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 [43], defined as "[...] *the development of a single individual, or a system within the individual, from the fertilized egg to maturation and death*" [39]. 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 include the cellular organization in space and the consecutive temporal stages characterizing the process. Each step corresponds to a different conformation and regulative set-up involving multiple interacting cells. Since these cells' regulatory states depend on their neighbors' relations, such conformations dictate the communication schemes they engage in [19]. This complexity contributes to creating a multi-dimensional and dynamic landscape of inter-dependent regulative states in which cells can fall into [34, 20]. 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 this hierarchical regulative *milieu*'s dynamic properties, a computational model must explicitly account for their evolution in space and time. Moreover, it must be able to simulate the underlying nonlinear regulatory dynamics in a quantitative way, which often encompasses different system levels. This requirement imposes the need of modeling the hierarchical organization of the system. Eventually, biological systems are intrinsically stochastic, i.e., modeling of their complexity must include stochastic behaviors.

A variety of modeling approaches based on different formalisms exists, and each of them has strengths that better fit specific aspects of the system [6, 13]. However, current approaches often fail to adequately capture spatial elements, limiting their capability to analyze patterns emerging from local interactions between biological entities [6, 5].

One of the approaches for handling model heterogeneity is the composition of existing models into a more extensive scope [6]. However, this raises consistency issues [38] and introduces additional requirements to create multi-level and hybrid models [3].

Differently, this paper introduces a computational modeling approach based on the Nets-Within-Nets (NWN) formalism that relies on a single formalism to cover all requirements posed by the modeling of complex ontogenesis processes. NWN can efficiently model complex dynamics, stochastic processes, hierarchical organizations, and spatial structures. They go beyond the qualitative representation of states typical of agent-based systems, leveraging the same expressive power of discrete mathematical models, specifying such states as dynamic and complex information structures. Moreover, they support compositional and integration processes typical of hybrid models, supporting the integration of different sources of knowledge. In other words, NWN models

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

leverage the strengths of modeling paradigms from various existing computational tools and methods yet maintaining uniformity in formalism usage.

After introducing the NWN formalism, its relations to other Petri Nets-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*.

2. Background

A low-level Petri Net (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. Transitions can move tokens from their input to their output places, following explicit rules. These rules define when a transition is enabled (e.g., the marking of the input places must fulfill specific requirements) and what happens when a transition fires (e.g., tokens can move). The marking of a PN is the token configuration in its places. It can dynamically change when transitions fire. The initial marking of a PN is the initial configuration of tokens. Together with the network architecture, it defines a PN model. Different Petri Nets exist. Depending on their characteristics, they can support biological systems modeling to different extents.

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.

Considering different biochemical reactions together, like, for instance, in the case of metabolic and gene regulation networks, poses further requirements to computational models, such as more complex architectures and concurrency [24]. Low-level PNs can still support this complexity. In a metabolic network, places can model molecular species and states, while transitions can model enzymatic reactions. Tokens can represent discrete quantities of molecules, while the architecture of the network models 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, while tokens can model the involved molecular species (DNA, RNA, aminoacids, proteins) and the conditions in which they can fall.

The phenomena considered so far interest a single system-level and dimensional range. To handle different timescales, timed PNs [40] introduce a time delay associated with each transition's activation as a tunable parameter. Once a transition is enabled, deterministic time delays can occur before actual firing, ordering different activations during the net evolution. The introduction of time delays increases time resolution when including diverse yet intertwined mechanisms in a model.

To handle the stochasticity of all biological systems (e.g., gene expression level random fluctuations [17]), stochastic PNs [28] extend Timed PNs with the use of probabilistic time delays. The delay becomes a random variable, and it can also depend on the current marking of the net [42].

In general, low-level PNs do not scale with system complexity. The lack of scalability limits their use to modeling only 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 PNs extend the low-level formalism, supporting multi-level and nested models that properly handle information diversity and complexity [27].

Colored PNs (CPNs) [21] are a class of high-level PNs allowing tokens to be associated with arbitrarily complex data structures defined as *colors*. Each place can accept a number of colors, defining the colors multi-set for that place. In this way, it is possible to separate the identity of resources from their location, modeling the same condition for different categories of resources [31]. CPNs 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 PNs, which can be generated from CPNs models by automatic unfolding [26, 27].

Models considered so far flatten information from different system levels into a single one. To represent a multi-level system structure, hierarchical (or nested) PNs 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 [29]. Nevertheless, like CPNs, hierarchical PNs 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 parts.

Complex biological processes such as ontogenesis challenge the limitations of most high-level PNs. 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, a developing embryo 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. These different views introduce further requirements to computational models that sum on the ones

posed by complex biological systems in general.

The Nets-Within-Nets (NWNs) formalism [25, 45] supports all capabilities of other high-level PNs and can face the additional modeling requirements posed by ontogenetic processes [4, 2]. As depicted in Figure 1, NWNs can express token colors using the Petri Net formalism. Tokens specified in this way (the net-tokens) evolve dynamically, just like the net holding them (the system-net). This hierarchical organization can be reiterated in a boundless way, allowing for open recursion in specifying system levels' hierarchical organization with dedicated model layers [9].

