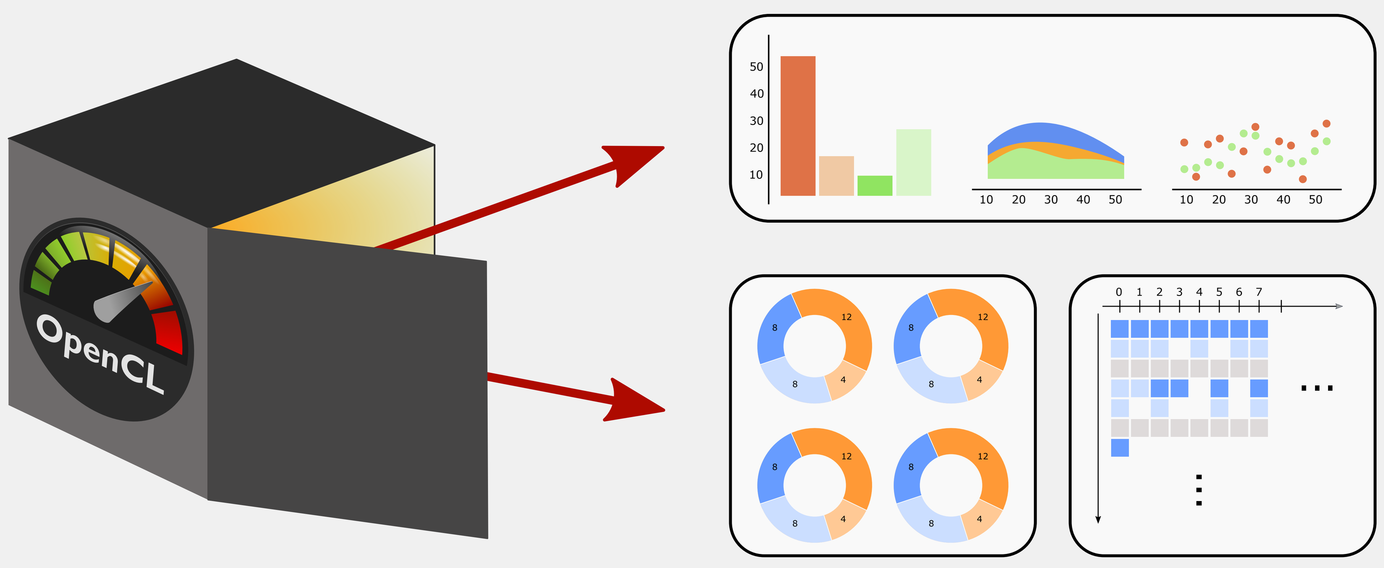
Traditional performance tools usually read back hardware counters and display statistics that correspond to the question "*What is going wrong*". In contrast to these tools, the aim of this work is to allow programmers to quickly test their hypothesis on "*Why is something going wrong*".

APPROACH

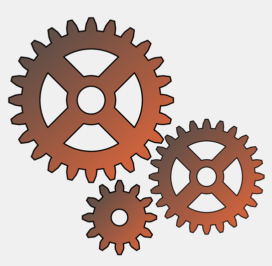
Domain-specific Language (DSL): We have developed a DSL based on ViSlang [1] and designed in a way that it closely resembles OpenCL C language with extra annotations to specify recordings of intermediate data.

Source-to-Source Compiler: We have developed a compiler that transforms programs, written in our DSL, to OpenCL programs with additional instrumentations to record the program's behavior during the execution.

Visual Explorer: Our visual explorerer reveals the behavior of the execution of the program and interactions that occur with the underlying hardware. The visualizations are based on the well-known D3 Framework [2].



