

**Tomofumi Yuki**  
INRIA Chargé de Recherche  
IRISA  
263 Avenue Général Leclerc  
35000 Rennes, FRANCE  
*tomofumi.yuki@inria.fr*  
(+33) 02 99 84 74 89

## Research Interests

Static analysis techniques for parallelization (more broadly, high performance computing). Modeling trade-offs between different metrics of performance (e.g., energy and throughput). Synthesis of efficient hardware from high-level specifications. Optimizations based on semantic knowledge, e.g., associativity/commutativity and/or approximations. Programming model/language to communicate user knowledge to the compiler for performing all of the above.

## Education

Doctor of Philosophy, Computer Science Colorado State University, Fort Collins, CO	Defended Fall 2012
Master of Science, Computer Science Colorado State University, Fort Collins, CO	December 2009
Bachelor of Science, Computer Science Colorado State University, Fort Collins, CO	May 2007

## Dissertation Topic

MPI Code Generation in the Polyhedral Model. We proposed a method for automatic parallelization targeting distributed memory architecture in the polyhedral model. The parallelization is based on *parametric* tiling, where many of the existing techniques for polyhedral programs are not directly applicable due to non-affine nature of parametrized tiling. The core idea is in utilizing uniformization/localization techniques that are rarely used in the multi-core era for simplifying communication.

## Memberships

ERCIM “Alain Bensoussan” Fellow	Mar. 2013 – Mar. 2014
---------------------------------	-----------------------

## Professional Experience

Chargé de Recherche, INRIA (CAIRN team)	IRISA	Nov. 2016 – present
Chargé de Recherche, INRIA (CompSys team)	LIP/ENS Lyon	Oct. 2014 – Nov. 2016
Post-doc, University of Rennes 1 (CAIRN team)	IRISA	July 2014 – Sep. 2014
INRIA Post-doctoral Fellow (CAIRN team)	IRISA	Nov. 2012 – June 2014
Research Assistant	Colorado State University	Spring 2010 – 2012
Summer Internship	IBM Research T.J. Watson Lab	May-Aug. 2008
Teaching Assistant	Colorado State University	2007-2009, Spring 2011

## Professional Activities

### Program Committees

- Program Committee member SuperComputing (SC16, SC17)
- Program Committee member for International Workshop on Polyhedral Compilation Techniques (IMPACT '15, '17, '18)
- Co-chair of the 6th International Workshop on Polyhedral Compilation Techniques (IMPACT 2016)
- Program Committee member for Reliable Software Technologies and Communication Middleware Track of 31st Symposium on Applied Computing (RST Track, SAC 2016)
- Publicity co-chair of the 14th International Conference on Generative Programming: Concepts & Experience (GPCE 2015)

### Teaching

- Lecture at Ecole Jeunes Chercheurs en Programmation (EJCP 2015-2017), “Research in Compilers and How it Relates to Software Engineering”
- M2 Course at ENS Lyon, “Advanced Compilers: Loop Transformations and High-Level Synthesis”, Fall 2015

### On-going Phd

- Phu Ha-Van, “Application-Level Tuning of Accuracy”, 2017-, co-advising with Olivier Sentieys
- Thibaut Marty, “Compiler support for speculative custom hardware accelerators”, 2017-, co-advising with Steven Derrien

### Master Projects

- Thibaut Marty, “Exploitation de la spéculation temporelle pour réseaux de neurones convolutifs”, 2017, co-advised with Steven Derrien
- Julien Versaci, “Optimisation et parallélisation OpenMP d’un programme calculant la section efficace d’un processus 6 photons en théorie quantique des champs”, 2016, co-advised with Jean-Philippe Guillet

## Projects

**AlphaZ** The AlphaZ system is an open-source system for polyhedral analyses, transformations, and code generators. The system models reductions (associative and commutative operator applied to a set of points) as first-class objects, enabling powerful reasoning capabilities that can lead to equivalent programs with reduced asymptotic complexity. Another strength of the system is that it completely decouples the specification of the computation and how it should be executed (i.e., schedule, memory allocation).

**Analysis and Transformations for High-Level Synthesis** High-Level Synthesis is based on the idea of improving hardware design productivity by allowing behavioral specification of the desired hardware in higher level languages (often C) and automatically generating corresponding hardware description. However, the input program needs to be tailored in order to obtain efficient hardware. This gives rise for the need to develop new analyses and transformations that bridge compilers and hardware design, which is one of the areas that I have been working on in the recent years.

**Verification of Parallel Programs** We have developed a tool that verifies the legality of OpenMP pragmas using polyhedral machinery as a Eclipse plug-in. Recently, we have been collaborating with Dr. Vijay Saraswat, one of the core members involved in the X10 language development, to provide safety guarantee of X10 programs. Parallel constructs in X10 can express parallelism that cannot be expressed in OpenMP, or by the conventional polyhedral notion of time. This project entails extending the polyhedral notion of time such that any possible in-deterministic behavior introduced by parallel constructs can be found.

