Constraint-based modeling of the *Drosophila* Gap-gene regulatory network

Eric Fanchon · Fabien Corblin · Laurent Trilling

Received: date / Accepted: date

**Abstract** We present a constraint-based assistant for the modeling of the dynamics of biological networks. The global aim is to automate some modeling tasks by providing high-level functionalities to biologists. Network dynamics is represented by the discrete abstraction of R. Thomas (Thomas and Kaufman (2001)), which is briefly introduced. We explain how this formalism is cast into Constraint Logic Programming (CLP) and boolean Satisfiability (SAT). An originality of this work is that not only the dynamical rules of network evolution are encoded as constraints, but also the available knowledge on network behaviour. The method is flexible and allow to express high-level queries useful in the discovery process: proof of model (in)consistency, automatic model revision, inference of model parameters, etc. The formalization of experimental data and the search for a minimal model is illustrated with the case of the *Drosophila* gap-gene module, which is involved in the first step of embryo segmentation.

**Keywords** Gene regulatory networks · *Drosophila* segmentation · Constraint Logic Programming · SAT

1 Introduction: modeling biological networks

In recent years, it has become increasingly clear that the right level to understand the inner workings of cells is the level of networks. It is exceptional that an macroscopic observable property (phenotypic trait) can be explained as the result of the action of a single protein. The general rule is that phenotypes are the result of the action of networks involving many actors interacting one with the other. The interest is thus shifting, with the rise of systems biology, from the study of single molecular objects (genes and proteins) to the study of networks of interacting molecules and genes.

A gene regulatory network is a particular kind of such network in which each gene is able to produce a specific regulatory protein at a rate which depends on the cellular context. In other words, the expression rate of a given gene depends on the quantities of regulatory proteins in the system. A single gene can be influenced by several proteins, including the protein encoded by the single gene itself. The set of all the influences between genes, mediated by the proteins produced, can be represented as an interaction graph (see section 1.1). The proteins are generally abstracted away and one speaks of interaction between genes.

These molecular networks have complex behavior because the interactions are highly non-linear and because there are generally many feedback loops. This makes human reasoning on such systems impossible for anything but the smallest networks (2 or 3 nodes), and calls for automated formal methods. Another aspect is that the data often have a qualitative flavour. It may be sufficient to represent a protein concentration by two (low/high) or three (low/medium/high) levels. The number of levels required rarely exceeds 4 or 5. The formalism described in section 1.2 is well suited to the level of knowledge which is generally available in systems biology. All the state variables, as well as all the model parameters, are integers, and simple rules define the evolution of the system.

Modeling is a part of a discovery process which is generally composed of many rounds of experimentation and modeling. The basic question addressed here is: how to build a model of a biological phenomenon, based on some partial knowledge about the network architecture and behaviour? Along these lines, an originality of our work is the fact that everything is represented as constraints: the generic evolution rules, the architecture of the particular network at hand, and the knowledge on its behaviour. Consequently a wide array of questions can be asked.

We describe here how we exploit constraint-based technologies to assist the process of building dynamic models of molecular networks using experimental data as constraints. We have developed a tool, GNBox, which provides high-level functionalities to biologists engaged in the modeling process. Among these:

- proof of inconsistency between the assumed architecture and the behaviour,
• inference of model parameters,
• search for minimal (in a sense to be defined) consistent models,
• automatic model revision,
• inference of properties (about behaviours, kinetics of reactions, thresholds) shared by all solutions.

More precisely, GNBox makes use of two technologies: Constraint Logic Programming (CLP) and boolean satisfiability (SAT). The network evolution rules and the queries describing the above functionalities are first expressed in CLP. To increase the efficiency of our tool, a translation scheme to SAT has been devised. We explain in section 2 how the evolution rules of Thomas networks are modelled in CLP, and how the CLP formulation is translated into Conjunctive Normal Form (CNF), which is the format used by most SAT solvers. The queries can thus be submitted to a CLP solver, or translated into CNF and submitted to a SAT solver. The use of GNBox is illustrated in section 3 with a type a question that arised during the course of the modeling of *Drosophila* gap-gene module.