APPLICATION



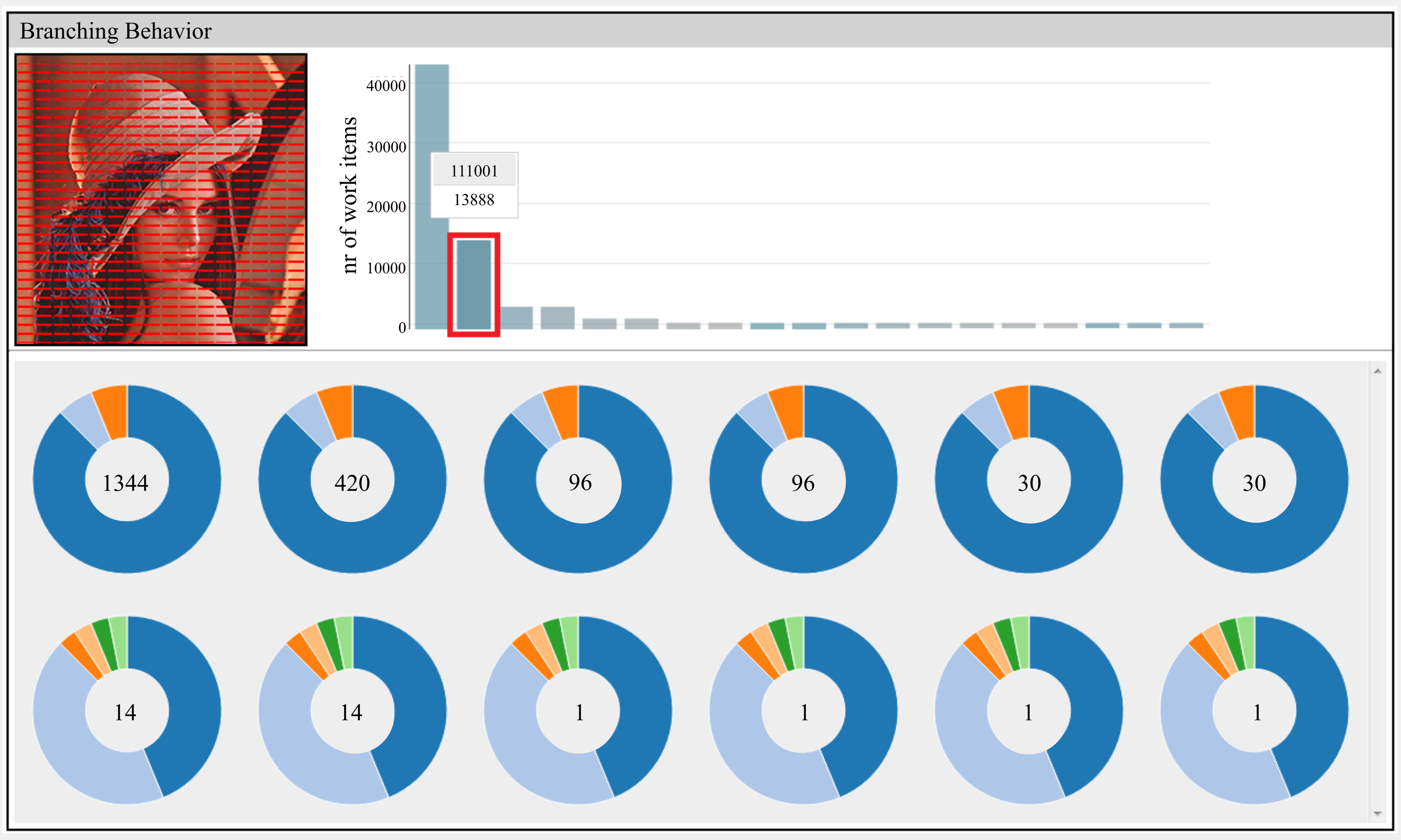
Rapid Prototyping: Our framework automatically creates the setup and boilerplate code that is required for the execution of parallel programs and benefits from fast turnaround times. This facilitates the rapid integration and testing of new algorithms.

Debugging: The investigation of intermediate data is usually realized with manual code changes, which record data and send it back to the host. Our system provides simple code annotations, which automatize this step and enable the direct use of visualizations.

Profiling: Our visualizations reveal common performance issues, such as branch divergence and bank conflicts and furthermore support the detection of their causes.

