Limit cycle analysis of a class of hybrid gene regulatory networks

Honglu Sun, Maxime Folschette, Morgan Magnin

Limit cycle analysis of a class of hybrid gene regulatory networks*

Honglu Sun[1][0000−0002−8265−0984], Maxime Folschette[2][0000−0002−3727−2320], and Morgan Magnin[1][0000−0001−5443−0506]

[1] Nantes Université, École Centrale Nantes, CNRS, LS2N, UMR 6004, F-44000 Nantes, France — *Corresponding author:* honglu.sun@ls2n.fr
[2] Univ. Lille, CNRS, Centrale Lille, UMR 9189 CRIStAL, F-59000 Lille, France

Abstract. Many gene regulatory networks have periodic behavior, for instance the cell cycle or the circadian clock. Therefore, the study of formal methods to analyze limit cycles in mathematical models of gene regulatory networks is of interest. In this work, we study a pre-existing hybrid modeling framework (HGRN) which extends René Thomas' widespread discrete modeling. We propose a new formal method to find all limit cycles that are simple and deterministic, and analyze their stability, that is, the ability of the model to converge back to the cycle after a small perturbation. Up to now, only limit cycles in two dimensions (with two genes) have been studied; our work fills this gap by proposing a generic approach applicable in higher dimensions. For this, the hybrid states are abstracted to consider only their borders, in order to enumerate all simple abstract cycles containing possible concrete trajectories. Then, a Poincaré map is used, based on the notion of transition matrix of the concrete continuous dynamics inside these abstract paths. We successfully applied this method on existing models: three HGRNs of negative feedback loops with 3 components, and a HGRN of the cell cycle with 5 components.

Keywords: Hybrid modeling · Celerity · Transition matrix · Limit cycle · Gene regulatory networks · Poincaré map.

1 Introduction

Using mathematical models to study the dynamics of gene regulatory networks is fundamental because of the complex nature of biological systems. Two widely used formalisms are discrete models (like Boolean networks [29]) and continuous models (differential equations [5], stochoastic models [28]). The dynamics of discrete models are easy to analyze but sometimes not precise enough (for example, it is hard to identify damped oscillation in discrete models). Continuous models are more precise but their dynamics are sometimes hard to analyze. To make a bridge between discrete and continuous models, hybrid models were proposed [34,13,14]: they can be seen as a simplification of the continuous models or an

* Supported by China Scholarship Council.

extension of the discrete model. These hybrid models contain both continuous and discrete components.

In this work, we study a class of hybrid models: *hybrid gene regulatory networks* (HGRN) [6,15] which is an extension of Thomas' discrete modeling framework [35,36]. HGRNs have been used to model the circadian clock [15] and the cell cycle [6]. In HGRNs, the state space is separated into several discrete states, as for discrete models, and in each discrete state, the temporal derivative of the system is described by a constant vector making the system evolve continuously over time, as for differential equations. The most important property of HGRNs is that the sliding mode is allowed, which means that when a trajectory reaches a black wall (a boundary of the discrete state which cannot be crossed by trajectories) it is forced to move along the black wall.

Previous studies of HGRNs mostly focused on parameters identification [7]. Another important aspect is the dynamical analysis of HGRNs, such as the location of the attractors and their nature (fixed point, limit cycle, etc.). Dynamical properties can be used for model verification and for the discovery of new possible biological behaviors. For now, few results about analysis of HGRNs have been published. We can mainly cite [15] which discussed necessary and sufficient conditions of the existence of a limit cycle of a HGRN in two dimensions, that is, containing two genes. In higher dimensions, limit cycles of HGRNs are more complex. However, no result about the analysis of limit cycles of HGRNs in N dimensions has been published yet, although many genetic oscillators contain several genes. In this work, we seek to fill this gap by studying limit cycles of HGRNs in N dimensions. The main contribution of this work is a new formal method to find all simple limit cycles that do not visit several times the same discrete state in one loop, in a HGRN of N dimensions, and to analyze their stability. The limitations are: we do not consider trajectories that reach several borders simultaneously, which is a very particular case, and we do not consider trajectories containing states which can potentially reach several discrete states and would introduce non-determinism.

The main idea of this new method is based on the notion of *Poincaré map*. The Poincaré map was initially proposed to study periodic orbits of nonlinear dynamical system and has also been used later to study limit cycles of hybrid systems [8,20]. The Poincaré map describes the intersection of a periodic orbit of the system with a lower dimension subspace which is called the Poincaré section. In other words, the Poincaré map allows to witness the shift made by an oscillatory trajectory in a chosen hyperplane of the state space. Thus, by using a Poincaré map, the study of the limit cycle in the original system is transformed into the study of the related fixed point in another system in lower dimensions. One major problem of the application of the Poincaré map to study limit cycles of hybrid systems is that the computation of the Poincaré map can be difficult and the shape of a Poincaré map can be complex, making it hard to analyze. In HGRNs, the shape of the Poincaré map is a simple affine map, but its calculation is still complex because of the existence of sliding modes (two different trajectories that cross the same sequence of discrete states have

different Poincaré maps if they have different sliding modes). It is the major difficulty of this method, and to deal with this problem, a new abstraction based on the new concept of *discrete domain* is proposed. Relying on such discrete domains, we also (re)define the notions of *discrete trajectory*, *transition matrix* and *compatible zone* to calculate the Poincaré map. After the Poincaré map is obtained, the fixed point of the Poincaré map is computed to find the limit cycle and an eigenanalysis is applied to analyze the stability of the limit cycle found.

Most of the works about dynamical analyses of hybrid systems focus on reachability analyses [16]. Among these analysis methods, our method is most similar to the discrete abstraction method [2,1] of which the main purpose is to obtain a finite state transition system from a hybrid automaton. The study of periodic orbits in hybrid system is also a lively field: we can cite, for example, works [11,33,12] based on the Poincaré-Bendixson theorem for systems in two dimensions, and works [26,27,37,21,23] based on the Poincaré map for hybrid systems in N dimensions.

Even though few works exist about limit cycles analysis in HGRNs, limit cycles were studied in other hybrid models of gene regulatory networks. Most of these works are also based on the Poincaré map. In [8,20], the Poincaré map is used to study the limit cycle of simple piecewise affine systems in two dimensions. In these works, since the system is planar, it is easy to compute and analyze the Poincaré map. In [30,17,18], methods are proposed to find and analyze limit cycles in higher dimensions of piecewise affine system with a uniform decay rate. The hypothesis of a uniform decay rate in these works makes it always possible to calculate a Poincaré map because they have a simple shape. However, for a general piecewise affine system, it is difficult to prove theoretically the existence of limit cycle except for some particular examples such as negative loops [19,10].

Compared to previous works about limit cycles in hybrid system, our work has two major novelties: (1) We consider limit cycles with sliding modes, and (2) We use an abstraction method in order to find cycles of discrete regions, which might contain limit cycles (in other words, cycles of discrete regions which contain at least one continuous trajectory).

It is worth mentioning that our work is similar to [30,17,18]. These works use similar methods based on the Poincaré map but apply it to a different kind of hybrid framework, in which temporal derivatives are affine functions in each discrete region, while in HGRNs they are constants. Following these works, [24] proposes a framework that is close to ours, in which temporal derivatives are constants in each discrete region, but is not focused on enumerating limit cycles. Our work also uses a similar approach than [7] to compute trajectories based on constraints, which constitutes one of the steps our method.

There are other works which are based on piecewise constant derivative systems, which are similar to HGRNs, but study different problems, for example, the decidability of the reachability problem [3,4], the stability of fixed point [32].

This paper is organized as follows. In Section 2, we define HGRNs. In Section 3, we use a simple example to describe our method to find and analyze limit cycles of HGRNs. In Section 4, we apply the method on three HGRNs

of negative feedback loops in 3 dimensions and a HGRN of the cell cycle in 5 dimensions. And finally in Section 5, we make a conclusion by discussing the merits and limit of this method and our future work.

2 Hybrid gene regulatory networks

This section defines Hybrid Gene Regulatory Networks (HGRNs). Compared to the original paper regarding HGRNs [6], we introduce here some new notions about HGRNs, including (closed) trajectory and (input/output/attractive/neutral) boundary.

Consider a gene regulatory network with N genes, the i^{th} gene has $n_i + 1$ discrete levels which are represented by integers: $\{0, 1, 2, ..., n_i\}$. A discrete state s is obtained by attributing a valuation for each gene among its discrete levels. We denote d_s the integer vector which describes the discrete levels of all genes in s in order; in the following, for simplicity, we also call d_s a discrete state. The set of all discrete states is $E_d = \left\{d_s \in \mathbb{N}^N \mid \forall i \in \{1, 2, ..., N\}, d_s^i \in \{0, 1, ..., n_i\}\right\}$, where d_s^i is the i^{th} component of d_s.