## Peer Reviewed Publications

1. Gaël Deest, Tomofumi Yuki, Sanjay Rajopadhye, and Steven Derrien, “One Size Does Not Fit All: Implementation Trade-offs for Iterative Stencil Computations on FPGAs”, 27th International Conference on Field-Programmable Logic and Applications, 2017
2. Xinyu Niu, Nicholas Ng, Tomofumi Yuki, Shaojun Wang, Nobuko Yoshida, and Wayne Luk, “EURECA compilation: Automatic optimisation of cycle-reconfigurable circuits”, 26th International Conference on Field-Programmable Logic and Applications, 2016
3. Alain Darte, Alexandre Isoard, and Tomofumi Yuki, “Liveness Analysis in Explicitly-Parallel Programs”, 6th International Workshop on Polyhedral Compilation Techniques, January 19, 2016
4. Gaël Deest, Nicolas Estibals, Tomofumi Yuki, Steven Derrien, and Sanjay Rajopadhye, “Towards Scalable and Efficient FPGA Stencil Accelerators”, 6th International Workshop on Polyhedral Compilation Techniques, January 19, 2016
5. Tomofumi Yuki, “Revisiting Loop Transformations with X10 Clocks”, X10 Workshop 2015, June 14, 2015
6. Gaël Deest, Tomofumi Yuki, Olivier Sentieys, and Steven Derrien “Toward Scalable Source Level Accuracy Analysis for Floating-point to Fixed-point Conversion”, 33rd International Conference on Computer-Aided Design, November 3-6, 2014
7. Tomofumi Yuki “Understanding PolyBench/C 3.2 Kernels” (position paper), 4th International Workshop on Polyhedral Compilation Techniques (IMPACT’14), Vienna, Austria, January, 2014
8. Tomofumi Yuki, Antoine Morvan, and Steven Derrien “Derivation of Efficient FSM from Loop Nests” 2013 International Conference on Field-Programmable Technology, December 9-11, 2013
9. Tomofumi Yuki and Sanjay Rajopadhye “Folklore Confirmed: Compiling for Speed = Compiling for Power” 26th International Workshop on Languages and Compilers for Parallel Computing, September 25-27, 2013
10. Antoine Floc’h, Tomofumi Yuki, Ali El-Moussawi, Antoine Morvan, Kevin Martin, Maxime Naullet, Mythri Alle, Ludovic L’Hours, Nicolas Simon, Steven Derrien, François Charot, Christophe Wolinski, and Olivier Sentieys “GeCoS: A Framework for Prototyping Custom Hardware Design Flows” 13th IEEE International Working Conference on Source Code Analysis and Manipulation
11. Tomofumi Yuki, Paul Feautrier, Sanjay Rajopadhye and Vijay Saraswat “Array Dataflow Analysis for Polyhedral X10 Programs” 18th ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming, February 23-27, 2013
12. Tomofumi Yuki, Sanjay Rajopadhye “Memory Allocations for Tiled Uniform Dependence Programs” 3rd International Workshop on Polyhedral Compilation Techniques, January 21, 2013
13. Tomofumi Yuki and Sanjay Rajopadhye “AlphaZ and the Polyhedral Equational Model” Second International Workshop on Domain-Specific Languages and High-Level Frameworks for High Performance Computing, November 2012
14. Tomofumi Yuki, Gautam Gupta, DaeGon Kim, Tanveer Pathan, and Sanjay Rajopadhye. “AlphaZ: A System for Design Space Exploration in the Polyhedral Model”, 25th International Workshop on Languages and Compilers for Parallel Computing, September 11-13, 2012

15. Antoine Floch, Tomofumi Yuki, Celement Guy, Steven Derrien, Benoit Combemale, Sanjay Rajopadhye, and Robert France. “Model-driven engineering and optimizing compilers: a bridge too far?”, International Conference on Model Driven Engineering Languages and Systems, October 16-21, 2011
16. Vamshi Basupalli, Tomofumi Yuki, Sanjay Rajopadhye, Antoine Morvan, Steven Derrien, Patrice Quinton, and David G. Wonnacott. “ompVerify: Polyhedral Analysis for the OpenMP Programmer”, International Workshop on OpenMP, June 13-15, 2011
17. Tomofumi Yuki, Lakshminarayanan Renganarayanan, Sanjay Rajopadhye, Charles Anderson, Alexandre E. Eichenberger, and Kevin O’Brien. “Automatic Creation of Tile Size Selection Models”, International Symposium on Code Generation and Optimization, April 24-28, 2010

## Technical Reports

1. Tomofumi Yuki, Paul Feautrier, Sanjay Rajopadhye, and Vijay Saraswat, “Checking Race Freedom of Clocked X10 Programs”, Technical Report arXiv:1311.4305, 2013
2. Tomofumi Yuki and Sanjay Rajopadhye. “Diminishing Returns of Frequency Scaling”, Technical Report CS-13-107, August 27, 2013
3. Tomofumi Yuki and Sanjay Rajopadhye. “Parametrically Tiled Distributed Memory Parallelization of Polyhedral Programs”, Technical Report CS-13-105, June 10, 2013
4. Sanjay Rajopadhye, Guillaume Iooss, Tomofumi Yuki, and Dan Connors. “The Stencil Processing Unit: GPGPU Done Right”, Technical Report CS-13-103, March 1, 2013
5. Tomofumi Yuki, Tanveer Patahan, Gautam Gupta, and Sanjay Rajopadhye. “Systematic Implementation of fast-i-loop in UNAFold using AlphaZ”, Technical Report CS-12-102, May 31, 2012
6. Tomofumi Yuki, Vamshi Basupalli, Gautam Gupta, Guillaume Iooss, DaeGon Kim, Tanveer Pathan, Pradeep Srinivasa, Yun Zou and Sanjay Rajopadhye. “AlphaZ: A System for Analysis, Transformation, and Code Generation in the Polyhedral Equational Model”, Technical Report CS-12-101, May 31, 2012
7. Tomofumi Yuki and Sanjay Rajopadhye. “Canonic Multi-Projection: Memory Allocation for Distributed Memory Parallelization”, Technical Report CS-11-106, September 20, 2011