A First Attempt to Model Notch Signalling by Means of P Systems

Manuel García-Quismondo, Beverley M. Henley,
Ignacio Pérez-Hurtado, Agustín Riscos-Núñez

Research Group on Natural Computing
University of Seville (Spain)
mangarfer2@alum.us.es, {bhenley,perezh,ariscosn}@us.es

Summary. During mammalian central nervous system development, an enormous variety of cell types are generated. This cell diversity is due in part to asymmetrical cell division. Indeed, in some sense Notch signals link the fate decisions of one cell to those of its neighbours. This fundamental signalling pathway has not yet been modeled within membrane computing framework.

1 Introduction

In the last years, an increasing number of results are being obtained in the field of using membrane systems to model different biological phenomena. This has been done both at the microscopical level as well as at the macroscopic level.

The purpose of the present work is to adequately model the activation of the Notch pathway (using membrane computing). Notch activation is described in detail in [9], which is clearly summarised in [8], including all relevant references and diagrams (the important Notch activation diagram is in [9]).

Of course, the first task that is required to be done is the specification of the membrane system model that is going to be used. This will be done within the software framework P-Lingua 2.0 [1], in order to allow an easy route towards a software simulation of the designed model.

Biologists usually use static model of pathways in an aid to understanding, so it will be really beneficial for them to get access to an effective representation of the activation of Notch in a dynamic computer model. Ideally (and in the long term) the model should be able to shed light on some important questions (listed at the end of the present document), specially if computer simulations can be run.

[1] P-Lingua 2.0 is a software package including several built-in simulators for a number of different P system models. It includes a specification language also called P-Lingua which is used to define the P systems to be simulated. More information at [6]

2 Cell Diversity

The mammalian central nervous system (CNS) contains an enormous variety of cell types each with a unique morphology, physiology and function [10]. Understanding how neuroepithelial cells (stem cells) of the developing CNS choose between alternative cell fates to generate cell diversity is a challenge [3]. During development, cell-fate diversity is brought about, in part, by asymmetric cell divisions [7]. Asymmetric segregation of cell determinants, such as Numb, can result in the differential activation of the Notch pathway, which can generate cell diversity [5].

In invertebrates, asymmetric segregation of cell-fate determining proteins or mRNAs to the two daughter cells during precursor cell division plays a crucial part in cell diversification. There is increasing evidence that this mechanism also operates in vertebrate neural development and that the Numb protein, which functions as cell-fate determinant during Drosophila development, may also function in this way during vertebrate development [3]. A very clear illustration of symmetric and asymmetric segregation of a cell fate-determining protein can be found in Figure 2 of [3].

3 Modelling

The Notch pathway is a fundamental pathway in metazoan development and the design and implementation of a good dynamic model of this pathway, and of crosstalk between Notch and other signalling pathways, may be beneficial to developmental biologists.

Besides, from a computer science point of view, if a software counterpart of the membrane system model is developed, capturing the interaction between neighbours and the relevance of asymmetric distribution of proteins, then certainly such a tool will be very valuable to facilitate future designs of similar models.

A reasonable choice to initiate the modelling task is to follow the work already done for other signalling cascades (e.g. FAS-induced apoptosis [4], gene regulation in *Lac Operon* [13]). In this sense, it is advisable to use stochastic P systems that use a *Multi-compartmental Gillespie Algorithm* to govern their evolution. Let us summarize next the types of rules used in this framework

- Protein-Protein rules (Multiset rewriting):

$$r : u[w]_l \xrightarrow{c_r} u'[w']_l$$

  where u, w, u', w' are finite multisets of objects and l a label. The multiset u placed outside of the membrane labelled with l and the multiset w placed inside of that membrane are simultaneously replaced by a multiset u' and w' respectively. These rules are referred to as *boundary rules* in [1].
- Genetic rules (String rewriting):

$$r : [u,\, s]_l \xrightarrow{c_r} [u',\, s']$$

These rules allow the interaction between a multiset of objects u (e.g. a multiset of proteins) and a string s (representing e.g. a sequence of DNA binding sites).

Note that both types of rules have associated with them a constant c_r that represents the kinetic constants associated with reactions in molecular biochemistry.

Using such P system setting has several advantages. On one hand, P-Lingua is able to handle this rules, and thus the possibility to run simulations of any designed model is at hand. On the other hand, in order to investigate in the future crosstalk between Notch and other signalling pathways, all the rules involved should follow the same syntax and semantics (e.g. stochastic P systems as implemented in P-Lingua).

