Modeling and analysis of qualitative behavior of gene regulatory networks

Alvis Brazma[1], Karlis Cerans[2], Dace Ruklisa[3], Thomas Schlitt[4], and Juris Viksna[2⋆]

[1] European Molecular Biology Laboratory, European Bioinformatics Institute, EMBL-EBI, Hinxton, UK brazma@ebi.ac.uk
[2] Institute of Mathematics and Computer Science, Riga, Latvia
karlis.cerans@lumii.lv, juris.viksna@lumii.lv
[3] University of Cambridge, Cambridge, UK dr320@medschl.cam.ac.uk
[4] King's College London, London, UK thomas@thomas-schlitt.net

Abstract. We describe a hybrid system based framework for modeling gene regulation and other biomolecular networks and a method for analysis of the dynamic behavior of such models. A particular feature of the proposed framework is the focus on qualitative experimentally testable properties of the system. With this goal in mind we introduce the notion of the frame of a hybrid system, which allows for the discretisation of the state space of the network. We propose two different methods for the analysis of this state space. The result of the analysis is a set of attractors that characterize the underlying biological system.
Whilst in the general case the problem of finding attractors in the state space is algorithmically undecidable, we demonstrate that our methods work for comparatively complex gene regulatory network model of λ-phage. For this model we are able to identify attractors corresponding to two known biological behaviors of λ-phage: *lysis* and *lysogeny* and also to show that there are no other stable behavior regions for this model.

Introduction

Hybrid systems (HS) are a natural choice for modeling biomolecular networks for at least two reasons: 1) they can model processes that are relevant to behavior of biomolecular networks – they can describe both discrete aspects (e.g. states of activity of specific promoters) and continuous aspects (e.g. concentrations of biological substances in a cell); 2) well established mathematical techniques and supporting software tools exist for analysis of such hybrid system models. One of the first explicit applications of a HS based approach to the modeling of biomolecular networks has been described by Alur et al in [2], where the authors discuss a rather general class of HS models and show that such models are adequate for description and simulation of biological networks.

⋆ The authors are listed in alphabetical order and have equally contributed to the paper. The work was supported by Latvian Council of Science grant 258/2012 and Latvian State Research programme project NexIT (2014-2017).

There is a significant number of other studies discussing applications of HS to biomolecular network modeling, often proposing somewhat more restricted formalisms than the one used in [2] and providing examples of applications of these models to description of specific biological systems (see for example [7], [9], [4], [15] and [1], which is by no means a comprehensive list). One of the most recent of such studies [8] describes an HS based Temporal Evolution Model and applies it to modeling of *Drosophila* circadian cycle. Multiaffine Hybrid Automata models ([10], [3]) that correspondingly have been applied to cardiac cell and bone cell modeling technically are similar to our approach, however the emphasis is on the simulation and identification of parameter values.

Whilst not stated in terms of HS explicitly, a related approach has been presented in [16, 17]. These models describe a biological system using differential equations and then analyze stability of specific cyclic behaviors ('circuits') at a logical level. Notably, by using this approach the stability of several regulatory circuits for λ-phage has been shown ([17]).

Our work presented here is generally in line with previous studies of application of hybrid systems to biomolecular network modeling and is motivated by two observations. First, it can be experimentally difficult to measure the quantitative parameters of biological systems accurately and experimental results often are closer to a qualitative assay than a quantitative measurement (e.g., it may be possible to detect if the concentration of a particular substance is increasing or decreasing, while measuring the exact rate is much more difficult). Second, in some cases it is possible to separate the structure of the underlying regulatory network from its quantitative parameters. In this case it is natural to ask to what extent the qualitative behavior of the system depends on the structure of the network alone, and to what extent on the exact quantitative values (relative or absolute) of the parameters.

Driven by these assumptions we propose a Hybrid System Model (HSM) tailored to the description of biomolecular networks and gene regulatory networks in particular. HSM can be viewed as a restricted version of hybrid system that still provides sufficient power for modeling of biological systems, while the restrictions imposed upon HSM facilitate the analysis of the models. HSM is a generalization of the authors' previous work on Finite State Linear Model (FSLM) ([5, 12, 13]).

A variation of HSM model and its application to the analysis of behavior of gene network of λ-phage ([11]) has been previously described by the authors in [6]. Here we present a more developed mathematical formalism for separation between quantitative and qualitative parameters of the system: 1) we assume that a biological system is correctly represented by a HSM, however the parameter values are unknown; 2) known however is a structure (modes and transitions between them) of HSM represented by its *frame*; 3) the analysis of HSM behavior is done at the level of *constrained frames* in which the exact parameter values are replaced by discrete constraints on them.

We also present a new algorithm for analysis of the *universal state space* of the constrained frames of HSM. This allows us to derive the constraints that affect the behavior of the system in single process of universal state space analysis,

avoiding exhaustive analysis of state spaces for all the possible sets of constraints that has been done previously. The mathematical formalism is presented here also in mathematically more rigorous terms than in the previous work.

For the assessment of the merits of the modeling and analysis methods described here, we have applied the model to a well-studied gene network of λ-phage [11]. λ-phage is a bacterial virus, which when invading its host can exhibit two different stable behaviors *lysis* and *lysogeny*.

It should be noted that due to the undecidability of the reachability problem for HSM we can not guarantee that HSM state space analysis will provide any results. Therefore it is noteworthy that for a comparatively complicated gene regulatory network model of λ-phage our method was able to identify two regions of stable behavior in the model's state space that correspond to the two biologically known behaviors – lysis and lysogeny. Moreover, these are the only regions of stable behavior and their existence does not depend on the exact quantitative parameters, but only on the structure of the network and the experimentally known qualitative information.

Hybrid systems for modeling gene regulatory networks

Like in most previous approaches that use HS for modeling gene regulatory networks (GRN), we use *modes* to represent different combinations of transcription factor binding site states (a binding site may be either vacant or in an occupied state) and continuous variables for representing the concentrations of various biological substances (e.g. proteins) in a cell. We assume that in each mode substance concentrations change according to continuous functions from a given set of possible functions. The change of mode is defined by a condition in transition diagram; mode is changed when a concentration of a substance reaches a certain threshold, i.e. one of the predicates 'guarding' a transition from the given mode is satisfied. In gene regulatory networks these thresholds correspond to association or dissociation concentrations of proteins that have to be reached in order to bind or dissociate from a particular binding site.

We keep our HSM formalism as simple as possible, but still sufficient for modelling biological processes. Most notably we disallow instantaneous resetting of continuous values to 0, since such resetting does not seem to have any valid biological interpretation.