Based on the notion of discrete state, HGRNs are defined as follows:

Definition 1 (Hybrid gene regulatory network (HGRN)). *A* hybrid gene regulatory network *(HGRN) is noted $\mathcal{H} = (E_d, c)$. E_d is the set of all discrete states. c is a function from E_d to $\mathbb{R}^N$. For each $d_s \in E_d$, $c(s)$, also noted c_s, is called the* celerity *of discrete state d_s and describes the temporal derivative of the system in d_s.*

In HGRNs, a hybrid state is used to fully describe the state of the system: it contains the discrete state in which the system currently is, and a fractional part that represents the (normalized) position of each variable inside this discrete state.

Definition 2 (Hybrid state of HGRN). *A* hybrid state *of a HGRN is a couple $h = (\pi, d_s)$ containing a* fractional part *π, which is a real vector $[0, 1]^N$, and a discrete state d_s in E_d. E_h is the set of all hybrid states.*

In the following, we will use simply *state* to denote a hybrid state. Based on this notion of state, a trajectory and a boundary are defined as follow.

Definition 3 (Trajectory). *A trajectory τ is a function from a time interval $[0, t_0]$ to $E_\tau = E_h \cup E_{sh}$, where $t_0 \in \mathbb{R}^+ \cup \{\infty\}$, E_h is the set of all states, and E_{sh} is the set of all finite sequences of states ($E_{sh} = \left\{(h_0, h_1, ..., h_m) \in (E_h)^{m+1} \mid m \in \mathbb{N}\right\}$).*

A trajectory represents a simulation of the system over time. Consider a trajectory τ on $[0, t_0]$. For any $t \in [0, t_0]$, if $\tau(t) \in E_{sh}$, this means that there is a sequence of instant transitions at t; otherwise, if $\tau(t) \in E_h$, then the trajectory in t is made of a regular point. A trajectory τ defined on $[0, \infty[$ is called a *closed trajectory* if $\exists T > 0, \forall t \in [0, \infty[, \tau(t) = \tau(t + T)$. The smallest T is the period of τ.

Definition 4 (Boundary). *A boundary in a discrete state d_s is a set of states defined by $e(i, \pi_0, d_s) = \{(\pi, d_s) \in E_h \mid \pi^i = \pi_0, \}$, where $i \in \{1, 2, ..., N\}, d_s \in E_d$ and $\pi_0 \in \{0, 1\}$. The boundary $e(i, \pi_0, d_s)$ is inside the discrete state d_s. In the rest of this paper, we simply use e to represent a boundary.*

A toy example of HGRN, not based on any real-world biological system, is shown in Fig 1. This example is related to a negative feedback loop with two genes: A (first dimension) and B (second dimension), where A activates B and B inhibits A. Each gene has two discrete levels, so there are four discrete states in this system. In the right part of the figure representing the model's dynamics, black arrows represent the celerities of each discrete state and red arrows represent a possible trajectory of this system, which happens to be a closed trajectory.

The state $h_M = ((\pi^1_M, 1)^T, (1, 1)^T)$ of point M belongs to the upper boundary e_1 in the second dimension of the discrete state $(1, 1)^T$. Since there is no other discrete state on the other side of e_1, the trajectory from h_M cannot cross e_1 and has to slide along e_1. Boundaries like e_1, which can be reached by trajectories but cannot be crossed, are defined as *attractive boundaries*. If there was another discrete state on the other side of e_1, in which the celerity is negative in the second dimension (towards the boundary), then the trajectory from h_M could still not cross it, and in this case e_1 would also be an attractive boundary.

The state $h_P = ((\pi^1_P, 0)^T, (0, 1)^T)$ of point P belongs to the lower boundary e_2 in the second dimension of the discrete state $(0, 1)^T$. The trajectory from h_P will reach instantly $h_Q = ((\pi^1_Q, 1)^T, (0, 0)^T)$, which belongs to the upper boundary e_3 in the second dimension of $(0, 0)^T$, because the celerities on both sides allow this (instant) discrete transition. e_2 is defined as an *output boundary* of $(0, 1)^T$ and e_3 is defined as an *input boundary* of $(0, 0)^T$.

When a trajectory reaches several output boundaries at same time, a priori, it is non-deterministic because it can cross any of them. In this work, Constraint 1 is proposed to make HGRNs deterministic in any state. This is convenient for simulation purposes.

Formal details about the simulation of general HGRNs are presented as follows. Consider a state $h = (\pi, d_s)$ and a trajectory τ which reaches h at $t > 0$.

- If h does not belong to any boundary, then $\frac{d\tau(t)}{dt} = c_s$ (the temporal derivative of a hybrid state $h = (\pi, d_s)$ is defined as $\frac{dh}{dt} = \frac{d\pi}{dt}$).
- If h only belongs to one boundary e, let us consider that e is the upper boundary in i^{th} dimension (the result is easily adapted when e is the lower boundary). In case d^i_s is not the maximal discrete level of i^{th} gene, the discrete state on the other side of e is noted as d_r, where $d^k_s = d^k_r$ for all $k \neq i$, and $d^i_s + 1 = d^i_r$. There are four possible cases:
  - If $c^i_s < 0$, then the trajectory from the current state will enter the interior of the current discrete state. e called an *input boundary* of d_s. $\frac{d\tau(t^+)}{dt} = c_s$, $\frac{d\tau(t^-)}{dt} = c_r$ and $\tau(t) = ((\pi', d_r), (\pi, d_s))$, where $\pi'^k = \pi^k$ for all $k \neq i$, and $\pi'^i = 0$, which means that there is an instant transition from (π', d_r) to (π, d_s) at t.

- If $c_s^i = 0$, then the trajectory from the current state will slide along the boundary e, which is then called a *neutral boundary* of d_s. $\frac{d\tau(t^+)}{dt} = \frac{d\tau(t^-)}{dt} = c_s$ and $\tau(t) = (\pi, d_s)$.
- If $c_s^i > 0$, and either d_s^i is the maximal discrete level of the i^{th} gene, or d_s^i is not the maximal discrete level of the i^{th} gene but the i^{th} component of c_r is negative, then the trajectory from the current state will slide along the boundary e, which is called an *attractive boundary* of d_s. If τ reaches e at t, then: $\frac{d\tau(t^+)}{dt}^k = c_s^k$ for all $k \neq i$, $\frac{d\tau(t^+)}{dt}^i = 0$, $\frac{d\tau(t^-)}{dt} = c_s$ and $\tau(t) = (\pi, d_s)$. If τ reaches e at $t_0 < t$, then: $\frac{d\tau(t)}{dt}^k = c_s^k$ for all $k \neq i$, $\frac{d\tau(t)}{dt}^i = 0$, and $\tau(t) = (\pi, d_s)$.
- If $c_s^i > 0$, d_s^i is not the maximal discrete level of the i^{th} gene, and the i^{th} component of c_r is positive, then the trajectory from the current state will cross instantly the boundary e and enter the discrete state d_r. e is called an *output boundary* of d_s. $\frac{d\tau(t^+)}{dt} = c_r$, $\frac{d\tau(t^-)}{dt} = c_s$ and $\tau(t) = ((\pi, d_s), (\pi', d_r))$, where $\pi'^k = \pi^k$ for all if $k \neq i$, and $\pi'^i = 0$.

– If h belongs to several boundaries, then the previous cases can be mixed:
- If in these boundaries there is no output boundary, then the trajectory from the current state will exit all input boundaries and slide along all attractive or neutral boundaries.
- If in these boundaries there is only one output boundary, then the trajectory from the current state will cross this output boundary.
- If in these boundaries there are several output boundaries, then the trajectory from the current state will cross one of them following Constraint 1.

