Randomized Gandy-Păun-Rozenberg machines

Adam Obtułowicz
Institute of Mathematics, Polish Academy of Sciences
Warsaw, Poland

Mathematics is a powerful tool that helps people achieve new goals as well as understand what is impossible to do.
Mark Burgin

Foundations

The paper *Logically Possible Machines* by Eric Steinhart (*Mind and Machines* 12 (2002), pp. 259–280) contains a discussion of logical foundations of computation theory including quantum computation which gives rise to the following family of questions:

(?) what is it an $\mathcal{X}$ possible machine?

for $\mathcal{X} \in \{$set-theoretically, discrete topologically, continuous topologically, geometrically, biologically inspired, physically, cognitive and intelligent$\}$.

Gandy's machine

We point out here that Robin Gandy's machines (cf. Gandy's paper *Church Thesis and Principles of Mechanisms* in: *The Kleene Symposium*, ed. J. Barwise et al., 1980, pp. 123–148) yield some answer to (?) for $\mathcal{X} \equiv$ set-theoretically in discrete case. The physically possible machines are discussed in the papers about physical limitations of computing devices by Scott Aaronson, Jacob Bekenstein, Charles H. Bennett, Rolf Landaurer, Stockmeyer and Meyer, among others.

Gandy's machine

We point out here that Robin Gandy's machines (cf. Gandy's paper *Church Thesis and Principles of Mechanisms* in: *The Kleene Symposium*, ed. J. Barwise et al., 1980, pp. 123–148) yield some answer to (?) for $\mathcal{X} \equiv$ set-theoretically in discrete case. The physically possible machines are discussed in the papers about physical limitations of computing devices by Scott Aaronson, Jacob Bekenstein, Charles H. Bennett, Rolf Landaurer, Stockmeyer and Meyer, among others.
The paper *Can Cognitive and Intelligent Systems Outperform Turing Machines* by Jiři Wiedermann inspired to formulate (?) for $\mathcal{X} \equiv$ cognitive and intelligent.

G–P–R machine

An idea of a Gandy–Păun–Rozenberg machine, briefly G–P–R machine, is aimed to provide an answer to (?) for
$\mathcal{X} \equiv$ set-theoretically,
$\mathcal{X} \equiv$ discrete topologically, and
$\mathcal{X} \equiv$ biologically inspired.

G–P–R machine

The G–P–R machines are the constructs which have common features with or are related to:

- Gandy's machines,

G–P–R machine

The G–P–R machines are the constructs which have common features with or are related to:

- Gandy's machines,
- P systems due to Gheorghe Păun (cf. *Membrane Computing. An Introduction*, Berlin 2002),

G–P–R machine

The G–P–R machines are the constructs which have common features with or are related to:

- Gandy's machines,
- P systems due to Gheorghe Păun (cf. *Membrane Computing. An Introduction*, Berlin 2002),
- parallel rewriting systems of graphs investigated by Grzegorz Rozenberg himself with scientists cooperating with him, among others, in preparation of multi-volume *Handbook of graph grammars and computing by graph transformation*.

Core

The core of a G–P–R machine is a finite set of rewriting rules for certain finite directed labelled graphs, where these graphs are instantenous descriptions for the computation process realized by the machine.

Parallelism

The conflictless parallel (simultaneous) application of the rewriting rules of a G–P–R machine is realized in Gandy's machine mode (according to Local Causality Principle), where (local) maximality of "causal neighbourhoods" replaces (global) maximality of, e.g. conflictless set of evolution rules applied simultaneously to a membrane structure which appears during the evolution process generated by a P system. Therefore one can construct a Gandy's machine from a G–P–R machine in an immediate way.

NP problems

The NP complete problems can be solved by G–P–R machines in a polynomial time (but with an exponential number of indecomposable processors), where one constructs the G–P–R machines solving these problems in a polynomial time in a similar way to (families of) P systems solving these problems also in a polynomial time (cf. the pioneering Păun's paper *P systems with active membranes: Attacking NP complete problems*, Journal of Automata Languages and Combinatorics 6 (2000), pp. 75–90).

Goal of randomization

Randomized G–P–R machines are aimed to solve NP problems in a polynomial time with subexponential number of indecomposable processors and with possibly small error probability.

Randomization

Randomization of a G–P–R machine consists in a modification of initial instantaneous descriptions of the machine by introducing those randomly chosen configurations to them which correspond to (even simultaneously many) randomly chosen instances of input data.

Randomization

Randomization of a G–P–R machine consists in a modification of initial instantaneous descriptions of the machine by introducing those randomly chosen configurations to them which correspond to (even simultaneously many) randomly chosen instances of input data.
These randomly chosen configurations cause an uncertainty of the final result of the machine computation which is measured by an error probability.

Computation phases

Computation of a randomized G–P–R machine (for 3-SAT problem) consists of two phases:

Computation phases

Computation of a randomized G–P–R machine (for 3-SAT problem) consists of two phases:

- an assembly phase of a distributed system (biologically inspired),

Computation phases

Computation of a randomized G–P–R machine (for 3-SAT problem) consists of two phases:

- an assembly phase of a distributed system (biologically inspired),
- a massively parallel computation phase realized by this distributed system.

Future investigations

Characterization of NP complexity class in terms of randomized G–P–R machines to link Heisenberg Uncertainty Principle with $P \neq NP$ hypothesis like C. S. Calude linked this Principle with undecidability in *From Heisenberg to Gödel via Chaitin* in Internat. J. of Theor. Physics vol. 46, no. 8 (2007).

Future investigations

Simulation of quantum computer computations by the computations of randomized G–P–R machines.
The relations between G–P–R machines and computability in Bounded Set Theory (V. Yu. Sazonov).

Hypothesis

Computations realized by multi-dimensional, multi-face graph rewriting systems (even cellular automata) reduce (with polynomial delay) to computations of G–P–R machines.