Sebastian POP

E-mail: sebastian.pop@amd.com
Phone: +1 512 602 6010

Mail:

General

- Software Engineer - Member of Technical Staff - Advanced Micro Devices. August 2007 to present.
- Consultant for Advanced Micro Devices - self employed. May 2007 to August 2007.
- Expert Compiler Engineer - INRIA Futurs - Alchemy Group. October 2006 to May 2007.
- Ph.D. in Computer Science - École nationale supérieure des mines de Paris - CRI - 2006.
- M.Sc. in Computer Science (DEA) - Université Louis Pasteur, Strasbourg - ICPS/LSIIT - 2003.
- M.Sc. in Computer Science (DESS) - Université Louis Pasteur, Strasbourg - ICPS/LSIIT - 2002 - Erasmus student at the University of Liverpool.
- Born in Cluj-Napoca (Romania), October the 14th 1978, French Nationality.

Publications

- Graphite: Towards a Declarative Polyhedral Representation. Sebastian Pop. Workshop on GCC for Research in Embedded and Parallel Systems, GREPS'07, September 16, Brasov, Romania.
- Enabling Interactivity in GCC for Fine-Grain Optimizations. Cupertino Miranda, Grigori Fursin, Sebastian Pop, Albert Cohen. 3rd HiPEAC Summer School, ACACES'07, July 15-20, L'Aquila, Italy.
- Practical Run-time Adaptation with Procedure Cloning to Enable Continuous Collective Compilation. Grigori Fursin, Cupertino Miranda, Sebastian Pop, Albert Cohen, Olivier Temam. GCC Summit 2007, Ottawa, Canada.
- The SSA Representation Framework: Semantics, Analyses and GCC Implementation. Sebastian Pop. Ph.D. thesis, December 2006.
- Denotational Semantics for SSA Conversion. Sebastian Pop, Albert Cohen, Pierre Jouvelot, Georges-André Silber. Research report, June 2006.
- GRAPHITE: Polyhedral Analyses and Optimizations for GCC. Sebastian Pop, Albert Cohen, Cédric Bastoul, Sylvain Girbal, Georges-André Silber, Nicolas Vasilache. GCC Summit 2006, Ottawa, Canada.
- The New Framework for Loop Nest Optimizations in GCC: from Prototyping to Evaluation. Sebastian Pop, Albert Cohen, Pierre Jouvelot, Georges-André Silber. The 12th Workshop on Compilers for Parallel Computers, CPC2006, January 2006, A Coruña, Spain.
- Induction Variable Analysis with Delayed Abstractions. Sebastian Pop, Albert Cohen, Georges-André Silber. First International Conference, High Performance Embedded Architectures and Compilers, HiPEAC2005, November 2005, Barcelona, Spain.
- High-Level Loop Optimizations for GCC. Daniel Berlin, David Edelsohn (IBM T.J. Watson Research Center), Sebastian Pop. GCC Summit 2004, Ottawa, Canada.
- Fast Recognition of Scalar Evolutions on Three-Address SSA Code. Sebastian Pop, Philippe Clauss (ICPS-LSIIT), Albert Cohen (INRIA), Vincent Loechner (ICPS-LSIIT), Georges-André Silber. Research report, October 2004.
- Analysis of Induction Variables Using Chains of Recurrences: Extensions. Sebastian Pop. M.Sc. thesis, June 2003.
- Interface and Extension of the Open Research Compiler. Sebastian Pop. M.Sc. thesis, Sept. 2002.
- Optimisations de code dans GCC. Sebastian Pop. M.Sc. thesis report, July 2002.

Reviews

GREPS'07, CASES'07, PPoPP'07, Parallel Computing Journal.

