Nets-within-nets for modeling emergent patterns in ontogenetic processes

Roberta, Bardini[a], Alfredo Benso[a], Gianfranco Politano[a] and Stefano Di Carlo[a]

[a]*Politecnico di Torino, Control and Computer Engineering Department, Corso Duca degli Abruzzi 24, 10129 Torino, Italy*

ARTICLE INFO

Keywords:
Petri Nets
Ontogenesis
Systems Biology
C. Elegans

ABSTRACT

Ontogenesis is the development of an organism from its earliest stage to maturity, including homeostasis maintenance throughout adulthood despite environmental perturbations. Almost all cells of a multicellular organism share the same genomic information. Nevertheless, phenotypic diversity and complex supra-cellular architectures emerge at every level, starting from tissues and organs. This is possible thanks to a robust and dynamic interplay of regulative mechanisms.

To study ontogenesis, it is necessary to consider different levels of regulation, both genetic and epigenetic. Each cell undergoes a specific path across a landscape of possible regulative states affecting both its structure and its functions during development. This paper proposes using the Nets-Within-Nets formalism, which combines Petri Nets' simplicity with the capability to represent and simulate the interplay between different layers of regulation connected by non-trivial and context-dependent hierarchical relations.

In particular, this work introduces a modeling strategy based on Nets-Within-Nets that can model several critical processes involved in ontogenesis. Moreover, it presents a case study focusing on the first phase of Vulval Precursor Cells specification in *C. Elegans*. The case study shows that the proposed model can simulate the emergent morphogenetic pattern corresponding to the observed developmental outcome of that phase, in both the physiological case and different mutations. The model presented in the results section is available online at https://github.com/sysbio-polito/NWN_CElegans_VPC_model/

1. Introduction

Ontogenesis is one of the key concepts at the base of developmental biology [51], defined as "[...] *the development of a single individual, or a system within the individual, from the fertilized egg to maturation and death*" [46]. Ontogenetic processes comprise complex and intertwined mechanisms at different levels, from the embryonic development of the organism as a whole to the differentiation of single cells.

A particularly challenging task when modeling ontogenesis is to predict the outcome of a developmental process, simulating the formation of emergent morphological and phenotypic patterns from local inter-cellular interactions. A model serving this purpose must describe the cellular organization in space and the consecutive temporal stages characterizing the process. Each stage corresponds to a different conformation and regulative set-up involving multiple interacting cells. Since the regulatory states of these cells depend on their neighbors' relations, such conformations dictate the communication schemes they engage in [21]. This complexity contributes to creating a multi-dimensional and dynamic landscape of inter-dependent regulative states in which cells can fall into [39, 22]. Moreover, one has to consider that physiology and phenotype not only emerge from the system's subparts. The subparts themselves self-reproduce, determining internal and system-level regulative and structural changes.

To holistically consider the dynamic properties of this hierarchical regulative environment, a computational model must explicitly account for their evolution in space and time. Moreover, it must quantitatively simulate the underlying non-linear regulatory dynamics across different system levels. This requirement imposes the need to model the hierarchical organization of the system. Eventually, biological systems are intrinsically stochastic, i.e., modeling of their complexity must include stochastic behaviors.

Several modeling approaches for Systems Biology based on different formalisms exist, and each of them has strengths that better fit specific aspects of the target problem [6, 14]. Mathematical models based on systems of differential equations have been among the first proposed to represent continuous biological quantities in chemical reactions at the metabolic level [24, 17]. Nevertheless, higher-level phenomena of interest for the description of ontogenetic mechanisms (e.g., cell fate determination) are better described by qualitative information. Also, several interactions (e.g., biological signaling, omics regulation, etc.) can be described by discrete information related to the involved entities (e.g., molecules, cells). Discrete representations are also valuable to overcome the computational complexity affecting continuous biological models, which strongly limit their application when the complexity of the system scales up [3]. Moreover, current approaches often focus on modeling intracellular mechanisms and thus fail to adequately capture supra-cellular spatial elements, limiting their capability to analyze patterns emerging from local interactions between biological entities [6, 5].

Comprising both intracellular and supra-cellular information can broaden the scope of representations. Yet, it poses the challenge of combining heterogeneous informa-

roberta.bardini@polito.it (R. Bardini); stefano.dicarlo@polito.it (S. Di Carlo)
ORCID(s):

tion from multiple system levels in a single model. One of the approaches for handling model heterogeneity is the composition of existing models into a more extensive scope [6]. However, this approach raises consistency issues [45] and introduces additional requirements to create multi-level and hybrid models [3].

This paper introduces a computational modeling methodology based on Nets-Within-Nets (NWN), a formalism based on Petri Nets (PN). NWN rely on a single formalism to cover all requirements posed by the modeling of complex ontogenetic processes. The paper does not aim at introducing a new formal extension of the NWN formalisms. NWN are instead used as a powerful instrument to propose a methodology to model complex dynamics, stochastic processes, hierarchical organizations, and spatial structures in biology. NWN support composition and integration processes typical of hybrid models, supporting the integration of heterogeneous sources of knowledge. In other words, NWN leverage the strengths of modeling paradigms from various existing computational tools and methods, with the advantage of maintaining uniformity in formalism usage. This paper explicitly refers to the NWN implementation provided by Renew, an extensible editor and simulation engine for Petri nets [29, 10, 53]. The main advantage of this framework is to add the full power of an object-oriented language (i.e., Java) to the NWN formalism, thus allowing the description of more complex functionalities.

After introducing the NWN formalism, its relations to other PN-based formalisms, and our usage of its capabilities for modeling ontogenetic processes, this paper provides a working example of the proposed approach. This example models a well-characterized and straightforward ontogenetic process: the Vulval Precursor Cells (VPC) specification process in *C. Elegans*. Eventually, the paper provides an appendix proposing a library of NWM modeling basic biological processes that interested readers can use as a starting point to build models of different ontogenetic processes.

2. Background

The goal of this section is not to provide a complete and formal review of the PN theory. Interested readers may refer to [43] for a detailed introduction on this topic. This section provides an overview of PN applications in the biological domain and, most importantly, introduces the basic characteristics of the NWN formalism exploited to model ontogenetic processes in the following parts of the paper.

2.1. Petri nets definition

Petri Nets (PN) are discrete event system models first introduced in the early 1960s by Carl Adam Petri in his Ph.D. dissertation [41]. PN combine a well-defined mathematical theory with a graphical representation of the dynamic behavior of the systems. This combination is among one of the main reasons for the great success of PN, which have been used to model various kinds of dynamic event-driven systems, including biological systems [11].

A PN is a directed bipartite graph having two types of nodes: *places* representing states or conditions and *transitions* representing events that may produce, consume or take resources from one state to another. Directed arcs can only link places to transitions, or transitions to places. Places can contain a discrete number of *tokens*, each one representing a resource unit. The marking of a PN is the token configuration in its places. It can dynamically change when transitions fire. Formally, a PN is a 4-tuple $N = (P, T, W, M_0)$, where:

1. $P = \{p_1, p_2, ..., p_m\}$ is a finite set of places;
2. $T = \{t_1, t_2, ..., t_n\}$ is a finite set of transitions, $P \cup T \neq \emptyset$, and $P \cap T = \emptyset$;
3. $W : (S \times T) \cup (T \times S) \rightarrow \mathbb{N}$ assigns to each arc a non-negative integer arc multiplicity (or weight); note that no arc may connect two places or two transitions;
4. $M_0 : P \rightarrow N$ is the initial marking, i.e., the initial configuration of tokens. Together with the network architecture, it defines the PN model.

Figure 1 shows an example of the graphical formalism used to represent a simple PN. Specific symbols are assigned to represent places, transitions, arcs, and tokens. Transitions can move tokens from their input to their output places. Firing a transition consumes $W(p_i, t)$ tokens from each input place p_i and produces $W(t, p_j)$ tokens in each of its output places p_j. When not indicated, the weight of an arc is equal to 1. A transition t is enabled (it may fire) in a marking M if there are enough tokens in its input places for the consumptions to be possible, i.e., if and only if $\forall p : M(p) \geq W(p, t)$.

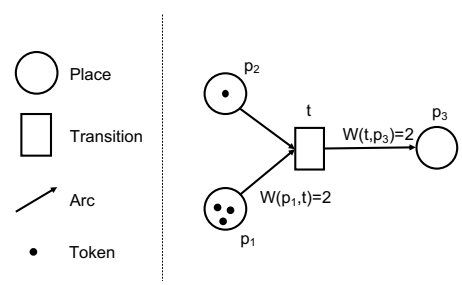

Figure 1: Graphical representation of a simple Petri net. Petri nets consist of places, transitions, arcs and tokens.

Starting from this basic definition, also referred to as *low-level PN*, several PN variants have been defined. Some of them found application in supporting biological systems modeling to different extents.

2.2. Petri nets in biological applications

Simple biological mechanisms such as isolated biochemical reactions require a quantitative representation of kinetic parameters and stoichiometric relations between the involved molecular species. In a low-level PN, places can represent species (substrates and products), while transitions can model reactions. Tokens can express in a discrete way relative quantities of molecules.

Moving from a single reaction to groups of biochemical reactions (e.g., metabolic and gene regulation networks)

introduces further requirements such as more complex architectures and concurrency [27]. Low-level PN can still support this complexity. In a metabolic network, places can model molecular species and states, while transitions can model enzymatic reactions. Tokens represent discrete quantities of molecules, while the architecture of the network represents how resources flow within a network of competitive or sequential reactions. Similarly, in a regulation network, places can refer to genes and gene products. Transitions can cover transcription, translation, and regulation processes. At the same time, tokens can model elements from the relevant omic pools (e.g., genes, transcripts, proteins) and the regulatory conditions in which each of them can fall.

The phenomena considered so far refer to a single system level (i.e., intracellular mechanisms) and consider a limited temporal scale. Expressing different time scales provides greater expressivity which is often required, for example, to model differential transcriptional rates. To handle different timescales, timed PN [48] introduce a time delay associated with the activation of each transition. Once a transition is enabled, deterministic time delays can occur, ordering the different activation events during the net evolution. The introduction of time delays increases time resolution when including diverse yet intertwined mechanisms in a model.

Stochastic PN [32] extend Timed PN with the use of probabilistic time delays. They can model the stochasticity of a biological system (e.g., gene expression level random fluctuations [19]). The delays become random variables that can depend on the current marking of the net [50].

Time continuous PN have also been proposed in the literature [12]. They can be a valuable tool to model selected biological processes that are not discrete in nature. While these PN can potentially be employed in the methodology presented in this paper, they have not been exploited since not supported by the selected simulation framework.

In general, low-level PN do not scale with system complexity. The lack of scalability limits their use to the modeling of small systems. *High-level* Petri Nets can increase the model's expressivity, including different system levels and dimensional scales to provide a more systemic view of the biological complexity. High-level PN extend the low-level formalism, supporting multi-level and nested models that properly handle information diversity and complexity [31].

