



## Preface to Hagiya-*sensei*'s 2<sup>6</sup>-th Birthday Special Issue

Shin-ya Nishizaki<sup>1</sup> · Shinnosuke Seki<sup>2</sup>

Published online: 21 July 2022

© Ohmsha, Ltd. and Springer Japan KK, part of Springer Nature 2022



Professor Masami Hagiya, The University of Tokyo

This special issue is dedicated to Professor Masami Hagiya on the occasion of his retirement from the University of Tokyo and his 2<sup>6</sup>-th birthday, in recognition of his outstanding contributions to programming, reasoning, DNA computing, and molecular computing.

Masami was born in 1957 in Tokyo, Japan. After receiving his M. Sc. from the University of Tokyo in 1982, he worked for the Research Institute for Mathematical Sciences, Kyoto University, as an assistant professor. Subsequently, he received a Dr. Sc. in 1988 and was promoted to associate professor. In 1993, he moved to the Department of Computer Science, the University of Tokyo, and was promoted to full professor in 1995.

---

✉ Shin-ya Nishizaki  
nisizaki@c.titech.ac.jp

<sup>1</sup> Tokyo Institute of Technology, 2-12-1-W8-69 Ookayama, Meguro, Tokyo 152-8552, Japan

<sup>2</sup> University of Electro-Communications, 1-5-1 Chofugaoka, Chofu, Tokyo 182-8585, Japan

He has been working on programming language theory and formalization and verification of computer systems. He has been dealing not only with systems composed of electronic computers, but also with biological and molecular systems, and has been working on DNA and molecular computing.

He has developed various language processors for various programming languages, namely symbolic processing languages represented by Lisp, functional languages idealized from Lisp, logic programming languages represented by Prolog, and object-oriented languages represented by Java. He has also conducted extensive research on the semantics of programming languages, type theory, formal system verification, and other fundamental studies in computer science.

As a student at the University of Tokyo, Masami was involved in the implementation of Hyperlisp with Masahiko Sato. As an assistant professor at Kyoto University, he focused on the development of Kyoto Common Lisp, which fully follows the Common Lisp specification, with Taichi Yuasa. He also participated in the development of the GMW window system and designed the G language, a concurrent object-oriented language that anticipated Java. In theoretical computer science, he has made numerous contributions in type theory, higher-order unification, partial evaluation, abstract interpretation, bytecode verification, and model checking. He also proposed “Proving-as-Editing,” a paradigm that combines document creation and programming.

In recent years, he has also been motivated to “program” natural systems, especially molecular and cellular systems, and has continued his research with the goal of constructing general-purpose programmable molecular systems, mainly using DNA molecules.

We have known Masami for many years and we have had the chance to collaborate with him and appreciate his intellectual standard, his remarkable insight, his notable software design and coding skills, and his excellent sense of humor.

In the open call for papers, manuscripts were solicited in areas of computer science, with special emphasis on those close to his research interests, including programming language theory and implementation, software verification, DNA computing, and molecular programming. After a rigorous peer-review process, 12 excellent papers were accepted, and we have organized them into the following two parts of this special issue.

Part 1 covers the topics of programming and reasoning.

Bisimilarity is one of the most important notions in process theory. “Codensity Games for Bisimilarity” by Yuichi Komorida, Shin-ya Katsumata, Nick Hu, Bartek Klin, Samuel Humeau, Clovis Eberhart, and Ichiro Hasuo gives a general categorical game characterization of bisimilarity, which covers known bisimilarity-like notions.

Reversible computing is an unconventional computing paradigm, in which there is both forward and reverse determinism. In “Making Programs Reversible with Minimal Extra Data”, Robert Glück and Tetsuo Yokoyama, present their study of one of the important issues of reversible computing, the existence of reversible programs with minimal extra output, and propose new methods comparing existing ones.

Pattern matching is a basic programming construct that is considered important for data abstraction in various programming paradigms. In “Embedding Non-linear Pattern Matching with Backtracking for Non-free Data Types into Haskell”, Satoshi Egi, Akira Kawata, Mayuko Kori, and Hiromi Ogawa report their study of a new method that allows us to introduce the pattern-match facility proposed by Egi and Nishiwaki into a statically typed language using meta-programming in Haskell.

Blockchain technology is a core technology that supports cryptocurrencies in a smart contract and is a program executed on a blockchain. In “Helmholtz: A Verifier for Tezos Smart Contracts Based on Refinement Types”, Yuki Nishida, Hiromasa Saito, Ran Chen, Akira Kawata, Jun Furuse, Kohei Suenaga, and Atsushi Igarashi introduce a type-based static verification tool, Helmholtz, for the statically typed stack-based language Michelson for writing smart contracts that are executed on the blockchain platform Tezos. They explain its design and present successful examples of verification using Helmholtz.