Definition 1. *A Hybrid System Model (HSM) is a 6-tuple* $\mathcal{H} = \langle M, X, C, T, F, MF \rangle$, *where:*

1. $M = \{\mu_1, \ldots, \mu_k\}$ *is a finite set of* modes.
2. $X = \{x_1, \ldots, x_m\}$ *is a finite set of* continuous variables *that can assume real non-negative values.*
3. $C = \{c_1, \ldots, c_r\}$ *is a finite set of real non-negative* transition constants.
4. T *is a set of* mode transitions*, where each transition* $\tau \in T$ *has the form* $\tau = \alpha \rightarrow_p \beta$*, where* $\alpha, \beta \in M$ *and* $p = p(\tau)$ *is a predicate that has a form* $x \leq c$ *or* $x \geq c$ *for some variable* $x = x(\tau) \in X$ *and some constant* $c = c(\tau) \in C$*. Predicate* p*, called a* guard*, is a function* $p : \mathbf{R}_+ \rightarrow \{true, false\}$ *the value of which depends on the value of variable* x*.*

5. $F = \{f_1, \ldots, f_n\}$ *is a set of real non-negative two argument* growth/degradation functions $f_i : \mathbf{R}_+ \times \mathbf{R}_+ \to \mathbf{R}_+$ *that are continuous and monotonous in both arguments and for which* $f_i(z, 0) = z$ *for every* $z \in \mathbf{R}_+$.
6. $MF : M \times X \to F$ *is a mapping providing* mode-function assignments *assigning to each mode* $\alpha \in M$ *and each variable* $x \in X$ *a function* $g \in F$.

A 'toy example' of hybrid system model of a GRN with two genes and three protein binding sites is shown in Figure 1.

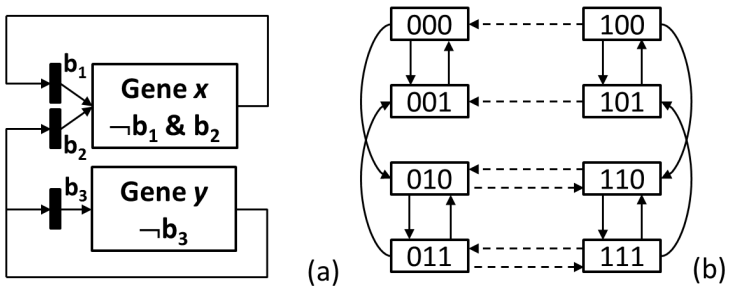

Fig. 1. A 'toy example' of GRN with two genes and three protein binding sites (a), and the corresponding HSM with 8 modes (b). Each binding site b becomes occupied if the concentration of gene product binding to it reaches an association constant a_b and becomes vacant if the concentration drops to dissociation constant d_b (where $a_b > d_b$). HSM modes are denoted by binding site states (e.g. 100 represents the situation when site b_1 is occupied and b_2 and b_3 are vacant); no concrete growth functions are shown, but functions are increasing if the value of a Boolean function from the states of binding sites is $true$ and decreasing if $false$. Transitions triggered by changes of variable x are shown with dashed and transitions triggered by changes of variable y with solid lines, constants in transition guards are not shown.

Intuitively a mode-function assignment assigns to each mode $\alpha \in M$ and each variable $x \in X$ a growth/degradation function that describes the change of this variable in time. Whilst these functions could be viewed as a part of the corresponding modes, if we are concentrating on qualitative aspects of such functions (e.g. are they increasing or decreasing) and not on their precise form, it may be useful to distinguish between pairs of modes with identical functional assignments for a given variable and pairs of modes with different ones.

Each growth/degradation function f has two real valued arguments. The first of them corresponds to the value of a variable x at the time of switching to a particular mode for which f is assigned to be the function regulating x, while the second is the time Δt elapsed since the switch. For instance, in a special case when the growth rates are constant, the functions are linear and have a form $g(x, \Delta t) = x + c \cdot \Delta t$, where c is a constant. For these particular functions the condition $g(z, 0) = z$ is also satisfied. This condition ensures the continuity of trajectories when the system switches from one mode to another.

In comparison with more general definitions of hybrid systems we have placed several restrictions on the set of functions according to which the substance concentrations can change. One such restriction is that only the mode of the system and the value of a particular substance determines the rate of concentration change until a switch to a different mode happens. In addition in a given mode the concentration of a particular substance can only increase, decrease or stay constant. In practice we also need to impose a few additional restrictions (mostly too technical to merit inclusion in HSM definition) to exclude some 'undesirable behaviors' of the system. Most notably we require that if concentration of some substance is moving towards a transition constant triggering a guard then eventually this constant will be reached. The formalism is strong enough to describe biological systems, but does not provide more freedom for the behavior of the system than is necessary.

To describe the behavior of a HSM starting at a given initial mode with a given set of initial variable values we use a concept of *run*. A run describes an evolution of HSM over time by specifying a concrete sequence of modes through which the system is evolving and assigning well-defined values $x(t)$ to each of the variables $x \in X$ at time moments t.

Let $\mathcal{H} = \langle M, X, C, T, F, MF \rangle$ be a HSM with given initial mode α_0, initial time moment t_0 and initial values of variables $X(t_0) = (x_1(t_0), \ldots, x_m(t_0))$. We define a *run* of the system $\mathcal{H}$ as a (finite or infinite) sequence of modes $\alpha_i \in M$ and times t_i: $\mathcal{R}(\alpha_0, t_0, X(t_0)) = (\alpha_0, t_0) \to (\alpha_1, t_1) \to (\alpha_2, t_2) \to \cdots$.

For each (α_i, t_i) t_i is the time point when $\mathcal{H}$ switches to the mode α_i. While the system is in the mode α_i its variables change as defined by the mode function assignment in that mode (i.e. for all j: $x_j(t) = g_j(x_j(t_i), t - t_i)$, where $g_j = MF(\alpha_i, x_j)$ – a function assigned to x_j in mode α_i). Such an evolution continues until one of the guards is satisfied (if several are satisfied simultaneously, we can assume that one is selected by some deterministic procedure).

A particular run $\mathcal{R}(\alpha_0, t_0, X(t_0))$ assigns well-defined functions describing changes of the variables in time. For each $x_j \in X$ and $t \in [t_i, t_{i+1}]$ it defines $x_j(t) = g_j(x_j(t_i), t - t_i)$ and thus also defines a vector function $X = (x_1, \ldots, x_m)$ in the whole interval $[t_0, \infty[$.

Thus a specific run of $\mathcal{H}$ describes a precise and (in principle) experimentally measurable behavior of the system, and given an appropriate HSM for some biological system, a run can be regarded as simulation of the behavior of this system starting from some known initial conditions. Still, if we want to describe all the possible behaviors of the system in such a way, we (normally) need a continuum of different runs.