Colored PN (CPN) [23] are the simplest class of high-level PN. They are important since they allow to associate tokens with arbitrarily complex data structures defined as *colors* encoding complex information. Moreover, in CPN, each place and transition can be designed to accept a limited number of colors. In this way, it is possible to separate the identity of resources from their location, modeling the same condition for different categories of resources [36]. CPN lead to non-redundant and more compact representations of the system. This compact representation improves readability and averts modeling errors while preserving the modeling capabilities of low-level PN, which can be generated from CPN models by automatic unfolding [30, 31].

Models considered so far flatten information from different system levels into a single one. To represent a multi-level system structure, hierarchical PN organize system parts and sub-parts in nested net architectures with explicit hierarchical relations. The nested architecture allows for arbitrarily high resolution when describing mechanisms from different system levels [34]. Nevertheless, like CPN, hierarchical PN stick to a static paradigm: token colors are static data structures, and nets have a fixed model architecture. Resources can change state only by moving from place to place, i.e., changing their position in the net, preserving the information they carry with them unaltered. Also, mobility is devised for tokens, i.e., for resources, but not for other model elements.

Complex biological processes such as ontogenesis challenge the limitations of most high-level PN. Ontogenesis comprises architectural and functional system changes across different phases of the same process. These changes include the movement and generation of new parts and decision-making processes based on previous process stages. For example, an embryonic development process can be considered at different system levels, from the organismal structural rearrangements to local molecular and cellular interactions from which morphological and functional patterns emerge at each developmental stage.

The NWN formalism is an extension of PN in which tokens themselves have the structure of a PN. Tokens specified in this way (net-tokens) evolve dynamically, just like the net holding them (system-net). This hierarchical organization can be reiterated in a boundless way, allowing for open recursion in specifying the system's hierarchical organization [9]. The idea of tokens being PN goes back to R. Valk [54], and NWN approaches are extensively studied in the PN literature [28, 33, 47, 9]. The Nets-Within-Nets (NWNs) formalism supports all capabilities of other high-level PN and can face the additional modeling requirements posed by ontogenetic processes [4, 2] as will be explained in the next sections. Among the different flavors of the NWN formalisms, this paper adopts the one provided by the Renew simulator [29, 10, 53]. One of the main motivations behind this choice is that it provides a robust framework that allows extending the NWN formalism with the full power of the Java object-oriented programming language (i.e., Java), thus allowing the implementation of more complex functionalities.

2.3. An introduction to the Nets-Within-Nets formalism

The NWN formalism introduces the possibility to specify tokens in terms of PN models. Such tokens are called *net-tokens*, or *object nets*, while the net holding them takes the name of *system-net*. This schema can grow recursively to an arbitrarily large number of levels: net-tokens from a level may function as system-nets for net-tokens at a lower one.

Figure 2 shows a simple NWN model used to introduce the basic modeling elements exploited in this paper. The figure uses the NWN notation offered by Renew, the simulation environment this work relies on. This notation is

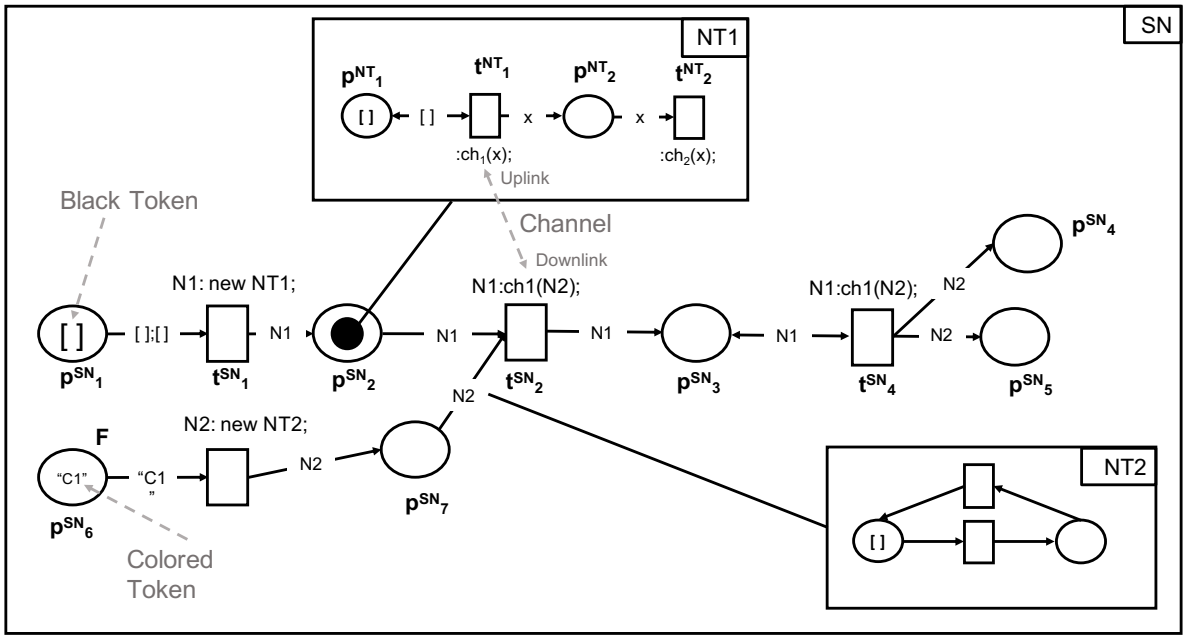

Figure 2: Simple example of the NWN formalism usage. Interested readers can refer to [53] for a complete and detailed description of the formalism.

used consistently throughout the paper. The proposed system is composed of a system-net (SN) in which four types of tokens coexist. As in low-level PN, the simplest category of tokens is black tokens, denoted as [] in place p_1^{SN}. The consumption or creation of tokens by a transition is denoted by inscriptions on its arcs, listing the involved tokens separated by semicolons. Black tokens do not bring specific associated information other than the presence of a unit of a generic resource. Whenever more specific information is required (e.g., a string or an integer value), colored tokens can be used, as shown in place p_6^{SN}. NT1 and NT2 define the architecture of two net-tokens. The inscription N1:new NT1 in transition t_1^{SN} denotes that a new instance N1 of the net NT1 is created, and a reference to this object is then located in place p_2^{SN}. Following the object-oriented paradigm, a reference to the same instance of a net-token can be instantiated in different places. Therefore a net-token can exist at the same time in different places or even in different nets. In Figure 2, the transition t_4^{SN} creates two references to the same instance of the same net-token N2 in places p_4^{SN} and p_5^{SN}. Finally, the bidirectional arc connecting p_3^{SN} with t_4^{SN} denotes that token N1 is used to fire the transition and then moved back to its original place.

This short overview shows how NWN can express a range of possible scenarios. The following section presents the way these capabilities respond to the modeling requirements posed by complex ontogenetic processes.

3. NWN applied to ontogenesis modeling

This section applies the NWN object-oriented paradigm to the problem of modeling ontogenetic processes. It focuses on describing the most relevant and complex biological semantics involved in every ontogenetic process as depicted in Figure 3. After presenting these mechanisms, Appendix A proposes an extensive set of biological processes modeled using NWN. These models represent a library of basic building blocks that can serve to build complex models of generic ontogenetic processes.

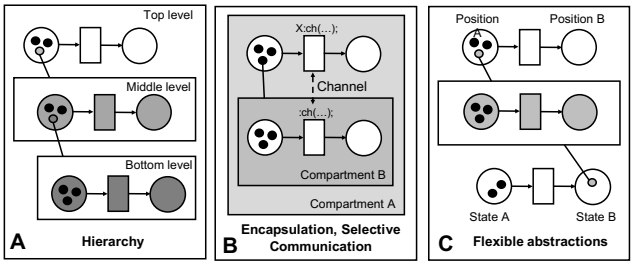

Figure 3: NWN modeling capabilities. NWN models express (A) multiple hierarchical levels; (B) encapsulation of model parts and selective communication between them; (C) coexistent and intertwined regulatory layers with different annotations.

In our approach, the net-tokens represent differentiating biological cells described through their internal regulative networks. The system-net instead represents the landscape of external factors that affect the cells' functioning. The system-net includes the microenvironment (e.g., environmental factors, spatial organization, and relative cell positions) and the relevant developmental phases. The presentation of the proposed modeling strategy resorts to a simple abstract example, depicted in Figure 4. The target biological system is composed of a Petri dish divided into four subspaces (A) where two cells of the same type and able to exchange and react to signals (B) coexist. In the first stage of the process

(A), cell NT1 moves in a subspace closer to cell NT2 (C), and the two cells start exchanging biological signals mediated by mutual activation of signaling mechanisms (D). As a result of this interaction, the process enters a new stage (E) where the cells change their state (F).

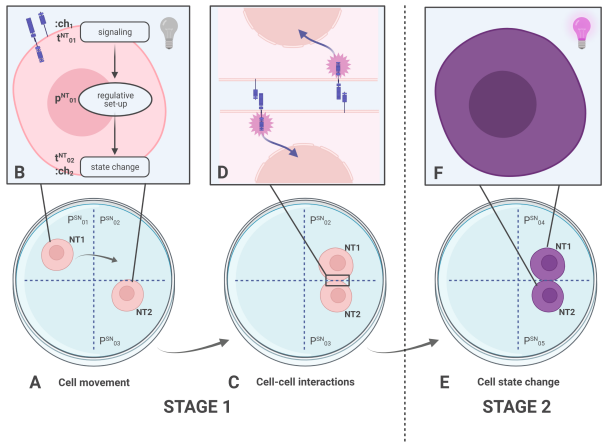

Figure 4: A simple abstract example of an ontogenetic process. The system involves (A) the movement of a cell (NT1) between different subspaces of a Petri dish, considered in reason of (B) the signals it exchanges when in contact with other cells, the regulative set-up it acquires after exchanging signals, and the following changes in cell state. (C) The interactions between cells in spatial proximity, which are mediated by (D) mutual activation of signaling mechanisms, lead to (E) a change in the cells state (F) that moves them to the second stage of the ontogenetic process (E).

Figure 5 shows a simple NWN modeling the mechanisms presented in Figure 4. NT1 and NT2 model the two cells as instances of the net-token (NT). The system-net (SN) models the spatial organization of the Petri dish and the process organization in the two considered stages.

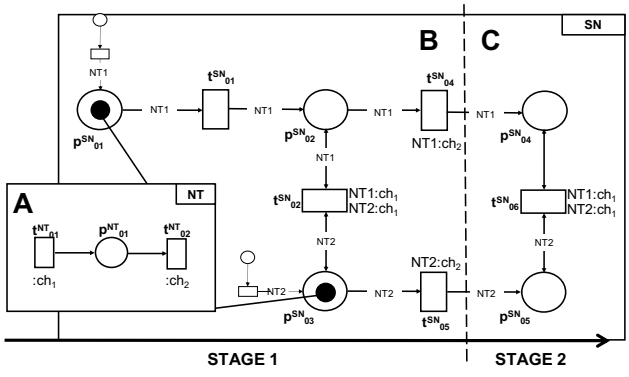

Figure 5: A simple NWN modeling the biological process presented in Figure 4.

3.1. Spatiality and mobility

NWN models can provide an explicit representation of spatiality, using places to represent actual locations that various actors occupy. For example, in Figure 5, places p^{SN}_{01}, p^{SN}_{02}, and p^{SN}_{03} represent the three subspaces of the Petri dish in which cells (net-tokens) can live during stage 1 of the ontogenetic process. Similarly, p^{SN}_{04} and p^{SN}_{05} are the two locations of interest during stage 2.