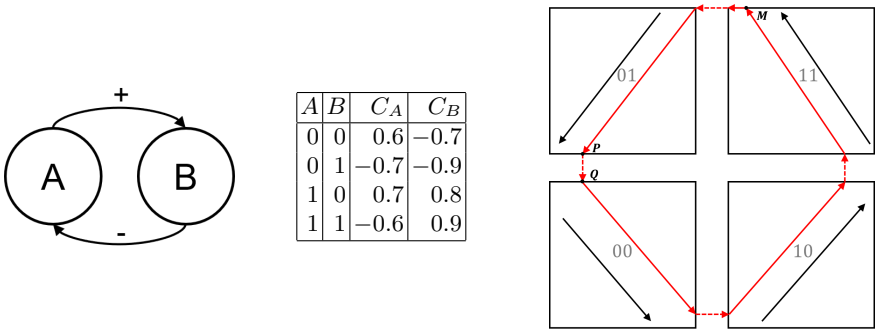
Constraint 1 *If a state of an HGRN is on several output boundaries of dimensions $dim_1, dim_2, ..., dim_m$, where dim_i is the gene number, such that $dim_1 < dim_2 < ... < dim_m$, then from this state the trajectory will only cross the output boundary of dimension dim_1 (the dimension of lowest value).*

An attractive boundary can also be considered as a black wall which is a boundary that attracts neighbor trajectories and cannot be crossed. In general hybrid systems, the behavior on black wall is not easy to define because the derivatives might be different on the different sides of a black wall. In HGRNs, by using hybrid states, a black wall is separated into two boundaries, therefore the system can have different derivatives on the different sides of the wall. There exist other methods to define behaviors of the system on a black wall [25,31].

3 Limit cycle analysis

This section presents new methods to find closed trajectories (potential limit cycles) (Section 3.1) and to analyze their stability (Section 3.2).

In this paper, we make two assumptions about limit cycles in HGRNs.



A	B	C_A	C_B
0	0	0.6	−0.7
0	1	−0.7	−0.9
1	0	0.7	0.8
1	1	−0.6	0.9

Fig. 1: Example of HGRN in 2 dimensions. Left: Influence graph (negative feedback loop with 2 genes). Middle: Example of corresponding parameters (celerities). Right: Corresponding example of dynamics; abscissa represents gene A and ordinate represents gene B.

Assumption 1 *Any non instant transition on closed trajectory does not reach more than one new boundary at the same time.*

Assumption 2 *For any instant transition on the closed trajectory (from state h_i to state h_j), there is at most one output boundary to which h_j belongs.*

For Assumption 1, in real-life systems, it is indeed very unlikely for parameters to be that constrained due to measurement noise. Assumption 2 can be satisfied if we assume that a threshold of one gene only influences at most one another gene. A counter example for Assumption 1 is: $((0.3, 0.7)^T, (a, b)^T) \rightarrow ((1, 1)^T, (a, b)^T)$, for any values of a and b, which is a non instant transition reaching two new boundaries at the same time. A counter example for Assumption 2 is: $((1, 1, 1)^T, (a, b, c)^T) \rightarrow ((0, 1, 1)^T, (a + 1, b, c)^T)$, for any values of a, b and c, where the upper boundaries in the second and third dimensions of $(a + 1, b, c)^T$ are output boundaries.

3.1 Identification of closed trajectories

In this section, we describe our method to find closed trajectory using the example in Fig 1. This method has three steps which are described in order.

(1) Abstract the HGRN with discrete domains
First, the HGRN is transformed into a graph of discrete domains. A discrete domain is a new concept proposed in this work which is defined as follows.

Definition 5 (Discrete domain). *A discrete domain $\mathcal{D}(d_s, S_-, S_+)$ is a set of states inside one discrete state d_s, defined by:*

$$\mathcal{D}(d_s, S_-, S_+) = \{(\pi, d_s) \mid \forall i \in \{1, 2, ..., N\}, \pi^i \in \begin{cases} \{1\} & \text{if } i \in S_+ \\ \{0\} & \text{if } i \in S_- \\ ]0, 1[& \text{if } i \notin S_- \cup S_+ \end{cases} \}$$

where S_+ and S_- are power sets of $\{1, 2, ..., N\}$ such that $S_+ \cap S_- = \emptyset$ and $S_+ \cup S_- \neq \emptyset$. In fact, S_+ (S_-) represents the dimensions in which the upper (lower) boundaries are reached by any state $h \in \mathcal{D}(d_s, S_-, S_+)$. In the rest of this paper, we simply use $\mathcal{D}$ to represent a discrete domain when there is no ambiguity.

In the rest of this paper, as a notation, we add exponents to the vector representation of a discrete state to indicate which upper (lower) boundaries are reached. For instance, $(1, 1^+)^T$ denotes the discrete domain inside discrete state $(1, 1)$ where the upper boundary is reached for the second dimension and no boundary is reached for the first dimension, that is: $\mathcal{D}((1,1)^T, \emptyset, \{2\}) = \left\{(\pi, (1,1)^T) \mid \pi^1 \in]0,1[\wedge \pi^2 = 1\right\}$ Actually, the discrete state $(1,1)^T$ contains 8 discrete domains: $(1^-, 1^-)^T$, $(1, 1^-)^T$, $(1^+, 1^-)^T$, $(1^-, 1^+)^T$, $(1, 1^+)^T$, $(1^+, 1^+)^T$, $(1^-, 1)^T$ and $(1^+, 1)^T$.

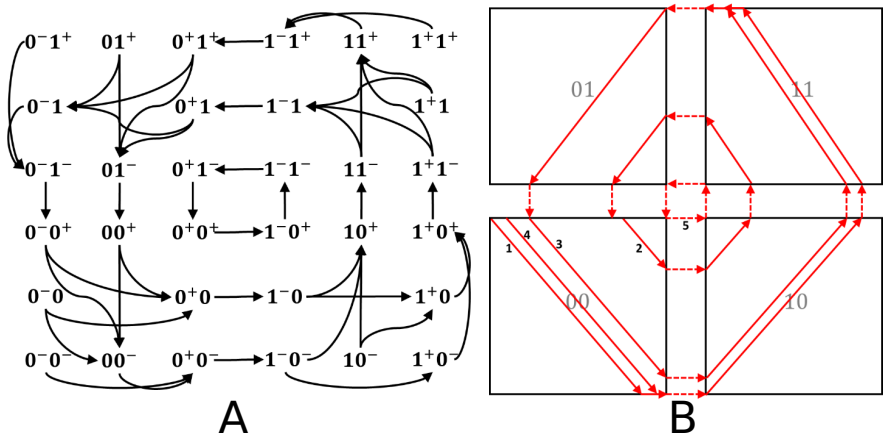


Fig. 2: A: Graph of discrete domains of the HGRN of Fig 1. B: Examples of trajectories inside different discrete trajectories.

It is then possible to build the *graph of discrete domains* of a HGRN, such as in Fig 2 A, where the nodes are the discrete domains and the edges are computed by considering only the signs of the celerities. In this graph, a discrete domain $\mathcal{D}_j$ is a successor of discrete domain $\mathcal{D}_i$ if:

- There are instant transitions from $\mathcal{D}_i$ to $\mathcal{D}_j$, which means that trajectories from $\mathcal{D}_i$ will cross a boundary and instantly reach $\mathcal{D}_j$; see for example $(0^+, 0)^T$ and $(1^-, 0)^T$ in Fig 2 A.
- Only considering the sign of celerities, it is possible that there is a trajectory which begins from $\mathcal{D}_i$ and reaches $\mathcal{D}_j$ without going through another boundary; see for example $(0, 0^+)^T$ and $(0^+, 0)^T$ in Fig 2 A: since the celerity of $(0, 0)^T$ is positive in the first dimension and negative in the second, it is

possible that there is trajectory from $(0, 0^+)^T$ which reaches $(0^+, 0)^T$. We exclude cases where two new boundaries are reached at the same time; for instance, there is no edge between $(0, 0^+)^T$ and $(0^+, 0^-)^T$.

(2) Find the closed discrete trajectories

Based on the graph of discrete domains, we consider a sequence of discrete domains $\mathcal{T} = (\mathcal{D}_0, \mathcal{D}_1, ...\mathcal{D}_p)$ which is a walk in the graph of discrete domains. A trajectory is said to be *inside* such a sequence of discrete domains if it begins from the first discrete domain and reaches by order all discrete domains in the sequence. Based on this, we define two new notions on such a sequence: the transition matrix, which allows to compute the final state of a trajectory inside a given sequence of discrete domains, when it exists, and the compatible zone, which is the set of initial states so that such a trajectory exists.

