This folder contains Haskell implementations (`www.haskell.org`) of the operational semantics and the denotational semantics presented in the paper:

> "A Semantic Investigation of Spiking Neural P Systems with Mute Rules"
> Authors: Eneia Nicolae Todoran and Gabriel Ciobanu
>
> Submitted to 28th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC 2026)

The implementations for the operational semantics and the denotational semantics are available in the files `Lsnpmr-osem-cps.hs` and `Lsnpmr-dsem-cps.hs`, respectively. The semantic interpreters `Lsnpmr-osem-cps.hs` and `Lsnpmr-dsem-cps.hs` are engineered to generate all execution traces (irrespective of their number or size) assembled in an element of a linear-time powerdomain [1]. An element of a powerdomain is exponential in the size of execution traces.[1] Thus, the implementations available in the files `Lsnpmr-osem-cps.hs` and `Lsnpmr-dsem-cps.hs` can only be used to test (toy) programs like the $\mathcal{L}_{snpmr}$ program π_1 considered in examples 1 and 2 presented in the paper, but become intractable in case the number of execution traces is large (possibly infinite).

By using techniques presented in [2, 3], one can develop semantic interpreters that provide support for testing complex concurrent and nondeterministic programs, based on either random choice or finite execution traces. The implementations available in the files `Lsnpmr-osem-cps-fin.hs` and `Lsnpmr-dsem-cps-fin.hs` employ the latter approach: each execution trace is pruned after a specified finite number of execution steps. Thus, only a finite prefix of specified length is preserved for each execution trace.

For illustration we examine the $\mathcal{L}_{snpmr}$ program π_2 presented below. We note that the $\mathcal{L}_{snpmr}$ program π_2 implements the SNPMR system Π given in [4] (Section 2, Fig. 1), which generates the set of all even natural numbers. The $\mathcal{L}_{snpmr}$ program π_2 is similar to $\mathcal{L}_{snpmr}$ program π_1 considered in examples 1 and 2 given in the paper. However, the declaration for neuron ν_1 is extended with the spiking rule $a^2/[a] \to a$ which provides support for iteration as explained in [4]. The $\mathcal{L}_{snpmr}$ program π_2 is defined by $\pi_2 = (N_2, s_2)$, where:

$$
\begin{aligned}
N_2 = {} & \mathsf{neuron}\ \nu_0\,\{\,\rho_\epsilon \ \mathbin{|}\ \{\nu_1, \nu_2\}\,\};\\
& \mathsf{neuron}\ \nu_1\,\{\,a/+\,[a],\ [a^2] \to a \parallel a,\ a^2/[a] \to a \ \mathbin{|}\ \{\nu_2\}\,\};\\
& \mathsf{neuron}\ \nu_2\,\{\,a/-\,[a],\ [a^2] \to a \ \mathbin{|}\ \{\nu_0\}\,\}, \qquad \text{and}\\
s_2 = {} & \mathsf{send}\,\{\nu_1\}\,a \ \parallel\ \mathsf{send}\,\{\nu_2\}\,a \ \parallel\ \mathsf{send}\,\{\nu_2\}\,a\,.
\end{aligned}
$$

The semantic interpreters `Lsnpmr-osem-cps-fin.hs` and `Lsnpmr-dsem-cps-fin.hs` can be used to test the $\mathcal{L}_{snpmr}$ program π_2, which generates the set of all even natural numbers in a nondeterministic manner. If for testing the $\mathcal{L}_{snpmr}$ program π_2 it is specified that the length of the prefix preserved for each execution trace is $2n$ then the output neuron will produce spikes (in alternative execution traces) in steps: 2 and 4, 2 and 6,..., or 2 and $2n$, respectively; hence the even natural numbers $2, 4, ..., 2(n-1)$ are generated in this way.[2]

References

[1] J.W. De Bakker and E.P. De Vink, Control Flow Semantics, MIT Press, 1996.

[2] G. Ciobanu, E.N. Todoran, "Spiking neural P systems and their semantics in Haskell," Natural Computing, vol. 22(1), pp. 41–54, 2023.

[3] G. Ciobanu, E.N. Todoran, "Variants of spiking neural P systems and their operational semantics in Haskell," J. of Membr. Comput., vol. 5(2), pp. 81–99, 2023.

[4] T. Wu, L. Valencia-Cabrera, M.J. Pérez-Jiménez, and L. Pan, "Spiking neural P systems with mute rules," Information and Computation, vol. 299, 105179, 2024.

[1] An element of a powerdomain is a structure resembling a tree, or a set of execution "traces" that can be seen as an unfolding of such a tree.

[2] In addition, one execution trace where the output neuron produces a spike only in step 2 is generated, representing the prefix of length $2n$ of any execution trace corresponding to an even natural number $\geq 2n$.