A natural and well explored alternative is to disregard the exact values of function X describing changes of variables, but consider only sequences of modes $\alpha_0, \alpha_1, \ldots$ that can occur in runs. Provided that modes of HSM have well defined biological interpretation, modes can also be more easily determined in experiments than exact concentrations of substances. Still, even experimental measurement of modes may not be a simple task.

We define a *path* of HSM in order to describe a finite sequence of modes that can occur in a particular run: a finite sequence $\alpha_0, \ldots, \alpha_n$ is called a *path* if there is a run $\mathcal{R}(\alpha_0, t_0, X(t_0)) = (\alpha_0, t_0) \to \cdots \to (\alpha_n, t_n) \to \cdots$.

Qualitative behavior of networks and frames of HSM

An appropriate HSM $\mathcal{H}$ can provide a good approximation of a biological system. However such a model also involves a large number of quantitative parameters: the set of growth/degradation functions and the set of the guards governing transitions, the knowledge of which is only rarely a realistic assumption.

Nevertheless usually we can assume that a set of modes M is known – a separate mode can be assigned to each state of gene activity (active or not) and/or binding site state (occupied or not). In addition it is often possible to define the set of variables X (e.g. the set of substances in whose concentrations we are interested), and gather the information about the transitions in T (the modes and variables involved) and a partial information about the growth and degradation functions (e.g. whether a particular concentration is growing or decreasing). Guards in our model have a very simple form: either $x \leq c$ or $x \geq c$. Often we know the type of the inequality, but not the exact constant c.

To specify a HSM using such a limited information (more of qualitative than quantitative nature) about the system we introduce the notion of *frame*. A single frame is intended to represent a whole set of HSMs that are consistent with the existing knowledge about the system.

A *frame* of HSM is defined as a 5-tuple $\mathcal{F} = \langle M, X, C', T, MF' \rangle$, where the requirements on M, X, T are the same as in the definition of HSM, but we don't have a set of concrete functions F and use an assignment $MF' : M \times X \to \{\nearrow, \searrow, \to\}$ instead*. In addition for frames a set C' is not a set of constants, but instead is a set of *variables* assuming real non-negative values. For two different transitions τ_1 and τ_2 the notation $c(\tau_1) = c(\tau_2)$ indicates that the 'constants' involved in the guards of these transitions are the same, otherwise it is assumed that these 'constants' are distinct. Normally we will use $c(\tau_1) = c(\tau_2)$ only for transitions with $x(\tau_1) = x(\tau_2)$.

Thus essentially a frame is a simplified HSM, for which only qualitative information about the growth/degradation functions and the guards is specified.

For a given HSM $\mathcal{H}$ we can easily construct the frame by using the same sets of modes M, variables X and transitions T and simply replacing C and MF with C' and MF' providing less restricted information about the transition guards and growth functions. We call such a frame an *induced* frame of $\mathcal{H}$ and denote it by $\mathcal{F}(\mathcal{H})$.

The concept of *run* can be extended to frames. Since for frames we lack information about substance concentrations, their change rates and their relation to transition guards, for frames each run is specified only by an initial mode α_0

* For biomolecular networks the value $''\to''$ describing the situation where concentration of some substance does not change is generally reserved for the cases in which concentration is either 0 or the maximal biologically feasible saturation value.

and is just a sequence of modes $\mathcal{R}(\alpha_0) = \alpha_0 \to \alpha_1 \to \alpha_2 \to \cdots$, where $\alpha_i \to \alpha_{i+1}$ is allowed if and only if there is a transition $\tau = \alpha_i \to_p \alpha_{i+1} \in T$.

Runs for frames are generally non-deterministic, i.e. not uniquely specified by initial α_0. Usually runs will be infinite sequences of modes, unless they terminate with some mode β that does not have outgoing transitions. Consistently with the terminology used for HSM a finite sequence $\alpha_0, \ldots, \alpha_n$ is called a *path* if there is a run $\mathcal{R}(\alpha_0)$ containing this sequence as an initial fragment.

In general frame $\mathcal{F} = \mathcal{F}(\mathcal{H})$ will have a multitude of paths $\alpha, \ldots, \beta$ in $\mathcal{F}$ for which there are no corresponding paths (i.e. with identical mode sequences) in $\mathcal{H}$. Moreover, for some paths $\alpha, \ldots, \beta$ in $\mathcal{F}$ there may not exist any HSM $\hat{\mathcal{H}}$ with $\mathcal{F} = \mathcal{F}(\hat{\mathcal{H}})$ in which $\alpha, \ldots, \beta$ is a path (the reason is that for frames we do not have the means of enforcing a consistent behavior of growth/degradation functions each time a particular mode is encountered in the run). Thus in general frame runs may describe behaviors that are not consistent with the qualitative information about the growth/degradation functions of the system included in the specification of its frame. However, it is evident that for each run $\mathcal{R}(\alpha_0, t_0, X(t_0))$ of $\mathcal{H}$ there always will be a run $\mathcal{R}(\alpha_0)$ of $\mathcal{F}(\mathcal{H})$ with the same sequence of modes (the same will hold also for paths). Therefore, we do not lose any of systems' behaviors when considering frames instead of HSM.

Whilst for a given HSM $\mathcal{H}$ its behavior can be represented by its induced frame, for analysis purposes it may be useful to characterize $\mathcal{H}$ with other frames that have more fine grained structure. To do this we introduce the notion of frame *refinement*.

Definition 2. *A frame* $\mathcal{F}_1 = \langle M_1, X, C', T_1, MF_1' \rangle$ *is a* refinement *of frame* $\mathcal{F}_2 = \langle M_2, X, C', T_2, MF_2' \rangle$ *(denoted* $\mathcal{F}_1 < \mathcal{F}_2$*), if there are surjective mappings* $m : M_1 \to M_2$ *and* $t : T_1 \to T_2$ *such that:*

1. *for all* $\alpha \to_p \beta \in T_1$*:* $t(\alpha \to_p \beta) = m(\alpha) \to_p m(\beta) \in T_2$*;*
2. *for all* $\alpha \in M_1$ *and all* $x \in X_1$*:* $MF_1'(\alpha, x) = MF_2'(m(\alpha), x)$*;*
3. *the sequence* $m(\alpha_0), \ldots, m(\alpha_n)$ *is a path in frame* $\mathcal{F}_2$ *for any path* $\alpha_0, \ldots, \alpha_n$ *in frame* $\mathcal{F}_1$*.*

Thus we allow partitioning of modes and removal of some transitions from partitioned modes in a frame refinement as long as all the paths of the original frame are preserved. Therefore by the definition of a refinement all the 'behaviors' of the initial frame will also be exhibited by its refinement.

In a similar manner we say that a frame $\mathcal{F}$ *supports* HSM $\mathcal{H}$ (denoted by $\mathcal{F} \lhd \mathcal{H}$), if for all paths in $\mathcal{H}$ there are corresponding paths in supporting frame $\mathcal{F}$. It is evident that frame refinement is transitive and that any HSM $\mathcal{H}$ will always be supported by the induced frame $\mathcal{F}(\mathcal{H})$.