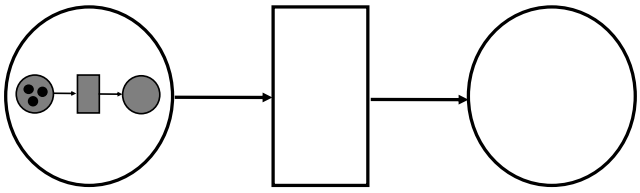

Figure 1: NWN models devise the use of the PNs formalism for specifying the structure of tokens, making it possible to set up multiple levels of nested and encapsulated nets in a recursive way. In this image a simple example of this is provided, where a simple low-level PN exists as a token in a place from another PN. This corresponds to the specification of two levels in the hierarchy of nets.

3. Methods

This section introduces features and capabilities of the NWNs formalism proposing their usage to model ontogenetic processes.

3.1. The Nets-Whithin-Nets formalism

As thoroughly described in [45], besides supporting all functionalities from other high-level PNs, NWNs innovate the modeling strategy providing tokens with a PNs structure. 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. Also, net-token instances can exist at the same time in different system-nets.

Following [45], NWNs bring the object-oriented paradigm into the PNs domain by introducing the concept of *net class*. A Net class defines a generic network that can be instantiated within other net instances. This organization creates a system of nets, in which relations between instances can have different forms, especially when concerning communication.

3.1.1. Transports and interactions

In the simplest case, transitions in the system-net *transport* net-tokens from a place to another without determining any other change while they function independently and concurrently.

In other cases, transitions from the different nets can *interact*. Interactions between different nets rely on communication mechanisms such as *synchronous communication channels*, which interlock transitions across nets [11]. Each channel has two ends: the *down-link* and the *up-link*. The transition containing the *down-link*, when enabled, checks for the presence of the corresponding *up-link* in the nets system. If available, the two transitions activate synchronously. Transitions containing an *up-link*, on the other hand, wait for the corresponding *down-link* to evoke their joint activation. A single *down-link* can activate multiple *up-links* at the same time. This defines a directionality for channels. Channels can also pass arguments, which can support tokens to flow across nets. Tokens can flow in both directions between the *down-link* and the *up-link* transitions.

3.1.2. Intra- and cross-layer interactions

In a hierarchical nets system, interactions can apply to model parts both from the same and from different layers. Interactions include both *writing* and *reading* mechanisms. Therefore, considering these possibilities, four interaction building blocks are possible as depicted in Figure 2:

- **Intra-layer reading**: a transition considers the marking at input places of another transition from the same layer. This situation occurs even in low-level PNs when a transition to be enabled checks its input places.
- **Intra-layer writing**: a transition affects the marking at output places of another transition from the same layer. In low-level PNs as well, a firing transition affects the marking of its output places.
- **Cross-layer reading**: a transition considers the marking of an input place of another transition from a different layer.
- **Cross-layer writing**: a transition affects the marking at output places of another transition from a different layer.

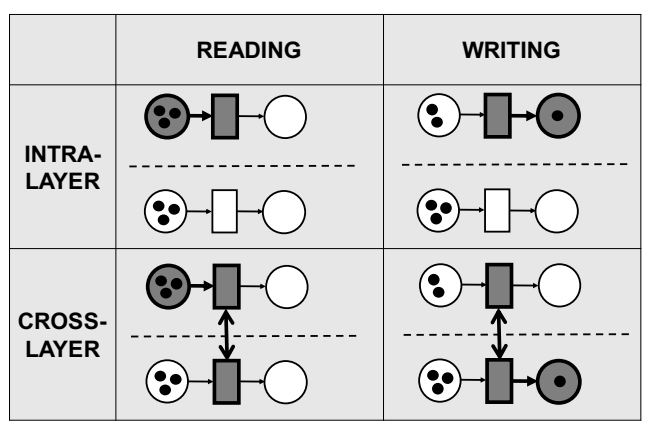

Figure 2: Four types of Petri Nets-based building blocks for implementing communication mechanisms between model parts.

This short overview shows how NWNs can express a range of possible scenarios. The following section presents the way these capabilities can support representing the biological mechanisms of interest.

3.2. NWN applied to ontogenesis modeling

This paper applies the NWN object-oriented paradigm to the problem of modeling ontogenetic processes. It exploits different usages of the formalism elements to describe biological semantics as depicted in Figure 3.

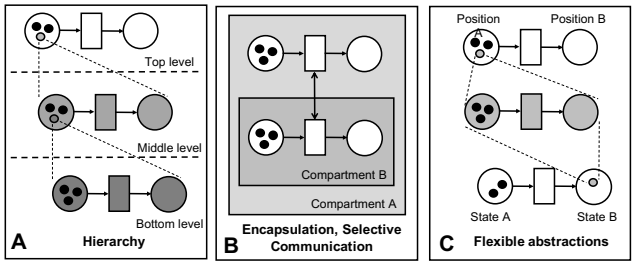

Figure 3: NWN models easily express (A) multiple hierarchical levels; (B) encapsulation of model parts and selective communication between them; (C) coexistent and intertwined regulatory layers with different annotation.

In our approach, the net-tokens represent differentiating biological cells described through their internal regulative networks. The system-net instead represents the landscape of regulations external to the cells and affecting their functioning, like their microenvironment (including environmental factors, spatial organization, and relative positions between them) and the developmental phase they are going through.

The modeling strategy presentation resorts to a simple working example, depicted in Figure 4 and Figure 5. The system-net reported in Figure 4 depicts a biological process composed of two stages. In each stage, places identify spatial locations holding instances of the net-token reported in Figure 5.

