On String Languages Generated by Sequential Spiking Neural P Systems Based on Maximum Spike Number

Keqin Jiang$^{1,2}$, Yuzhou Zhang$^2$, and Linqiang Pan$^{1,*}$

1 Key Laboratory of Image Information Processing and Intelligent Control, School of Automation, Huazhong University of Science and Technology, Wuhan 430074, Hubei, China
   jiangkq0519@163.com, lqpan@mail.hust.edu.cn
2 School of Computer and Information, Anqing Normal University, Anqing 246133, Anhui, China
   zhangyuzhou@aqtc.edu.cn

Abstract. Spiking neural P systems (SN P systems, for short) are a class of distributed parallel computing devices inspired from the way neurons communicate by means of spikes. In this work, we consider SN P systems with the restriction: at each step the neuron with the maximum number of spikes among the neurons that can spike will fire (if there is a tie for the maximum number of spikes stored in the active neurons, only one of the neurons containing the maximum is chosen non-deterministically). We investigate the computational power of such sequential SN P systems that are used as language generators. We prove that recursively enumerable languages can be characterized as projections of inverse-morphic images of languages generated by that sequential SN P systems. The relationships of the languages generated by these sequential SN P systems with finite and regular languages are also investigated.

Keywords: Membrane computing, Spiking neural P system, Sequentiality, Maximum spike number.

1 Introduction

Spiking neural P systems were introduced in [4] as a class of distributed parallel computing models which were abstracted from the way neurons process information and communicate to each other by sending spikes along synapses. Since then, many computational properties of SN P systems have been studied. SN P systems were proved to be computationally complete as number generating or accepting devices [3,4,12,13], language generators [1,2,14], and function computing devices [8,9,15]. SN P systems can be also used to (theoretically) solve computationally hard problems in a feasible time [5,7]. Readers can refer to the
handbook [10] for the details of SN P systems, and the up-to-date information is available at the membrane computing website http://ppage.psystems.eu.

Briefly, an SN P system consists of a set of neurons, which are placed in the nodes of a directed graph whose arcs represent the synapses. Each neuron can contain a number of copies of a single object type, called the spike, spiking rules and forgetting rules. Using its rules, a neuron can send information (in the form of spikes) to all neurons connected by an outgoing synapse from it. The applicability of a rule is determined by checking the total number of spikes contained in the neuron against a regular expression associated with the rule. One of the neurons is the output neuron and its spikes are sent to the environment. The moments of time when a spike is emitted by the output neuron are marked with 1, the other moments are marked with 0. This binary sequence is called the spike train of the system. A result can be associated with a computation in various ways: for example, as the number of spikes sent to the environment or as the time elapsed between the first two consecutive spikes sent to the environment by the system. An SN P system can be used as a computing device in various ways, for example, as an acceptor, a transducer or as a language generator.

In [3], SN P systems were used as number generating devices and as number accepting devices with the restriction: at each step the neuron with the maximum number of spikes among the neurons that can spike will fire; if there is a tie for the maximum number of spikes stored in the active neurons, only one of the neurons containing the maximum is chosen non-deterministically. Such restriction under which SN P systems work is called “max-sequentiality”. The computational power of SN P systems working in the max-sequentiality manner used as number generating devices and as number accepting devices was already investigated [3].

In this work, we investigate the computational power of SN P systems working in the max-sequentiality manner used as language generators. We prove that recursively enumerable languages can be characterized as projections of inverse-morphic images of languages generated by sequential SN P systems. The relationships of the languages generated by sequential SN P systems with finite and regular languages are also investigated.

2 SN P Systems Working in Max-sequentiality Manner

We recall the definition of SN P systems working in max-sequentiality manner. In the definition of the systems, the notion of regular expression is used, readers can refer to [11] for the details.

An SN P system working in max-sequentiality manner, of degree $m \geq 1$, is a construct of the form

$$\Pi = (O, \sigma_1, \sigma_2, \ldots, \sigma_m, \text{syn}, \text{out}),$$

where:

- $O = \{a\}$ is a singleton alphabet ($a$ is called spike);
- $\sigma_1, \sigma_2, \ldots, \sigma_m$ are neurons, of the form $\sigma_i = (n_i, R_i)$ with $1 \leq i \leq m$, where
  - a) $n_i \geq 0$ is the initial number of spikes contained in $\sigma_i$;