In HSM models of biological networks guards of transitions $x \geq c$ ($x \leq c$) can correspond to an event when the concentration of a protein described by a variable x reaches an association constant of some site and binds to it (or drops below a dissociation constant and vacates the site). The exact values of binding site affinities usually are unknown, however in cases where there are several binding sites for the same protein at least a partial ordering of binding affinities may be known. To characterize such affinity orderings we use the notion

of *constraints*. For a given frame $\mathcal{F} = \langle M, X, C', T, MF' \rangle$ we define a *constraint* $O(C')$ as a transitive directed acyclic graph with a set of vertices C'. A constraint effectively specifies a *strict* (i.e. non-reflexive) partial ordering of values in C': for $c_1, c_2 \in C'$ an edge (c_1, c_2) in constraint $O(C')$ is interpreted as inequality $c_1 < c_2$. Normally graphs $O(C')$ will consist of a number of non-connected components, a separate one for each variable in X. We denote by $Cons(\mathcal{F})$ the set of all constraints of frame (i.e. set of all partial orderings of C'). By $O_\emptyset(C')$ we denote constraint with no edges.

Given two constraints $O_1(C')$ and $O_2(C')$ with the same set of vertices C' and sets of edges $E(O_1)$ and $E(O_2)$ the union $O_1 \cup O_2$ denotes a graph with the vertex set C' and the set of edges a transitive closure of $E(O_1) \cup E(O_2)$. If $O_1 \cup O_2$ contains a cycle, it is not a constraint and we say that constraints O_1 and O_2 are *incompatible*. Otherwise we say that O_1 and O_2 are *consistent*. Similarly the intersection of constraints $O_1 \cap O_2$ denotes a graph with the set of edges $E(O_1) \cap E(O_2)$. $O_1 \cap O_2$ is always a constraint.

Definition 3. *A* constrained frame *is a pair* $(\mathcal{F}, CA)$*, where* $\mathcal{F} = \langle M, X, C', T, MF' \rangle$ *is a frame and* $CA : T \to Cons(\mathcal{F})$.

We say that two constraint assignments CA_1 and CA_2 are *incompatible* if for some transition τ constraints $CA_1(\tau)$ and $CA_2(\tau)$ are incompatible. Otherwise we say that CA_1 and CA_2 are *consistent*. For consistent CA_1 and CA_2 it is convenient to denote by $CA_1 \cup CA_2$ the assignment of constraint $CA_1(\tau) \cup CA_2(\tau)$ to each transition τ. We can define $CA_1 \cap CA_2$ analogously. By CA_O we denote the assignment of the same constraint O to all transitions.

In constrained frame each transition is annotated with a partial ordering of C' and we consider transition as 'available' only if its transition constant has the highest priority among all the transitions from the same mode and involving the same variable. Moreover we require that in constrained frames a sequence of modes forming a run should be 'passable' by a sequence of transitions that does not include any transition pairs with incompatible constraint assignments.

To make this more precise we define constraint $O(\alpha, \beta)$ as an intersection of all constraints $CA(\tau)$ for transitions $\tau = \alpha \to_p \beta \in T$ for a particular pair of modes α, β in constrained frame $(\mathcal{F}, CA)$. Furthermore, for each sequence of modes $P = \alpha_0, \ldots, \alpha_n$ we define a graph $O(P)$ as a union of all constraints $O(\alpha_i, \alpha_{i+1})$, $i = 0 \ldots n-1$. Informally, if $O(P)$ is a constraint (i.e. without cycles) it can be regarded as the least restrictive *single* constraint under which the sequence of modes P is 'passable'.

For constrained frames it is convenient to start with a definition of path instead of run – a finite sequence $P = \alpha_0, \ldots, \alpha_n$ is a *path* if $O(P)$ is a constraint. Then $\mathcal{R}(\alpha_0) = \alpha_0 \to \alpha_1 \to \alpha_2 \to \cdots$ is a *run* if each initial fragment of sequence of its modes $\alpha_0, \ldots, \alpha_n$ is a path. For finite runs $\mathcal{R}(\alpha_0) = \alpha_0 \to \cdots \to \alpha_n$ we also require that $\mathcal{R}(\alpha_0)$ is not a proper prefix of some other run $\hat{\mathcal{R}}(\alpha_0)$.

The notion of frame refinement can be extended to constrained frames. The definition of a constrained frame $(\mathcal{F}_1, CA_1)$ being a refinement of $(\mathcal{F}_2, CA_2)$ (denoted by $(\mathcal{F}_1, CA_1) < (\mathcal{F}_2, CA_2)$) is analogous to Definition 2 with an additional requirement that a mapping t of transitions should be consistent with the con-

straint assignments (i.e. $CA_1(t(\tau)) = CA_2(\tau)$) and the requirement for path conservation now referring to paths in constrained frames.

Let us consider HSM $\mathcal{H}$ describing some gene regulatory network. The complete ordering of affinities C of binding sites in $\mathcal{H}$ is represented by some constraint O and it would be adequate to describe this GRN with induced frame $\mathcal{F}(\mathcal{H})$ and constraint assignment CA_O. We call such constraint assignment CA_O *maximal* and for given $\mathcal{H}$ denote it by $Max(\mathcal{H})$. The induced constrained frame will be denoted by $(\mathcal{F}(\mathcal{H}), Max(\mathcal{H}))$. This notation allows to extend definition of frame support for HSM to constrained frames: constrained frame $(\mathcal{F}, CA)$ supports $\mathcal{H}$ (denoted by $(\mathcal{F}, CA) \lhd \mathcal{H}$) if $(\mathcal{F}, CA) < (\mathcal{F}(\mathcal{H}), Max(\mathcal{H}))$.

However usually we do not have complete knowledge of ordering of C. In the worst case when we do not possess any knowledge about the orderings of affinities we can only consider the least constrained frame that will support $\mathcal{H}$ regardless of the affinity ordering in $\mathcal{H}$. In such frame we assign to all transitions in $\mathcal{F}$ constraints that give them the highest priority – i.e. to each transition $\alpha \rightarrow_p \beta \in T$ with guard p of the form $x(\tau) \leq c(\tau)$ we assign a constraint with edge set $\{(c(\tau), c(\hat{\tau}))|\hat{\tau} = \alpha \rightarrow_p \gamma \in T, \gamma \in M, \tau \neq \hat{\tau}, x(\tau) = x(\hat{\tau})\}$ (or edge set $\{(c(\hat{\tau}), c(\tau))|\hat{\tau} = \alpha \rightarrow_p \gamma \in T, \gamma \in M, \tau \neq \hat{\tau}, x(\tau) = x(\hat{\tau})\}$ if p is of form $x(\tau) \geq c(\tau)$). We call the resulting constraint assignment *minimal* and for frame $\mathcal{F}$ denote it by $Min(\mathcal{F})$. In contrast to maximal assignment, which is based on underlying HSM $\mathcal{H}$, minimal assignment is defined solely by the properties of $\mathcal{F}$.