1.1 Interaction graph of a Gene Regulatory Networks (GRN)

The interactions between the genes are usually represented by an interaction graph: see Fig. 1 for the interaction graph of the gapGRN module (adapted from Sánchez and Thieffry (2001)). In such oriented graphs the nodes represent the genes and the arrows represent the interactions. Remember that each gene produce one or several proteins which are specific of this gene. In systems like the one considered here, the correspondence between genes and proteins is one to one. We will here make an abuse of language by not always distinguishing genes and proteins. The variable associated to a given node (gene) is the concentration of the corresponding protein (produced by this gene).

In most cases the influence of one gene on another can be represented by a sigmoid function: there is a threshold in the concentration of the regulatory protein at which the effect on the production of the target protein changes steeply from efficient to inefficient (or more generally between two qualitative levels of production rates). In a discrete modeling, sigmoids are approximated by step functions. An interaction can be either an activation or an inhibition. For instance, in the gapGRN, the gene *gt* inhibits the gene *kr*: this means that it is possible that the concentration of protein *gt* being low (under a certain threshold $t_{gt}$), the concentration of protein *kr* tends to increase, whereas the concentration of *kr* tends to decrease when *gt* is high (above the same threshold $t_{gt}$).

Each arrow carries one label: an index identifying the threshold at which the interaction changes its activity status. We depart here from usual practice by not labeling the arrows with a sign indicating whether the interaction is an activation or an inhibition. This kind of information is represented differently in our framework. The thresholds, noted $t^c_p$ (where $c$ is the gene and $p$ is the identifier of a threshold associated to $c$) are model parameters. The integer values of the threshold parameters define the order between them, which are sometimes determined from experiments. In gapGRN, $t^1_{hb_z}$ is the first threshold of the gene *hb_z* and is associated to the interactions on *gt* and *kr*; $t^2_{hb_z}$ and $t^3_{hb_z}$ are the second and third thresholds of the same gene for the interactions on *kni* and *kr*, respectively (cf. Fig. 1). The order between these thresholds is assumed to be known and is the following: $t_{hb_z} < t^2_{hb_z} < t^3_{hb_z}$. Note that several arrows can label a given pair of nodes.

![Fig. 1 Interaction graph for rosophila gap-gene system.](image)
1.2 Thomas discrete formalism for GRNs

We now introduce the asynchronous multivalued formalism created by the biologist R. Thomas and his collaborators (see Thomas and Kaufman (2001) for a pedagogical review). It relates the interaction graph of a GRN and its dynamical behaviour. The main goal of this formalism is to obtain a qualitative understanding of the network dynamics by reasoning on discrete entities. It can be described as follows:

1. All the entities of a model are discrete (abstraction of continuous variables):
   - The thresholds \( t^l_k \) take discrete values between 1 and \( \max_c \). In the gapGRN, we have for the species \( hb_z \):
     \[
     t^1_{hb_z} < t^2_{hb_z} < t^3_{hb_z}.
     \]
     Then \( t^1_{hb_z} = 1, t^2_{hb_z} = 2, t^3_{hb_z} = 3 \) and \( \max_{hb_z} = 3 \). In the general case the values of \( t^l_k \) and \( \max_c \) may not be known.
   - The concentration of the protein produced by the gene \( c \) is modeled by a discrete variable, noted \( S_c \), which ranges over \( [0, \max_c] \).
   - A discrete concentration state \( S \), or just state, of the system is represented by an ordered list of discrete values, each value representing the concentration of a protein. For instance a possible state \( S \) of the gapGRN is \( S = (S_{gt} = 0, S_{hb_z} = 1, S_{kr} = 0, S_{kni} = 0) \), or \( S = (0,1,0,0) \) for short. The three other nodes (namely \( cad, hb_z \) and \( bcd \)) are input nodes representing the environment state (see Section 3). The state \( S \) is interpreted as follows: the concentration of proteins \( gt, kr \) and \( kni \) are all below their lowest threshold, while the concentration of \( hb_z \) is between its first and second thresholds. In the gapGRN we have 36 different states (not counting environment states).

