Chapter IV
Spiking Neural P Systems: An Overview

Gheorghe Păun
Institute of Mathematics of the Romanian Academy, Romania

Mario J. Perez-Jimenez
University of Sevilla, Spain

ABSTRACT

This chapter is a quick survey of spiking neural P systems, a branch of membrane computing which was recently introduced with motivation from neural computing based on spiking. Basic ideas, examples, some results (especially concerning the computing power and the computational complexity/efficiency), and several research topics are discussed. The presentation is succinct and informal, meant mainly to let the reader having a flavour of this research area. The additional references are an important source of information in this respect.

THE GENERAL FRAMEWORK

Learning computing ideas (data structures, operations with them, ways to control these operations, computing architectures, new types of algorithms—in general, of heuristic ways to search for fast solutions to complex problems, and so on) from biology was a permanent concern for computer science, but in the last decades this became a real fashion. Genetic algorithms, in general, evolutionary computing, neural computing, DNA computing, and several other research directions are already well established theoretically, sometimes also with numerous applications (this is especially the case of genetic algorithms). All these areas form what is now called *natural computing*.

Membrane computing is one of the youngest branches of natural computing. It has been initiated in (Păun, 2000) and soon became a "fast emerging research front of computer science", as Thomson Institute for Scientific Information, ISI, called it – see http://esi-topics.com. In a few words, the goal is to start from the living cell, as well as from the way the cells cooperate in tissues, organs, or other structures, and to devise computational devices. *Membrane systems,* currently called *P systems*, were introduced in this context, taking as basic architecture the structure of a cell, with chemical objects evolving in compartments delimited by membranes. In a cell, the membranes are hierarchically arranged (hence placed in the nodes of a tree), but also P systems with membranes placed in the nodes of a general graph were considered.

The (mathematical) theory of membrane computing is well developed, and a lot of classes of P systems were introduced and investigated – details can be found at the web site from http://psystems.disco.unimib.it, as well in the monograph (Păun, 2002). The main directions of research concern the computing power of P systems and their usefulness for solving computationally hard problems in a feasible time, by making use of the parallelism intrinsic to the model. Thus, most classes of P systems were proved to be Turing complete, and, in the cases where an exponential workspace can be produced in a linear time (e.g., by dividing membranes – *mitosis*), polynomial solutions to **NP**-complete problems were devised.

Recently, a series of applications of P systems were reported, expecially using membrane computing as a framework for building models of biological phenomena (approached at a micro level, thus contrasting with the models based on differential equations, which approach the reality at a macro level). The above mentioned web page contains information in this respect; details can be also found in the volume (Ciobanu et al., eds., 2006),

Also the neural cell was considered as an inspiration source for membrane computing. The present chapter is a quick introduction to so-called *spiking neural P systems* (in short, *SN P systems*), a class of P systems defined in (Ionescu et al., 2006), starting from the way the neurons cooperate in large neural nets communicating by electrical impulses (*spikes*). Neurons are sending to each others electrical impulses of identical shape (duration, voltage, etc.), with the information "encoded" in the frequency of these impulses, hence in the time passed between consecutive spikes. For neurologists, this is nothing new, related drawings already appear in papers by Ramon y Cajal, a pioneer of neuroscience at the beginning of the last century, but in the recent years "computing by spiking" became a vivid research area, with the hope to lead to a neural computing "of the third generation" – see, e.g., (Gerstner and Kistler, 2002), (Maass and Bishop, eds., 1999).

For membrane computing it is natural to incorporate the idea of spiking neurons (already neural-like P systems exist, based on different ingredients, efforts to compute with a small number of objects were recently made in several papers, using the time as a support of information, for instance, taking the time between two events as the result of a computation, was also considered), but still important differences exist between the general way of working with multisets of objects in the compartments of a cell-like membrane structure – as in usual P systems – and the way the neurons communicate by spikes. A way to answer this challenge was proposed in (Ionescu et al., 2006): neurons taken as single membranes, placed in the nodes of a graph corresponding to synapses, only one type of objects present in neurons, the spikes, with specific rules for handling them, and with the distance in time between consecutive spikes playing an important role (e.g., the result of a computation being defined either as the whole spike train of a distinguished output neuron, or as the distance between consecutive spikes). Details will be given immediately.

What is obtained is a computing device whose behaviour resembles the process from the neuron nets, meant to generate strings or infinite sequences (like in formal language theory), to recognize or translate strings or infinite sequences (like in automata theory), to generate or accept natural numbers, or to compute number functions (like in membrane computing). Results of all these types will be mentioned below. Nothing is said