Analysis of dynamic behavior of hybrid system models

Given a constrained frame $(\mathcal{F} = \langle M, X, C', T, MF' \rangle, CA)$ describing some system, all its possible behaviors are represented by runs that are allowed under the specified constraints. We can conveniently characterize all such runs by a graph whose vertices correspond to modes of $\mathcal{F}$ and edges correspond to the transitions in $\mathcal{F}$ that are allowed in runs by a constraint assignment CA. We call such graph a *state space graph* of $\mathcal{F}$ and denote it by $G = G(\mathcal{F}, CA)$. The vertex set of $V(G)$ is simply M, the set of edges $E(G)$ is subset of $\{(\alpha, \beta)|\alpha \rightarrow_p \beta \in T\}$. There is a simple algorithm that computes $G(\mathcal{F}, CA)$ for given $\mathcal{F}$ and CA.

There are notable similarities between frame state space graphs and state space graphs in Boolean gene network models. For Boolean models the analysis of their space graphs is also simple and gives unambiguous characterization of the system's behavior – the graphs decompose into cyclic *attractor* subgraphs, each of which can be regarded as a descriptor of one of the possible behaviors of the system. The information that a frame state space graph provides about the system's behavior is very similar to that given by state space graphs of Boolean models. Unfortunately, however, frame state space graphs can be much more complex and do not allow for a simple partitioning into attractor basins.

We are interested in identifying the regions in frame state space graphs $G = G(\mathcal{F}, CA)$ that will characterize 'stable behaviors' of the system. In order to achieve this task we propose to partition G into *strongly connected components* (SCC) and to relate these components to stable behaviors (similar generalization of attractors has been already used for Random Boolean Networks in [14]).

In the worst case the whole G can consist just of a single SCC. However we can perform a more detailed analysis of the dynamics than just compute a partition of G into SCCs. Let us consider a SCC $S \subseteq V(G)$ and a variable x, such that for all modes $\alpha \in S$: 1) we have $MF'(\alpha, x) = \nearrow$ (or $MF'(\alpha, x) = \searrow$), and 2) there is an edge $(\alpha, \beta) \in E(G)$ derived from transition $\tau = \alpha \rightarrow_p \beta$ with the guard of the form $x \leq c(\tau)$ (correspondingly $x \geq c(\tau)$). In such a situation we can conclude that the system can stay in S only for a limited time, since eventually one of guards for these transitions will get satisfied (due to restrictions we impose on growth functions). In such cases we say that SCC S is *transitional*.

Definition 4. *A strongly connected non-transitional component of constrained state space graph $G(\mathcal{F}, CA)$ is called an* attractor.

Finding of attractors requires splitting state space graphs in SCCs and checking whether each SCC is or is not transitional. The latter task can be achieved in linear time with respect to the size of graph.

Suppose that we have HSM model $\mathcal{H}$ for some gene regulatory network. We assume that we have only qualitative knowledge about parameters of $\mathcal{H}$, i.e., we have complete knowledge whether growth/degradation functions are increasing or decreasing and possibly a partial knowledge about the ordering of transition constants. Such amount of available information seems to be typical for many biological networks. In terms of constrained frames our knowledge about the system is represented by an induced constrained frame $(\mathcal{F}(\mathcal{H}), CA)$ with a set of constraints CA ranging somewhere between $Min(\mathcal{F}(\mathcal{H}))$ (no knowledge about transition constants) and $Max(\mathcal{H})$ (complete knowledge about transition constants). Usually however some information about constant ordering is available, e.g. if constants correspond to association and dissociation affinities of binding sites, then for the same binding site association affinity must be the largest of these two. Additional constraints also can be derived from the known biological facts. We denote these known constraint assignments by $External(\mathcal{H})$.

In order to make some judgments about possible $\mathcal{H}$ behaviors we are interested in finding the sets of all attractors of all frames $(\mathcal{F}(\mathcal{H}), CA)$ for which CA is consistent with $External(\mathcal{H})$. There are two natural ways to do this.

Firstly, we could consider all *complete orderings* $Ord \in Cons(\mathcal{F}(\mathcal{H}))$ of transition constants that are consistent with $External(\mathcal{H})$ and analyze attractor structure of all the corresponding graphs $G(\mathcal{F}(\mathcal{H}), CA_{Ord})$ (thus essentially we check for attractor structure of graphs corresponding to all the possible choices of $Max(\mathcal{H})$). Such an approach has been used by the authors in [6] for analysis of λ-phage model. Despite comparatively large size of this model (11664 modes and 32 transition constants) the number of different orderings that have to be considered is comparatively small – only 42. The main reason for this is (easily provable) fact that attractor structure is influenced only by orderings of subsets of constants $\{c(\tau) | x(\tau) = x\}$ for each variable $x \in X$. Therefore for HSM models where transitions are defined by binding site affinities, in the special case when for each binding factor there is only single binding site it affects, there will be only a single ordering Ord to consider. For our λ-phage model there are multiple binding sites for most of the binding factors and 'biologically known' constraints

are used to reduce the number of possible orderings from a few thousands to 42. One of the results we have presented in [6] is the fact that the attractor structure remains the same for all these 42 orderings and contains only 2 attractors that correspond to two known biological behaviors of λ-phage: *lysis* and *lysogeny*.

An alternative approach on which we focus in this paper is to try to analyze directly attractor structure $G(\mathcal{F}, CA)$ using only known limited knowledge of CA that is given by $External(\mathcal{H})$ and $Min(\mathcal{F}(\mathcal{H}))$ without explicit consideration of all the possible choices of CA. Such an approach has several advantages: first, if two different assignments CA_1 and CA_2 have similar state space graphs we probably can save some work by noticing shared parts in these graphs; second this could help to decide whether the behaviors described by the attractors yielded by different choices of CA are essentially the same or different; third such an approach could help to derive automatically the conditions (i.e. constraint assignments) that separate different qualitative behaviors.

