Boolean network analysis through the joint use of linear algebra and algebraic geometry

(Article begins on next page)

18 January 2022

Boolean network analysis through the joint use of linear algebra and algebraic geometry

Laura Menini[a], Corrado Possieri[b,*], Antonio Tornambè[c]

[a] *Dipartimento di Ingegneria Industriale,*
Università di Roma Tor Vergata, 00133 Roma, Italy
[b] *Dipartimento di Elettronica e Telecomunicazioni,*
Politecnico di Torino, 10129 Torino, Italy
[c] *Dipartimento di Ingegneria Civile e Ingegneria Informatica,*
Università di Roma Tor Vergata, 00133 Roma, Italy

Abstract

Among the various phenomena that can be modeled by Boolean networks, *i.e.*, discrete-time dynamical systems with binary state variables, gene regulatory interactions are especially well known. Therefore, the analysis of Boolean networks is critical, *e.g.*, to identify genetic pathways and to predict the effects of mutations on the cell functionality. Two methodologies (*i.e.*, the semi-tensor product and the Gröbner bases over finite fields) have recently been proposed to tackle the problem of determining cycles and attractors (with the corresponding basin of attraction) for such systems. Here, it is shown that, by suitably coupling methodologies taken from these two fields (*i.e.*, linear algebra and algebraic geometry), it is not only possible to determine cycles and attractors, but also to find closed-form solutions of the Boolean network. Such a goal is pursued by finding an immersion that recasts the Boolean dynamics in a linear form and by computing the closed-form solution of the latter system. The effectiveness of this technique is demonstrated by fully computing the solutions of the Boolean network modeling the *differentiation of the Th-lymphocyte*, a type of white blood cells involved in the human adaptive immune system.

Keywords: Boolean networks, linear systems, algebraic geometry, linear algebra

1. Introduction

One of the most simple ways to explain the flow of genetic information in a biological system is the *central dogma of molecular biology*, which asserts that

---

*Corresponding author
Email addresses:* `laura.menini@uniroma2.it` (Laura Menini),
`corrado.possieri@polito.it` (Corrado Possieri), `tornambe@disp.uniroma2.it` (Antonio Tornambè)

the information contained in each single gene is transcribed into a sequence of mRNA, which, in turn, is translated into a protein that —possibly interfacing with other proteins — regulates the replication and the transcription of several genes, giving rise to complex interactions (Crick, 1958, 1970). Since the pioneering work of Kauffman (1969), one of the most intuitive, yet effective, models that have been used in the literature (Albert and Barabási, 2000; Bansal et al., 2006; Davidich and Bornholdt, 2008) to represent these complex dynamics is the *Boolean network*, i.e., a discrete-time dynamical system, whose state variables admit only two operational levels: ON (or 1) when active, and OFF (or 0) otherwise. Such systems have recently gained a growing interest thanks to the availability of high-throughput experimental technologies (as, *e.g.*, gene expression micro-arrays) that allow to obtain qualitative and quantitative data about the molecular make-up of cells, which permit to identify gene interactions through simple computational analysis (Covert et al., 2001; Papin et al., 2003; Bansal et al., 2007; Busetto and Lygeros, 2014).

Several tools are available in the literature to represent and analyze the dynamics of Boolean networks as, *e.g.*, linear systems obtained through the semi-tensor product (Cheng and Qi, 2010), logical models (Chaouiya et al., 2012), Boolean functions (Müssel et al., 2010), Petri nets (Rohr et al., 2010), directed graphs obtained through exhaustive enumeration of the states (Menini et al., 2017; Possieri and Teel, 2017), and truth tables (Wuensche, 2011; Shah et al., 2018) (for illustrative purpose, Figure 1 shows several representations of a simple Boolean network representing the dynamics of a 3-repressilator (Dilão, 2014)). Each of these methods presents its own advantages and disadvantages for the representation and the analysis of Boolean dynamics. For instance, the approach based on semi-tensor product has been proved successful to characterize the structural properties of a Boolean network (Cheng et al., 2010; Fornasini and Valcher, 2013), but the dynamical matrices of the corresponding linear system have large dimensions, whereas the approach based on a polynomial representation (Veliz-Cuba et al., 2010; Hinkelmann et al., 2011b; Menini and Tornambe, 2013b) can successfully identify attractors and the corresponding basins of attraction, but the involved tools are rather complex from a computational point of view (Hinkelmann et al., 2011a).