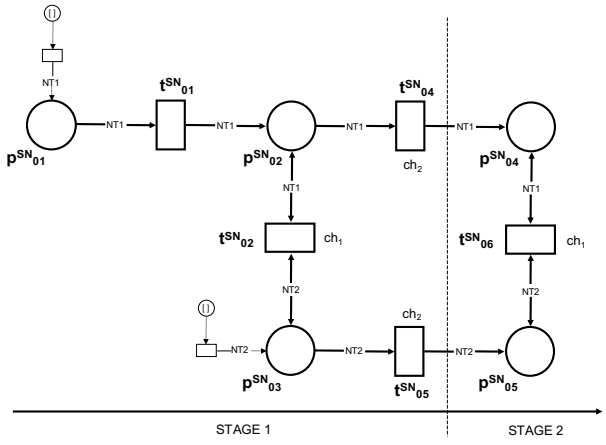

Figure 4: This figure shows a simple system-net summarizing the mechanisms presented in this section: spatial mobility (t^{SN}_{01}), proximity-enabled interactions between net-tokens from different system places (t^{SN}_{02}, t^{SN}_{06}), net-token marking evaluation for checkpoint analysis and transition to a different process stage (t^{SN}_{04}, t^{SN}_{05}).

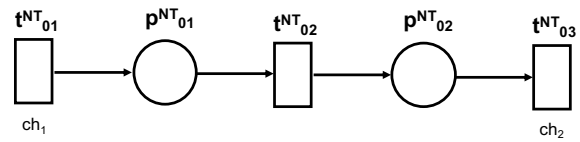

Figure 5: In this picture, a simple net-token architecture shows the two up-links to the system-net, located in transition t^{NT}_{01}, and another one for cross-level reading by the system-net, located in transition t^{NT}_{03}.

3.2.1. Spatiality and mobility

NWN can provide an explicit representation of spatiality, using places to represent actual locations that various actors occupy. For example, in Figure 4, places p^{SN}_{01}, p^{SN}_{02}, and p^{SN}_{03} represent three locations in which net-tokens can live during stage 1 of the considered process. Similarly, p^{SN}_{04} and p^{SN}_{05} are two locations available during stage 2 of the ontogenetic process.

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 or paracrine interactions between neighbor cells in a biological context, which occur only when the cells are in more or less close spatial proximity.

Net-tokens that move across system-net places having spatial semantics model mobility of biological entities. For example, in Figure 4, 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 a system place to a different 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.

3.2.2. 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 are instances

of a net class describing cells' inner functioning intended as biological compartments. For example, net-tokens NT1 and NT2 in the system-net of Figure 4 are two instances of the same net class, depicted in Figure 5. 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 4) and ch_2 (transitions t^{SN}_{04} and t^{SN}_{05} in Figure 4), using cross-layer writing and reading mechanisms, respectively. 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.

3.2.3. 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. Combining the communication building blocks presented in Figure 2, one can assemble biologically relevant network motifs modeling the cross-membrane communication mechanisms of interest and their regulation at different levels.

As an additional example, Figure 6 depicts two modules based on such communication building blocks, assembled in a convenient way to support the modeling of ontogenetic processes.

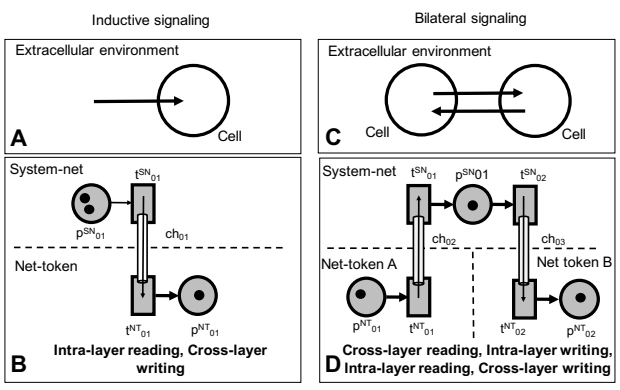

Figure 6: Two signaling mechanisms relevant for modeling ontogenetic processes: (A) a cell receiving a signal from the extracellular environment, modeled combining (B) an intra-layer with a cross-layer writing building block; (C) two cells exchanging signals through the intra-cellular space, modeled (D) combining the four basic building blocks from Figure 2.

In the first case, the combination of an intra-layer reading and cross-layer writing models the cross-membrane activation of an intracellular signaling pathway from an extracellular cue sent out by an external source (Figure 6.A).

In Figure 6.B, the place p^{SN}_{01} in the system-net represents the signal source, which is the input place for transition t^{SN}_{01}. If the marking of this place is compatible with the transition's enabling rule, the transition is enabled. This transition is linked to transition t^{NT}_{01} in the net-token through a synchronous channel ch_{01}. It is then possible to transfer the resources from the extracellular (system-net) to the intra cellular (net-token) environment.

The second example combines all of the basic building blocks: cross-layer reading, intra-layer writing, intra-layer reading, and cross-layer writing. This structure enables the communication between two net-tokens through the system-net, modeling a unilateral communication mechanism between two cells (Figure 6.C).

In Figure 6.D, transition t^{SN}_{01} in the system-net reads the marking from the input place p^{NT}_{01} in net-token A, through the synchronous channel ch_{02}. According to its activation function, if t^{SN}_{01} is enabled, it determines a marking in its output place p^{SN}_{01} at the system-net level. p^{SN}_{01} is an input place for transition p^{SN}_{02} in the system-net, which interlocks with transition p^{NT}_{02} in net-token B through the synchronous channel ch_{03}. When the two transitions fire, they determine the marking of the output place p^{NT}_{02}. This schema can model the flow of resources or information from a cell to another cell through the extracellular environment. Moreover, using two of these constructs, the same schema can also model bilateral communication between two cells.