Interactions between the actors existing at different positions can model proximity-enabled communication between neighbor cells involving communication channels. For example, transitions t^{SN}_{02} and t^{SN}_{06} express proximity between p^{SN}_{02} and p^{SN}_{03}, and p^{SN}_{04} and p^{SN}_{05}, respectively. When these transitions fire, they involve the net-tokens from the connected places, creating mutual interactions. This mechanism can represent juxtacrine interactions between neighbor cells in a biological context, which occur only when the cells are in close spatial proximity.

The movement of net-tokens across system-net places having spatial semantics models mobility of biological entities. For example, in Figure 5, transition t^{SN}_{01} can move net-token NT1 from the place p^{SN}_{01}, representing a position in space, to the place p^{SN}_{02}, representing a different position. In a biological system, this corresponds to a cell's movement from a location to another. In general, biological actors, including cells, are capable of active movement. Developmental processes often comprise cell migration phases across different microenvironments.

Net-tokens that move from one system place to another one change the set of interactions they can engage. When involving places with a spatial connotation, this mechanism models the regulatory action of the spatial context for biological entities. The shift from a biological microenvironment to another corresponds to a regulatory change since it comes with the potential engagement in interactions with a different set of actors.

3.2. Semi-permeability of biological compartments

The object-oriented paradigm implemented by the NWN formalism is powerful to express *encapsulation* and *selective communication* mechanisms. These features, coupled with the ability to handle spatial information, make it easy to describe *compartmentalization* and *semi-permeability* of membranes between biological compartments.

In our modeling approach, the net-tokens describe the inner functioning of cells intended as biological compartments. For example, net-tokens NT1 and NT2 in the system-net of Figure 5 are two instances of the same net class, depicted in Figure 5-A. The system-net can interact with the net-tokens dynamics of both instances through the synchronous channels ch_1 (transition t^{SN}_{02} and t^{SN}_{06} in Figure 5) and ch_2 (transitions t^{SN}_{04} and t^{SN}_{05} in Figure 5). Biological compartments are semi-permeable: communication mechanisms between the net-tokens and the system-net represent the selective permeabilities of biological membranes, including the mechanisms regulating them.

3.3. Inter-cellular communication

In developing multicellular organisms, each cell behaves independently from the others yet can exchange signals and

resources with them. These interactions create the complex and highly dynamical regulative landscape in which cells live and move.

Paracrine and juxtacrine signaling depicted in Figure 6 are among the most relevant intercellular communication mechanisms underlying ontogenetic processes. In biology, paracrine signaling is a form of cell signaling in which a cell produces a signal and sends it to the extracellular environment, affecting nearby cells. Juxtacrine signaling (or contact-dependent signaling) is instead a type of cell signaling that requires close contact.

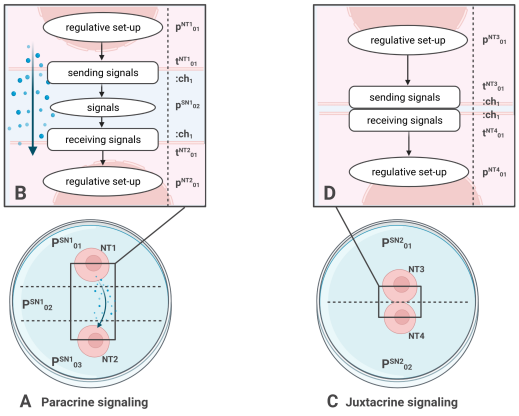

Figure 6: Paracrine and juxtacrine signaling. (A) Paracrine signaling between two cells exchanging a signal through the extracellular environment. (B) The regulative state of cell NT1 causes the cell to secrete a signal in the extracellular environment; cell NT2 receives the signal, which affects its regulative set-up. (C) Juxtacrine signaling between two cells exchanging signals through the intercellular space. (D) The regulative state of cell NT3 causes the cell to secrete a signal, which directly affects the regulative set-up of cell NT4.

The two communication mechanisms can be modeled according to the NWN reported in Figure 7. In paracrine signaling (Figure 7-A) a cell (NT1 in p_{01}^{SN1}) uses channel ch_1 to send a signal represented by a black token in the extracellular environment (p_{02}^{SN1}). The extracellular environment here represented for simplicity as a single place can be also modeled with multiple places, based on the spatial organization of the cells. Another cell NT2 of a different type and in spatial proximity (p_{03}^{SN1}) receives the signal using channel ch_2. In juxtacrine signaling (Figure 7-B), two adjacent cells (NT3 and NT4 in p_{01}^{SN2} and p_{02}^{SN2}, respectively) engage in direct communication through channels ch_1 and ch_2.

These two network motifs are particularly relevant for modeling ontogenetic processes when combined with spatial information. During development, architectural complexity at the supra-cellular level emerges from local interactions between cells, which are often influenced by different distance ranges [38][37].

For example, in paracrine communication, sender cells engage in long-range interactions, diffusing soluble signal molecules in the extracellular environment, targeting receiver

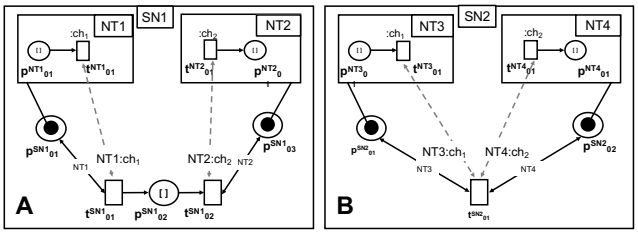

Figure 7: NWN representation of the two signaling mechanisms relevant for modeling ontogenetic processes: (A) Paracrine signaling and (B) Juxtacrine signaling.

cells. The signal starts with a higher concentration close to its source, and the concentration decreases with distance according to diffusion laws specific to the target molecule. Since the reaction to a signal is often dose-dependent, this mechanism translates into a distance-dependent effect over target cells.

In unilateral juxtacrine communication, sender cells communicate with receiver cells in their very proximity through their trans-membrane signal and receptor proteins, respectively. The cells involved in bilateral juxtacrine communication act through the same mechanisms, but each of them acts both as a sender and a receiver. Finally, in autocrine signaling, the cell sends out a signal intended to be self-received, acting both as a sender and as a receiver.

3.4. Dynamic hierarchy

The ability of recursively instantiating net-tokens within another net expresses the hierarchical and dynamic regulatory structures of biological systems (Figure 3.A). Let us consider the two-level model architecture of Figure 5:

- the system-net at the top-level describes the spatial organization and regulative landscape for two cells in two different process phases;
- the net-tokens existing as instances within the system-net represent cells living in such landscape.

One important aspect to highlight is that the hierarchy is not static but instead dynamically defined. At first, net-token instances are created in p_{01}^{SN} and p_{03}^{SN}, respectively. Instantiation per se does not carry any biological meaning, but the resulting marking reflects the system's initial hierarchical organization. The system-net then evolves, and net-tokens can be created or destroyed, simulating the considered biological process, and modifying the system's hierarchical architecture.

Moreover, NWN support the flexible specification of different abstraction layers (Figure 3-C). The same instance of a net-token can live into multiple higher-level nets, each one describing a different aspect of the considered process. This is an important characteristic that poses the basis for multi-level and multi-scale modeling.

Eventually, the cross-layer communication mechanisms presented in the previous sections provide a high degree of flexibility when deciding case by case which layer is in control of the evolution of a specific mechanism.

3.5. Process stages

Ontogenetic processes have sequential stages, with checkpoints involving regulatory states and sets of biological actors governing the passage from a stage to the next. Looking at Figure 5, most of the dynamics considered so far belong to the first of the two stages of the considered process. Transitions t_{04}^{SN} and t_{05}^{SN} can transport net-tokens NT1 and NT2 to p_{04}^{SN} and p_{05}^{SN} , respectively. In this case, the transport represents the passage to a new regulatory setup. To occur, this passage needs some requirements at the net-token level to be satisfied. Both t_{04}^{SN} and t_{05}^{SN} carry a down-link for the synchronous channel ch_2. The up-link of this channel is in transition t_{03}^{NT} of the net-tokens. The channel creates an interplay between the two layers. To move a net token from p_{02}^{SN} to p_{04}^{SN} and therefore advance to the next process stage, it is not enough to have a net-token in a specific location of the system-net. The net-token must also satisfy an internal condition, i.e., one token must be present in p_{02}^{NT}. This condition represents the checkpoint for this developmental stage that models the gatekeeper mechanisms organizing ontogenetic processes into subsequent phases, i.e., some phenotypic traits need to be expressed by the cells for them to access the next stage. The ways each checkpoint evaluates net-tokens evolution and state can range from a simple read of a static value in the net to dynamic tracking to extract complex information about cell phenotype. The checkpoint may then include robust decision-making tools implementing classification routines able to handle complex information and dynamically label complex net-token behaviors.

3.6. Dynamic regulatory landscape

The organization in different process stages also contributes to the definition of a dynamic regulatory landscape. If places p_{04}^{SN} and p_{05}^{SN} represent the counterpart of p_{02}^{SN} and p_{03}^{SN} in a new developmental stage, the movement of a cell (net-token) from one stage to another one changes the interaction schema that engages the cell. This modeling approach covers both intracellular (net-token) and supra-cellular (system-net) regulations, the latter intended as any regulation existing on top of intracellular regulation, including environmental and epigenetic context. Also, it allows expressing a priority scheme among such regulation layers and how that may change in different stages of the same process. This modeling approach allows representing a complex regulative landscape. Each point corresponds to a particular set of interactions that may take place in a specific process phase and context, and each cell can undergo multiple paths across it, passing through different process stages.

4. Results and discussion

In this section, we apply our modeling approach to a well-characterized developmental process: the Vulval Precursor Cells (VPC) specification in the development of *C. Elegans* larva. This aims to show the proposed modeling strategy at work providing an overview of its capabilities. Differently from other models of the same process proposed in the literature (e.g., [8]), the proposed model includes different hierarchical levels, explicitly combines spatial information with cell differentiation and cell interaction, taking advantage from the proposed NWN modeling methodology.

4.1. Biological process

VPC specification in *C. Elegans* is an ontogenetic process involving a small number of cells but still including all characteristics of more complex processes. Also, it is one of the most widely characterized processes of this kind at the intracellular level.

As extensively described in [52, 49], VPC specification occurs between the L3 and L4 stages of larval development in *C. Elegans*. At this stage, each of six multi-potent stem cells, the Pn.p cells, acquire one of three fates (1°, 2° or 3° fate), which guide the subsequent phases of organ development. Different actors contribute to fate decision: the Anchor Cell (AC), residing in the adjacent developing uterus district, the underlying hypodermal syncytium, and the neighbor Pn.p cells.