Definition 6 (Transition matrix). *Consider two different discrete domains $\mathcal{D}_i$ and $\mathcal{D}_j$ such that there exists a sequence of discrete domains $\mathcal{T}$ from $\mathcal{D}_i$ to $\mathcal{D}_j$. If there exists a state $h_i = (\pi_i, d_{s_i})$ in $\mathcal{D}_i$ so that from h_i there is a trajectory τ (defined on $[0, t_0]$) which is inside $\mathcal{T}$ and reaches $\mathcal{D}_j$ on $h_j = (\pi_j, d_{s_j})$ at t_0, then there exists a* transition matrix *M which describes the relation between π_i and π_j, that is: $\pi_j = s^{-1}(Ms(\pi_i))$, where s is a function that adds an extra dimension and the value in the extra dimension is always 1: $s((a_1, a_2, ..., a_N)^T) = (a_1, a_2, ..., a_N, 1)^T$. The transition matrix M only depends on $\mathcal{T}$.*

Considering the HGRN in Fig 1, the transition matrix of $((0, 0^+)^T, (0^+, 0)^T)$ is $\begin{bmatrix} 0 & 0 & 1 \\ -\frac{c^2_{00}}{c^1_{00}} & 1 & \frac{c^2_{00}}{c^1_{00}} \\ 0 & 0 & 1 \end{bmatrix}$ and the transition matrix of $((0^+, 0)^T, (1^-, 0)^T)$ is $\begin{bmatrix} 1 & 0 & -1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$.

Definition 7 (Compatible zone). *Consider a sequence of discrete domains $\mathcal{T} = (\mathcal{D}_0, \mathcal{D}_1, \mathcal{D}_2, ..., \mathcal{D}_m)$. The* compatible zone *$\mathcal{S}$ is the maximal subset of $\mathcal{D}_0$ such that any trajectory starting from $\mathcal{S}$ contains a sub-trajectory that is inside $\mathcal{T}$. More formally, for any state $h \in \mathcal{S}$, if τ is the trajectory defined on $[0, \infty]$ and beginning from h, then there exists t_0 such that the restriction of τ on $[0, t_0]$ is a trajectory inside $\mathcal{T}$.*

The compatible zone $\mathcal{S}$ of a sequence of discrete domains $\mathcal{T} = (\mathcal{D}_0, \mathcal{D}_1, \mathcal{D}_2, ..., \mathcal{D}_m)$ can be expressed with linear inequalities: $\mathcal{S} = \left\{(\pi, d_{s_0})^T \mid (\pi, d_{s_0}) \in \mathcal{D}_0 \wedge A\pi < b\right\}$ where A is a square matrix and b a vector. The idea to calculate compatible zone is based on Theorem 1.

Theorem 1. *A state $h = (\pi, d_{s_0})$ belongs to the compatible zone $\mathcal{S}$ of $\mathcal{T} = (\mathcal{D}_0, \mathcal{D}_1, \mathcal{D}_2, ..., \mathcal{D}_m)$ if and only if $(\pi, d_{s_0}) \in \mathcal{D}_0$, $(s^{-1}(M_{(\mathcal{D}_0, \mathcal{D}_1)} s(\pi)), d_{s_1}) \in \mathcal{D}_1$, $(s^{-1}(M_{(\mathcal{D}_0, \mathcal{D}_1, \mathcal{D}_2)} s(\pi)), d_{s_2}) \in \mathcal{D}_2$, ..., $(s^{-1}(M_{(\mathcal{D}_0, \mathcal{D}_1, ..., \mathcal{D}_{m-1})} s(\pi)), d_{s_{m-1}}) \in \mathcal{D}_{m-1}$ and $(s^{-1}(M_{(\mathcal{D}_0, \mathcal{D}_1, ..., \mathcal{D}_m)} s(\pi)), d_{s_m}) \in \mathcal{D}_m$, where $M_{(\mathcal{D}_0, \mathcal{D}_1, ..., \mathcal{D}_i)}$ is the transition matrix of $(\mathcal{D}_0, \mathcal{D}_1, ..., \mathcal{D}_i)$ and $\mathcal{D}_i$ is inside discrete state d_{s_i} ($i \in \{0, 1, ..., m\}$).*

Proof. Proof of sufficient condition: We can easily see that if $h = (\pi, d_{s_0})$ belongs to the compatible zone $\mathcal{S}$ of $\mathcal{T} = (\mathcal{D}_0, \mathcal{D}_1, \mathcal{D}_2, ..., \mathcal{D}_m)$, then $\forall i \in \{1, 2, ..., m\}$, h also belongs to the compatible zone of $(\mathcal{D}_0, \mathcal{D}_1, \mathcal{D}_2, ..., \mathcal{D}_i)$, so $(s^{-1}(M_{(\mathcal{D}_0, \mathcal{D}_1, ..., \mathcal{D}_i)} s(\pi)), d_{s_i}) \in \mathcal{D}_i$.

Proof of necessary condition: By induction. Consider a sequence of discrete domains of length 2: $(\mathcal{D}_0, \mathcal{D}_1)$, $(s^{-1}(M_{(\mathcal{D}_0, \mathcal{D}_1)} s(\pi)), d_{s_1}) \in \mathcal{D}_1$ means that, when the trajectory from h reaches the new boundary $\mathcal{D}_1$ in all dimensions in which an attractive boundary is not reached and which are not part of the boundaries related to $\mathcal{D}_1$, the fractional parts are all strictly between 0 and 1, so h belongs to the compatible zone of $(\mathcal{D}_0, \mathcal{D}_1)$. Now suppose that it is true for any sequence of discrete domains of length $k + 1$, and consider a sequence of discrete domains of length $k + 2$: $(\mathcal{D}_0, \mathcal{D}_1, \mathcal{D}_2, ..., \mathcal{D}_{k+1})$. Since it is true for a sequence of discrete domains of length $k+1$, h belongs to the compatible zone of $(\mathcal{D}_0, \mathcal{D}_1, \mathcal{D}_2, ..., \mathcal{D}_k)$, so the trajectory from h will stay inside $(\mathcal{D}_0, \mathcal{D}_1, \mathcal{D}_2, ..., \mathcal{D}_k)$ and will reach $\mathcal{D}_k$ at $h_k = (s^{-1}(M_{(\mathcal{D}_0, \mathcal{D}_1, ..., \mathcal{D}_k)} s(\pi)), d_{s_k})$. Let $h_k = (\pi_k, d_{s_k})$. We can easily see that $s^{-1}(M_{(\mathcal{D}_0, \mathcal{D}_1, ..., \mathcal{D}_{k+1})} s(\pi)) = s^{-1}(M_{(\mathcal{D}_k, \mathcal{D}_{k+1})} s(\pi_k))$. So we have $(s^{-1}(M_{(\mathcal{D}_k, \mathcal{D}_{k+1})} s(\pi_k)), d_{s_{k+1}}) \in \mathcal{D}_{k+1}$. Similarly to the case of length 2, h_k belongs to the sable zone of $(\mathcal{D}_k, \mathcal{D}_{k+1})$. Therefore, h belongs to the compatible zone of $(\mathcal{D}_0, \mathcal{D}_1, \mathcal{D}_2, ..., \mathcal{D}_{k+1})$. □

A sequence of discrete domains $\mathcal{T}$ is called a *discrete trajectory* if the compatible zone of $\mathcal{T}$ is not empty. A discrete trajectory $\mathcal{T} = (\mathcal{D}_1, \mathcal{D}_2, ...\mathcal{D}_m)$ is said *closed* if $\mathcal{D}_1 = \mathcal{D}_m$.

In order to find closed discrete trajectories, we use a depth first algorithm. For this, we rely on the notion of *Poincaré section* which, in our case, is a boundary of dimension $N - 1$ that a given closed trajectory always crosses. We first choose one or several input boundaries of discrete states as Poincaré sections by studying the cycles in the transition graph of discrete states, and then on each discrete domain on the Poincaré section, we apply this depth first algorithm. In each step of this depth first algorithm the compatible zone is calculated and the search will continue if the compatible zone is not empty. This algorithm finds all discrete trajectories which begin from a discrete domain and return to the initial discrete state without crossing the same discrete state more than once. An execution of this algorithm on discrete domain $(0, 0^+)^T$ is illustrated in Fig 3. Among these discrete trajectories, we can easily find the closed ones.