These two network motifs are particularly relevant for modeling ontogenetic processes when combined with spatiality information. During development, architectural complexity at the supra-cellular level emerges from local interactions between cells often influenced by different distance ranges [33][32].

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

3.2.4. Dynamic hierarchy

The ability of recursively instantiating net-tokens within another net naturally expresses the hierarchical and dynamic regulatory structures of biological systems (Figure 3.A). Let us consider the two-levels model architecture of Figure 4 and Figure 5:

- the system-net at the top level (Figure 4) 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 (Figure 5) 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, NWNs 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, describing various aspects 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.

3.2.5. Process stages

Ontogenetic processes are a sequence of subsequent stages, with checkpoints involving regulatory assets and sets of biological actors governing the passage from a stage to the next. Looking at Figure 4, 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 signifies the passage to a new regulatory set-up. To occur, it 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.

3.2.6. 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 genetic (net-token) and epigenetic (system-net) regulations, the latter intended as any regulation existing on top of genetic regulation. 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 developing of *C. Elegans* larva. This aims to show the proposed modeling strategy at work providing an overview of its capabilities.

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 [44, 41], 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 Pnp 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 Pnp cells.

More specifically, as in Figure 7, in the physiological case, the AC sends out a LIN-3 (EGF-like) signal reaching Pnp cells with distance-dependent intensity: the closest cell, P6p, receives juxtacrine signaling, its neighbors P5p, and P7p cells receive paracrine signaling. The signal does not target P3p, P4p, and P8p, the farthest cells. The hypodermal syncytium (hyp7) sends uniformly intense paracrine LIN-3 signals to all Pnp cells. Each of the Pnp 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 P6p 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. P3p, P4p, and P8p 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.

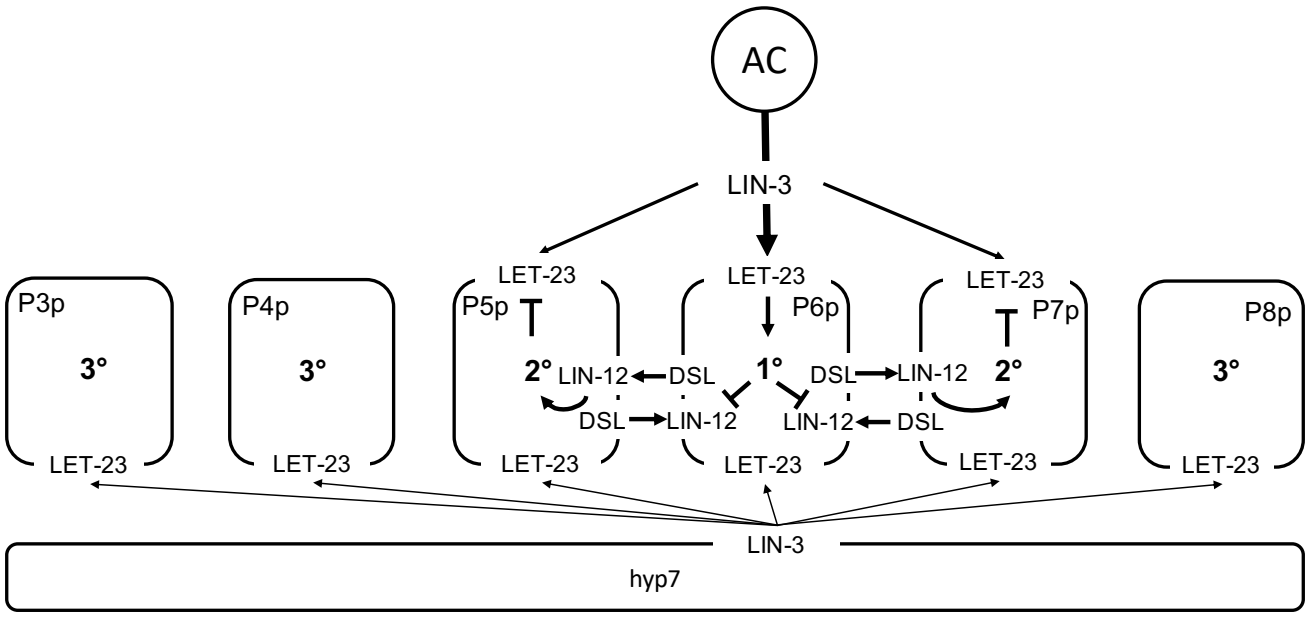

Figure 7: Diagram of signaling involved in VPC pattern formation

4.2. The model

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

- a bi-dimensional Interactive Spatial Grid (ISG) (Figure 8);
- a Pnp specific Differentiative Landscape (DL) (Figure 9) with complex checkpoint functionalities;
- three cell models, for the Anchor Cell, the hypodermal syncytium (hyp7) and Pnp cells (Figure 13, Figure 12, and Figure 14, respectively).

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 10 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 Pnp cell fates (Fates Manager net) live and interact as net-tokens within this system-net.

