Seminars: 1st Semester - 2016

Compilers Laboratory

The seminars of the Compilers Lab happen every Friday, at ICEx room 2014, at 2pm. Usually they end around 3:30pm. Each attendee must read the abstract, introduction (and ideally the Section 2) of each paper. Everyone should participate in the discussions. When reading a paper, try to find answers for the following questions: what is the problem that the paper solves? Why is this problem important? What is the key idea of the paper? How does the paper improve on previous work?

March 4th – PLDI'15

- Algorithmic Debugging of Real-World Haskell Programs: Deriving Dependencies from the Cost Centre Stack
- Automatic Error Elimination by Multi-Application Code Transfer
- Light: Replay via Tightly Bounded Recording
- Many-Core Compiler Fuzzing

March 11th – PLDI'15

- Efficient Synthesis of Network Updates
- Efficient Synthesis of Probabilistic Programs
- FlashRelate: Extracting Relational Data from Semi-Structured Spreadsheets Using Examples
- Synthesizing Data Structure Transformations from Input-Output Examples

April 1st – PLDI'15

- LaminarIR: Compile-Time Queues for Structured Streams
- Optimizing Off-Chip Accesses in Multicores
- Improving Compiler Scalability: Optimizing Large Programs at Small Price

April 8th – PLDI'15

- A Formal C Memory Model Supporting Integer-Pointer Casts
- Defining the undefinedness of C
- KJS: A Complete Formal Semantics of JavaScript
- Verdi: A Framework for Formally Verifying Distributed System Implementations

May 6th – PLDI'15

- DAG Inlining: A Decision Procedure for Reachability-Modulo-Theories in Hierarchical Programs
- Exploring and Enforcing Security Guarantees via Program Dependence Graphs
- Making Numerical Program Analysis Fast
- Tree Dependence Analysis

May 13th – PLDI'15

- Automated Detection of Performance Bugs via Static Analysis
- Autotuning Algorithmic Choice for Input Sensitivity
- Helium: Lifting High-Performance Stencil Kernels from Stripped x86 Binaries to Halide DSL Code
- Profile-Guided Meta-Programming

May 20th – ASPLOS'15

- Targeted Automatic Integer Overflow Discovery Using Goal-Directed Conditional Branch Enforcement
- Architectural Support for Software-Defined Metadata Processing
- A Hardware Design Language for Timing-Sensitive Information-Flow Security
- SPECS: A Lightweight Runtime Mechanism for Protecting Software from Security-Critical Processor Bugs

May 27th– MICRO'15

- Enabling Coordinated Register Allocation and Thread-level Parallelism Optimization for GPUs
- Free Launch: Optimizing GPU Dynamic Kernel Launches through Thread Reuse
- Efficient Warp Execution in Presence of Divergence with Collaborative Context Collection
- Control Flow Coalescing on a Hybrid Dataflow/von Neumann GPGPU

June 3rd – CGO'16

- A Black-box Approach to Energy-Aware Scheduling on Integrated CPU-GPU Systems
- Portable and Transparent Software Managed Scheduling on Accelerators for Fair Resource Sharing
- Communication-Aware Mapping of Stream Graphs for Multi-GPU Platforms
- GPUCC: An Open-Source GPGPU Compiler

June 10th – CGO'16

- Cheetah: Detecting False Sharing Efficiently and Effectively
- AutoFDO: Automatic Feedback-directed Optimization for Warehouse-scale Applications
- Portable Performance on Asymmetric Multicore Processors