Some Quick Research Topics

Gheorghe Păun

Institute of Mathematics of the Romanian Academy
PO Box 1-764, 014700 Bucureşti, Romania
gpaun@us.es, curteadelaarges@gmail.com

Summary. Some research topics are suggested, in a preliminary form, in most cases dealing with (somewhat nonstandard) extensions of existing types of P systems.

1 Introduction

Almost at every edition of the Brainstorming, lists of open problems and research topics were circulated – the present note should be seen as a step in this tradition, part of a ritual. The interested reader can check the Brainstorming volumes, or, for a more systematic list of research suggestions, (s)he should consult the "mega-paper" [7]. Of course, many open problems and research topics can be found in [11] and at the domain web site [14].

The list which follows contains several suggestions (this time I do not count them...) which might look strange at the first sight, but which have their motivation, they are natural at least from a mathematical point of view. Just one example: multisets with negative multiplicity seems to be artificial objects, but they appear already in computer science, see, e.g., [2].

Of course, the reader is assumed to be familiar with membrane computing, so that the presentation is minimal, both in what it concerns the details and the references.

2 "Negative" Extensions

It is about negative numbers, as already mentioned above...

In several places in membrane computing we have functions $f : X \longrightarrow \mathbf{N}$. The multiplicity of objects in multisets, the time associated with rules in timed P systems, weights associated with synapses in SN P systems, life duration of objects in P systems with object decay (what else?) are of this form. At least as a mathematical challenge, we can try to extend these functions to $f : X \longrightarrow \mathbf{Z}$, where $\mathbf{Z}$ is the set of integers, both positive and negative.

Negative multiplicities can be interpreted in various ways – see also [2] where such multisets are considered. A connection can be made with anti-spikes in SN P systems, in general, with anti-matter, in the sense of [9] – with the interesting observation that anti-matter (plus the priority of the annihilation rules over evolution rules, with the annihilation rules applied in no time) is useful, it speeds-up the functioning of P systems, see, e.g., [6]. What happens when the annihilation does not have priority?

Of course, passing to negative integers in other cases raises problems concerning the definition of the functioning of the systems. For instance, what means to apply a rule having associated a negative time? Moving back in time the produced objects is a possibility, but this means separating the objects associated to several time moments, which would imply that objects can travel back and forth in time and only objects of the same time can react. What happens with the observer time? Naturally, it has to grow continuously with unit steps. How the internal times of objects interact with the external time of the observer? Finding a good definition of the computations in such a system is already a first task. I forecast, however, that the interplay of the internal and the external times will lead to interesting results. Remember also that the observer plays a crucial role in computations – see, e.g., [3].

Of course, one further extension is to replace natural numbers with numbers in larger classes than **Z**, why not?, with real numbers or even with complex numbers. The results are not easy to forecast, but the next section can give some hints and motivation.

3 Hypercomputing

Going beyond the "Turing barrier" is a constant preoccupation of computer scientists. I recall only three surveys, [4], [8] and [13]. In membrane computing there are only a few attempts to achieve a hypercomputation power, see [1] and [12].

No result of this type was reported for SN P systems, in spite of the fact that the motivation of these systems comes from the brain, and the brain is (supposed to be) a non-Turing "computing device".

The problem is much more general: in the hypercomputability area there are several tricks (Martin Davis would say even "dishonest tricks", as the power is introduced from the beginning in the system and then we prove that the system is powerful...) used in order to increase the power of of the obtained machineries beyond the power of usual Turing machines. I list the ten ideas mentioned in [8]: 1. O-machines (Turing machine with oracles), 2. Turing machines with initial inscriptions (infinitely many cells of the input tape contain already symbols), 3. Coupled Turing machines (input channels are provided which bring information into the machine during the computation, as a possibly non-recursive sequence of bits), 4. Asynchronous networks of Turing machines (timing functions are provided, not necessarily recursive), 5. Error prone Turing machines (the errors appear according to a function, again, not necessarily recursive), 6. Probabilistic Turing machines (not so easy to describe), 7. Infinite state Turing machines (infinite sets of states and transitions, but only a finite number of transitions leading from a given state), 8. Accelerated Turing machines (each step takes half of the time needed for performing the previous step – like in [1]), 9. Infinite time Turing machines, 10. Fair non-deterministic Turing machines.

Summarizing: infinite resources, specification or functioning, real numbers, non-recursive functions involved in the computations. Similar tricks are described in [13].

Which of these ideas can be (naturally) extended to P systems? Which of them have even a remote biological support/motivation? Which if these ideas can also speed-up the computations so that computationally hard problems could be solved in a polynomial time? (At least the acceleration is doing it, as in two external time units any computation halts...)

4 Extensions of SN P Systems

Many modifications of the structure and the functioning of SN P systems can be imagined. Here are a few of them.

Consider SN P systems as devices computing functions $f : \mathbf{N}^k \longrightarrow \mathbf{N}^l$, for some $k, l \geq 1$ (take k input neurons and l output neurons etc.). What about the efficiency of this way to compute functions? Any application (similar to the application of numerical P systems in robot control)?