More specifically, as in Figure 8, in the physiological case the AC sends out a LIN-3 (EGF-like) signal reaching Pn.p cells with distance-dependent intensity: the closest cell, P6.p, receives juxtacrine signaling, its neighbors P5.p, and P7.p cells receive paracrine signaling. The signal does not target P3.p, P4.p, and P8.p, the farthest cells. The hypodermal syncytium (hyp7) sends uniformly intense paracrine LIN-3 signals to all Pn.p cells. Each of the Pn.p cells can engage in mutual juxtacrine lateral signaling via trans-membrane DSL/LIN-12 (DSL/Notch-like) signaling. At the intracellular level, intense LIN-3 signaling induces the 1° fate in P6.p via the activation of a LET-23-mediated RAS/MAPK signaling pathway. This condition is marked by high concentrations of the active form of MPK-1, which activates strong DSL lateral signaling to the neighbors. This activation causes them to switch off the 1° fate traits induced by LIN-3 paracrine signals from the AC, activating 2° fate traits, corresponding to high concentrations of the active LIN-12 protein. P3.p, P4.p, and P8.p cells do not receive any LIN-3 other than that from hyp7, this causing them to undergo the 3° fate.

Different non-physiological cases for VPC specification exist due to various mechanisms, including genetic mutations.

4.2. The model

The NWN model of the VPC specification process comprises the following elements:

- a bi-dimensional Interactive Spatial Grid (ISG) (Figure 9);
- a Pn.p specific Differentiative Landscape (DL) (Figure 10) with complex checkpoint functionalities;
- three cell models, for the Anchor Cell, the Pn.p cells and the hypodermal syncytium (hyp7) (Figure 13, Figure 14 and Figure 15, respectively).

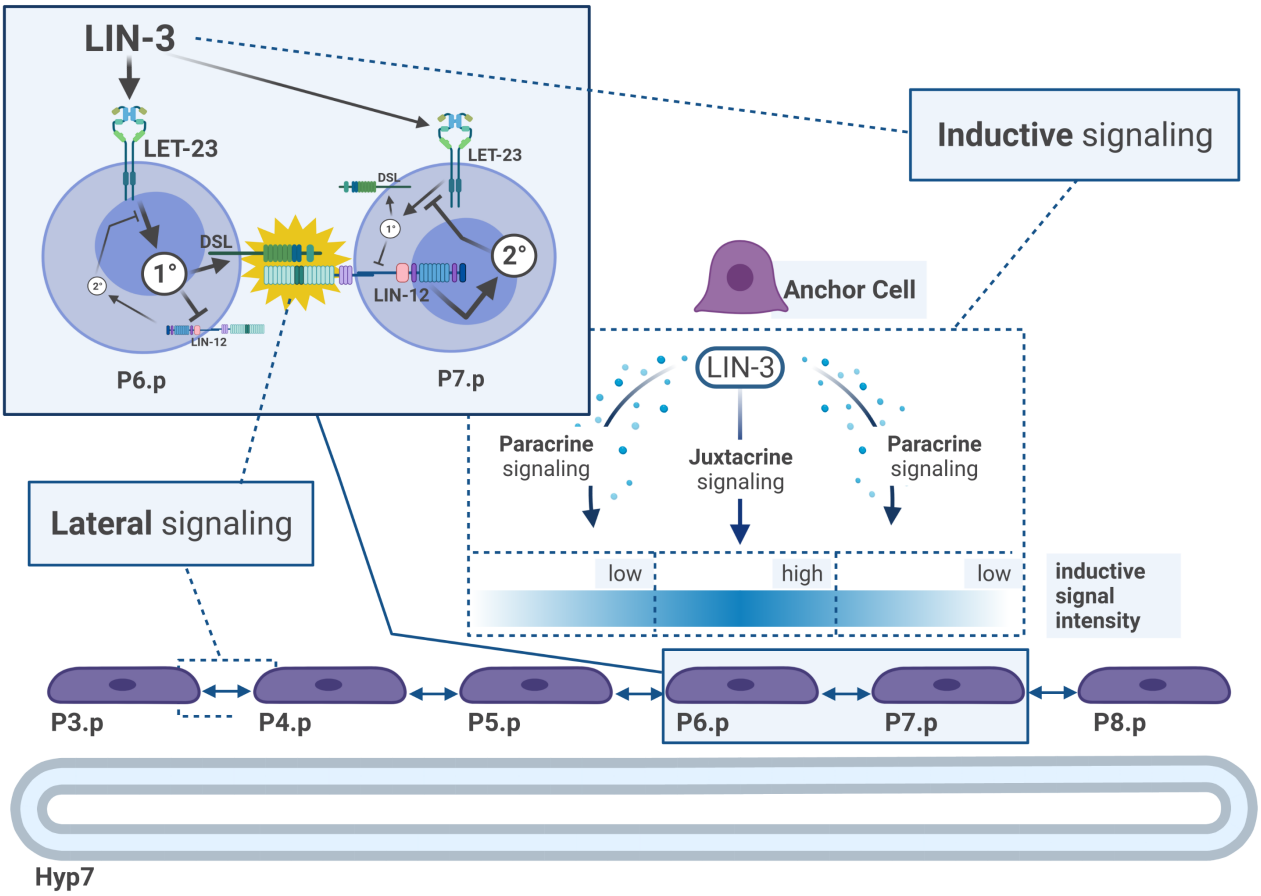

Figure 8: Diagram of signaling mechanisms involved in VPC pattern formation.

For readability, this paper resorts to simplified figures helping the reader understand the modeling approach. The full model and instructions to reproduce the experiments are instead provided in a public GitHub repository at `https://github.com/sysbio-polito/NWN_CElegans_VPC_model/`.

Figure 11 helps to understand the hierarchical organization of the model's elements. The system-net at the top-level combines ISG and DL functionalities. The nets that model the cells and implement the rules for classification of Pn.p cell fates (Fates Manager net) live and interact as net-tokens within this system-net.

Figure 12 reports a screenshot of the full system-net model implementation. The figure highlights the different portions of this model using the dashed boxes, while the complete source file is available on GitHub.

The top part of Figure 12 represents the ISG depicted in a more schematic way in Figure 9. It provides a two-dimensional representation of the VPC specification spatial environment and models the involved cells' interactions. In this model, places represent the positions cells can occupy, and transitions model the potential interactions between cells in such positions (e.g., cell-cell communication mechanisms involved in the emergence of the morphogenetic pattern). Hyp7 sends LIN-3 paracrine signals uniformly to all Pn.p cells, modeling that the hypodermal syncytium lies under the array of Pn.p cells.

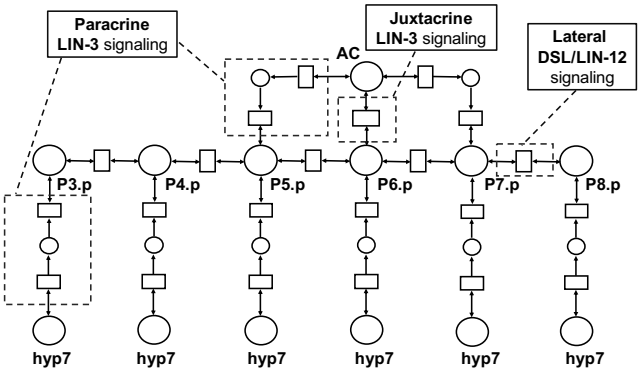

Figure 9: A schematic representation of the Interactive Spatial Grid (ISG) model for the VPC specification example, highlighting communication building blocks: *juxtacrine* LIN-3 signaling from the Anchor Cell (AC) to the P6.p cell; *paracrine* LIN-3 signaling from the Anchor Cell (AC) to the P5.p and P7.p cells; *neighbor* communication between pairs of Pn.p cells; *paracrine* LIN-3 signaling from the hypodermal syncytium (hyp7) to the Pn.p cells.

The DL model (Figure 10) represents instead the mecha-

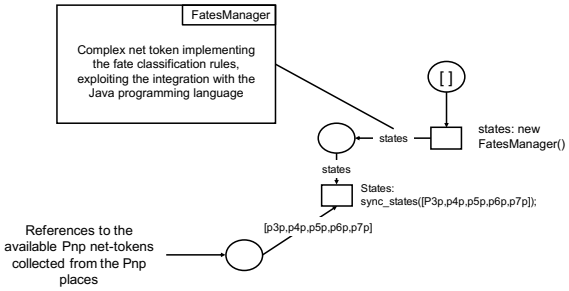

Figure 10: A schematic representation of the DL model for the VPC specification example.

nism governing the Pn.p cells' states along the developmental step from L3 to L4. Possible conditions are: Pn.p state, and Primary (1°), Secondary (2°), or Tertiary (3°) fates. Since this developmental stage does not imply a change in the system's architecture, the different fates are modeled as colors of the net-tokens representing the Pn.p cells. From the Pn.p places holding the cells, specific transitions create an array of references to the Pn.p token-nets. This array is passed to the Fates Manager net-token using the sync_state channel. The Fates Manager is a special token-net that monitors the places modeling active MPK-1 and active LIN-12 of the Pn.p net-tokens and uses this information to predict the cell's fate. The bottom part of Figure 12 represents the actual implementation of this mechanism connected to the remaining parts of the system-net.

In the presented implementation, the system net holds three main types of net-tokens.

The AC net-token describes the LIN-3 production and signaling in the Anchor Cell as reported in Figure 13. The AC Cell model has two LIN-3 communication modes implemented using channels: one for *juxtacrine* signaling with P6.p (*neighbor communication*), and the other one for *paracrine* signaling with P5.p and P7.p (*signal sending*). Both signals rely on the LIN-3 gene for production (*transcription and translation*).

The Pn.p net-token describes the regulation of the Pn.p cells, all sharing the same architecture. This net-token is described using an adaptation of the Petri Nets model from [8]. As shown in Figure 14, the Pn.p cell model receives LIN-3 signals through channels that describe *paracrine* or *juxtacrine* signaling (*signal sensing* and *neighbor communication*). LIN-3 activates a cascade of enzymatic activation reactions (the MAPK signaling cascade, *enzymatic reactions*), resulting in the production of active MPK-1. This active protein causes the DSL signal to increase and affect neighbor cells. Pn.p Cells receive DSL signals from neighbors, and this activates LIN-12 (DSL/LIN-12 lateral signaling via *neighbor communication*). Increased active LIN-12 causes LST and DPY23 inhibitors to switch off the MAPK signaling cascade (*inhibitory post-transcriptional regulation*).

Eventually, the hyp7 net-token describes the hypodermal syncytium lying below the Pn.p cells. It provides uniform and low-intensity LIN-3 paracrine signals to all of them (*signal sending*), modeling the fact that LIN-15 can shut down LIN-3 transcription (*inhibitory transcriptional regulation*), as described in [8].

The Fates Manager (FM) net-token is a complex net able to assign one of the three possible fates (Primary, Secondary and Tertiary Fate) to the different Pn.p cells. The full network is available on GitHub. However, for the sake of readability, Figure 16 focuses on presenting the overall behavior of this network.

The first task of the FM is sampling the marking evolution of the MPK1_ACT and LIN12_ACT places of all Pn.p cells. A channel transfers an array of Pn.p net-token object references to the FM that can then communicate with them to obtain their marking. The ability to track the dynamic evolution of places available in various instances of a net-token is a crucial characteristic of this modeling strategy. Apart from its usage in the presented model, this mechanisms can be used as a general and powerful inspection instrument to understand the considered process.