2. A transition rule defines the relationship between a state and its successors. It is based on the following notions:
   (a) **Tendency**: it is clear that the concentrations evolve continuously. Consequently a successor of a given state \( S \) is necessarily a state adjacent to \( S \).
   (b) **Tendency**: in a given state, the system can be thought of as tending to evolve towards a state, called focal state. The complex mix of influences between the genes of the system is therefore reduced to abstractions of the form "in state \((0,1,0,0)\), the system tends to evolve towards the focal state \((2,0,0,0)\)". The focal state of a state \( S \) is noted \( F(S) \) and its components \( c \) are denoted \( F_c(S) \).
   (c) **Asynchronicity**: in a given transition, the system cannot cross two or more thresholds simultaneously. This is a translation of the fact that, in the continuous version of the model, the number of trajectories that pass through the intersection of two or more threshold hyperplanes is negligible.

The transition rule can now be expressed as follows:
   - If the system’s current state \( S \) is different from its focal state \( F(S) \), then the concentration of only one of its components \( c \) will change, by one unit, in the direction indicated by \( F_c(S) \). This is done independently for each component so that a state \( S \) can have several successors. We thus obtain a non-deterministic transition system.
   - If the system current state is equal to its focal state, the system is said to be in a steady state and no concentration changes.

For instance if the focal state of \((0,1,0,0)\) is \((2,0,0,0)\), then the set of possible successors is constituted of \((1,1,0,0)\) (move along the first dimension towards value 2), and \((0,0,0,0)\) (second dimension, towards value 0).

3. This leaves the question of how each component \( F_c(S) \) of the focal state of a state \( S \) is defined. A gene \( c \) is influenced by a number of regulatory proteins produced by a subset of the genes constituting the network. Let say that gene \( a \) is influenced by \( b \) and \( d \) which are acting on \( a \) at thresholds \( t^1_k \) and \( t^2_k \), respectively. From the point of view of gene \( a \), there are only four cases to distinguish. Each case will be called a cellular context, and corresponds to a region of state space. For each cellular context we must specify the position of the focal component \( F_a(S) \) (defining the evolution trend of gene \( a \) or, in other words, the expression rate of \( a \) in state \( S \)). We consequently need four parameters for this gene, one for each context. \( F_a(S) \) can be formally expressed as follows:

\[
F_a(S) = \begin{cases} 
K^1_a & \text{if } S_b < t^1_k \land S_d < t^2_k \\
K^2_a & \text{if } S_b < t^1_k \land S_d \geq t^2_k \\
K^3_a & \text{if } S_b \geq t^1_k \land S_d < t^2_k \\
K^4_a & \text{if } S_b \geq t^1_k \land S_d \geq t^2_k 
\end{cases}
\]

The parameters \( K^l_a \) (where \( l \) is the index of the corresponding cellular context and \( c \) is the gene identifier) are related to the kinetic parameters of the underlying differential description. More precisely, \( K^l_a \) is related to the expression rate of gene \( c \) in the cellular context identified by \( l \). These parameters are in general not measured directly, thus are not known (the same is true for the parameters \( \max_c \) and \( t^l_k \)).

However, there are constraints on these parameters which are imposed by additional hypotheses about composition of interactions and properties on these compositions (in Corblin et al (2009a,b) we give the details of a lanugage in GNBox which permits to enforce such hypotheses). For instance, the observation of an inhibition of gene \( b \) by gene \( a \) acting at threshold \( t^2_k \) corresponds to an observability constraint of the form: \( (K^3_a < K^1_a) \lor (K^4_a < K^2_a) \). Another kind of constraints between kinetic parameters is the additivity constraint which expresses the observation (or hypothesis) that when two regulatory proteins \( a \) and \( b \), acting on \( d \), are present simultaneously, their combined effect is additive. An important point to note is that the \( K^l_a \) parame-


tors contains the information about how influences combine on node \( c \). Consequently, in this formalism, the inference of parameter values entail the inference of combination functions (one per node).