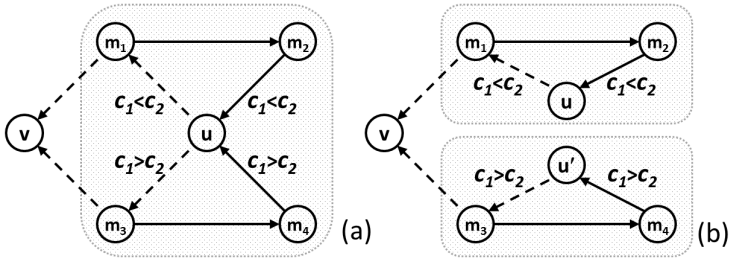

Fig. 2. A simple example showing that refinement of constrained frame (b) can have attractors with fewer number of states than in original frame (a). In original frame there is a SCC containing states m_1, m_2, m_3, m_4, u, which have incompatible constraints for two different paths between states u and v. In refinement this SCC has been split in two SCCs with states m_1, m_2, u and m_3, m_4, u'. Solid and dashed lines represent correspondingly transitions and paths, in the latter case the constraints shown refer to the whole paths.

To perform such analysis we can start with initially given state space graph $G_0 = G(\mathcal{F}(\mathcal{H}), Min(\mathcal{F}(\mathcal{H})) \cup External(\mathcal{H}))$, check for parts of G_0 that will 'behave' differently if different additional restrictions are imposed on existing constraints, and try to partition these parts in such a way that different behaviors are represented by different parts of these partitions. Let G_1 be a graph obtained by such a process from G_0. By repeating this process we will obtain (hopefully finite) sequence of graphs $G_0, G_1, \ldots, G_n$, where G_n can not be further partitioned. By analyzing attractor structure of G_n we can then expect to find attractors for all different choices of constraints compatible with initial constraints $Min(\mathcal{F}(\mathcal{H})) \cup External(\mathcal{H})$ and, moreover, hope, although that is not guaranteed, that some attractors will be shared by several orderings of constants.

It turns out that the process of computing space graph sequence $G_0, G_1, \ldots, G_n$ essentially can be regarded as a process of constructing appropriate constrained frame refinements $(\mathcal{F}_0, CA_0), \cdots, (\mathcal{F}_n, CA_n)$, where $(\mathcal{F}_0, CA_0) = (\mathcal{F}(\mathcal{H}), Min(\mathcal{F}(\mathcal{H})) \cup External(\mathcal{H}))$, $(\mathcal{F}_n, CA_n) < \cdots < (\mathcal{F}_0, CA_0)$ and $G_n = G(\mathcal{F}_n, CA_n)$. An example in Figure 2 shows that such a refinement process indeed can reduce the number of states in the attractors of a state space graph. For computing the sequence of graphs we propose the *RefineStateSpace* algorithm. We use assignments $init(\alpha)$ and $constr(\alpha)$ to refer correspondingly to the initial mode from which α has been derived and to the last transition constraint that has triggered creation of α.

Algorithm *RefineStateSpace*
Input: State space graph $G(\mathcal{F}_0, CA_0)$
Output: Refined state space graph $G(\mathcal{F}, CA)$

1. $\mathcal{F} = \langle M, X, C', T, MF' \rangle \leftarrow \mathcal{F}_0$, $CA \leftarrow CA_0$
2. for each $\alpha \in M$ assign $init(\alpha) \leftarrow \alpha$ and $constr(\alpha) \leftarrow O_\emptyset(C')$
3. **while** there exist $\alpha, \beta \in M$ and different paths $P_1 = \alpha, \gamma_1, \ldots, \beta$, $P_2 = \alpha, \gamma_2, \ldots, \beta$ such that: $O(P_1)$ and $O(P_2)$ are incompatible **and** $init(\delta_1) \neq init(\delta_2)$ for all pairs of internal vertices $\delta_1 \in P_1$ and $\delta_2 \in P_2$ **and** there is no alternative pair P_1' and P_2' of paths with $O(P_1')$ or $O(P_2')$ being a proper subgraph of either $O(P_1)$ or $O(P_2)$
4. **do** the following:
5. let $\tau_1 = \alpha \rightarrow \gamma_1 \in T$ and $\tau_2 = \alpha \rightarrow \gamma_2 \in T$ be transitions corresponding to P_1 and P_2
6. $\hat{M} \leftarrow M$, $\hat{T} \leftarrow T$, $\hat{CA} \leftarrow CA$
7. remove α from $\hat{M}$ and add to $\hat{M}$ two new modes α_1 and α_2
8. assign $init(\alpha_1) \leftarrow init(\alpha)$, $init(\alpha_2) \leftarrow init(\alpha)$, $constr(\alpha_1) \leftarrow CA(\tau_1)$, $constr(\alpha_2) \leftarrow CA(\tau_2)$
9. remove transitions and constraint assignments involving α from $\hat{T}$ and $\hat{CA}$
10. for each $\tau = \alpha \rightarrow \delta \in T$, with $\tau \neq \tau_1, \tau \neq \tau_2$ add transition $\hat{\tau} = \alpha_1 \rightarrow \delta$ to $\hat{T}$ and assign $\hat{CA}(\hat{\tau}) \leftarrow CA(\tau)$
11. for each $\tau = \alpha \rightarrow \delta \in T$, with $\tau \neq \tau_1, \tau \neq \tau_2$ add transition $\hat{\tau} = \alpha_2 \rightarrow \delta$ to $\hat{T}$ and assign $\hat{CA}(\hat{\tau}) \leftarrow CA(\tau)$
12. add transition $\hat{\tau} = \alpha_1 \rightarrow \gamma_1$ to $\hat{T}$ and assign $\hat{CA}(\hat{\tau}) \leftarrow CA(\tau_1)$
13. add transition $\hat{\tau} = \alpha_2 \rightarrow \gamma_2$ to $\hat{T}$ and assign $\hat{CA}(\hat{\tau}) \leftarrow CA(\tau_2)$
14. for each $\tau = \delta \rightarrow \alpha \in T$ if $CA(\tau)$ and $CA(\tau_1)$ are consistent then add transition $\hat{\tau} = \delta \rightarrow \alpha_1$ to $\hat{T}$ and assign $\hat{CA}(\hat{\tau}) \leftarrow CA(\tau) \cup CA(\tau_1)$
15. for each $\tau = \delta \rightarrow \alpha \in T$ if $CA(\tau)$ and $CA(\tau_2)$ are consistent then add transition $\hat{\tau} = \delta \rightarrow \alpha_2$ to $\hat{T}$ and assign $\hat{CA}(\hat{\tau}) \leftarrow CA(\tau) \cup CA(\tau_2)$
16. $M \leftarrow \hat{M}$, $T \leftarrow \hat{T}$, $CA \leftarrow \hat{CA}$
17. **return** $G(\mathcal{F}, CA)$

Algorithm *RefineStateSpace* is greedy and heuristic by its nature and its output depends from the order in which pairs of paths P_1 and P_2 are selected in *Step 3*. Nevertheless we can guarantee that the algorithm terminates and outputs state space graph of constrained frame $(\mathcal{F}, CA)$.