The FM stores the marking evolution of the MPK1_ACT and LIN12_ACT places of each cell into a set of circular buffers, creating a sliding window where the markings of the six Pn.p cells are stored during the simulation. This information builds a set of features for each specific cell submitted to a classifier able to determine the fate of the cell (Figure 16.B). The resulting fates are sent to the system-net (Figure 16.C) and used to assign the corresponding color to the Pn.p cells.

The classification process relies on a Weka random forest model [25]. The classification model is trained with a supervised mechanism. A training dataset is generated by collecting the marking of the MPK1_ACT and LIN12_ACT places in several simulations of the Pn.p net-tokens in a physiological condition. As will be detailed later in this section, simulations include stochastic behaviors that lead to different dynamics. Each set of features was manually labeled to a fate by comparing the evolution of the MPK1_ACT and LIN12_ACT markings with the expected values from biological knowledge. In particular, cells exhibiting high values of MPK1_ACT and low values of LIN12_ACT were labeled with the Primary fate, cells having low MPK1_ACT and high LIN12_ACT with the Secondary fate, and cells having low levels of both markers with the Tertiary fate.

4.3. Model tuning and simulations

The proposed implementation of the Pn.p, AC, and hyp7 net-tokens follows the semi-quantitative paradigm proposed in [8]. The tuning proceeded manually by trial-and-error until the marking evolution of markers recapitulated physiology for all cell models.

Renew supports different high-level Petri Nets formalisms and their simulation. Experiments use the Java Net Compiler [10], which supports stochastic simulation of NWN models. The simulation engine selects all enabled transitions at each cycle and then selects a subset of them randomly for activation. This mechanism makes the transitions activation order non-deterministic, mimicking the stochasticity of the biological processes. In other words, this means that starting from

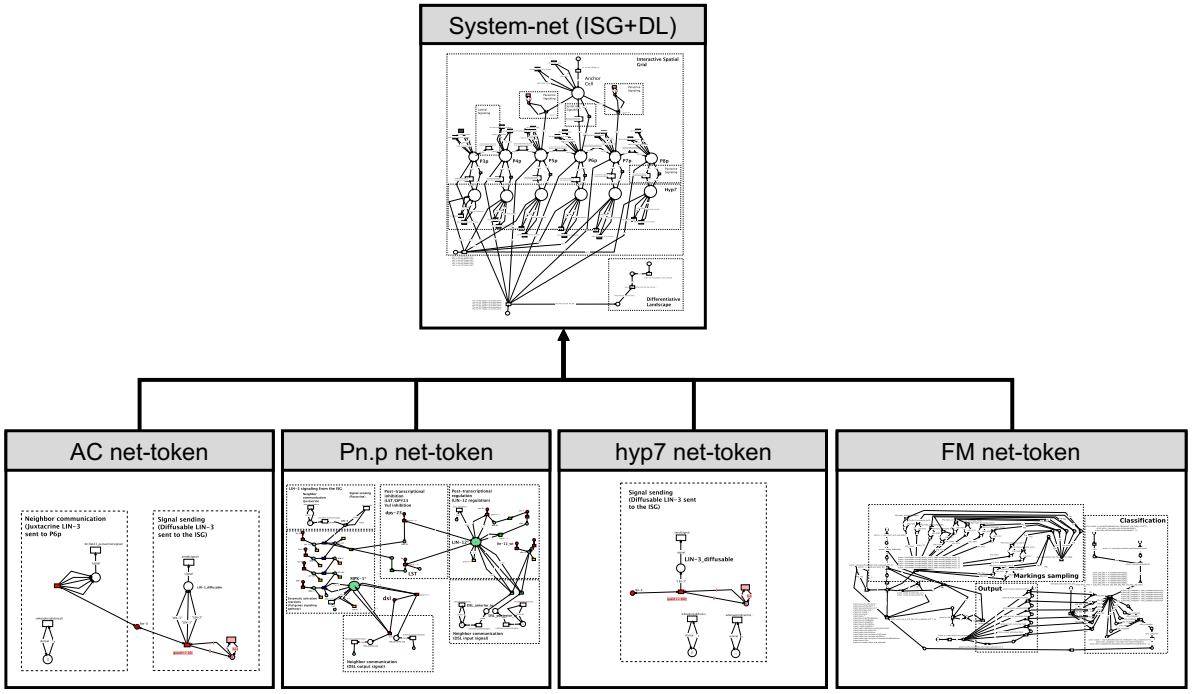

Figure 11: Nets hierarchy for the VPC specification model implementation in Renew. The system net having ISG and DL functionalities is at the top level, and it hosts the net tokens modeling different cells (AC, Pn.p and hyp7), plus the Fates Manager (FM) modeling complex checkpoint rules.

the same initial conditions, simulations of the same model may generate different outcomes.

Moreover, the model structure introduces delays for some mechanisms to provide better expressivity of graded signaling intensity. For example, expressing the delayed onset of *paracrine* signals from the AC compared to *juxtacrine*.

This modeling approach expresses temporal dynamics, which are relevant for modeling ontogenesis since the timing of functional activation often has a regulatory meaning in that context.

Experiments involved two simulation campaigns.

The first simulation campaign aimed at collecting features to train the FM classifier. In this campaign, the Weka classifier was detached from the model. The FM only collected the features from the marking of the Pn.p cells and stored them in a file. Collected data were then labeled and used to train the model. Given the manual work required in this fase, a small training set of 63 experiments was collected (see cells_fated_v3.arff file on GitHub). Nevertheless, even wit only few training data the accuracy of the model is acceptable.

The second campaign shows the model at work.

Different initial markings represent different experimental conditions in the model, chosen by the scheme proposed in [8]. In particular, this paper considers the following conditions:

1. *wt*: the wild-type physiological condition;
2. *lin_ko*: knock-out of the lin-12 gene;
3. *lst_lf*: knock-out of the lst set of genes (lip-1, lst-1, lst-2, lst-3 and lst-4);
4. *vul_ko*: knock-out of the Vul set of genes (let-23, sem-5, let-60 and mpk-1).

For each experimental condition, 100 simulations were collected following recommendations in [44]. Simulations provide as outputs both the temporal evolution of the markers of interest from cell models and the six labels resulting from Pn.p cells classification.

Table 1 summarizes, for each cell, the ability of the model to predict the correct fate of the different cells in the considered conditions. It is important to remark that the model was trained and tuned on the *wt* case and then used to evaluate new unknown conditions. This is a simple example to show how this modeling strategy can be exploited to inspect different conditions, therefore, performing in-silico knowledge exploration.

5. Conclusions and future perspectives

This paper presented an NWN-based modeling approach responding to modeling requirements posed by ontogenesis applied to the case study of a VPC specification in C. Elegans. Considering model performance in terms of predictive power, compared to other state-of-the-art approaches [8, 13, 26, 15], our modeling framework has similar or better predictive performance, together with the advantage of supporting the integration and composition of heterogeneous biological information and models, combining the advantages

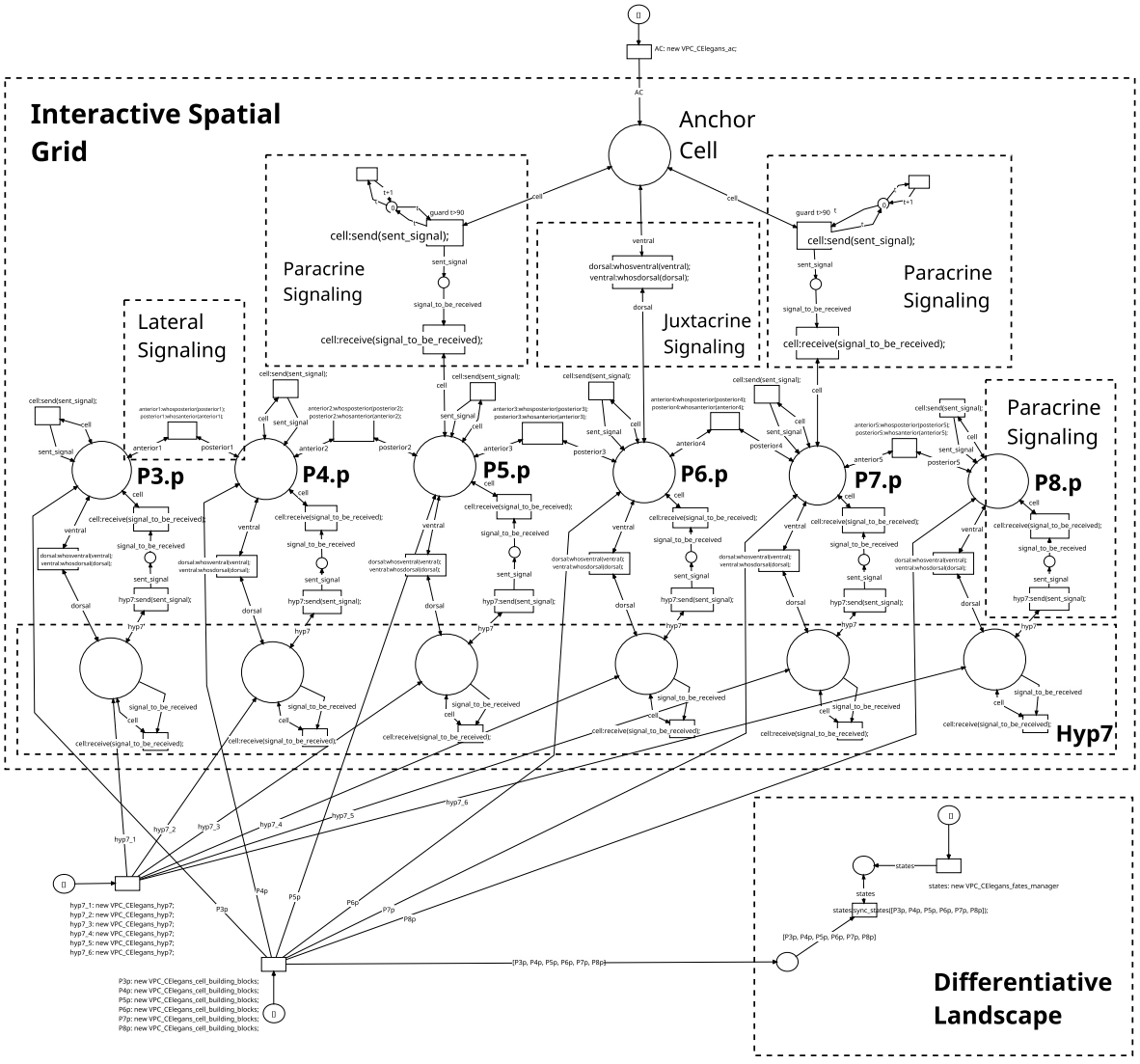

Figure 12: The Renew implementation of the system net for the VPC specification example, combining ISG and DL functionalities.

experiment	Pn.p pattern	accuracies
wt	3 3 2 1 2 3	100% 95% 82% 94% 81% 96%
lin12_ko	3 3 1 1 1 3	100% 100% 87% 100% 97% 100%
lst_lf	3 3 1 1 1 3	99% 90% 86% 100% 83% 81%
vul_ko	3 3 3 3 3 3	100% 100% 100% 100% 100% 100%

Table 1
Results of the pattern formation simulations. For each experimental condition, the Pn.p fates pattern observed is reported, together with the accuracy of the prediction for each cell generated in the simulation outcomes. Reported results refer to 100 simulation runs.

of multi-level hybrid models [3] with formalism uniformity, which facilitates model analysis, and knowledge representation and exchange.