## 2 Modeling in CLP and SAT

The transition rule described previously is formalized in terms of multi-valued variables and arbitrary numerical/boolean constraints in Section 2.1. This formal description can be used directly with finite domain CLP solvers. But SAT solvers being nowadays extremely powerful, we explain briefly in Section 2.2 how the CLP formulae are translated into a boolean encoding in order to be able to use SAT solvers.

### 2.1 The CLP encoding of the Thomas formalism

The key point is the encoding of the transition rule. We must express the formal relation between a state and its successors by using intermediate variables relative to the focal state. The relation between a state and its associated focal state is given by the focal equations introduced informally in 1.2. To make this formal we use the notion of cellular context introduced previously.

The first step is to create auxiliary boolean variables, noted \( B_{i,S} \), defining the truth value of each elementary condition in focal equations for a state \( S \) and an interaction \( i \). An elementary condition is an inequation associated to an interaction from \( c \) to \( c' \) with the threshold \( t^c_i \) (arrow in the interaction graph, as in figure 1) and reflects the position of the concentration \( S_c \) compared to \( t^c_i \). Formally an interaction \( i \) is described by a triple: an effector \( \text{effector}(i) \), a threshold \( \text{threshold}(i) \), a target gene \( \text{target}(i) \). For example, the interaction in gapGRN from \( kr \) to \( hbz \) (cf Fig. 1) is identified by \( i = (kr, t^c_{kr}, hbz) \) and \( B_{i,S} \) for this interaction \( i \) is defined by:

\[
B_{i,S} = (S_{\text{effector}(i)} \geq \text{threshold}(i))
\]

Now, it becomes easy to define formally the cellular context of index \( l \) for the component \( c \) and a state \( S \), noted \( \text{Cell}_c^l,S \). These conditions \( \text{Cell}_c^l,S \) depend only on the variables \( B_{i,S} \) such that \( \text{target}(i) = c \). More precisely, \( \text{Cell}_c^l,S \) is a conjunction of such \( B_{i,S} \) or negation of such \( B_{i,S} \). Note that, by construction of the focal equations, for all state \( S \) and component \( c \), we have exactly one condition \( \text{Cell}_c^l,S \) which is true.

The second step of the encoding is the definition of the relation between cellular contexts \( \text{Cell}_c^l,S \), parameters \( K^c_l \) and focal components \( F_c(S) \) for a current state \( S \). It is given by:

\[
\forall c, \forall l, \text{Cell}_c^l,S \Rightarrow F_c(S) = K^c_l
\]

The third step is to define the possible successor states of a current state. We introduce the (multivalued) variable \( S_i' \) which is the value of the component \( c \) of a possible successor state \( S_i' \) of a state \( S \). We introduce several boolean variables: \( \text{Bst true if and only if } S \) is steady, \( \text{Bdown}_c \) and \( \text{Bup}_c \) corresponding to a decrease and an increase of the value of \( S_c \), respectively, for the transition from \( S \) to \( S_i' \). With the help of these variables, the components \( S_i' \) are defined by the predicate \( \text{successor}(S, S_i', F) \) which is true if \( S_i' \) is a successor of \( S \) according to the focal state \( F \) of \( S \). More precisely, it is true if the conditions a, b and c in the informal presentation of the transition rule in 1.2 are true. Formally,

\[
\text{successor}(S, S_i', F) \iff (\text{Bst} \Rightarrow S = S_i') \land (\begin{cases} \text{Bdown}_c \Rightarrow S_i' = S_c - 1 & \text{defines } \text{Bdown}_c \\ \text{Bup}_c \Rightarrow S_i' = S_c + 1 & \text{defines } \text{Bup}_c \\ S_c - 1 \leq S_i' \leq S_c + 1 & S_i' \text{ adjacent to } S_c (a) \\ \text{Bdown}_c \Rightarrow S_c > F_c & \text{tendency to go down (b)} \\ \text{Bup}_c \Rightarrow S_c < F_c & \text{tendency to go up (b)} \\ \text{Bst} \Rightarrow S = F & \text{stationary tendency (c)} \\ \end{cases}) \land \text{exactly_one(<list of } \text{Bdown}_c, \text{Bup}_c \text{ and } \text{Bst}>)}
\]

