Reading List

Standard optimizations & analysis

[Aho *et al.* 86] Alfred V. Aho, Ravi Sethi, and Jeffrey D. Ullman. *Compilers: Principles, Techniques, and Tools*. Addison-Wesley, Reading, MA, 1986.

Data flow analysis

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

Intermediate representations

[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*.

[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*.

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

[Chaitin 82] G. J. Chaitin. Register Allocation & Spilling via Graph Coloring. In *PLDI '82*.

[Chow & Hennessy 84] Frederick Chow and John Hennessy. Register Allocation by Priority-Based Coloring. In *PLDI'84*.

[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*.

Garbage collection

[Wilson 95] Paul R. Wilson. Uniprocessor Garbage Collection Techniques. Submitted to *Computing Surveys*.

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*.

[Shivers 88] Olin Shivers. Control Flow Analysis in Scheme. In *PLDI'88*.

(Probably another paper here.)

Object-oriented languages

[Stroustrup 89] Bjarne Stroustrup. Multiple Inheritance for C++. In *Computing Systems 2(4)*, pp. 367-395, Fall, 1989.

[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*.

[Dean *et al.* 95] Jeffrey Dean, David Grove, and Craig Chambers. Optimization of Object-Oriented Programs Using Static Class Hierarchy Analysis. In *ECOOP '95*.