In the future, performance needs to be tested over a larger number of mutations for the VPC specification case, starting from those tested in [7], and performing predictions over unexplored experimental conditions to be subsequently verified experimentally, in order to better demonstrate the actual predictive power of the model.

At the moment, NWN models have been mainly used through simulation since the formal analysis of such complex graphs is not a trivial task due to the state-space explosion problem [20]. In the future, we intend to apply or develop formal analysis and model checking techniques specifically designed for NWN models, like the one presented in [55], so to gain deeper insights into the modeled system.

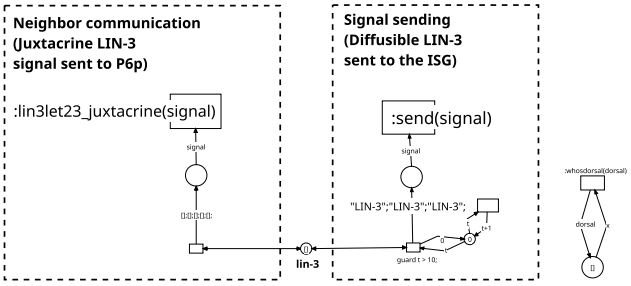

Figure 13: The AC Cell model.

Our models combine hypothesis-driven and data-driven approaches, but a necessary improvement is to make parameter identification rely on experimental data only, possibly generating them *ad hoc*.

Computational capabilities could limit the model complexity: if computational complexity is too high, analyses and simulations take too long to complete [40]. It is possible to face this problem both at the model level, with complexity reduction [1], and at the hardware level, with state-of-the-art architectures for parallel and distributed computing to speed up computational times [16, 18].

Finally, significant effort is required to improve the modeling strategy usability from life scientists with limited experience in model development and computer engineering. An effort in this direction is already in place with the development of the Biological System Description Language (BiSDL), a domain-specific high-level, modular programming language naturally accessible by purely biological semantics and automatically generating simulation-ready NWN models [35].

In this way, we intend to improve the capability of modeling biological complexity and make the resulting tools available to a large and diverse user base to fulfill the systems biology scientific community's needs [42].

References

[1] Antoulas, A.C., Sorensen, D.C., Gugercin, S., 2001. A survey of model reduction methods for large-scale systems. Contemporary mathematics 280, 193–220.

[2] Bardini, R., Politano, G., Benso, A., Carlo, S.D., 2017a. Using multi-level petri nets models to simulate microbiota resistance to antibiotics, in: 2017 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), pp. 128–133. doi:10.1109/BIBM.2017.8217637.

[3] Bardini, R., Politano, G., Benso, A., Di Carlo, S., 2017b. Multi-level and hybrid modelling approaches for systems biology. Computational and structural biotechnology journal 15, 396–402.

[4] Bardini, R., Politano, G., Benso, A., Di Carlo, S., 2018. Computational tools for applying multi-level models to synthetic biology, in: Synthetic Biology. Springer, pp. 95–112.

[5] Bartocci, E., Bortolussi, L., Milios, D., Nenzi, L., Sanguinetti, G., 2015. Studying emergent behaviours in morphogenesis using signal spatio-temporal logic, in: International Workshop on Hybrid Systems Biology, Springer. pp. 156–172.

[6] Bartocci, E., Lió, P., 2016. Computational modeling, formal analysis, and tools for systems biology. PLoS computational biology 12, e1004591.

[7] Bonzanni, N., Feenstra, K.A., Fokkink, W., Heringa, J., 2014. Petri nets are a biologist's best friend, in: International Conference on Formal Methods in Macro-Biology, Springer. pp. 102–116.

[8] Bonzanni, N., Krepska, E., Feenstra, K.A., Fokkink, W., Kielmann, T., Bal, H., Heringa, J., 2009. Executing multicellular differentiation: quantitative predictive modelling of c.elegans vulval development. Bioinformatics 25, 2049–2056. URL: http://dx.doi.org/10.1093/bioinformatics/btp355, doi:10.1093/bioinformatics/btp355.

[9] Cabac, L., Duvigneau, M., Moldt, D., Rölke, H., 2005. Modeling dynamic architectures using nets-within-nets, in: International Conference on Application and Theory of Petri Nets, Springer. pp. 148–167.

[10] Cabac, L., Haustermann, M., Mosteller, D., 2016. Renew 2.5–towards a comprehensive integrated development environment for petri net-based applications, in: International Conference on Applications and Theory of Petri Nets and Concurrency, Springer. pp. 101–112.

[11] Carvalho, R.V., Verbeek, F.J., Coelho, C.J., 2018. Bio-modeling using petri nets: a computational approach, in: Theoretical and Applied Aspects of Systems Biology. Springer, pp. 3–26.

[12] David, R., Alla, H., 2010. Timed Continuous Petri Nets. Springer Berlin Heidelberg, Berlin, Heidelberg. pp. 159–229. URL: https://doi.org/10.1007/978-3-642-10669-9_5, doi:10.1007/978-3-642-10669-9_5.

[13] Dawes, A.T., Wu, D., Mahalak, K.K., Zitnik, E.M., Kravtsova, N., Su, H., Chamberlin, H.M., 2017. A computational model predicts genetic nodes that allow switching between species-specific responses in a conserved signaling network. Integrative Biology 9, 156–166.

[14] Fisher, J., Henzinger, T.A., 2007. Executable cell biology. Nature biotechnology 25, 1239.

[15] Fisher, J., Piterman, N., Hajnal, A., Henzinger, T.A., 2007. Predictive modeling of signaling crosstalk during c. elegans vulval development. PLoS computational biology 3, e92.

[16] Fox, G.C., Williams, R.D., Messina, G.C., 2014. Parallel computing works! Elsevier.

[17] Frank, T., Beek, P., 2001. Stationary solutions of linear stochastic delay differential equations: Applications to biological systems. Physical Review E 64, 021917.

[18] Fujimoto, R., 2015. Parallel and distributed simulation, in: Proceedings of the 2015 Winter Simulation Conference, IEEE Press. pp. 45–59.

[19] Goss, P.J., Peccoud, J., 1998. Quantitative modeling of stochastic systems in molecular biology by using stochastic petri nets. Proceedings of the National Academy of Sciences 95, 6750–6755.

[20] Groote, J.F., Kouters, T.W., Osaiweran, A., 2015. Specification guidelines to avoid the state space explosion problem. Software Testing, Verification and Reliability 25, 4–33.

[21] Guglielmi, G., Renzis, S.D., 2017. Chapter 9 - optogenetic inhibition of apical constriction during drosophila embryonic development, in: Lecuit, T. (Ed.), Cell Polarity and Morphogenesis. Academic Press. volume 139 of *Methods in Cell Biology*, pp. 167 – 186. URL: http://www.sciencedirect.com/science/article/pii/S0091679X16301509, doi:https://doi.org/10.1016/bs.mcb.2016.10.007.

[22] Huang, S., 2012. The molecular and mathematical basis of waddington's epigenetic landscape: A framework for post-darwinian biology? Bioessays 34, 149–157.

[23] Jensen, K., 1987. Coloured Petri nets. Springer Berlin Heidelberg, Berlin, Heidelberg. pp. 248–299. URL: https://doi.org/10.1007/BFb0046842, doi:10.1007/BFb0046842.

[24] Jones, D.S., Plank, M., Sleeman, B.D., 2009. Differential equations and mathematical biology. CRC press.

[25] Kalmegh, S.R., 2015. Comparative analysis of weka data mining algorithm randomforest, randomtree and ladtree for classification of indigenous news data. International Journal of Emerging Technology and Advanced Engineering 5, 507–517.

[26] Kam, N., Kugler, H., Marelly, R., Appleby, L., Fisher, J., Pnueli, A., Harel, D., Stern, M.J., Hubbard, E.J.A., 2008. A scenario-based approach to modeling development: a prototype model of c. elegans vulval fate specification. Developmental biology 323, 1.

[27] Koch, I., 2015. Petri nets in systems biology. Software & Systems Modeling 14, 703–710.

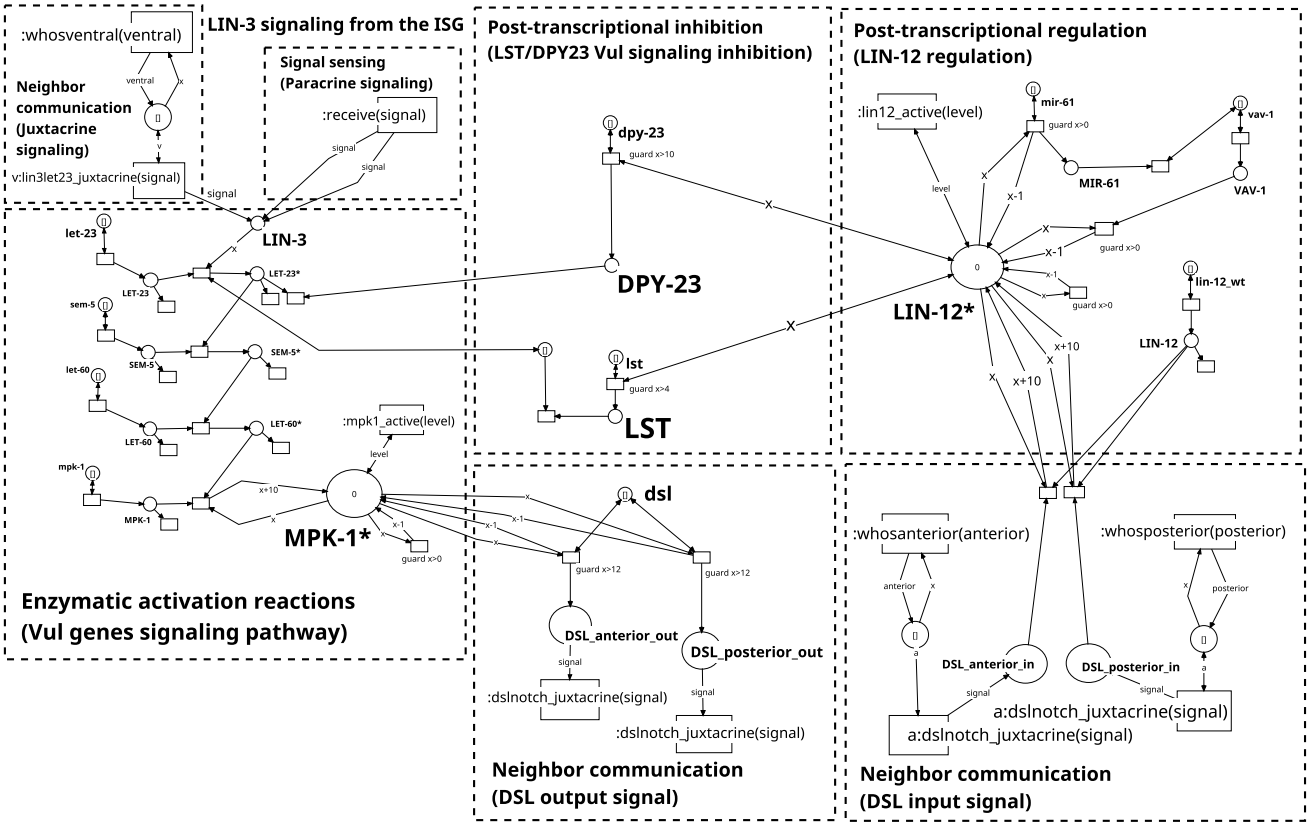

Figure 14: The Pn.p Cell model.