Consider the HGRN in Fig 1, we can easily see that there is only one cycle of discrete states in this system, which is:
$(0, 0)^T \to (1, 0)^T \to (1, 1)^T \to (0, 1)^T \to (0, 0)^T$
Therefore, for this system, we only need one Poincaré section and any boundary in this cycle can take this role. Let us choose for instance the input boundary of discrete state $(0, 0)^T$ from $(0, 1)^T$ as Poincaré section, that is, the union of the three discrete domains $(0^-, 0^+)^T$, $(0, 0^+)^T$ and $(0^+, 0^+)^T$. We thus apply the depth first algorithm on each of these three discrete domains. As a result, we can find 5 discrete trajectories which begin from the Poincaré section and returns to the initial discrete state:
$1 : (0^-, 0^+)^T \to (0, 0^-)^T \to (0^+, 0^-)^T \to (1^-, 0^-)^T \to (1, 0^+)^T \to (1, 1^-)^T \to$

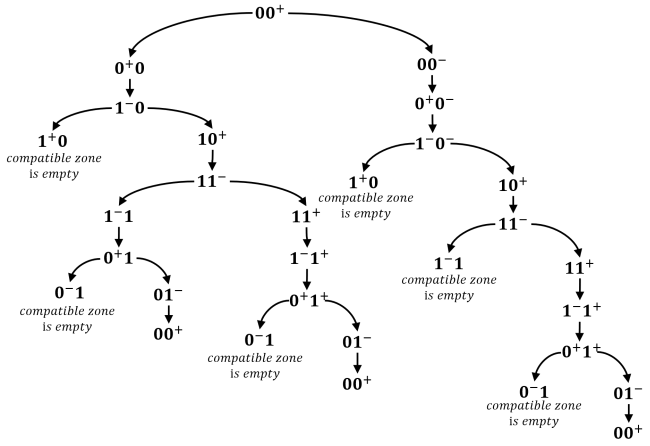


Fig. 3: Illustration of the depth first algorithm on discrete domain $(0, 0^+)^T$

$(1, 1^+)^T \to (1^-, 1^+)^T \to (0^+, 1^+)^T \to (0, 1^-)^T \to (0, 0^+)^T$
$2 : (0, 0^+)^T \to (0^+, 0)^T \to (1^-, 0)^T \to (1, 0^+)^T \to (1, 1^-)^T \to (1^-, 1)^T \to (0^+, 1)^T \to (0, 1^-)^T \to (0, 0^+)^T$
$3 : (0, 0^+)^T \to (0^+, 0)^T \to (1^-, 0)^T \to (1, 0^+)^T \to (1, 1^-)^T \to (1, 1^+)^T \to (1^-, 1^+)^T \to (0^+, 1^+)^T \to (0, 1^-)^T \to (0, 0^+)^T$
$4 : (0, 0^+)^T \to (0, 0^-)^T \to (0^+, 0^-)^T \to (1^-, 0^-)^T \to (1, 0^+)^T \to (1, 1^-)^T \to (1, 1^+)^T \to (1^-, 1^+)^T \to (0^+, 1^+)^T \to (0, 1^-)^T \to (0, 0^+)^T$
$5 : (0^+, 0^+)^T \to (1^-, 0^+)^T \to (1^-, 1^-)^T \to (0^+, 1^-)^T \to (0^+, 0^+)^T$
Examples of trajectories inside each of these 5 discrete trajectories are shown in Fig 2 B. We note that there always exists at least one trajectory inside a discrete trajectory since, by definition, its compatible zone is not empty. Among the 5 discrete trajectories above, only the first one is not closed.

(3) Find a closed trajectory inside each closed discrete trajectory
Consider a closed discrete trajectory $\mathcal{T} = (\mathcal{D}_0, \mathcal{D}_1, ...\mathcal{D}_m, \mathcal{D}_0)$. A *closed trajectory* inside $\mathcal{T}$ is a looping trajectory inside $\mathcal{T}$, that is, a trajectory which begins from a state $h \in \mathcal{D}_0$, reaches by order all discrete domains of $\mathcal{T}$ and finally reaches back state h. To check if there is a closed trajectory inside $\mathcal{T} = (\mathcal{D}_0, \mathcal{D}_1, ...\mathcal{D}_m, \mathcal{D}_0)$, we only need to verify the two following properties:

- $\exists(\pi_0, d_{s_0}) \in \mathcal{D}_0$ such that $s^{-1}(M_{\mathcal{T}} s(\pi_0)) = \pi_0$, and
- (π_0, d_{s_0}) belongs to the compatible zone of $\mathcal{T}$.

Then (π_0, d_{s_0}) is called a fixed point of $\mathcal{T}$.

Under Assumption 1, any closed trajectory which crosses the Poincaré section must be inside one of the closed discrete trajectories found by the depth first algorithm. Meanwhile, if a closed trajectory reaches more than one new boundary at the same time, then it is not inside any closed discrete trajectories found by the algorithm.

Among the five closed discrete trajectories in the HGRN in Fig 1, we find only one closed trajectory of interest, inside the third closed discrete trajectory; it is the trajectory labeled "3" in Fig 2 B. For this closed trajectory, $\pi_0 = (0.222, 1)^T$, $M = \begin{bmatrix} 0 & 0 & 0.222 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix}$, and the compatible zone is $\left\{(\pi, (0,0)^T) \mid \pi^2 = 1, \pi^1 \in]0.1428, 0.3469[\right\}$. Actually, there is another closed trajectory inside the fifth closed discrete trajectory; it is the trajectory labeled "5" in Fig 2 B. This trajectory only contains instant transitions (transition crossing a boundary), so all states in this trajectory are related to the same point in the euclidean space. It could be called a Zeno fixed point. Our analysis method of limit cycles does not consider this type of closed trajectory.

By using the method above, it is possible to find some isolated closed trajectories (closed trajectories which can not be reached or converged to by any trajectory) which are not limit cycles. They can be identified by the analysis method proposed in the next section.

3.2 Stability analysis

Before introducing our stability analysis method of limit cycles in HGRNs, firstly we define the stability of limit cycles in HGRNs.

Definition 8 (Neighborhood in the same discrete state). *The* neighborhood in the same discrete state *of a state* $h = (\pi_0, d_s)^T$ *is a set of states defined as:* $N_d(h, r) = \left\{(\pi, d_s) \mid d(\pi, \pi_0) < r, \pi \in [0,1]^N\right\}$, *with* $r > 0$ *the radius of this neighborhood, and* d *the maximum norm between vectors:* $d(\pi, \pi_0) = \max_{i \in \{1,2,\ldots,N\}} \mid \pi^i - \pi_0^i \mid$.

Definition 9 (Stability of limit cycles in HGRNs). *A limit cycle* $\mathcal{C}_\tau$ *is* stable *if, for any state* h *on* $\mathcal{C}_\tau$*, there exists a neighborhood in the same discrete state of radius* r *such that any trajectory* τ_0 *that begins from this neighborhood* $N_d(h, r)$ *satisfies:* $\lim_{t \to \infty} (Dis_{min}(\tau_0(t), \mathcal{C}_\tau)) = 0$ *where* $Dis_{min}(h', \mathcal{C}_\tau)$ *is defined as* $Dis_{min}(h', \mathcal{C}_\tau) = \min_{h_0 \in \mathcal{C}_\tau} d(x(h'), x(h_0))$*, with* $h' \in E_h$*,* $x(h')$ *the sum (dimension by dimension) of the fractional part and the discrete state of state* h'*, and* d *the maximum norm.*

It is noteworthy that in most cases, a value of t high enough is sufficient to obtain $Dis_{min}(\tau_0(t), \mathcal{C}_\tau) = 0$, without needing a limit computation.

In the following, we call *neighborhood of a trajectory* a union of neighborhood in the same discrete state of all the states in this trajectory. A limit cycle is said to respect the *continuity of neighborhood* if there exists a neighborhood of this cycle that is small enough so that all trajectories starting from this neighborhood remain in this neighborhood. When a limit cycle does not have continuity of neighborhood, some trajectories in the neighborhood may undergo a "disruption" by touching another boundary and thus follow another sequence of discrete states. Without Assumption 1 and Assumption 2, some neighborhoods of

a limit cycle might not respect this continuity, no matter how small they are. For example, consider a limit cycle that contains a state $((1,1)^T,(a,b)^T)$, for given values of a and b, where the upper boundaries in the first and second dimensions are both output boundaries. According to Constraint 1, the trajectory from this state crosses the boundary in the first dimension at first. However, in the neighborhood of $((1,1)^T,(a,b)^T)$, no matter how small it is, we can always find a state which reaches the boundary in the second dimension at first, and as it will reach a different discrete state, it might never return to the neighborhood of the limit cycle. We claim that Assumption 1 and Assumption 2 together are sufficient conditions for the continuity of neighborhood of any limit cycle in HGRNs, although we do not show a proof of this. In the following, the continuity of neighborhood is thus assumed for any limit cycle.

