

Biblio CR10

Laure Gonnord & David Monniaux

17 octobre 2016

A useful link : <http://www.complang.tuwien.ac.at/nino/cs-bibles/bibtex.html>

1 Course #1 : Introduction, Model Checking, Program Analysis

Biblio

- On Program Analysis
- On Binary Decision Diagrams
- On Program Analysis with SAT/SMT

2 Course # 2 : Data Flow, Abstract Interpretation, Numerical Domains

Biblio

- Dataflow analyses in compilation :
- A book about program Analysis, quite formal [23].
- The seminal paper in AI[7], ranges[], polyhedra[8]
- Abstract interpretation of probabilistic programs : [20]
- Acceleration [3] Abstract acceleration : [12]
- AI + SMT : [22]
- See the intro of [9] for a basic crash course on AI with polyhedra. Same for applications in [21].

3 Cours #3 : Data Structures in Abstract Interpretation

Biblio

- Filters
- Octagons [19]

4 Course #4 : Astree and Worst-Case Execution Time

Biblio

- Synchronous languages : Lustre [14], Esterel [4], Signal [11]. Analyses of synchronous languages : [10]

- Astree [6], ellipsoids, trace abstract domain, ...
- Real time estimation in a book (fr) about real-time systems [18]
- Cache analyses : [25], [2]
- WCET with smt [15]

5 Course #5 : termination and compilation

Biblio

- Hoare and all that in the Nielson Book[23]
- SMT-solving is described in [16, 17]. If you are interested in the nitty-gritty of SAT-solving, read [5].
- Termination : synthesis of multidimensional ranking functions[1] ans [13]
- SSA, SSI. [24] to appear. Type “ssa book” to find a pdf draft on internet.

Références

- [1] Christophe Alias, Alain Darte, Paul Feautrier, and Laure Gonnord. Multi-dimensional Rankings, Program Termination, and Complexity Bounds of Flowchart Programs. In *Static Analysis Symposium*, Perpignan, France, September 2010.
- [2] Sebastian Altmeyer, Robert I. Davis, and Claire Maiza. Improved cache related pre-emption delay aware response time analysis for fixed priority pre-emptive systems. *Real-Time Systems*, 48(5) :499–526, 2012.
- [3] S. Bardin, A. Finkel, J. Leroux, and L. Petrucci. Fast : Fast acceleration of symbolic transition systems. In *CAV’03*, pages 118–121, Boulder (Colorado), July 2003. LNCS 2725, Springer-Verlag.
- [4] Gérard Berry and Georges Gonthier. The esterel synchronous programming language : design, semantics, implementation. *Science of Computer Programming*, 19(2) :87 – 152, 1992.
- [5] Armin Biere, Marijn J. H. Heule, Hans van Maaren, and Toby Walsh, editors. *Handbook of Satisfiability*, volume 185. IOS Press, February 2009.
- [6] B. Blanchet, P. Cousot, R. Cousot, J. Feret, L. Mauborgne, A. Miné, D. Monniaux, and X. Rival. A static analyzer for large safety-critical software. In *PLDI 2003, ACM SIGPLAN SIGSOFT Conference on Programming Language Design and Implementation*, pages 196–207, San Diego (Ca.), June 2003.
- [7] P. Cousot and R. Cousot. Abstract interpretation : a unified lattice model for static analysis of programs by construction or approximation of fixpoints. In *4th ACM Symposium on Principles of Programming Languages, POPL’77*, Los Angeles, January 1977.
- [8] P. Cousot and N. Halbwachs. Automatic discovery of linear restraints among variables of a program. In *5th ACM Symposium on Principles of Programming Languages, POPL’78*, Tucson (Arizona), January 1978.
- [9] Laure Danthony, gonnord. *Abstract Acceleration to improve precision in Linear Relation Analysis*. Theses, Université Joseph-Fourier - Grenoble I, October 2007.
- [10] Paul Feautrier, Abdoulaye Gamatié, and Laure Gonnord. Enhancing the compilation of synchronous dataflow programs with a combined numerical-boolean abstraction. *CSI Journal of Computing*, 1(4) :8 :86 – 8 :99, 2012.

- [11] Abdoulaye Gamatié. *Designing Embedded Systems with the SIGNAL Programming Language - Synchronous, Reactive Specification*. Springer, 2010.
- [12] Laure Gonnord and Nicolas Halbwachs. Combining Widening and Acceleration in Linear Relation Analysis. In *Static Analysis (SAS 2006)*, pages 144–160, Seoul, South Korea, August 2006.
- [13] Laure Gonnord, David Monniaux, and Gabriel Radanne. Synthesis of ranking functions using extremal counterexamples. In *Programming Languages, Design and Implementation*, Portland, Oregon, United States, June 2015.
- [14] Nicolas Halbwachs, Paul Caspi, Pascal Raymond, and Daniel Pilaud. The synchronous data flow programming language lustre. *Proceedings of the IEEE*, 79, 1991.
- [15] Julien Henry, Mihail Asavoaie, David Monniaux, and Claire Maiza. How to compute worst-case execution time by optimization modulo theory and a clever encoding of program semantics. In Youtao Zhang and Prasad Kulkarni, editors, *LCTES*, pages 43–52. ACM, 2014.
- [16] Daniel Kroening and Ofer Strichman. *Decision Procedures : An Algorithmic Point of View*. Springer, 1 edition, May 2008.
- [17] Daniel Kroening and Ofer Strichman. *Decision Procedures : An Algorithmic Point of View*. Springer, 2 edition, November 2016.
- [18] Claire Maiza, Christine Rochange, and Pascal Raymond. Estimation de temps d’exécution et délais. In *Ordonnancement dans les systèmes temps réel*, chapter 13, pages 365–392. ISTE editions, 2014.
- [19] A. Miné. The octagon abstract domain. *Higher-Order and Symbolic Computation (HOSC)*, 19(1) :31–100, 2006. <http://www-apr.lip6.fr/~mine/publi/article-mine-HOSC06.pdf>.
- [20] David Monniaux. Backwards abstract interpretation of probabilistic programs. In *European Symposium on Programming Languages and Systems (ESOP '01)*, number 2028 in Lecture Notes in Computer Science, pages 367–382. Springer Verlag, 2001.
- [21] David Monniaux. *Analyse statique : de la théorie à la pratique*. Habilitation to direct research, Université Joseph Fourier, Grenoble, France, July 2009.
- [22] David Monniaux and Laure Gonnord. Using Bounded Model Checking to Focus Fixpoint Iterations. In Eran Yahav, editor, *Static analysis symposium (SAS)*, volume 6887 of *Lecture notes in Computer Science*, pages 369–385, Venezia, Italy, September 2011. Springer.
- [23] Flemming Nielson, Hanne R. Nielson, and Chris Hankin. *Principles of Program Analysis*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 1999.
- [24] Sebastian Pop and Albert Cohen. SSA-based Compiler Design. In Fabrice Rastello, editor, *SSA-based Compiler Design*, volume Loop tree and induction variables. springer, August 2016. ISBN 978-1-4419-6201-0.
- [25] Joerg Schneider. Cache and pipeline sensitive fixed priority scheduling for preemptive real-time systems. In *Real-Time Systemes Symposium*, 2000.