Part 2 of the book covers the topic of programming in the biologically conventional media called *molecules*. Molecular engineering has already advanced enough to allow us to design and synthesize molecules in such a way that they interact with each other as well as with natural molecules so as to achieve various computational tasks. A fundamental drive of its advancement is the ideal fusion of theory and practice: theoretical models facilitate experiments by reducing errors and costs, speeding up development time, and so on, while experiments in practice incorporate constraints into models.

Cellular automaton (CA) should be considered the first computational model inspired by nature and has brought various models into life including the abstract tile assembly model (aTAM). Isotropy and conservation of resources are significant constraints, and Hisamichi Ishizaka, Gil-Tak Kong, and Katsunobu Imai show in their article “The weak universality of two-dimensional five-state von Neumann neighborhood number-conserving cellular automaton” how to simulate logic circuits by an isotropic CA with 5 states—0, 1, 2, 3, and 4—such that the sum of these integral states never changes. Andrew Alseth, Jacob Hendricks, Matthew J. Patitz, and Trent A. Rogers present the results of their study “Replication of arbitrary hole-free shapes via self-assembly with signal-passing tiles” a variant of aTAM in which molecular tiles can propagate signals to change their reactivity in a restricted manner. In this model, they propose a system to replicate a given hole-free shape without any scaling in an exponential manner.

Chemical reaction network (CRN) and reaction system (R system) model a network of molecules and its dynamics via reactions among them. Chun-Hsiang Chan, Cheng-Yu Shih, and Ho-Lin Chen introduce in their article “On the computational power of phosphate transfer reaction networks” a constraint to CRNs being motivated by phosphate transfer and study the computational power of the resulting subclass called PTRNs. The work by Ryutaro Yako, Daihei Ise, Ken Komiya, Kenzo Fujimoto, and Satoshi Kobayashi reported in “Monotone control of R systems” introduces the monotonicity of the control in R system and shows that any control on an R system can be simulated by monotone control with just two control symbols. In “Chemical reaction regular grammars”, Fumiya Okubo, Kaoru Fujioka, and

Takashi Yokomori show chemical reactions rather as an acceptor and generator of a set of strings and propose a chemical reaction regular grammar (CRRG).

These theoretical works on a network of interactions between molecules are complemented by experimental works. “DNA concentration regulator that can be driven for a long time” by Takashi Nakakuki, Keiji Murayama, and Hiroyuki Asanuma tackles the problem of how to enable a molecular robot operate for a long time. They propose a methodology to provide a reaction scheme of strand-displacement-based DNA concentration regulator (DCR) with the capability to restore the initial concentration of its gate and fuel strands. “Toggling between two limit cycles in a molecular ecosystem” by Adrien Fauste-Gay, Nicolas Lobato-Dauzier, Alexandre Baccouche, Yannick Rondelez, Soo Hyeon Kim, Teruo Fujii, Nathanael Aubert-Kato, and Anthony J. Genot describes a system with two independent limit cycles implemented by combining a predator–prey oscillator (Lotka–Volterra oscillator) and a bistable switch to achieve cross inhibition. The system exhibits rich enough behaviors to serve as a basis for computing. Last but not the least, Ibuki Kawamata, Shin-ichiro M. Nomura, and Satoshi Murata propose in “Autonomous and programmable strand generator implemented as a DNA and enzymatic chemical reaction cascade” a system that generates single-stranded DNAs as programmed in an input single-stranded DNA. The order in which the strands are generated can be also thus programmed.

The papers submitted to this special issue have been kindly reviewed by Tatsuya Abe, James Aspnes, Ryohei Banno, Luca Cardelli, Da-Jung Cho, Corina Cirstea, Martin Delacourt, David Doty, Kotaro Funakoshi, Shigeki Hagihara, Masahiro Hamano, Koji Hasebe, Soichiro Hidaka, Jarkko Kari, Eli Kengmana, Jongmin Kim, Barbara Koenig, Steffen Kopecki, Hieu Hanh Le, Luca Manzoni, Sosuke Moriguchi, Tetsuya Murai, Keisuke Nakano, Susumu Nishimura, Ion Petre, Rebecca Schulman, Robert Schweller, Masaya Shimakawa, Darko Stefanovici, Izumi Takeuti, Takaaki Tateishi, Seiki Ubukata, György Vaszil, Takuo Watanabe, and Mitsuharu Yamamoto. We take this opportunity to express our sincere gratitude to them for their cordial support to ensure the scientific value of this special issue and enhance it further by their valuable comments.

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.