The constraint on the last line links together all the previous constraints. It enforces that exactly one boolean variable is true among the \( \text{Bdown}_c \) variables, the \( \text{Bup}_c \) variables and \( \text{Bst} \).

A rapid examination shows that the number of booleans necessary for expressing a transition between two states, namely \( B_{i,S}, \text{Cell}_c^l,S, \text{Bdown}_c, \text{Bup}_c, \text{Bst} \) is linear with respect to the number of species. In the case of the \( \text{Cell}_c^l,S \) one can note that their number grows exponentially according to the inside branching factor of the interaction graph. Fortunately, this factor is rarely greater than five. It should be also remarked that a path composed of successive states requires for its definition a number of constraints which is proportional to the size of the path.

The main predicate of our tool GNBox is \( \text{path}(M, Path, L) \) true if \( Path \) is a possible succession of \( L \) states for the model \( M \). Formally,

\[
\text{path}(M, Path, L) \iff \bigwedge_{i \in [1..L-1]} (\bigwedge_c (\text{Cell}_c^{Path_i,S} \Rightarrow F_c(\text{Path}_i) = K^c_i) \land \text{successor}(\text{Path}_i, \text{Path}_{i+1}, F(\text{Path}_{i+1}))
\]
2.2 The SAT encoding

The CLP formalization can be used directly. However, for efficiency reasons, we developed a scheme to translate this description into a CNF formula in order to use SAT solvers. We now present our procedure to replace a multivalued variable by a set of Boolean variables and clauses, and the ideas behind our boolean encoding of integer constraints.

2.2.1 Boolean representation of a multivalued variable.

We chose a straightforward manner to encode as boolean variables the multivalued variables of the model. For every variable $S_c, F, K_i$, $t^c_i$, and $max_c$, we introduce as many boolean variables as the size of their domain, and a set of clauses specifying that exactly one of these new boolean variables is true.

Example: For the variable $S_i$, having the domain $\{0, 1, 2\}$, we introduce three Boolean variables $x_0, x_1, x_2$ and the following links to the values of $S_i$: $x_0$ is equivalent to $S_i = 0$, $x_1$ is equivalent to $S_i = 1$, etc. Then, we add the clauses: $x_0 \lor x_1 \lor x_2$, $\neg x_0 \lor \neg x_1$, $\neg x_0 \lor \neg x_2$ and $\neg x_1 \lor \neg x_2$.

For a variable with a domain of $q$ values we obtain 1 clause of $q$ literals, and $q(q - 1)/2$ clauses of 2 literals (so with immediate propagation on these two literals in current SAT solvers). Since in our context the variable domains are small, we obtain finally a quite small set of clauses (of a non penalizing size).

This encoding is not efficient if $q$ is large. Another possible encoding is the one proposed in Guinčhiglia et al (2004). This encoding is linear according to $q$. More precisely, it introduces $q - 1$ new boolean variables and generates $4 \ast q - 5$ clauses of two or three literals. This encoding becomes interesting if $(q(q - 1)/2) + 1 \geq 4 \ast q - 5$, and this is true for $q \geq 8$. This case is rare in the context of genetic networks. In typical genetic networks each gene is influenced by a very small subset of the set of all the genes.

2.2.2 Introduction of relations between multivalued variables.

We have to encode only relations with at most two multivalued variables with small domains. In spite of the apparent lack of heavy encoding problems, we cannot use the trivial way, based on the truth table of the initial relation, to obtain a clausal form. We would face an exponential explosion of the number of generated clauses according to the size of the variable domains. By taking into account the fact that exactly one of the Boolean variables for each possible value of a multivalued variable is true, we get a more efficient encoding.

