Solving the SAT problem using spiking neural P systems with coloured spikes and division rule

24th Conference on Membrane Computing, CMC 2023

Authors: Prithwineel Paul, Petr Sosík

Institute of Computer Science, Faculty of Philosophy and Science, Silesian University in Opava, Czech Republic

Spiking neural P systems

- Neural-like P systems [1];
- Third-generation neural networks;
- Spiking neural P systems with colored spikes [2];
- Spiking neural P systems with neuron division and budding [3];

[1] Ionescu, M., Păun, G., Yokomori, T.: Spiking neural P systems. Fundamenta informaticae 71(2-3), 279–308 (2006)

[2] Song, T., Rodríguez-Patón, A., Zheng, P., Zeng, X.: Spiking neural P systems with colored spikes. IEEE Transactions on Cognitive and Developmental Systems 10(4), 1106–1115 (2017)

[3] Pan, L., Păun, G., Pérez-Jiménez, M.J.: Spiking neural P systems with neuron division and budding. Science China Information Sciences 54, 1596–1607 (2011)

Spiking neural P system with coloured spikes and neuron division

Definition

- $\Pi = (S, H, syn, \sigma_1, \sigma_2, \ldots, \sigma_m, R, in, out)$
- $m \geq 1$ (the number of neurons initially present in the system);
- $S = \{a_1, a_2, \ldots, a_g\}, g \in \mathbb{N}$ (the alphabet of spikes of different colours);
- H (the set containing labels of the neurons);
- $syn \subseteq H \times H$ (synapse dictionary between the neurons where $(i, i) \notin syn$ for $i \in H$);
- $\sigma_i = \langle n_1^i, n_2^i, \ldots, n_g^i \rangle$, $(1 \leq i \leq m)$ neuron σ_i contains initially $n_j^i \geq 0$ spikes of type a_j $(1 \leq j \leq g)$;

Definition

- R (set of the rules of Π);
- **Spiking rule:** $[E/a_1^{n_1} a_2^{n_2} \ldots a_g^{n_g} \to a_1^{p_1} a_2^{p_2} \ldots a_g^{p_g}; d]_i$ where $i \in H$, E is a regular expression over S; $n_j \geq p_j \geq 0$ $(1 \leq j \leq g)$; $d \geq 0$ (delay); $p_j > 0$ for at least one j, $1 \leq j \leq g$.
- **Forgetting rule:** $[a_1^{t_1} a_2^{t_2} \ldots a_n^{t_n} \to \lambda]_i$ where $i \in H$, and $a_1^{t_1} a_2^{t_2} \ldots a_n^{t_n} \notin L(E)$ for each regular expression E associated with any spiking rule in neuron i;
- **Neuron division rule:** $[E]_i \to []_j \,\|\, []_k$; E is a regular expression over S; $i, j, k \in H$.
- *in* (input neuron); *out* (output neuron)

A solution to the SAT problem

- SAT problem (or the Boolean satisfiability problem) [4] is a well-known NP-complete decision problem.
- $\gamma_{n,m} = C_1 \wedge C_2 \wedge \ldots \wedge C_m$
- C_i $(1 \leq i \leq m)$ (clauses).
- Each clause is a disjunction of literals of the form x_j or $\neg x_j$, where x_j are logical variables, $1 \leq j \leq n$.
- $SAT(n, m)$ = class of SAT instances with n variables and m clauses;
- $\gamma_{n,m} \in SAT(n, m)$;

[4] Rintanen, J.: Planning and SAT. Handbook of Satisfiability 185, 483–504 (2009)

A solution to the SAT problem

- At first, we encode an instance $\gamma_{n,m}$ using spikes in the SNPS, and we send it to the input neuron.

$$code(\gamma_{n,m}) = a^{n+1}(\alpha_{1,1}\alpha_{1,2}\ldots\alpha_{1,n})a_c(\alpha_{2,1}\alpha_{2,2}\ldots\alpha_{2,n})a_c$$

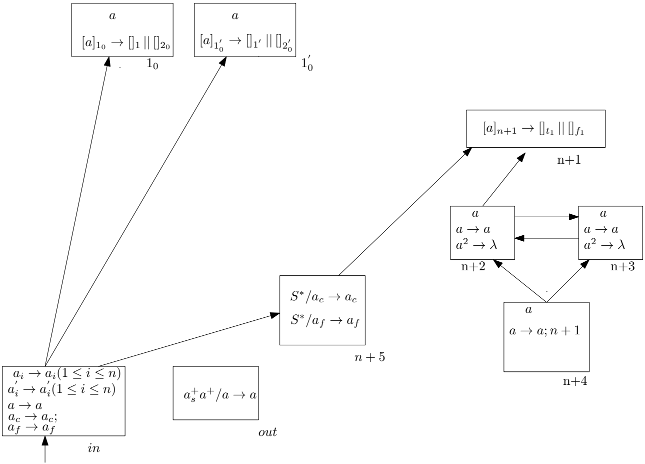
$$\ldots(\alpha_{m,1}\ldots\alpha_{m,n})a_c a_f,$$

$$\alpha_{i,j} = \begin{cases} a_j, & \text{if } x_j \in C_i, \\ a'_j, & \text{if } \neg x_j \in C_i, \\ a, & \text{otherwise.} \end{cases}$$