The main objective of this paper is to show that a comprehensive analysis of the dynamics of a Boolean network can be carried out by suitably coupling tools borrowed from algebraic geometry and from linear algebra. Some tools borrowed from algebraic geometry over finite fields (namely, over the Galois field of order 2) allow to easily encode the dynamics of a Boolean network in polynomial form, thus paving the way towards a simplified linear representation of the Boolean dynamics. In fact, it is shown that, by employing standard algebraic geometry tools (which essentially consist of determining a solution to a system of linear equalities modulo 2), it is possible to immerse any Boolean network into a linear, time-invariant, discrete-time system, which will be called the reduced linear representation. The main advantage is that the obtained linear system has, in general, a smaller dimension than the one obtained by using other approaches (as, *e.g.*, the semi-tensor product or the monomial im-
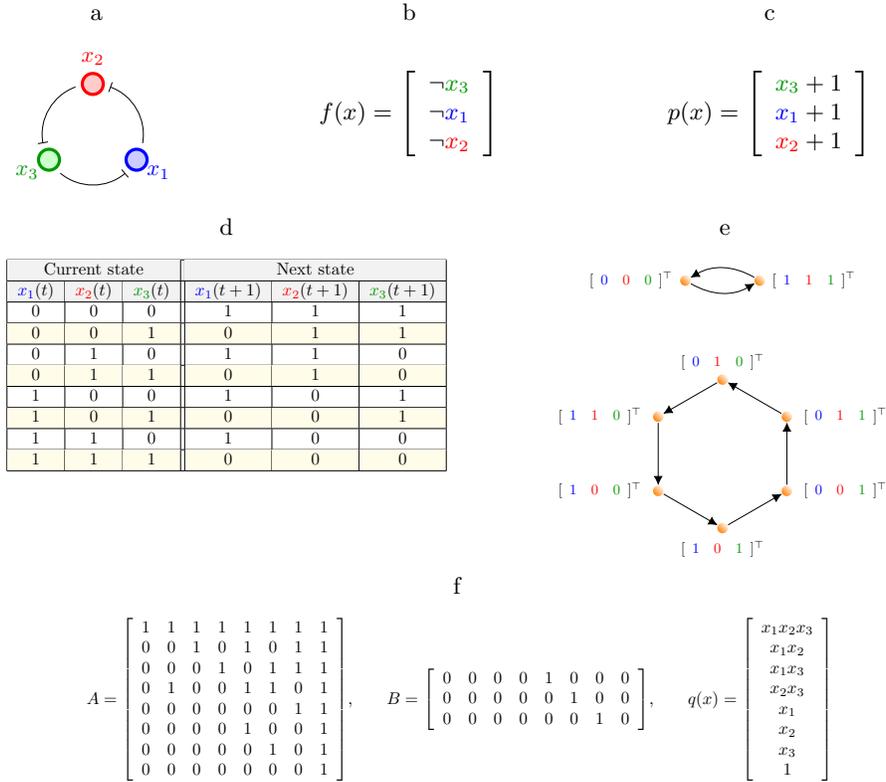
2

a

x_2

x_3 x_1

b

$$f(x) = \begin{bmatrix} \neg x_3 \\ \neg x_1 \\ \neg x_2 \end{bmatrix}$$

c

$$p(x) = \begin{bmatrix} x_3 + 1 \\ x_1 + 1 \\ x_2 + 1 \end{bmatrix}$$

d

Current state			Next state		
$x_1(t)$	$x_2(t)$	$x_3(t)$	$x_1(t+1)$	$x_2(t+1)$	$x_3(t+1)$
0	0	0	1	1	1
0	0	1	0	1	1
0	1	0	1	1	0
0	1	1	0	1	0
1	0	0	1	0	1
1	0	1	0	0	1
1	1	0	1	0	0
1	1	1	0	0	0

e

$[\, 0 \; 0 \; 0 \,]^\top$ $[\, 1 \; 1 \; 1 \,]^\top$

$[\, 0 \; 1 \; 0 \,]^\top$

$[\, 1 \; 1 \; 0 \,]^\top$ $[\, 0 \; 1 \; 1 \,]^\top$

$[\, 1 \; 0 \; 0 \,]^\top$ $[\, 0 \; 0 \; 1 \,]^\top$

$[\, 1 \; 0 \; 1 \,]^\top$

f

$$A = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}, \quad q(x) = \begin{bmatrix} x_1 x_2 x_3 \\ x_1 x_2 \\ x_1 x_3 \\ x_2 x_3 \\ x_1 \\ x_2 \\ x_3 \\ 1 \end{bmatrix}$$