Example: Let us consider a relation of the type $B \iff X = Y$, with the domains $\{0, 1\}, \{0, 1, 2\} \{1, 2, 3\}$ for $B$, $X$ and $Y$, respectively (like $Bup_i \iff S_i = S_i + 1$, see definition of the predicate successor in Section 2.1). Let $x_0, x_1, x_2, y_1, y_2, y_3$ and $b$ be the Boolean variables which are linked to the multivalued variables $X, Y$ and $B$ ($B$ being considered in CP as a finite domain variable). To encode this relation, we introduce six clauses relative to the conjunction of the two following formulæs:

- $B \Rightarrow X = Y$: $(-b \lor \neg x_1 \lor y_1) \land (-b \lor x_1 \lor \neg y_1) \land (-b \lor \neg x_2 \lor y_2) \land (-b \lor x_2 \lor \neg y_2)$
- $X = Y \Rightarrow B$: $(b \lor \neg x_1 \lor \neg y_1) \land (b \lor \neg x_2 \lor \neg y_2)$

To do the encoding of our model we need a procedure for every type of relation: $B_1 \lor B_2, B_1 \land B_2, B_1 \iff B_2, X = Y, X \neq Y, X \leq Y, X < Y, B \Rightarrow X = Y, X = Y \Rightarrow B, B \Rightarrow X < Y, X < Y \Rightarrow B, B \Rightarrow \bigwedge_i B_i$ and $\text{exactly\_one}(<\text{ list of } B_i >)$, where the $B_i$ variables are booleans and the $X$ and $Y$ variables are potentially multivalued. The size of the CNF representation for all these relations is linear according to the size of the multivalued variable domains (which is the number of species in the system), except for $\text{exactly\_one}(<\text{ list of } B_i >)$. This last constraint produces a quadratic number of clauses with respect to the number of species, but there is only two literals in each clause.

3 Looking for a minimal model of the Drosophila gap-gene module

We now show how the formalism developed in previous Sections is applied to solve a concrete modeling problem about a specific biological system, the Drosophila gap-gene network (gapGRN). The embryo of Drosophila melanogaster is segmented along the anterior-posterior axis into several regions. In each region the cells are characterized by the concentrations of a few specific proteins. A cell "knows" the region to which it belongs by sensing the values of these concentrations. Segmentation takes place in several successive stages, the first of which is controlled by the gapGRN. Four different regions will be distinguished for this early stage.

The starting model is taken from Sánchez and Thieffry (2001). From this paper we deduced the formal interaction graph of Fig. 1 and we extracted the observations about the behaviour of the network. These observations are constituted of data about the stationary state of each region, for the wild type and several mutants.

The interest of this biological problem is twofold in the context of our work. First we have to impose the existence of steady states for several regions and mutants. Remember that each cell carries its own copy of the network. In each region, due to differing environmental cues (provided by three inputs) the network reaches a different steady state. In addition, variants (mutants) of the network reach different steady states. We thus have to describe these variants and their behaviour. Second, the model proposed by Sánchez and Thieffry (2001) can be explored with respect to the minimum number of distinct thresholds necessary to explain the observations.
in the framework of Thomas' network. In other words, we want a method to identify the minimal model (with respect to the number of thresholds) consistent with the observations.

We now present the modeling hypotheses and biological observations from Sánchez and Thieffry (2001), and express them as constraints. The model of the gap-gene module contains 7 genes: giant (gt), hunchback zygotic (hb), krüppel (kr), knirps (kni), bicoid (bcd), hunchback maternal (hb_m), and caudal (cad). Three of them, bcd, hb_m and cad, are input genes the concentrations of which are determined by the mother. Thus, the concentration of the associated proteins is represented by an input parameter. In the interaction graph G the input genes are represented by dotted circles. The model of Sánchez and Thieffry (2001) takes into account four regions along the anterior-posterior axis of the embryo, denoted A, B, C and D. Data about steady states (phenotypes) are available for the wild type, noted wt, and for nine mutants: knockout (KO) on gt (the expression rate of gt is 0 everywhere), noted gt0; KO on both hb and hb_m, noted hb0; KO on kr (kr0); KO on kni (kni0); KO on bcd (bcd0); KO on cad (cad0); KO on hb_m (hbm0); ectopic expression equal to 1 on gt (the expression rate of gt is forced to be everywhere equal to 1), noted gt1; ectopic expression equal to 1 on kni (kni1).