In the provided implementation, which relies on Renew [10], a single system-net combines both the ISG and the DL. Figure 11 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 11 represents the ISG depicted in a more schematic way in Figure 8. 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,

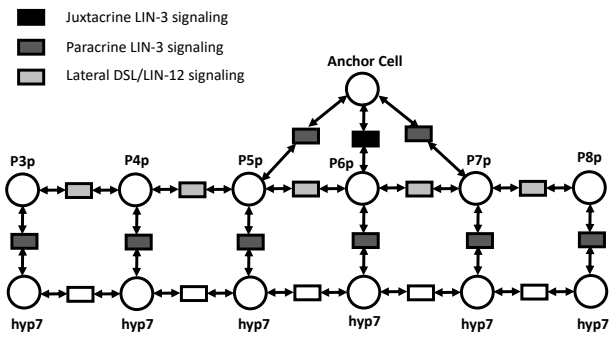

Figure 8: A schematic representation of the Interactive Spatial Grid (ISG) model for the VPC specification example. Black symbolizes *juxtacrine* communication, dark gray *paracrine* communication, light gray *neighbor* communication.

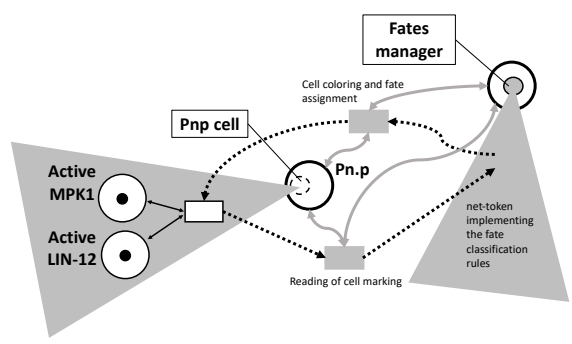

Figure 9: A schematic representation of the DL model for the VPC specification example.

and transitions model the potential interactions between cells in such positions (e.g., cell-cell communication mechanisms

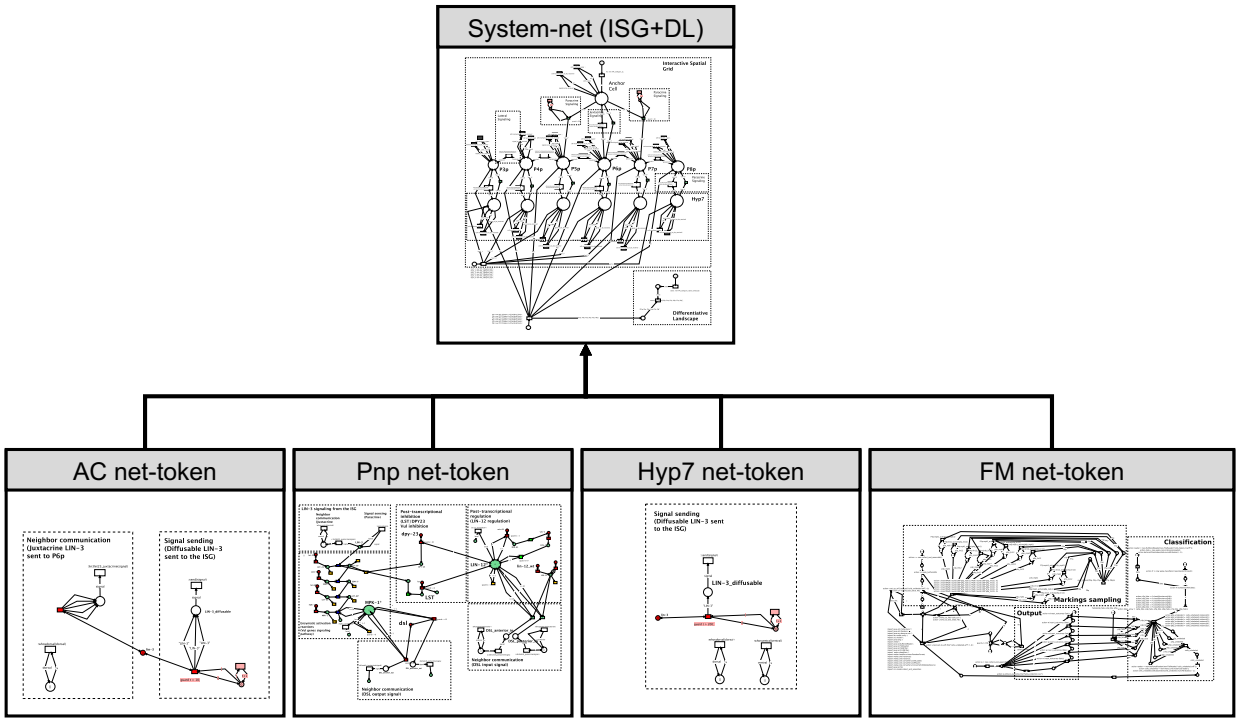

Figure 10: Nets hierarchy for the VPC specification model implementation in Renew. The system net having ISG and DL functionalities is at the top level, and it hosts the net tokens modeling different cells (AC, Pnp and hyp7), plus the Fates Manager (FM) modeling complex checkpoint rules.

involved in the emergence of a morphogenetic pattern). Hyp7 sends LIN-3 paracrine signals uniformly to all Pnp cells, modeling that the hypodermal syncytium lies under the array of Pnp cells.

The DL model (Figure 9) represents instead the mechanism governing the Pnp cells' states along the developmental step from L3 to L4. Possible conditions are: Pnp 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 Pnp cells. From the Pnp places holding the cells, specific transitions involving channels dynamically track the involved cells' marking (net-tokens). These channels monitor the places modeling active MPK-1 and active LIN-12, and transfer this information to a dedicated net-token named Fates Manager. The bottom part of Figure 11 represents the actual implementation of this mechanism. While this implementation is equivalent to the one presented in Figure 9, it exploits the use of object references available in Renew to reduce the number of places, channels, and transitions.