Teaching

- helped organisation of a HiPEAC Summer School Lecture: "Compiler Optimizations Research in GCC"
- working with a Google Summer of Code 2006 (Alexandru Plesco) on polyhedral transforms in GCC.
- helped organisation of "1st HiPEAC GCC Tutorial: Middle-end and Back-end Program Manipulation".
- working with a STMicroelectronics group (Erven Rohou, Roberto Costa) on a CIL back-end for GCC.
- worked with a Philips group (Alexandru Turjan, Sjoerd Meijer) on generating parallel code from GCC.
- UNIX (GNU/Linux) user, administration, shell scripting, Zope and Plone.
- Supervised a student (Olivier Trichot) working on a spatial locality optimizing compiler.

Domains of Interest

- Programming languages, compilers construction, parallel programming,
- Formal methods, discrete mathematics.

Languages

English, French, Romanian, Deutsch (notions).

Projects

Temporal locality optimization (2001)

Implementation of an algorithm designed at *ICPS* (Image and Parallel Scientific Computing) by Philippe Clauss, Vincent Loechner and Benoît Meister. In this project Frédéric Wagner and I have implemented an algorithm for optimizing data references locality in nested loops. The source to source program transformer has been developed as a module of the *Nestor* Fortran source to source compiler. One of the main specificities of this algorithm is that it uses a polyhedral representation of the data accesses in loop nests. The *PolyLib*, a polyhedral library developed at *ICPS*, is used to analyze the data accesses and to propose a better access order for increasing the reuse of data elements. The validation of the transformation is based on the *Omega* data dependence analyzer.

Towards high level optimizers in GCC (2001 - 2002)

One of the most important tool for implementing transformations on an intermediate representation is the pretty-printer. At the beginning of the *AST-optimizer-branch* of GCC, the tree pretty printer was limited to the printing of the internal structures that contain too much information for source to source transformations. The first step consisted in the implementation of a pretty printer for the C front-end that was able to print the tree representations into a C-like code.

In November 2001 I have tried to implement an analysis that extracts the data access functions from the tree representation of a C program in order to perform the data dependence tests. In this process I have discovered the extreme complexity of analyzing the unstructured abstract syntax trees. This complex task was described in one of the papers of Laurie Hendren, and the proposed solution was to reduce the expression composition into a three-address code using temporary variables.

My first contribution to the GCC project was the initial version of the C pretty printer and the initial version of the simplifier. This work has been included in the *AST-optimizer-branch* in January 2002 by Diego Novillo, one of the developers of GCC that helped me in the implementation of these first versions.

Internship on Open64/ORC (INRIA - summer 2002)

During this internship I have discovered the *Open64* and *Open Research Compiler*. This work has been supervised by Albert Cohen in the *A3* project at *INRIA* (French Institute for research in computer science). The second part of this internship was the implementation of a pass of partial redundancy elimination on predicated code using the back-end of the ORC compiler.

Analysis of scalar evolutions and data dependences (ICPS - ENSMP - 2003)

The aim of this project has been the study of the previous works on the analysis of scalar evolutions. I have proposed several extensions to the previous research works, and new algorithms for extracting the scalar evolutions informations from programs represented in a SSA form. I have implemented one of the proposed algorithms using the *tree-SSA* branch of GCC. The number of iterations in a loop is computed as part of the scalar evolution analyzer, opening the way to other loop optimizations that use this information, such as the loop unroller. The data dependence extracts the relations from the evolution functions of the array access variables. All these analyzers are used by the auto-vectorizer contributed to GCC by Dorit Nuzman from IBM Haifa.

Internship (IBM T.J.Watson Yorktown Heights - summer 2004)

From June to September 2004, I have worked in the Linux Technology Center at IBM with David Edelsohn and Daniel Berlin on high level loop nest optimizations, and on the integration of this new infrastructure in the production branch 4.0.0 of GCC. This shows in practice the scalability of the algorithms that I have proposed and implemented.

Ph.D thesis (CRI - ENSMP (2003 - 2006))

In my thesis I'm proposing for the first time a formal definition for the SSA language, an intermediate form that is commonly used in all the industrial compilers for imperative languages. I'm also providing a proof of the correctness of the translation from an imperative language to SSA form, and I provide different techniques for defining static analyzers on the SSA form.