A model \( M^T \) is defined for each type \( T \in \{wt, gt0, hb0, kr0, kni0, bcd0, cad0, hbm0, gt1, kni1\} \). All mutant models are structural variants of \( M^wt \). The formal specification of a given mutant model reflect the fact that most of the parameters are shared with the wild-type model (there is not a whole new set of parameters for each mutant).

The observations relate to the existence of a steady state for each mutant type and each region. They are formalized by constraints of existence of steady states and by constraints between these stationary states.

A state of gapGRN is represented by a list of four discrete concentrations, as explained in previous Sections. The steady state of gapGRN in region \( R \in \{A, B, C, D\} \) and mutant type \( T \) is denoted \( S_{R,T} = (S_{R,T}^{gt}, S_{R,T}^{hb}, S_{R,T}^{kr}, S_{R,T}^{kni}) \). Similarly, the concentrations of the proteins produced by input genes cad, hb_m and bcd for each region \( R \) and each mutant type \( T \) are respectively noted \( S_{cad}^{R,T} \), \( S_{hb_m}^{R,T} \) and \( S_{bcd}^{R,T} \) (environment state). The constraint corresponding to the observation of a steady state for each region \( R \) and each mutant type \( T \) is:

\[
\forall T, \forall R, \quad S_{R,T} = [S_{R,T}^{gt}, S_{R,T}^{hb}, S_{R,T}^{kr}, S_{R,T}^{kni}] \land \text{path}(M^T, [S_{R,T}^{gt}, S_{R,T}^{hb}], P)
\]

where \( P \) represents the set of parameters, common to all models (wild-type and mutants).

The three input nodes cad, hb_m and bcd correspond to mRNAs which are deposited at the egg extremities by the mother. This creates a polarization axis in the embryo which is discretized in Sánchez and Thieffry (2001) into four regions. The gapGRN of a given cell in the embryo feels the local concentrations of the proteins produced by these maternal mRNAs, and evolve toward a steady state which is specific of the region. These maternal concentrations are not known but are restricted by a set of constraints. For example, the models in which the maternal genes are not mutated, have concentration profiles along the anterior-posterior axis which are identical to the wild-type:

\[
\forall R, \quad S_{R,bcd} = S_{R,wt}, \quad S_{R,hb_m} = S_{R,wt}, \quad S_{R,kr} = S_{R,wt}, \quad S_{R,kni} = S_{R,wt}
\]

\[
\forall T \in \{gt0, kr0, kni0, gt1, kni1\}, \quad [S_{R,bcd}^{R,T}, S_{R,hb_m}^{R,T}, S_{R,kr}^{R,T}, S_{R,kni}^{R,T}] = [S_{R,wt}^{R,T}, S_{R,wt}^{R,T}, S_{R,wt}^{R,T}, S_{R,wt}^{R,T}]
\]

The genetic data reviewed in Sánchez and Thieffry (2001) allow us to generate many constraints relating the concentrations in regions A, B, C and D, for each mutant. There is no space to give the details here, but let us just say that these constraints are inequalities between finite-domain variables.