In the presented model 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 12. The AC Cell model has two LIN-3 communication modes implemented using channels: one for *juxtacrine* signaling with P6p (*neighbor communication*), and the other one for *paracrine* signaling with P5p and P7p (*signal sending*). Both signals rely on the LIN-3 gene for production (*transcription and translation*).

The Pnp net-token describes the regulation of the Pnp 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 13, the Pnp 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. Pnp 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 Pnp 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*).

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 Pnp cells. The full network is available on GitHub. However, for the sake of readability, Figure 15 focuses on presenting the overall behavior of this

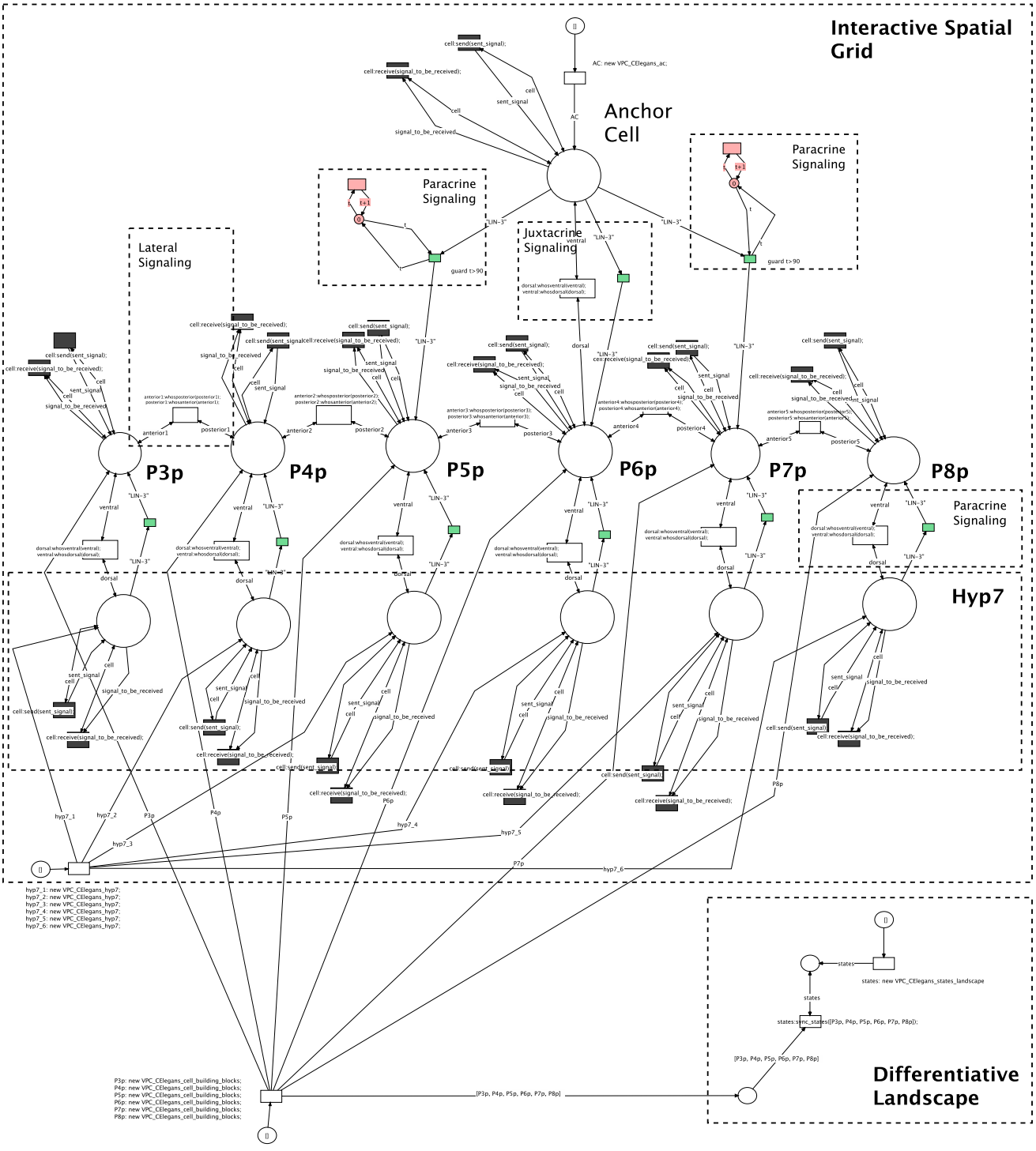

Figure 11: The Renew implementation of the system net for the VPC specification example, combining ISG and DL functionalities.

network.

The FM's first task is sampling the marking evolution of the MPK1_ACT and LIN12_ACT places of all Pnp cells. A channel transfers an array of Pnp 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 circular buffers set, creating a sliding window where the six Pnp cells' mark-

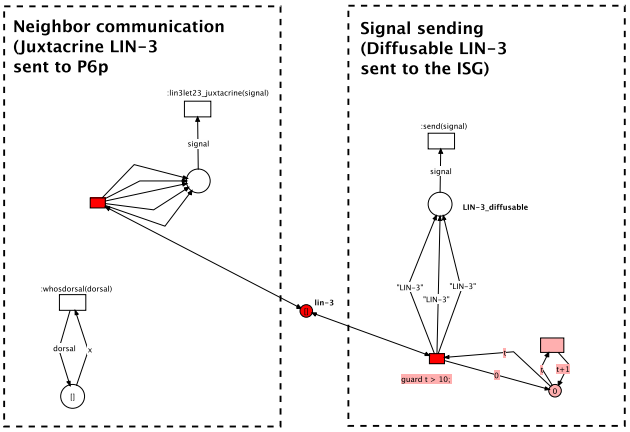

Figure 12: The AC Cell model.

ing values are retained 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 15.B). The resulting fates are sent to the system-net (Figure 15.C) and used to assign the corresponding color to the Pnp cells.