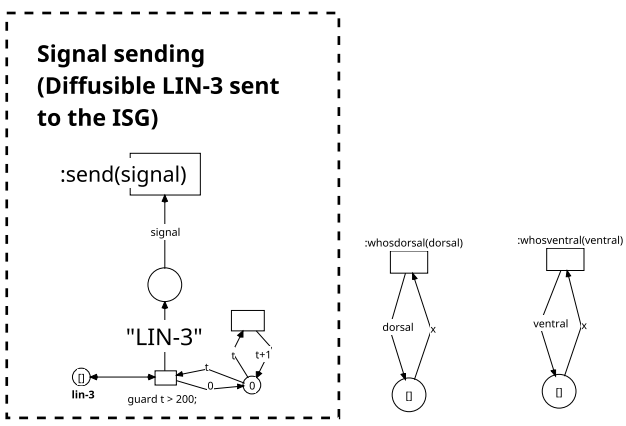

Figure 15: The hyp7 Cell model.

[28] Köhler, M., Moldt, D., Rölke, H., 2003. Modelling mobility and mobile agents using nets within nets, in: International Conference on Application and Theory of Petri Nets, Springer. pp. 121–139.

[29] Kummer, O., Wienberg, F., Duvigneau, M., Schumacher, J., Köhler, M., Moldt, D., Rölke, H., Valk, R., 2004. An extensible editor and simulation engine for petri nets: Renew, in: International Conference on Application and Theory of Petri Nets, Springer. pp. 484–493.

[30] Liu, F., Heiner, M., 2010. Colored petri nets to model and simulate biological systems. Recent advances in Petri Nets and concurrency 827, 71–85.

[31] Liu, F., Heiner, M., Gilbert, D., 2017. Coloured petri nets for multilevel, multiscale and multidimensional modelling of biological systems. Briefings in bioinformatics .

[32] Liu, F., Heiner, M., Yang, M., 2016. Fuzzy stochastic petri nets for modeling biological systems with uncertain kinetic parameters. PloS one 11, e0149674.

[33] Lomazova, I.A., 2008. Nested petri nets for adaptive process modeling, in: Pillars of computer science. Springer, pp. 460–474.

[34] Marwan, W., Wagler, A., Weismantel, R., 2011. Petri nets as a framework for the reconstruction and analysis of signal transduction pathways and regulatory networks. Natural Computing 10, 639–654.

[35] Muggianu, F., Benso, A., Bardini, R., Hu, E., Politano, G., Di Carlo, S., 2018. Modeling biological complexity using biology system description language (bisdl), in: 2018 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), IEEE. pp. 713–717.

[36] Narayanan, M., Cherukuri, A.K., 2018. Verification of cloud based information integration architecture using colored petri nets. International Journal of Computer Network and Information Security 10, 1.

[37] Nelson, C.M., 2009. Geometric control of tissue morphogenesis. Biochimica et Biophysica Acta (BBA)-Molecular Cell Research 1793, 903–910.

[38] Nelson, C.M., Jean, R.P., Tan, J.L., Liu, W.F., Sniadecki, N.J., Spector, A.A., Chen, C.S., 2005. Emergent patterns of growth controlled by multicellular form and mechanics. Proceedings of the National Academy of Sciences of the United States of America 102, 11594–11599.

[39] Nepal, C., Hadzhiev, Y., Previti, C., Haberle, V., Li, N., Takahashi, H., Suzuki, A.M.M., Sheng, Y., Abdelhamid, R.F., Anand, S., et al., 2013. Dynamic regulation of the transcription initiation landscape at single nucleotide resolution during vertebrate embryogenesis. Genome research 23, 1938–1950.

[40] Papadimitriou, C.H., 2003. Computational complexity. John Wiley and Sons Ltd.

[41] Petri, C.A., 1966. Communication with automata. Ph.D. the-

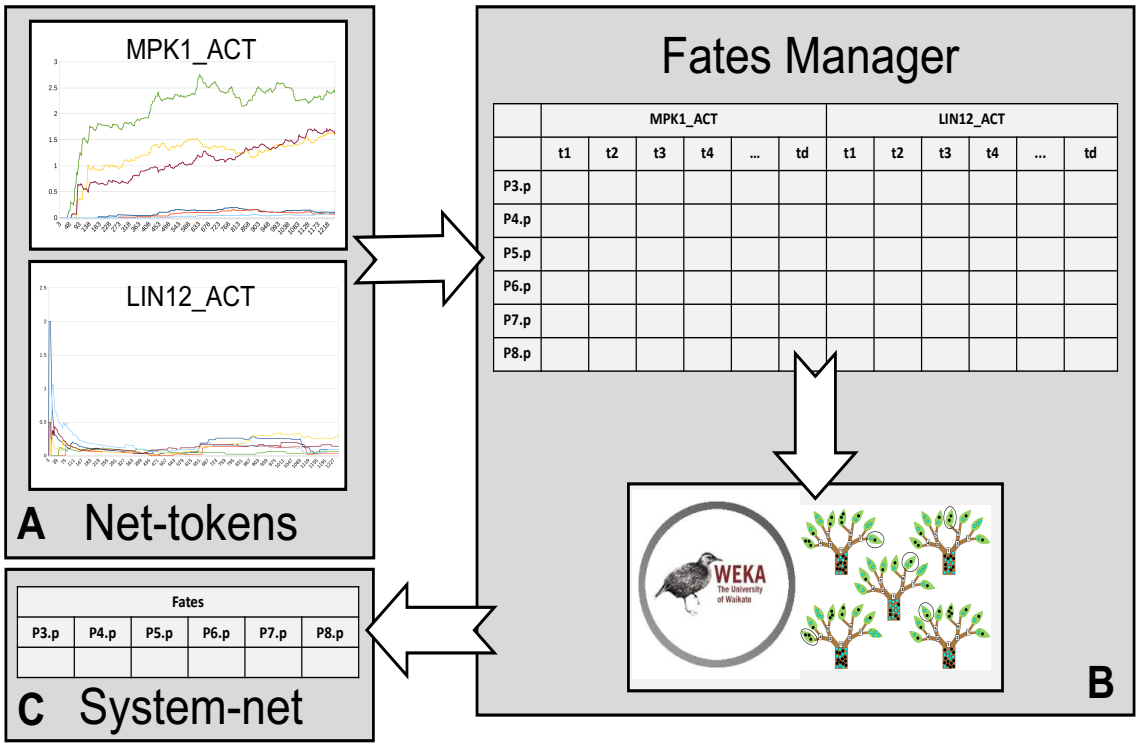

Figure 16: Overview of the cell fate classification process in the Fates Manager.

sis. Universität Hamburg. URL: https://edoc.sub.uni-hamburg.de/informatik/volltexte/2010/155/.

[42] R, B., 2019. A diversity-aware computational framework for systems biology. Ph.D. thesis. Politecnico di Torino.

[43] Reisig, W., 2012. Petri nets: an introduction. volume 4. Springer Science & Business Media.

[44] Ritter, F.E., Schoelles, M.J., Quigley, K.S., Klein, L.C., 2011. Determining the Number of Simulation Runs: Treating Simulations as Theories by Not Sampling Their Behavior. Springer London, London. pp. 97–116. URL: https://doi.org/10.1007/978-0-85729-883-6_5, doi:10.1007/978-0-85729-883-6_5.

[45] Rogojin, V., Petre, I., 2016. Large-scale executable biology using rapid integration of computational models. Computer Science Journal of Moldova 24.

[46] Said, H., 2018. Physiology of the Gastrointestinal Tract. Elsevier Science. URL: https://books.google.it/books?id=2_lQDwAAQBAJ.

[47] Sanchez-Herrera, R., Lopez-Mellado, E., 2004. Modular and hierarchical modeling of interactive mobile agents, in: 2004 IEEE International Conference on Systems, Man and Cybernetics (IEEE Cat. No.04CH37583), pp. 1740–1745 vol.2. doi:10.1109/ICSMC.2004.1399889.

[48] Scheidel, J., Lindauer, K., Ackermann, J., Koch, I., 2015. Quasi-steady-state analysis based on structural modules and timed petri net predict system's dynamics: The life cycle of the insulin receptor. Metabolites 5, 766–793.

[49] Schmid, T., Hajnal, A., 2015. Signal transduction during c. elegans vulval development: a neverending story. Current opinion in genetics & development 32, 1–9.

[50] Schulz-Trieglaff, O., 2005. Stochastic petri nets in systems biology. BMC Bioinformatics 6, P25.

[51] Scott, R.J., K., H.B., M., O.W., 2001. Bridging the gap between developmental systems theory and evolutionary developmental biology†. BioEssays 23, 954–962. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/bies.1136, doi:10.1002/bies.1136, arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/bies.1136.

[52] Sternberg, P., 2005. Vulval development (june, 25 2005), wormbook. URL: http://www.wormbook.org.

[53] Theoretical Foundations Group, Univ. of Hamburg, . The reference net workshop. [online] http://www.renew.de/.

[54] Valk, R., 1987. Nets in computer organization, in: Brauer, W., Reisig, W., Rozenberg, G. (Eds.), Petri Nets: Applications and Relationships to Other Models of Concurrency, Springer Berlin Heidelberg, Berlin, Heidelberg. pp. 218–233.

[55] Venero, M.L.F., da Silva, F.S.C., 2017. Model checking multi-level and recursive nets. Software & Systems Modeling 16, 1117–1144.

Roberta Bardini is a post-doc researcher at Department of Control and Computer Engineering of Politecnico di Torino. Her research focuses on computational approaches to understand and represent complex biological systems.

Alfredo Benso received the MS degree in computer engineering and the PhD degree in information technologies, both from Politecnico di Torino, Italy, where he is working as a tenured associate professor of computer engineering. His current research interests include bioinformatics, and system biology. He is also actively involved in the Computer Society, where he has been a leading volunteer for several projects. He is a Computer Society Golden Core Member, and a senior member of the IEEE.

Gianfranco Politano is an Assistant Professor in the department of Control and Computer Engineering at Politecnico di Torino (Italy). He holds a Ph.D. and an M.S. equivalent in Computer Engineering and Information Technology from the Politecnico di Torino. Politano's research contributions include complex biological networks analysis, data fusion and statistical data analysis.

Stefano Di Carlo is an associate professor in the department of Control and Computer Engineering at Politecnico di Torino (Italy) since 2014. He holds a Ph.D. (2003) and an M.S. equivalent (1999) in Computer Engineering and Information Technology from the Politecnico di Torino in Italy. Di Carlo's research contributions include biological network analysis and simulation, machine learning, image processing and evolutionary algorithms, as well as Reliability, Memory Testing and BIST.