A solution to the SAT problem

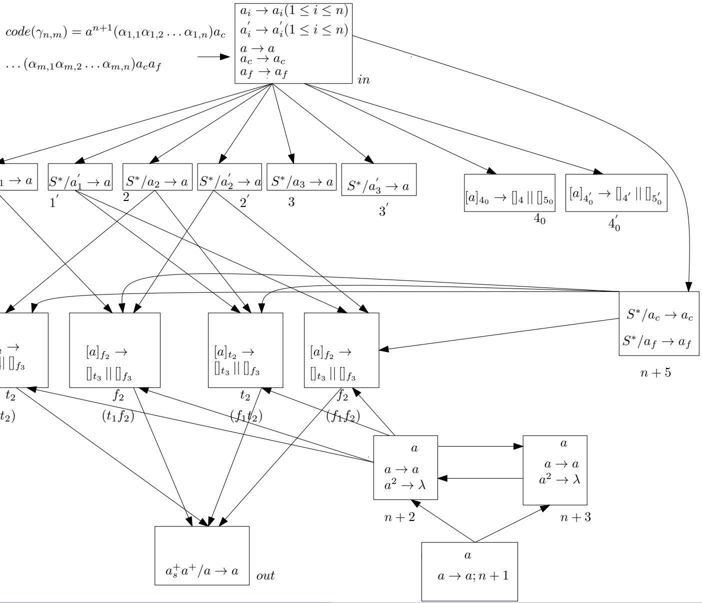
- a^{n+1} is added at the beginning to give the system a necessary initial period during which it generates an exponential workspace with $\mathcal{O}(2^n)$ neurons.
- The encoding of each clause is separated by a_c and the end of the encoding is identified by a_f.

Initial structure of the SNPS

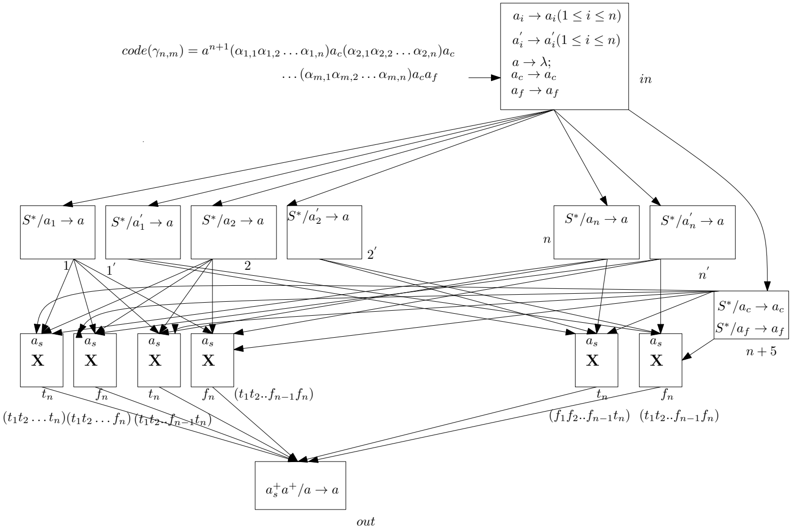


$code(\gamma_{n,m}) = a^{n+1}(\alpha_{1,1}\alpha_{1,2}\ldots\alpha_{1,n})a_c\ldots(\alpha_{m,1}\alpha_{m,2}\ldots\alpha_{m,n})a_c a_f$

Structure of the SNPS at time $t = 4$



Structure of the SNPS at time $t = n + 2$



$\mathbf{X} = \{(1)\ a_saa/a \to a_s;\ (2)\ a_sa_c \to \lambda;\ (3)\ a_sa_ca/a_ca \to a_s;\ (4)\ a_sa_f \to a\}$

Comparison of the resources

Resources	Wang et. al. [5]	Zhao et. al. [6]	This paper
Initial number of neurons	11	$3n+5$	9
Initial number of spikes	20	$2m+3$	5
Number of neuron labels	$10n+7$	2^n+11	$6n+7$

[5] Wang, J., Hoogeboom, H.J., Pan, L.: Spiking neural P systems with neuron division. In: Membrane Computing: 11th International Conference, CMC 2010, Jena, Germany, August 24-27, 2010, pp. 361–376. Springer (2011)

[6] Zhao, Y., Liu, X., Wang, W.: Spiking neural P systems with neuron division and dissolution. PLoS One 11(9), e0162882 (2016)

Comparison of the resources

Size of synapse dictionary	$6n + 11$	$5n + 5$	$2n + 12$
Number of rules	$2n^2 + 26n$ $+26$	$n2^n + \frac{1}{3}(4^n - 1)$ $+ 9n + 5$	$8n + 16$
Time complexity	$4n + nm + 5$	$2n + m + 3$	$nm + n+$ $m + 5$
Number of neurons generated throughout the computation	$2^n + 8n$	$2^{n+1} - 2$	$2^n + 2n$


Thank You