Now we present the method to analyze the stability of limit cycle. Consider a closed trajectory τ inside the closed discrete trajectory $\mathcal{T} = (\mathcal{D}_1, \mathcal{D}_2, ...\mathcal{D}_m, \mathcal{D}_1)$. τ begins from $h = (\pi, d_{s_1}) \in \mathcal{D}_1$. By definition of a closed trajectory, we have:

$$\pi = s^{-1}(M_{\mathcal{T}} s(\pi)) \tag{1}$$

For π, there might be some dimensions in which the values are 0 or 1 because in these dimensions the upper or lower boundaries are reached. If we only consider the dimensions in which the boundaries are not reached, Equation 1 becomes:

$$x = Ax + b \tag{2}$$

where x is a reduction of π which only contains the dimensions in which the boundaries are not reached. The matrix A is called the reduction matrix of $\mathcal{T}$ and vector b is called the constant vector of $\mathcal{T}$.

The stability analysis method of the limit cycle is based Theorem 2.

Theorem 2. *Consider a limit cycle τ inside the closed discrete trajectory $\mathcal{T} = (\mathcal{D}_1, \mathcal{D}_2, ...\mathcal{D}_m, \mathcal{D}_1)$, and $\lambda_1, \lambda_2, ..., \lambda_p$ the eigenvalues of the reduction matrix A of $\mathcal{T}$. If $\max_{i\in\{1,2,...,p\}} \mid \lambda_i \mid < 1$ then τ is stable, otherwise τ is not stable.*

Proof. For this proof, we define the *neighborhood in the same discrete domain* of a state $h = (\pi_0, d_{s_0})^T$ as the set of states: $N_{\mathcal{D}}(h, r) = \{(\pi, d_{s_0}) \mid d(\pi, \pi_0) < r \wedge (\pi, d_{s_0}) \in \mathcal{D}_0\}$, where $\mathcal{D}_0$ is the discrete domain which includes h.

Consider a closed trajectory $\mathcal{C}_\tau$ that exists inside a closed discrete trajectory $\mathcal{T}$. The intersection of $\mathcal{C}_\tau$ with the Poincaré section e is $h_0 = (\pi_0, d_{s_0})$. The Poincaré map in the compatible zone of $\mathcal{T}$ is noted as $x^{k+1} = Ax^k + b$, where x is the reduction of the fractional part considering only the dimensions in which the boundaries are not reached (the reduction of π_0 is x_0). The stability of the fixed point(s) of the system $x^{k+1} = Ax^k + b$ depends on the eigenvalues of A.

If the absolute values of all eigenvalues of A are less than 1, then x_0 is asymptotically stable for the system $x^{k+1} = Ax^k + b$. And since the neighborhood of $\mathcal{C}_\tau$ is continuous, we can find a neighborhood in the same discrete domain of h_0: $N_{\mathcal{D}}(h_0, r_0)$, such that any trajectory τ from $N_{\mathcal{D}}(h_0, r_0)$ stays inside the

neighborhood of $\mathcal{C}_\tau$ and converges asymptotically to or reaches $\mathcal{C}_\tau$. Also, based on the fact that the neighborhood of $\mathcal{C}_\tau$ is continuous, for any state h' on $\mathcal{C}_\tau$, we can find a neighborhood in the same discrete state of h': $N_d(h', r)$, such that any trajectory from $N_d(h', r)$ reaches $N_{\mathcal{D}}(h_0, r_0)$. Thus, for any trajectory τ from $N_d(h', r)$, we have: $\lim_{t\to\infty} Dis_{min}(\tau(t), \mathcal{C}_\tau) = 0$, which proves that $\mathcal{C}_\tau$ is a stable limit cycle.

If the maximum absolute value of all eigenvalues of A equals to or is greater than 1, then x_0 is marginally stable or unstable for system $x^{k+1} = Ax^k + b$; in both cases we cannot guarantee that any trajectory from a small neighborhood in the same discrete domain of h_0 converges to or reaches $\mathcal{C}_\tau$. Therefore, $\mathcal{C}_\tau$ is not stable. □

For the HGRN in Fig 1, the reduction matrix of the third closed discrete trajectory is $\begin{bmatrix}0\end{bmatrix}$, so the closed trajectory inside this closed discrete trajectory is a stable limit cycle. Consider for example the fourth trajectory in Fig 2 B which is a trajectory from the neighborhood of the limit cycle: we can see it finally reaches the limit cycle. In fact, in this HGRN the basin of attraction of this limit cycle is the set of all states of the system.

In fact, if all eigenvalues of A are equal to 1, then the relevant closed trajectory is an isolated closed trajectory, that is, a closed trajectory that can not be reached or converged to by any trajectory, which is not a limit cycle.

4 Application

In this section, we apply our proposed limit cycle analysis method on three HGRNs of negative feedback loop in 3 dimensions and one HGRN of cell cycle in 5 dimensions. The negative feedback loop in 3 dimensions can be used to describe real biological oscillators, for example the p53 system [22]. The signs of the celerities in these three HGRNs are determined by the influence graph (positive for an activation and negative for an inhibition) and their absolute values of celerities are randomly selected. The parameters of the HGRN in 5 dimensions are generated randomly respecting the constraints in Table 3 of [6]. The influence graphs of both systems can be found in Fig 4. Details about implementation can be found at `https://doi.org/10.5281/zenodo.6524936`.

4.1 HGRNs of negative feedback loop in 3 dimensions

The parameters of these three HGRNs of negative feedback loop in 3 dimensions are shown in Table 1. The signs of celerities in these three models are the same so they have the same graph of discrete states. There is only one cycle of discrete states in each of these systems, which is:
$(1,1,1)^T \to (0,1,1)^T \to (0,1,0)^T \to (0,0,0)^T \to (1,0,0)^T \to (1,0,1)^T \to (1,1,1)^T$
Therefore, for these three models, we choose the input boundary e of $(0,0,0)^T$ in the cycle as the Poincaré section. Simulations depicting the convergence to the

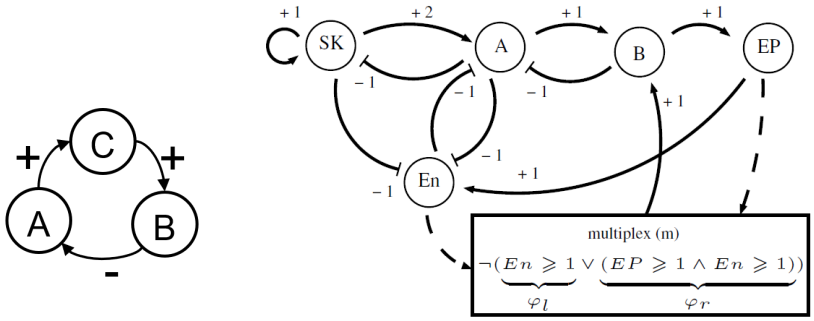


Fig. 4: Left: Influence graph of a negative feedback loop with 3 genes, used to build three models given in Table 1. Right: Influence graph of a cell cycle model with 5 genes from [6]; the multiplex (m) expresses constrains on the joint activation of En and Ep on B.

Table 1: Parameters of the three HGRNs of negative feedback loop in 3 dimensions. Left: First model. Middle: Second model. Right: third model.

A	B	C	C_A	C_B	C_C
0	0	0	1	-0.6	-0.7
0	0	1	1	0.7	-0.9
0	1	0	-0.8	-0.8	-0.7
0	1	1	-0.8	0.6	-0.9
1	0	0	0.7	-0.6	0.6
1	0	1	0.7	0.7	0.5
1	1	0	-0.9	-0.8	0.6
1	1	1	-0.9	0.6	0.5

A	B	C	C_A	C_B	C_C
0	0	0	3	-0.6	-0.7
0	0	1	3	0.7	-2.9
0	1	0	-2.8	-0.8	-0.7
0	1	1	-2.8	0.6	-2.9
1	0	0	2.7	-0.6	2.6
1	0	1	2.7	0.7	0.5
1	1	0	-2.9	-0.8	2.6
1	1	1	-2.9	0.6	0.5

A	B	C	C_A	C_B	C_C
0	0	0	3	-0.6	-0.7
0	0	1	3	0.7	-2.9
0	1	0	-0.8	-0.8	-0.7
0	1	1	-0.8	0.6	-2.9
1	0	0	0.7	-0.6	2.6
1	0	1	0.7	0.7	0.5
1	1	0	-2.9	-0.8	2.6
1	1	1	-2.9	0.6	0.5

stable cycle or to the fixed point (see below) in these three HGRNs are shown in Fig 5.