What about SN P systems with astrocytes, with the astrocytes controlling the flow of spikes not according to thresholds associated with them, but using regular expressions in a similar way as the spiking rules use them: couples $(E_i, action_i)$ are associated with the astrocytes and $action_i$ is performed when the number of spikes on the controlled synapses belongs to $L(E_i)$.

In [5], the notion of *white holes* is introduced in membrane computing, as regions where rules which expel all objects are present. What about SN P systems with "white hole neurons", i.e., containing spiking rules $a^n \rightarrow a^n$ for all n? Systems where all neurons are of this type have an interesting behavior: just consider the SN P system in Figure 1, with all neurons being white holes, and follow its functioning. Three increasing sequences of numbers describing the number of spikes in the three neurons are obtained, with an intriguing growth. Can you characterize these sequences? Ca you compute in this way known sequences, such as the Fibonacci one?

Finally, let us return to the brain. Usually, it is considered as working at two levels, the conscious one and the subconscious one. One model of the brain functioning claims that the cortex formulates problems to the subconscious level, this one works "silently", in a great extent nondeterministically, proposing solutions to the conscious level. The cortex evaluates the proposed solutions, accepts the good one, if any, or returns the problem to the "lower" level, and so on, until either the problem is solved or the problem is abandoned or... the brains gets into troubles.

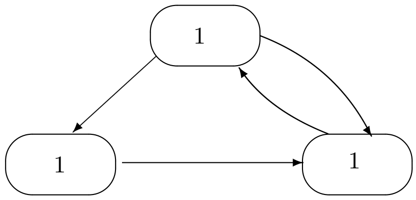

Fig. 1. An SN P system composed of white holes

Can this strategy be implemented in terms of SN P systems? Which is its computing power and, also, its computing efficiency? Nondeterminism is powerful, one can expect interesting results, provided that such an "SN P brain", with two modules (somewhat like in dP systems, [10]), a nondeterministic one and a deterministic one, connected with the environment/user, could be defined.

5 Numerical P Systems

This is a class of P systems which I feel still keeps undiscovered many nice results and, possibly, applications. Just to recall the attention about them, I am formulating here two problems, one theoretical (and also formulated in other contexts) – (1) consider numerical P systems as decidability devices and investigate their efficiency (complexity classes), both in the original setup and, if they are not efficient enough, after introducing membrane division or other tools for producing an exponential working space in linear time – and one applicative: (2) these systems were used to build controllers for robots. 2D robots. What about passing to 3D robots? This is mainly a programming issue, but it could find good applications – for instance, in controlling the drones, so popular in the last time.

6 Final Remarks

References

1. C. Calude, Gh. Păun: Bio-steps beyond Turing. *BioSystems*, 77 (2004), 175–194.
2. J. Carette, A.P. Sexton, V. Sorge, S.M. Watt: Symbolic domain decomposition. *AISC/Calculemus/MKM 2010* (S. Autexier et al., eds.), LNAI 6167, Springer, 2010, 172–188.
3. M. Cavaliere, P. Leupold: Evolution and observation. A new way to look at membrane systems. *Proc. WMC 2003*, LNCS 2933, Springer, 2003, 70–87.
4. B.J. Copeland: Hypercomputation. *Minds and Machines*, 12, 4 (2002), 461–502.
5. E. Csuhaj-Varjú, M. Gheorghe, Gy. Vaszil, M. Oswald: P systems for social networks. *Ninth Brainstorming Week on Membrane Computing*, Sevilla, 2011, 113–124.

6. D. Díaz-Pernil, F. Peña-Cantillana, A. Alhazov, M.A. Gutíerrez-Naranjo, R. Freund: Antimatter as a frontier of tractability in membrane computing. *Fundamenta Informaticae*, 134, 1-2 (2014), 83–96.
7. M. Gheorghe, Gh. Păun, M.J. Pérez-Jiménez, G. Rozenberg: Frontiers of membrane computing: Open problems and research topics, *Intern. J. Found. Computer Sci.*, 24, 5 (2013), 547–623 (first version in *Proc. Tenth Brainstorming Week on Membrane Computing*, Sevilla, January 30 – February 3, 2012, vol. I, 171–249).
8. T. Ord: *Hypercomputation: Computing More Than the Turing Machine*. Honours Thesis, Department of Computer Science, University of Melbourne, 2003.
9. Gh. Păun: Four (somewhat nonstandard) research topics. *12th BWMC*, Sevilla, February 2014, 305–309.
10. Gh. Păun, M.J. Perez-Jimenez: Solving problems in a distributed way in membrane computing: dP systems, *Int. J. of Computers, Communication and Control*, 5, 2 (2010), 238–252.
11. Gh. Păun, G. Rozenberg, A. Salomaa, eds.: *The Oxford Handbook of Membrane Computing*. Oxford University Press, 2010.
12. P. Sosík, O. Valik: On evolutionary lineages of membrane systems. *Membrane Computing, International Workshop, WMC6, Vienna, Austria, 2005, Selected and Invited Papers*, LNCS 3850, Springer, Berlin, 2006, 67–78.
13. A. Syropoulos: *Hypercomputation: Computing Beyond the Church-Turing Barrier*. Springer, Berlin, 2008.
14. The P Systems Website: `http://ppage.psystems.eu`.