The classification process relies on a Weka random forest model [22]. 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 Pnp 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 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 Pnp, 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. That means that the simulation engine selects all the transitions at each cycle whose enabling conditions are satisfied 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 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 ontogeny 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 collects the features from the marking of the Pnp cells and stores 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 [37]. Simulations provide as outputs both the temporal evolution of the markers of interest from cell models and the six labels resulting from Pnp 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, 12, 23, 14], our modeling framework has similar or better predictive performance, together with the advantage of naturally supporting the integration and composition of heterogeneous biological information and models, combining the advantages of multi-level hybrid models [3] with formalism uniformity, which facilitates model analysis, and knowledge representation and exchange.

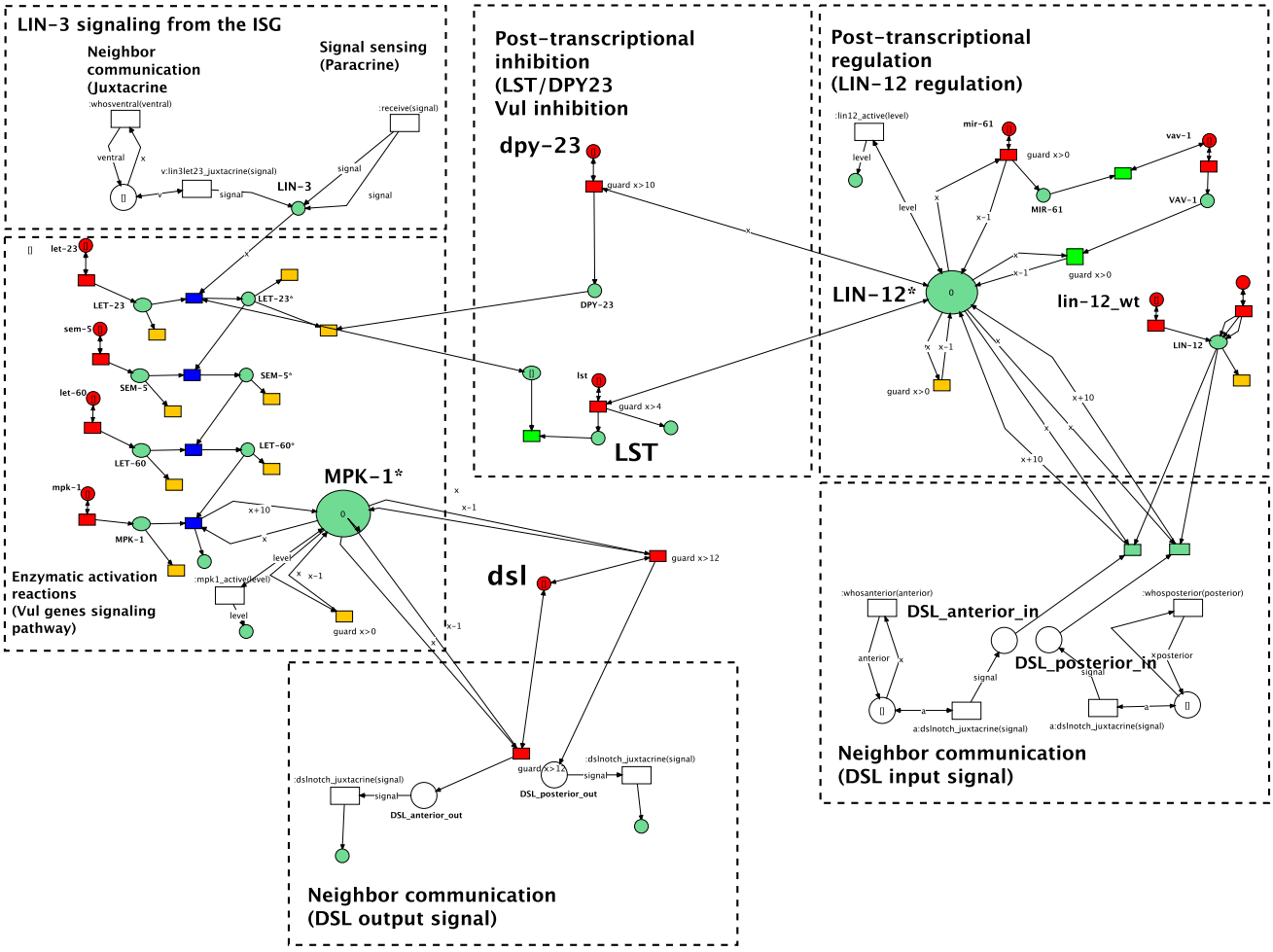

Figure 13: The Pnp Cell model.

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
For each experimental condition, the Pnp 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.

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 better to 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 [18]. 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 [46], so to gain deeper insights into the modeled system.

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 model complexity: if computational complexity is too high, analyses and simulations take too long to complete [35]. 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 [15, 16].