Let us denote by $n(\mathcal{H})$ and $m(\mathcal{H})$ correspondingly the number of modes and the number of transitions in $\mathcal{H}$. Let us denote by $N(\mathcal{H})$ the number of different complete (for each of the variables) orderings of transition constants of $\mathcal{H}$ that are consistent with $Min(\mathcal{F}(\mathcal{H})) \cup External(\mathcal{H})$.

Proposition 1. *Algorithm RefineStateSpace terminates after a finite number of steps and outputs a state space graph G with at most $n(\mathcal{H}) \times N(\mathcal{H})$ vertices.*

Proof. For each vertex of initial graph $\alpha \in V(G_0)$ consider set $S_\alpha = \{\beta \in V(G) | init(\beta) = \alpha\}$. By the design of the algorithm for each pair of different vertices $\beta_1, \beta_2 \in S_\alpha$ constraints $constr(\beta_1)$ and $constr(\beta_2)$ are incompatible. Thus there are at most $N(\mathcal{H})$ vertices in each such set S_α and therefore at most $n(\mathcal{H}) \times N(\mathcal{H})$ vertices in $V(G)$.

The straightforward implementation of each **while** step of the algorithm has $O(N(\mathcal{H})(n(\mathcal{H}) + m(\mathcal{H})))$ time, leading to comparatively high total time complexity $O((N(\mathcal{H})(n(\mathcal{H}) + m(\mathcal{H})))^2)$. However it is not difficult, although technically somewhat involved, to provide implementation with $O(N(\mathcal{H})n(\mathcal{H})(n(\mathcal{H}) + m(\mathcal{H})))$ running time, which compares well with running of state space analysis separately for each of $N(\mathcal{H})$ complete orderings and *analyzing* the whole set of $N(\mathcal{H})$ graphs afterwards.

The way in which algorithm constructs $(\mathcal{F}, CA)$ from initial $(\mathcal{F}_0, CA_0)$ closely resembles the way of defining a constrained frame refinement – essentially the algorithm explicitly constructs the mode and transition mappings m and t. Moreover, when mode α is split into modes α_1 and α_2 each path involving transition $\alpha \to \delta$ is preserved by replacing α either with α_1 or α_2. Thus, if all runs of frame $(\mathcal{F}_0, CA_0)$ are infinite we have $(\mathcal{F}, CA) < (\mathcal{F}_0, CA_0)$. For finite runs however there is a possibility in each step to lose the first mode of the sequence. In most cases this will not be a problem, since all SCCs, apart from the ones consisting of a single mode, will be preserved, however the algorithm can be adjusted to preserve such modes.

Proposition 2. *If there are no finite runs of initial frame $(\mathcal{F}_0, CA_0)$ then for $G(\mathcal{F}, CA)$ computed by algorithm RefineStateSpace the following holds:*

1. *For each complete (for each of the variables) ordering Ord of transition constants and an attractor S in state space graph $G(\mathcal{F}_0, CA_{Ord})$ there is an attractor $\hat{S}$ in $G(\mathcal{F}, CA)$ with the same number of vertices and preserving all paths in S.*
2. *For each attractor S in $G(\mathcal{F}, CA)$ there is a complete ordering Ord with state space graph $G(\mathcal{F}_0, CA_{Ord})$ containing attractor $\hat{S}$ with the same number of vertices and preserving all paths in S.*

Proof. 1. We have already shown that all infinite paths and thus the whole attractor $\hat{S}$ will be preserved in $G(\mathcal{F}, CA)$. Let $\hat{S}$ be one of the attractors with minimal number of modes such that subset $A \subseteq \hat{S}$ is mapped to S. If there exists a non-mapped mode $\alpha \in \hat{S}$ and $\alpha \notin A$ then due to minimality of $\hat{S}$ constraints for all transitions to α must be compatible with Ord, thus we should have $\alpha \in A$.

2. Consider an infinite path P in $\hat{S}$ and constraint $O(P)$. Then for every complete ordering Ord compatible with the $O(P)$ there will be a corresponding attractor S in $G(\mathcal{F}_0, CA_{Ord})$ to which $\hat{S}$ can be mapped. Clearly all modes of S should also have preimages in $\hat{S}$.

Computational experiments

For the toy example in Figure 1 we have number of modes $n(\mathcal{H}) = 8$ and number of different orderings $N(\mathcal{H}) = 6$. By performing analysis of space graphs for all the 6 orderings we find 5 different attractors: A_1 (8 states), A_2 (4 states), A_3, A_4 and A_5 (each with 2 states). The state graphs for 6 possible orderings correspondingly contain the following sets of attractors: $\{A_1\}$, $\{A_2\}$, $\{A_2, A_3\}$ (for two different orderings), $\{A_4\}$ and $\{A_4, A_5\}$.

When we use algorithm *RefineStateSpace* for analysis of this model, we find the same sets of attractors. In principle number of states in refined graph depends on the order in which the pairs of paths is chosen by the algorithm. However for this example the algorithm consistently produced graphs with 40 states (this can be shown to be the smallest possible) in which only attractor A_3 was present in 2 copies (a separate one for each of the orderings allowing A_3).

The constrained HSM frame of our λ-phage model is derived from the earlier model given in FSLM formalism. The model is described in [5] and [13] and has been derived from semi-formal yet very well developed biological model of λ-phage from [11].

The model includes 11 genes: *N, cI, cII, cIII, cro, xis, int, O, P, Q* and an artificial gene *Struc* that stands for all genes producing structural proteins. The activity of these genes is regulated by 10 binding sites, 4 of them each can bind one and other 6 can bind two different proteins. The initial constrained HSM frame $(\mathcal{F}(\mathcal{H}) = \langle M, X, C', T, MF' \rangle, Min(\mathcal{F}(\mathcal{H})) \cup External(\mathcal{H}))$ thus contains $3^6 \times 2^4 = 11664$ modes in M corresponding to all the possible combinations of the binding site states. Each of these modes has between 10 and 16 outgoing transitions in T – there is at least one transition for each binding site representing the change of its state and two outgoing transitions for each site binding two proteins, if this site is unoccupied in this particular mode. The set X contains 11 variables corresponding to the number of genes and there are 32 constants in C' for binding affinities (see [5, 13]). Mode-function assignments MF' are derived from FSLM model by replacing concrete linear growth/degradation functions by values from $\{\nearrow, \searrow, \rightarrow\}$.

Constraint assignments $Min(\mathcal{F}(\mathcal{H}))$ are defined by $\mathcal{F}(\mathcal{H})$ and there are 42 orderings of binding affinities that are consistent with known biological facts that define $External(\mathcal{H})$ (in the most compact form these biological constraints are described in [6]). Finally, from the frame are removed transitions for which constraints assigned by $External(\mathcal{H})$ and $Min(\mathcal{F}(\mathcal{H}))$ are incompatible.