In Sánchez and Thieffry (2001), hypotheses were made concerning the number of thresholds. These choices are reflected in the interaction graph of Fig. 1. It can be seen for example, that four arrows start on \( hb_2 \), but two of them carry the same threshold \( t_{hb_2}^1 \). Insofar as the subdividing of the concentration space is only speculative, it is interesting to ask whether all these different thresholds are necessary to satisfy the observations. The extreme case would be that all the observations and hypotheses can be rendered by a boolean model, i.e. all the thresholds pertaining to a given node are equal. It is easy with our tool to enforce equality constraints between threshold variables. For this, we fix all the threshold variables \( t_{c}^p \) to the value 1, and so \( \max c \) is fixed to 1 for any component \( c \). It appears that the set of all constraints on the stationary states of the 4 regions for each of the 10 types (wild-type and mutant) is inconsistent.

To identify the minimum number of different thresholds needed to satisfy all the hypotheses, we must build a query using a method of constraint relaxation. The relaxation takes place on the number of thresholds for each component \( c \) from a number of thresholds \( Min = 1 \) to a number of thresholds \( Max \) equals to the maximum value of \( p \) of the parameters \( t_{c}^p \). Moreover, we wish to keep the inequality constraints between the thresholds for this model:

\[
\forall p1, p2, \quad (p1 < p2 \Rightarrow t_{c}^{p1} \leq t_{c}^{p2})
\]

To summarize, we challenge the hypotheses about the number of thresholds for all component. For this, we introduce:

- two Booleans \( B_1 \) and \( B_2 \) for the gene \( hb_2 \) (the number of thresholds being between 1 and 3). \( B_1 \) is equivalent to "the number of thresholds for \( hb_2 \) is strictly less than 2". \( B_2 \) is equivalent to "the number of thresholds for \( hb_2 \) is strictly less than 3".
• one Boolean $B_3$ for $kr$ equivalent to “the number of thresholds for $kr$ is strictly less than 2”
• two Booleans $B_4$ and $B_5$ for $bcd$.
• one Boolean $B_6$ for $cad$.

Finally, we seek the complete predication $R_1$ obtained by disjunction of relaxation conditions. For this query we get in 63 seconds (mostly spent in the SAT solver) the following property $R_1$:

$$R_1 \iff (\neg B_1 \land \neg B_2 \land \neg B_3 \land \neg B_4 \land \neg B_5 \land \neg B_6)$$

This query indicates that all the thresholds introduced in Sánchez and Thieffry (2001) are necessary to account for the available genetic data. To give an idea of the size of the SAT instances generated, the last query is constituted of 55634 boolean variables and 239809 clauses. From this point we can determine the impact of each data item on the final result.

4 Conclusion and perspectives

We have explained how a class of discrete networks can be formalized with constraints in CLP, and the SAT. Not only the structural knowledge (network architecture, or interaction graph) is represented in this way, but also behavioral knowledge. This formalization of a whole body of knowledge generates a constraint system which is submitted to a solver. This allows to determine whether the assumed architecture is consistent with the observed data, and if so, to characterize the set of solutions (which may be huge if the system is underconstrained). The number of solutions is in itself a useful information for the biologist. It means that experiments generating stronger constraints are needed. We are currently developing methods to assist in the choice of such informative experiments. The formal representation of experimental data and network dynamic rules in an integrated framework opens the way to model-driven experimentation and automated procedures to reason on data and models.

For cases where the systems is inconsistent, a constraint relaxation procedure has been developed. A subset of unreliable constraints is defined, and one searches a satisfiable system by removing the smallest number of constraints in this subset. This is used in particular for automated model revision. The model parameters are treated as formal variables and can thus be inferred from experimental if they have not been measured directly. This is the case for the expression rates $K_{ic}$ and the thresholds. We have shown here how the minimum numbers of distinct thresholds can be inferred from genetic data. This is obtained without having to carry out numerous trials. As a last remark, let us say that our framework is not limited to the discrete formalism of R. Thomas. Other dynamic rules could be easily implemented, as long as they involve finite-domain quantities (integers). This includes S. Kauffman’s synchronous boolean networks (Kauffman (1969)), and the whole family of networks with block-sequential update rules.

References

Corblin F, Fanchon E, Trilling L (2009a) Applications of a formal approach to decipher discrete genetic networks Presently being written