Reading List

Optimizations & analysis

[Appel 98] Andrew W. Appel. *Modern Compiler Implementation*. Cambridge University Press, Cambridge, UK, 1998.

Data flow analysis

[Tjiang & Hennessy 92] Steven W.K. Tjiang and John L. Hennessy. Sharlit - A Tool for Building Optimizers. In *PLDI '92*.

[Chambers *et al.* 96] Craig Chambers, Jeffrey Dean, and David Grove. Frameworks for Intra- and Interprocedural Dataflow Analysis. UW TR.

Intermediate representations

[Ferrante *et al.* 87] Jeanne Ferrante, Karl J. Ottenstein, and Joe D. Warren. The Program Dependence Graph and Its Use in Optimization. In *TOPLAS 9(3)*, July 1987.

[Alpern *et al.* 88] Bowen Alpern, Mark N. Wegman, and F. Kenneth Zadeck. Detecting Equality of Variables in Programs. In *POPL '88*.

[Weise *et al.* 94] Value Dependence Graphs: Representation Without Taxation. In *POPL '94*.

Inlining and interprocedural optimizations

[Dean & Chambers 94] Jeff Dean and Craig Chambers. Towards Better Inlining Decisions Using Inlining Trials. In *L&FP '94*.

[Callahan *et al.* 86] David Callahan, Keith D. Cooper, Ken Kennedy, and Linda Torczon. Interprocedural Constant Propagation. In *PLDI '86*.

[Grove *et al.* 97] David Grove, Greg DeFouw, Jeffrey Dean, and Craig Chambers. Call Graph Construction in Object-Oriented Languages. In *OOPSLA '97*.

[Consel & Danvy 93] Charles Consel and Olivier Danvy. Partial Evaluation: Principles and Perspectives. Tutorial notes included in *POPL '93*.

Alias and pointer analysis

[Wilson & Lam 95] Robert P. Wilson and Monica S. Lam. Efficient Context-Sensitive Pointer Analysis for C Programs. In *PLDI '95*.

[Steensgaard 96] Bjarne Steensgaard. Points-to Analysis in Almost Linear Time. In *POPL '96*.

Dependence, vectorization, and parallelization

[Padua & Wolfe 86] David A. Padua and Michael J. Wolfe. Advanced Compiler Optimizations for Supercomputers. In *Communications of the ACM 29(12)*, December 1986.

Register allocation

[Briggs *et al.* 94] Preston Briggs, Keith D. Cooper, and Linda Torczon. Improvements to Graph Coloring Register Allocation. In *TOPLAS 16(3)*, May 1994.

[Wall 86] David W. Wall. Global Register Allocation at Link Time. In *PLDI '86*.

Instruction scheduling

[Gibbons & Muchnick 86] Phillip B. Gibbons and Steven S. Muchnick. Efficient Instruction Scheduling for a Pipelined Architecture. In *PLDI '86*.

(Maybe another paper here.)

Functional languages

[Kranz *et al.* 86] David Kranz, Richard Kelsey, Jonathan Rees, Paul Hudak, James Philbin, and Norman Adams. ORBIT: An Optimizing Compiler for Scheme. In *PLDI '86*.

[Tarditi *et al.* 96] David Tarditi, Greg Morrisett, Peter Cheng, Chris Stone, Robert Harper, and Peter Lee. TIL: A Type-Directed Optimizing Compiler for ML. In *PLDI '96*.

(Maybe another paper here.)

Object-oriented languages

[Agesen & Hölzle 95] Ole Agesen and Urs Hölzle. Type Feedback vs. Concrete Type Inference: A Comparison of Optimization Techniques for Object-Oriented Languages. In *OOPSLA '95*. (Or maybe an alternative paper.)

[Dean *et al.* 96] Jeffrey Dean, Greg DeFouw, David Grove, Vassily Litvinov, and Craig Chambers. Vortex: An Optimizing Compiler for Object-Oriented Languages. In *OOPSLA '96*.