In the first HGRN, by using our limit cycle analysis method, we find one stable limit cycle and one closed trajectory which only contains instant transitions (that we call a fixed point). Regarding the stable limit cycle, the fixed point of this limit cycle in discrete domain $(0^-, 0^+, 0)^T$ is $((0, 1, 0.125)^T, (0, 0, 0)^T)$, the transition matrix is $\begin{bmatrix} 0\,0\,0 & 0 \\ 0\,0\,0 & 1 \\ 0\,0\,0 & 0.125 \\ 0\,0\,0 & 1 \end{bmatrix}$, the compatible zone is $\left\{(\pi, (0,0,0)^T) \mid \pi^1 = 0, \pi^2 = 1, \pi^3 \in]0, 0.7[\right\}$ and the reduction matrix is $\left[0\right]$, therefore trajectories from the neighborhood of this limit cycle will reach this limit cycle very quickly (less than one turn if the neighborhood is small enough).

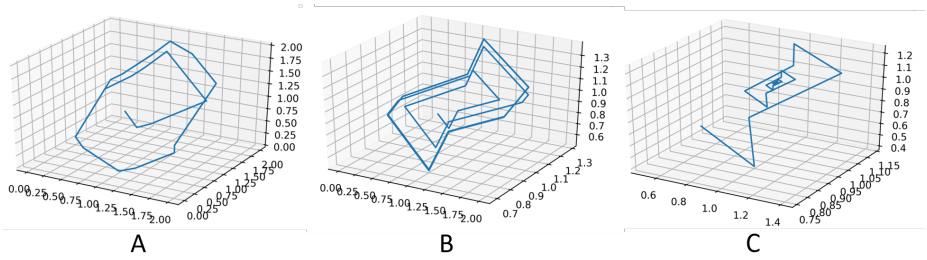


Fig. 5: Illustration of stable limit cycles and stable fixed point in HGRNs in 3 dimensions. A: Stable limit cycle in the first HGRN. B: Stable limit cycle in the second HGRN. C: Stable fixed point in the third HGRN.

In this HGRN we can also prove that all trajectories will reach this limit cycle except trajectories which can reach the fixed point of the system. All discrete trajectories which begin from the Poincaré section and return to the Poincaré section in this HGRN are shown in Fig 6 A. Since in this HGRN there is only one cycle of discrete states which is also a global attractor, any trajectory from the Poincaré section must return to the Poincaré section and it must begin from the compatible zone or the boundary of the compatible zone of one of the discrete trajectories in Fig 6 A. We see that all discrete trajectories which are not closed will finally reach closed discrete trajectories (31, 32, 33, 9, 10). Discrete trajectories 31, 32 and 33 have the same transition matrix and their reduction matrix is $\begin{bmatrix}0\end{bmatrix}$ so any trajectory from $(0^-, 0^+, 0)^T$ will reach the limit cycle. For the discrete trajectory 10, the two eigenvalues of the reduction matrix are 7.0306 and 0.0368, so trajectories inside discrete trajectory 10 will finally leave the compatible zone and reach $(0^-, 0^+, 0)^T$. From here, we can see that any trajectories from the Poincaré section will reach the limit cycle except the trajectories inside discrete trajectory 9 which are related to a fixed point. As any trajectory in this system will finally reach this Poincaré section, all trajectories will reach this limit cycle except trajectories which can reach the fixed point.

For the second HGRN, by using our method, we can also find one stable limit cycle and one fixed point. Unlike the first HGRN, trajectories from the neighborhood of the limit cycle converge asymptotically to the limit cycle: The limit cycle is inside the discrete trajectory which begins from $(0^-, 0^+, 0)^T$ and the reduction matrix of the limit cycle is $\begin{bmatrix}0.0298\end{bmatrix}$. We can also prove that all trajectories will converge to this limit cycle except trajectories which can reach the fixed point by using the same method as for the first HGRN.

Contrary to the first and the second HGRN, we cannot find a limit cycle in the third HGRN but only a fixed point which is related to the discrete trajectory 2 in Fig 6 C. By analyzing the eigenvalues and the fixed points of discrete trajectories 3 and 4, we can prove that all trajectories in this system will converge to the fixed point.

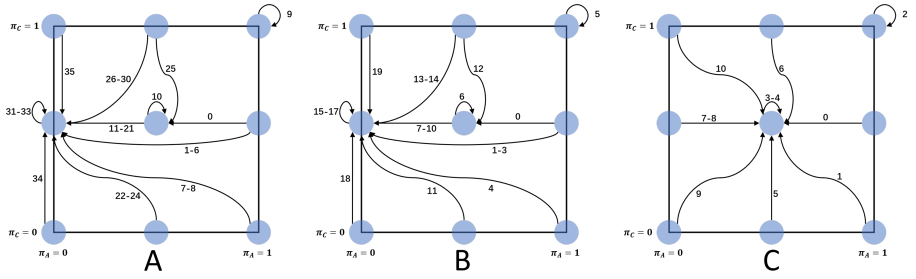


Fig. 6: Abstracted representations of the chosen Poincaré sections in the HGRNs in 3 dimensions, illustrating all possible discrete trajectories which start from and return to this Poincaré section. The blue dots represent the discrete domains and each arrow depicts one or several different discrete trajectories (each following a unique sequence of discrete domains). A: First HGRN. B: Second HGRN. C: Third HGRN.

4.2 HGRN of cell cycle in 5 dimensions

For the HGRN in 5 dimensions, the transition graph of discrete states is more complex. By using a depth first algorithm, we find that there are 1104 cycles of discrete states in which 930 cycles contain the discrete transition $(0,1,0,1,0)^T \to (0,1,0,1,1)^T$, 94 cycles contain $(0,0,0,1,1)^T \to (0,1,0,1,1)^T$ and all the rest contain $(0,0,1,1,0)^T \to (0,0,0,1,0)^T$. Therefore, for this model, we use the three input boundaries crossed by these transition as Poincaré sections, and perform as many analyses. Our method exhibits one stable limit cycle and one unstable limit cycle. The stable one is the same one studied in [6] to calculate the constraints of parameters. The simulations of both cycles are shown in Fig 7 A and B. We need to mention that for now we have not identified any biological behavior related to this unstable limit cycle yet.

For the stable limit cycle of cell cycle model, the fixed point of this limit cycle in the discrete domain $(0,1^+,0,1^+,1^-)^T$ is $((0.3714,1,0.8581,1,0)^T,(0,1,0,1,1)^T)$, and the reduction matrix is $\begin{bmatrix}0&0\\0&0\end{bmatrix}$.

For the unstable limit cycle of cell cycle model, the fixed point of this limit cycle in the discrete domain $(0,0,0^+,1,0^-)^T$ is $((0.6375,0.2552,1,0.3472,0)^T,(0,0,0,1,0)^T)$, and the reduction matrix A is $\begin{bmatrix} 4.95359512\cdot 10^3 & 0 & 1.37489884\cdot 10^{-13}\\ -5.25996267\cdot 10^2 & 0 & -1.45993292\cdot 10^{-14}\\ -7.15619779\cdot 10^2 & 0 & -1.98624389\cdot 10^{-14}\end{bmatrix}$. The eigenvalues of A are 0, $4.95359512\cdot 10^3$ and $3.15544362\cdot 10^{-30}$, making it unstable.

This current naive implementation in Python reaches its limits w.r.t. execution time when the size of the system increases: finding the limit cycles above takes less than one minute for the HGRNs in 3 dimensions, and 8 hours for

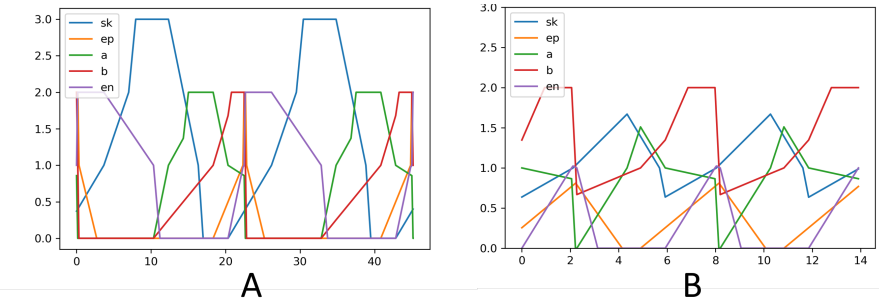


Fig. 7: Simulation of the two limit cycles found in the HGRN of 5 dimensions. A: Stable limit cycle. B: Unstable limit cycle.

the HGRN in 5 dimensions[3]. In future works, we plan to make adjustments to improve the implementation performance.

5 Conclusion

In this work, we proposed a formal method to find all limit cycles of HGRNs with some minor restrictions, mainly to remove non-deterministic behaviors and complex loops, and to analyze their stability. To our knowledge, this method is the first one to find and analyze limit cycles of HGRNs in N dimensions. We showed the merits of this method on random generated HGRNs of a negative feedback loop with 3 components and a HGRN of the cell cycle with 5 components taken from the literature.