Additionally we can remove all modes that are not reachable under any of 42 allowed threshold orderings (e.g. if for some gene G there are two binding sites: b_1 with dissociation constant c_1 and b_2 with association constant c_2, and according to $External(\mathcal{H})$ we should have $c_2 < c_1$, then state in which b_1 is

occupied but b_2 is free is not reachable). This reduces number of modes in $\mathcal{F}(\mathcal{H})$ to 2890 (the removal of these modes does not change the attractors found by algorithm *RefineStateSpace*).

Thus for our λ-phage model we have $n(\mathcal{H}) = 2890$ and $N(\mathcal{H}) = 42$. In [6] we have already shown that for each Ord of 42 orderings consistent with $External(\mathcal{H})$ all the state spaces of $G(\mathcal{F}(\mathcal{H}), CA_{Ord})$ contain only 2 attractors that correspond to *lysis* (12 states) and *lysogeny* (2 states) behaviors of λ-phage (for details about the structure of these attractors see [6]).

When instead of analyzing each consistent ordering Ord separately we applied algorithm *RefineStateSpace* to constrained frame $(\mathcal{F}(\mathcal{H}), Min(\mathcal{F}(\mathcal{H})) \cup External(\mathcal{H})$ it produced graph with 19693 states. In this case $n(\mathcal{H}) \times N(\mathcal{H}) = 121380$, so we needed to analyze around 6 times fewer states compared to individual analysis of space state graphs for all 42 orderings. Also, the total number of attractors was proportionally smaller – on average each of the attractors found was shared by 7 different orderings.

Conclusions and discussion

Whilst it is known that in general cases the method we propose here may not be able to produce non-trivial results (the whole state space can consist of just a single attractor), we have shown that it can be useful for analysis of specific models, including the relatively complex model of λ-phage. The results for the few existing models are encouraging – all the attractors we have found by analysis of qualitative information incorporated in frames describe behaviors that can be achieved by underlying HSM with explicitly defined quantitative parameter values. Moreover an interesting observation is that the same attractor structure can be shared by many different orderings of transition constants, even when the complete state space graphs for these orderings are different.

Regarding future work, firstly there is a range of questions concerning the formalism that we have developed here and algorithms for state space analysis. E.g. what can we say about general mathematical structure of frame refinements? Can we design a good heuristic for *RefineStateSpace* algorithm that will minimize the number of states in constructed frame refinement (or even, can we design an efficient algorithm that always computes a refinement with the minimal number of states)? Here we have not concentrated much on these questions, partially because the underlying mathematical formalism might still be adjusted and it is not clear as yet which parts of it will be really essential for analysis of HSM representing real biological systems.

Secondly, there are questions how our approach can be extended to better answer questions of biological nature. For instance, we have already indicated the objective to include in models additional information about comparative growth rates of growth/degradation functions (there are examples that show that such inclusion may be useful). Another question of biological significance is the identification and study of constraints that are crucial for directing the system's behavior towards a specific attractor – are such constraints consistent

over the whole state space, and can we link such constraints to known biological events within models of real biological systems?

References

1. J. Ahmad, J. Bernot, J. Comet, D. Lime, and O. Roux. Hybrid modelling and dynamical analysis of gene regulatory networks with delays. *Complexus*, 3:231–251, 2007.
2. R. Alur, C. Belta, F. Ivancic, V. Kumar, M. Mintz, G. Pappas, H. Rubin, and J. Schug. Hybrid modeling and simulation of biomolecular networks. In *Proc. of HSCC 2001*, volume 2034 of *LNCS*, pages 19–32, 2001.
3. E. Bartocci, P. Lio, E. Merelli, and N. Paoletti. Multiple verification in complex biological systems: the bone remodelling case study. In *Trans. on Comput. Syst. Biol. XIV*, volume 7625 of *LNBI*, pages 53–76, 2012.
4. G. Batt, R. Ben Salah, and O. Maler. On timed models of gene networks. In *Proc. of FORMATS 2007*, volume 4763 of *LNCS*, pages 38–52, 2007.
5. A. Brazma and T. Schlitt. Reverse engineering of gene regulatory networks: a finite state linear model. *Genome Biol.*, 4:P5:1–31, 2003.
6. R. Brazma, K. Cerans, D. Ruklisa, T. Schlitt, and J. Viksna. HSM – a hybrid system based approach for modelling intracellular networks. *Gene*, 518:70–77, 2013.
7. H. de Jong, J. Gouze, C. Hernandez, M. Page, T. Sari, and J. Geiselmann. Qualitative simulation of genetic regulatory networks using piecewise-linear models. *Bull. Math. Biol.*, 66:301–340, 2004.
8. J. Fromentin, D. Eveillard, and O. Roux. Hybrid modeling of biological networks: mixing temporal and qualitative biological properties. *BMC Syst. Biol.*, 4:79:11, 2010.
9. R. Ghosh and C. Tomlin. Symbolic reachable set computation of piecewise affine hybrid automata and its application to biological modelling: Delta-notch protein signalling. *Syst. Biol.*, 1:170–183, 2004.
10. R. Grosu, G. Batt, F. Fenton, J. Glimm, C. Le Guernic, S. Smolka, and E. Bertozzi. From cardiac cells to genetic regulatory networks. In *Proc. of CAV 2011*, volume 6806 of *LNCS*, pages 396–411, 2011.
11. H. McAdams and L. Shapiro. Circuit simulation of genetic networks. *Science*, 269:650–656, 1995.
12. D. Ruklisa, A. Brazma, and J. Viksna. Reconstruction of gene regulatory networks under the finite state linear model. *Genome Inform.*, 16:225–236, 2005.
13. T. Schlitt and A. Brazma. Modelling in molecular biology: describing transcription regulatory networks at different scales. *Philos. Trans. R. Soc. Lond. B*, 361:483–494, 2006.
14. R. Serra, M. Vilani, A. Barbieri, S. Kaufmfman, and A. Colacci. One the dynamics of random boolean networks subject to noise: attractors, ergodic sets and cell types. *J. Theor. Biol.*, 265:185–193, 2010.
15. H. Siebert and A. Bockmayr. Temporal constraints in the logical analysis of regulatory networks. *Theor. Comput. Sci.*, 391:258–275, 2008.
16. D. Thomas, R. Thieffrry and M. Kaufman. Dynamic behaviour of biological regulatory networks. i. biological role of feedback loops and practical use of the concept of the loop-characteristic state. *Bulletin of Mathematical Biology*, 57:247–276, 1995.
17. D. Thomas, R. Thieffrry and M. Kaufman. Dynamic behaviour of biological regulatory networks. ii. immunity control in bacteriophage lamda. *Bulletin of Mathematical Biology*, 57:277–297, 1995.