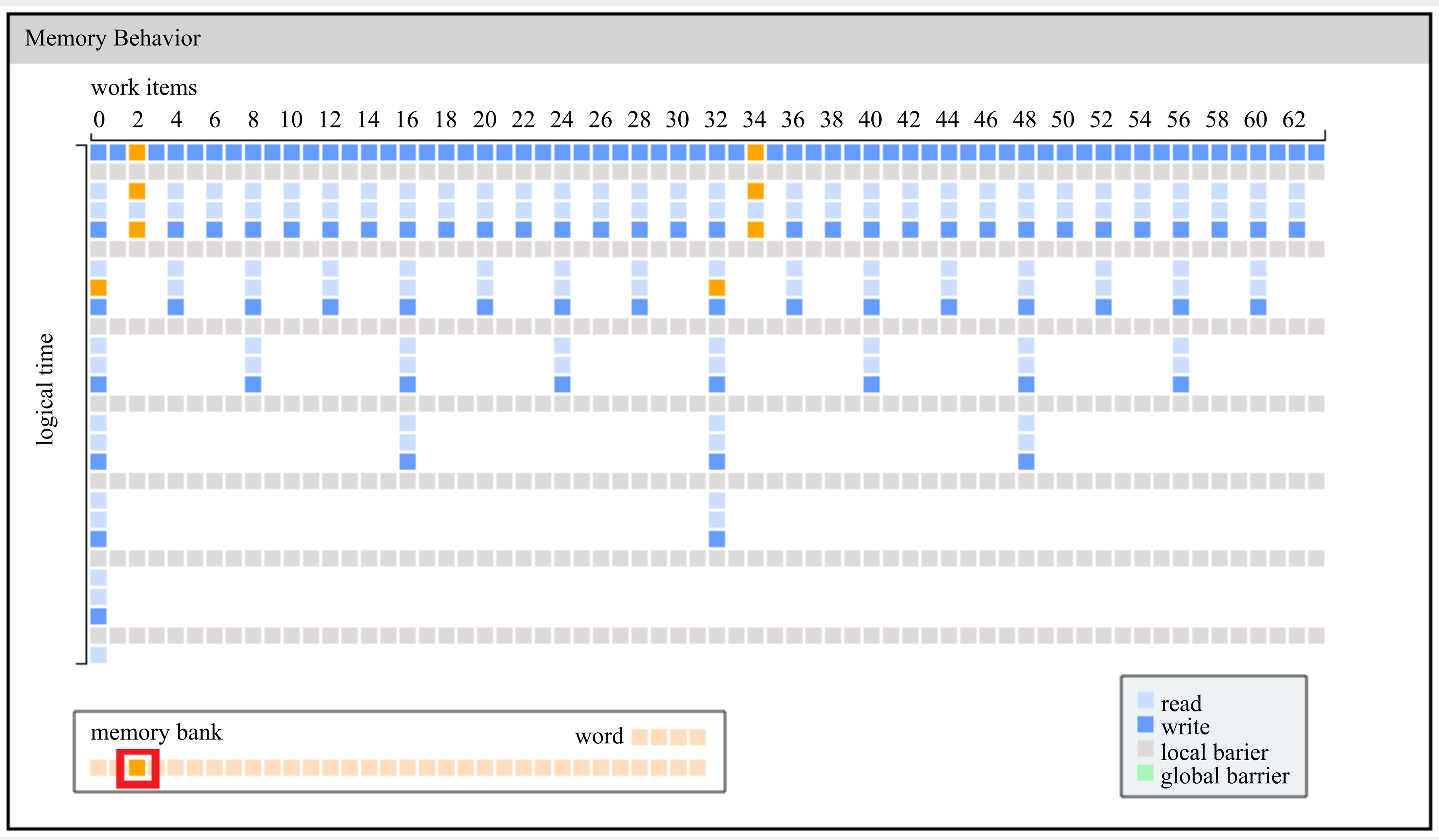
Education: Our visualizations can be utilized for educational purposes since they support the direct understanding of complex parallel algorithms through the visual representation and interactive exploration of their behavior.

VISUAL EXPLORER



Branching View

The branching view reveals different control flow patterns that occur during the execution of the program. It also visually encodes branch divergence, a common performance issue in parallel programs. Equal semantics are linked between different views so that the user is able to interactively explore the behavior of the algorithm.



Memory View

The memory view depicts memory accesses of a specified variable. Furthermore, it shows visual representations (memory bank) that follow the semantics of the underlying hardware architecture and their interaction with the execution. This view also enables to explore memory bank conflicts and different memory access patterns.

REFERENCES

[1] Peter Rautek, Stefan Bruckner, M. Eduard Gröller, and Markus Hadwiger. ViSlang: A system for interpreted domain-specific languages for scientific visualization. *IEEE Transactions on Visualization and Computer Graphics (Proc. SCIVIS '14)*, 20(12):2388-2396, 2014.

[2] Michael Bostock, Vadim Ogievetsky, and Jeffrey Heer. D3: Data-driven documents. *IEEE Transactions on Visualization and Computer Graphics (Proc. InfoVis '11)*, 17(12):2301-2309, 2011.

جامعة الملك عبدالله للعلوم والتقنية
King Abdullah University of Science and Technology

CONTACT *tobias.klein@tuwien.ac.at*