A. Appendix 1

This appendix reports a library of biological functional modules and their relative NWN model that can be used to model complex ontogenetic processes.

A.1. Interactive spatial grid

This section reports a set of cell movement and cell to cell communication mechanisms mediated by the relative spatial positions modeled in the ISG.

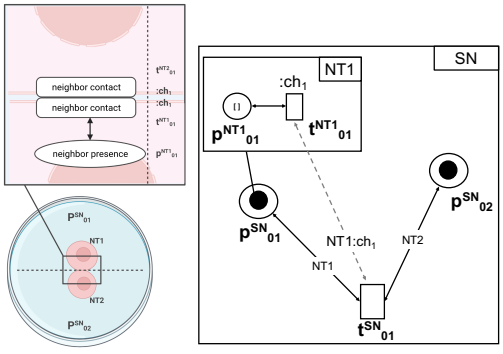

Figure 17: Neighbor detection: a cell detects the presence of a neighbor cell.

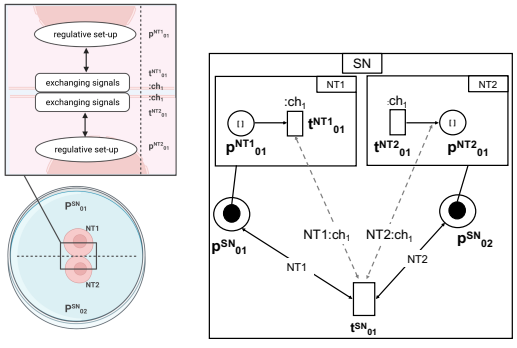

Figure 18: Neighbor communication: a cell communicates with a neighbor cell.

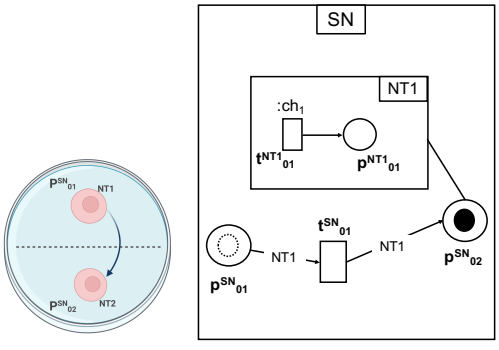

Figure 19: Cell movement: a cell moves from a position in space to another one.

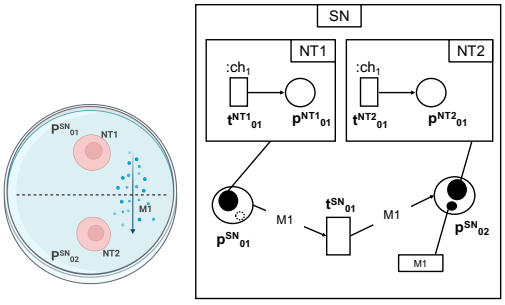

Figure 20: Molecular flow: a molecule moves from a position in space to another one.

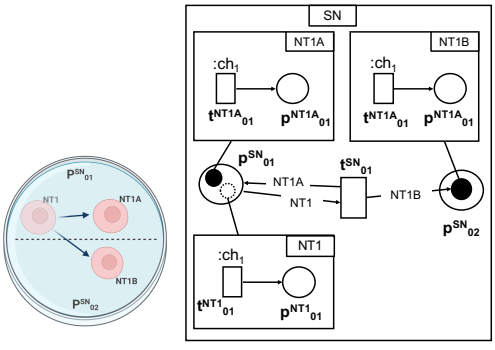

Figure 21: Mitosis: a cell undergoes mitosis and generates two daughter cells, one in its starting position, and another one in close spatial proximity.

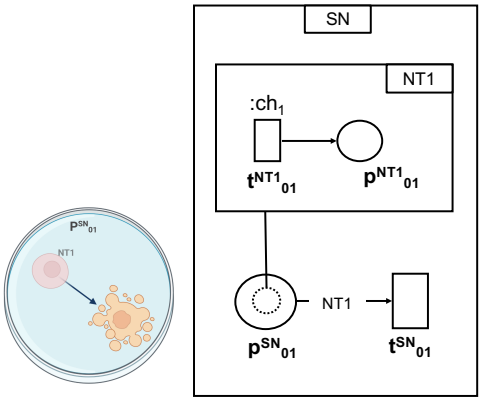

Figure 22: Apoptosis: a cell undergoes apoptosis and leaves the position it occupied empty.

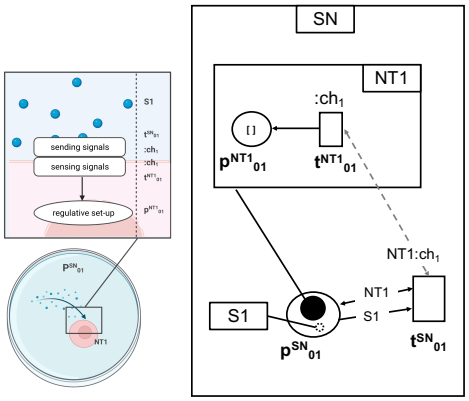

Figure 23: Signal sensing: a cell senses a signal from the extracellular environment.

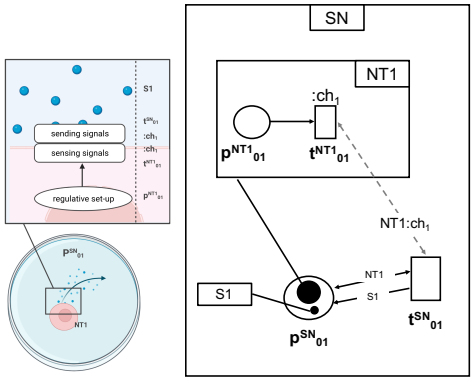

Figure 24: Signal sending: a cell sends a signal in the extracellular environment.

A.2. Differentiative landscape

This section reports a building block to model a single differentiative step in a Differentiative Landscape model.

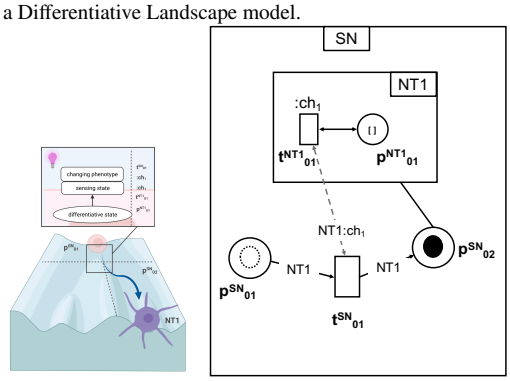

Figure 25: Differentiative step: a cell evolves from one phenotype state to another. This passage is regulated by a checkpoint evaluating the cell state.

A.3. Cells

This section reports a set of sample building blocks to model simple intracellular mechanisms in the Cell models.

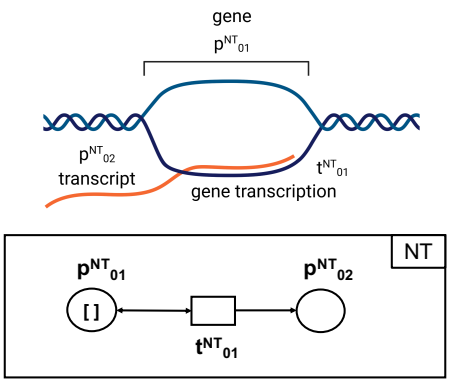

Figure 26: Transcription: the transcription of a gene (or a region of the genome) into coding or non-coding RNA transcripts.

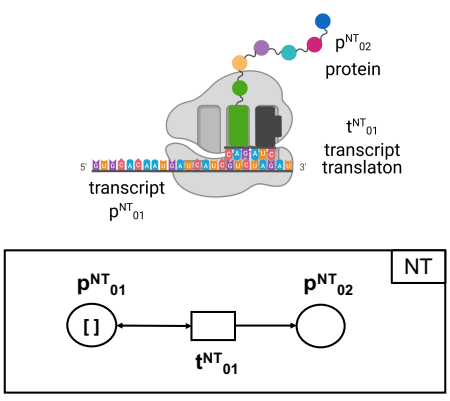

Figure 27: Translation: the translation of a a RNA transcript in a protein.

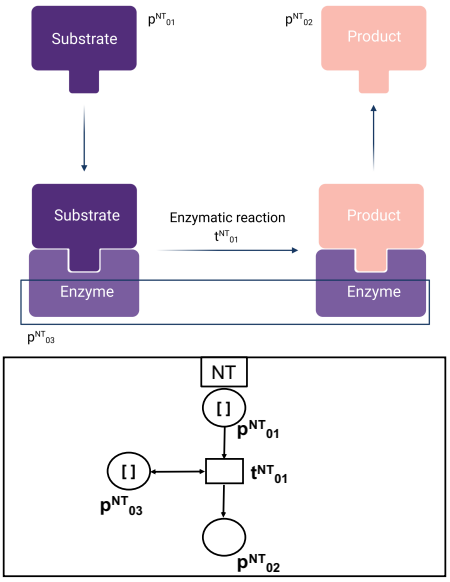

Figure 28: Enzymatic reaction: an enzyme catalyzes the transformation of substrates into products.

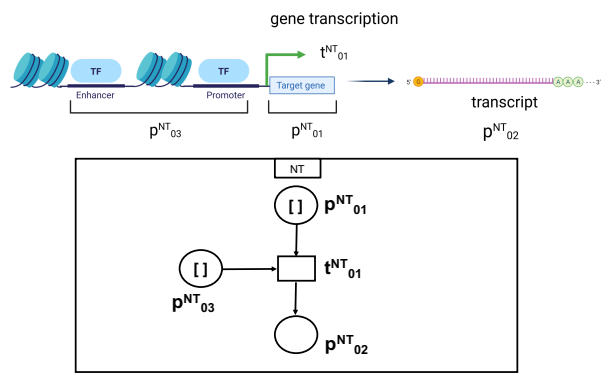

Figure 29: Activating gene regulation: a regulatory factor makes the transcription of a gene start.

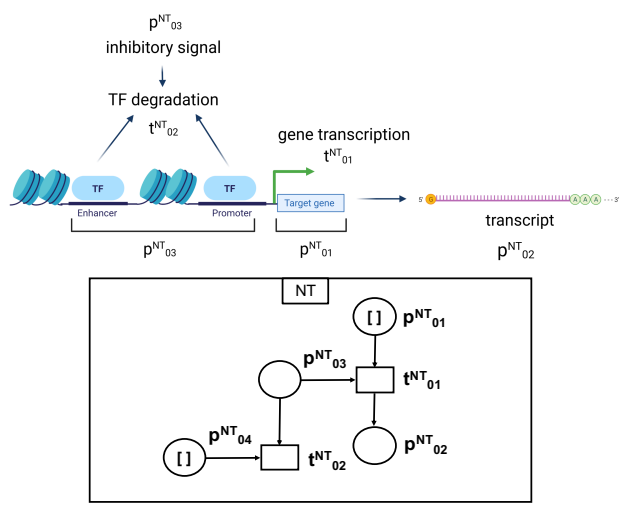

Figure 30: Inhibiting gene regulation: a regulatory factor inhibits the factors that make the transcription of a gene start.