However, there are still important difficulties to solve, since the asymmetric distribution of Notch and Delta ligands over the skin membrane seems to play a crucial role, although it is not possible to express this information in the above framework. Besides, in order to understand globally the effects of the Notch signalling on the cell diversification process mentioned in the preceding section, we need to consider in our model rules allowing the interaction between two neighbour cells. Furthermore, division rules should also be considered, as well as rules capturing the movement of cells. These new requirements remind us of other models in the literature where instead of focusing inside the cytoplasm of a single cell, a population of individuals is considered (see e.g. [12] where multienvironments are used to model the quorum sensing system in *Vibrio fischeri*).

Our proposal is to bridge stochastic P systems and multienvironments. Keep in mind that P-Lingua is a flexible language, and even if we tailor a new model fitting our expectations there are programming methods to easily extend the software in order to cover the new model. Moreover, the package includes a Java library implementing several built-in simulators and parsers. It is possible to develop an appropriate interface over the library in order to implement a specific simulator for the topic of this paper.

4 Notch pathway: questions

Notch signals affect specific cell fates in a context-specific manner, a schematic summarising the effects of Notch signalling and its effect on cell fate decisions can be found in [11], Figure 3. Understanding how and why different target genes are activated according to cell type and time is a very important question, in other words: how and why is Notch activation context dependant [2]? This and other important questions are posed by Bray in [2]. The response to Notch differs greatly between cell types, for example Notch promotes cell proliferation in some contexts and apoptosis in others. What is the reason for this? Bray also states that recent data reveals that the precise location of the Notch ligand and the receptor in the cell can have profound effects on signalling. How does the different ligand locations exactly impact on Notch activity? All of these questions are extremely important

in untangling the role of Notch during diverse developmental and physiological processes.

Acknowledgement

The authors acknowledge the support of the project TIN2006–13425 of the Ministerio de Educación y Ciencia of Spain, cofinanced by FEDER funds, and the support of the "Proyecto de Excelencia con Investigador de Reconocida Valía" of the Junta de Andalucía under grant TIC04200.

References

1. F. Bernardini, V. Manca. P systems with boundary rules. *Lecture Notes in Computer Science*, 2597 (2002), 107-118.
2. S. J. Bray. Notch signalling: a simple pathway becomes complex. *Nature Reviews Molecular Cell Biology*, 7, 9 (2006), 678-689.
3. M. Cayouette and M. Raff. Asymmetric segregation of Numb: a mechanism for neural specification from Drosophila to mammals. *Nat. Neurosci.*, 5, 12 (2002), 1265-1269.
4. S. Cheruku, A. Păun, F.J. Romero, M.J. Pérez-Jiménez, O.H. Ibarra. Simulating FAS-induced apoptosis by using P systems. *Progress in Natural Science*, 17, 4 (2007), 424–431.
5. P. Fichelson and M. Gho. Mother-daughter precursor cell fate transformation after Cdc2 down-regulation in the Drosophila bristle lineage. *Dev. Biol.*, 276, 2 (2004), 367-377.
6. M. García-Quismondo, R. Gutiérrez-Escudero, I. Pérez-Hurtado, M.J. Pérez-Jiménez, A. Riscos-Núñez. An overview of P-Lingua 2.0. In this volume.
7. M. Gho and F. Schweisguth. Frizzled signalling controls orientation of asymmetric sense organ precursor cell divisions in Drosophila. *Nature*, 393 (1998), 178-181.
8. B.M. Henley. Notch Signalling and Cellular Fate Choices: A Short Review Proc. Seventh Brainstorming Week on Membrane Computing (vol. I), 227–230. (available online through `http://www.gcn.us.es`)
9. R. Kageyama, T. Ohtsuka, J. Hatakeyama, and R. Ohsawa. Roles of bhlh genes in neural stem cell differentiation. *Experimental Cell Research*, 306, 2 (2005), 343-348.
10. E. S. Lein et al. Genome-wide atlas of gene expression in the adult mouse brain. *Nature*, 445 (2006), 168-176.
11. A. Louvi and S. Artavanis-Tsakonas. Notch signalling in vertebrate neural development. *Nature Reviews Neuroscience*, 7 (2006), 93-102.
12. F.J. Romero, M.J. Pérez-Jiménez. A model of the Quorum Sensing System in Vibrio Fischeri using P systems. *Artificial Life*, 14, 1 (2008), 95-109.
13. F.J. Romero, M.J. Pérez-Jiménez. Modelling gene expression control using P systems: The Lac Operon, a case study. *BioSystems*, 91, 3 (2008), 438–457.