Finally, significant improvements are required to improve the modeling strategy usability from life scientists with lim-

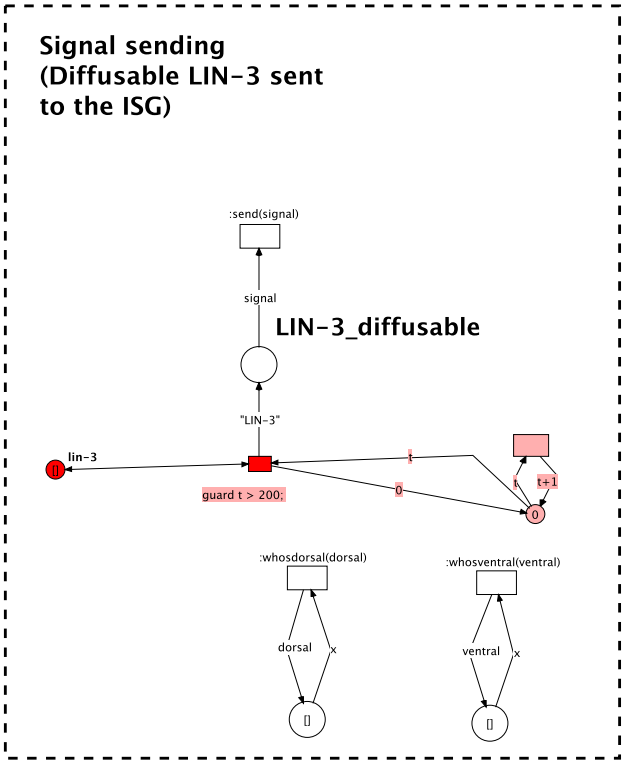

Figure 14: The hyp7 Cell model.

ited 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 [30].

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 [36].

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] Christensen, S., Hansen, N.D., 1994. Coloured petri nets extended with channels for synchronous communication, in: International Conference on Application and Theory of Petri Nets, Springer. pp. 159–178.

[12] 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.

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

[14] 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.

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

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

[17] 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.

[18] 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.

[19] 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.

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

[21] 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.

[22] 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.

[23] 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.

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

[25] 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.

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

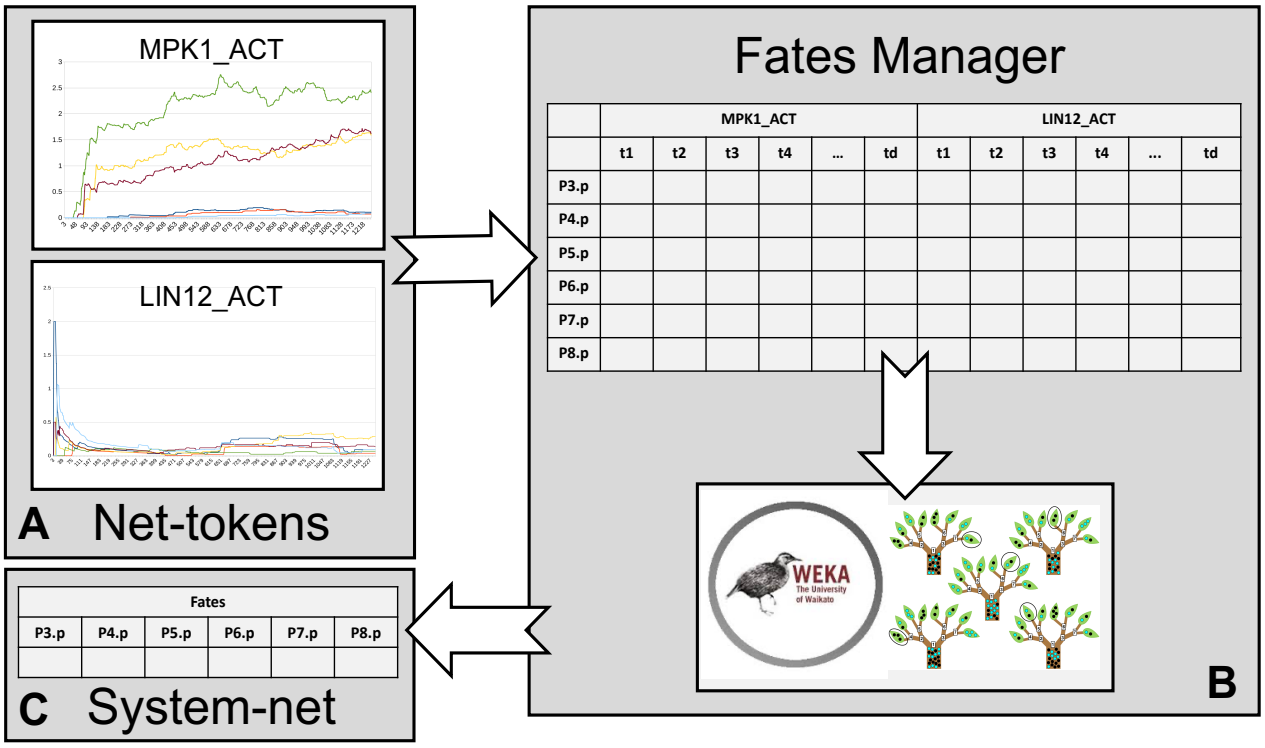

Figure 15: Overview of the cell fate classification process in the Fates Manager.

827, 71–85.

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

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

[29] 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.

[30] 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.

[31] 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.

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

[33] 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.

[34] 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.

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

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

[37] 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.

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

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

[40] 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.

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

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

[43] Scott, R.J., K., H.B., M., O.W., 2001. Bridging the gap between developmental systems theory and evolutionary developmental biologyâĂă. 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.

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

[45] Valk, R., 2003. Object petri nets, in: Advanced Course on Petri Nets, Springer. pp. 819–848.

[46] 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.