As stated above, a first limitation of this method is that we do not handle non-determinism, and we might thus miss some complex closed trajectories, consisting of a composition of several loops using states with a non-deterministic future. Considering closed trajectories inside more complex attractors by assessing non-determinism is thus an interesting continuation. Another limitation in the application of this method is that we first need to construct a HGRN of a specific gene regulatory network; however, the observation of real biological systems is limited and it is not always possible to determine all parameters. In some cases, some parameters can only be described by constraints, or remain unknown. Thus, considering extensions of this method that are parameterized or that take into account a set of constraints on parameters is also of interest.

Finally, in future works, we will also focus on the application of this method on the problem of the control of gene regulatory networks. Similar works have been done with other classes of hybrid models, for instance [8] for the control of oscillations and [9] for the control of bistable switches.

[3] Computations were performed on a standard laptop computer, with an Intel Core I7-8550U 1.80GHz processor and 16.0GB RAM.

Acknowledgements We would like to thank Gilles Bernot and Jean-Paul Comet for their fruitful discussions.

References

1. Alur, R., Dang, T., Ivančić, F.: Counter-example guided predicate abstraction of hybrid systems. In: International Conference on Tools and Algorithms for the Construction and Analysis of Systems. pp. 208–223. Springer (2003)
2. Alur, R., Henzinger, T.A., Lafferriere, G., Pappas, G.J.: Discrete abstractions of hybrid systems. Proceedings of the IEEE **88**(7), 971–984 (2000)
3. Asarin, E., Maler, O., Pnueli, A.: Reachability analysis of dynamical systems having piecewise-constant derivatives. Theoretical computer science **138**(1), 35–65 (1995)
4. Asarin, E., Mysore, V.P., Pnueli, A., Schneider, G.: Low dimensional hybrid systems–decidable, undecidable, don't know. Information and Computation **211**, 138–159 (2012)
5. Barik, D., Baumann, W.T., Paul, M.R., Novak, B., Tyson, J.J.: A model of yeast cell-cycle regulation based on multisite phosphorylation. Molecular systems biology **6**(1), 405 (2010)
6. Behaegel, J., Comet, J.P., Bernot, G., Cornillon, E., Delaunay, F.: A hybrid model of cell cycle in mammals. Journal of bioinformatics and computational biology **14**(01), 1640001 (2016)
7. Behaegel, J., Comet, J.P., Folschette, M.: Constraint identification using modified hoare logic on hybrid models of gene networks. In: 24th International Symposium on Temporal Representation and Reasoning (TIME 2017). Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik (2017)
8. Belgacem, I., Gouzé, J.L., Edwards, R.: Control of negative feedback loops in genetic networks. In: 2020 59th IEEE Conference on Decision and Control (CDC). pp. 5098–5105. IEEE (2020)
9. Chaves, M., Gouzé, J.L.: Exact control of genetic networks in a qualitative framework: the bistable switch example. Automatica **47**(6), 1105–1112 (2011)
10. Chaves, M., Preto, M.: Hierarchy of models: From qualitative to quantitative analysis of circadian rhythms in cyanobacteria. Chaos: An Interdisciplinary Journal of Nonlinear Science **23**(2), 025113 (2013)
11. Clark, W., Bloch, A.: A poincaré–bendixson theorem for hybrid dynamical systems on directed graphs. Mathematics of Control, Signals, and Systems **32**(1), 1–18 (2020)
12. Clark, W., Bloch, A., Colombo, L.: A poincaré-bendixson theorem for hybrid systems. Mathematical Control & Related Fields **10**(1), 27 (2020)
13. Comet, J.P., Bernot, G., Das, A., Diener, F., Massot, C., Cessieux, A.: Simplified models for the mammalian circadian clock. Procedia Computer Science **11**, 127–138 (2012)
14. Comet, J.P., Fromentin, J., Bernot, G., Roux, O.: A formal model for gene regulatory networks with time delays. In: International Conference on Computational Systems-Biology and Bioinformatics. pp. 1–13. Springer (2010)
15. Cornillon, E., Comet, J.P., Bernot, G., Enée, G.: Hybrid gene networks: a new framework and a software environment. advances in Systems and Synthetic Biology (2016)

16. Doyen, L., Frehse, G., Pappas, G.J., Platzer, A.: Verification of hybrid systems. In: Handbook of Model Checking, pp. 1047–1110. Springer (2018)
17. Edwards, R.: Analysis of continuous-time switching networks. Physica D: Nonlinear Phenomena **146**(1-4), 165–199 (2000)
18. Edwards, R., Glass, L.: A calculus for relating the dynamics and structure of complex biological networks. Adventures in Chemical Physics: A Special Volume of Advances in Chemical Physics **132**, 151–178 (2005)
19. Farcot, E., Gouzé, J.L.: Periodic solutions of piecewise affine gene network models with non uniform decay rates: the case of a negative feedback loop. Acta Biotheoretica **57**(4), 429–455 (2009)
20. Firippi, E., Chaves, M.: Topology-induced dynamics in a network of synthetic oscillators with piecewise affine approximation. Chaos: An Interdisciplinary Journal of Nonlinear Science **30**(11), 113128 (2020)
21. Flieller, D., Riedinger, P., Louis, J.P.: Computation and stability of limit cycles in hybrid systems. Nonlinear Analysis: Theory, Methods & Applications **64**(2), 352–367 (2006)
22. Geva-Zatorsky, N., Rosenfeld, N., Itzkovitz, S., Milo, R., Sigal, A., Dekel, E., Yarnitzky, T., Liron, Y., Polak, P., Lahav, G., et al.: Oscillations and variability in the p53 system. Molecular systems biology **2**(1), 2006–0033 (2006)
23. Girard, A.: Computation and stability analysis of limit cycles in piecewise linear hybrid systems. IFAC Proceedings Volumes **36**(6), 181–186 (2003)
24. Glass, L., Edwards, R.: Hybrid models of genetic networks: Mathematical challenges and biological relevance. Journal of theoretical biology **458**, 111–118 (2018)
25. Gouzé, J.L., Sari, T.: A class of piecewise linear differential equations arising in biological models. Dynamical systems **17**(4), 299–316 (2002)
26. Hiskens, I.A.: Stability of hybrid system limit cycles: Application to the compass gait biped robot. In: Proceedings of the 40th IEEE Conference on Decision and Control (Cat. No. 01CH37228). vol. 1, pp. 774–779. IEEE (2001)
27. Hiskens, I.A.: Stability of limit cycles in hybrid systems. In: Proceedings of the 34th Annual Hawaii International Conference on System Sciences. pp. 6–pp. IEEE (2001)
28. Karlebach, G., Shamir, R.: Modelling and analysis of gene regulatory networks. Nature reviews Molecular cell biology **9**(10), 770–780 (2008)
29. Kauffman, S.A.: Metabolic stability and epigenesis in randomly constructed genetic nets. Journal of theoretical biology **22**(3), 437–467 (1969)
30. Mestl, T., Lemay, C., Glass, L.: Chaos in high-dimensional neural and gene networks. Physica D: Nonlinear Phenomena **98**(1), 33–52 (1996)
31. Plahte, E., Kjøglum, S.: Analysis and generic properties of gene regulatory networks with graded response functions. Physica D: Nonlinear Phenomena **201**(1-2), 150–176 (2005)
32. Prabhakar, P., Garcia Soto, M.: Abstraction based model-checking of stability of hybrid systems. In: International Conference on Computer Aided Verification. pp. 280–295. Springer (2013)
33. Simic, S.N., Sastry, S., Johansson, K.H., Lygeros, J.: Hybrid limit cycles and hybrid poincaré-bendixson. IFAC Proceedings Volumes **35**(1), 197–202 (2002)
34. Sriram, K., Bernot, G., Képès, F.: Discrete delay model for the mammalian circadian clock. Complexus **3**(4), 185–199 (2006)
35. Thomas, R.: Boolean formalization of genetic control circuits. Journal of theoretical biology **42**(3), 563–585 (1973)
36. Thomas, R.: Regulatory networks seen as asynchronous automata: a logical description. Journal of theoretical biology **153**(1), 1–23 (1991)

37. Znegui, W., Gritli, H., Belghith, S., et al.: Design of an explicit expression of the poincaré map for the passive dynamic walking of the compass-gait biped model. Chaos, Solitons & Fractals **130**(C) (2020)