Figure 1: **Several representations of a 3-repressilator**. (**a**) Wiring diagram, where x_i represents a protein and the end symbols $\rightarrow$ and $\dashv$ indicate that the reactions are activating and inhibitory, respectively. (**b**) Logical representation, $x(t+1) = f(x(t))$. (**c**) Polynomial representation, $x(t+1) = p(x(t))$. (**d**) Representation through truth table. (**e**) Extensive graphical representation, where each node represent a state $x \in \mathbb{F}_2^n$ and there is an arc between the node x and the node y if and only if $y = f(x)$. (**f**) Representation through linear immersion, $z(t+1) = Az(t)$, $x(t) = Bz(t)$, $z(0) = q(x(0))$.

mersion(Menini and Tornambe, 2013a), which generically immerse the Boolean network into a linear system with 2^n states), thus enlarging the "size" of the Boolean networks that can be analyzed. Furthermore, the reduced linear representation allows to determine a closed-form expression for the solutions of the Boolean network and to characterize its limit cycles by using standard linear algebra methods.

The effectiveness of this framework is demonstrated by applying the proposed tools to the Boolean network modeling the differentiation of the Th-lymphocyte. Namely, a polynomial representation of the dynamics of such a Boolean network is derived and is used to determine an immersion that recasts it into a compact linear form. Such a representation is then used to characterize the cycles of this Boolean network and to determine a closed-form expression for its solutions.

2. Representation of Boolean networks through polynomials and binary matrices

2.1. Boolean networks

A Boolean network is a discrete-time dynamical system in which each state variable takes value in $\mathbb{F}_2 = \{0,1\}$. In this model, the current state determines uniquely the future evolution of the system since, at each time instant, the value of the next state is determined uniquely by the current value of the state, using a logical rule. Hence, a Boolean network with n state variables can be represented as a discrete-time system of the form

$$x(t+1) = f(x(t)),$$

where $x(t) \in \mathbb{F}_2^n$ is the state vector and $f : \mathbb{F}_2^n \to \mathbb{F}_2^n$ is a logical function assigning to each element of $\mathbb{F}_2^n$ a Boolean vector in $\mathbb{F}_2^n$.

2.2. Representation of Boolean networks through polynomial systems

Among all the models for describing Boolean dynamics, it is worth mentioning the possibility of representing a Boolean network through a polynomial system (Hinkelmann et al., 2011a). In fact, by defining the sum and product operators over $\mathbb{F}_2$ as

$$a + b = \begin{cases} 1, & \text{if } a \neq b, \\ 0, & \text{if } a = b, \end{cases} \qquad ab = \begin{cases} 1, & \text{if } a = 1 \wedge b = 1, \\ 0, & \text{otherwise,} \end{cases} \qquad (1)$$

for each $a, b \in \mathbb{F}_2$, one obtains that the set $\mathbb{F}_2$ has the structure of a finite field (it is usually known as the *Galois field of order 2*). Thus, in view of the usual Boolean relations

$$\neg 0 = 1, \qquad 0 \wedge 0 = 0, \qquad 0 \vee 0 = 0, \qquad 1 \wedge 1 = 1,$$
$$\neg 1 = 0, \qquad 0 \wedge 1 = 1 \wedge 0 = 0, \qquad 1 \vee 0 = 0 \vee 1 = 1, \qquad 1 \vee 1 = 1,$$

it can be noticed that $a \wedge b = ab$, $a \vee b = a + b + ab$, and $\neg a = a + 1$ for each $a, b \in \mathbb{F}_2$. Therefore, each Boolean function $f : \mathbb{F}_2^n \to \mathbb{F}_2^k$ can be represented through a polynomial vector in n variables with coefficients in $\mathbb{F}_2$. In particular, letting $x = [\begin{array}{ccc} x_1 & \cdots & x_n \end{array}]^\top$ be such variables, for each Boolean map $f(x)$, there exists a polynomial vector $p(x)$, with coefficients in $\mathbb{F}_2$, such that $p(x) = f(x)$ for all $x \in \mathbb{F}_2^n$ (in the remainder of the paper, following the notation usually employed when dealing with polynomials(Cox et al., 2015), the set of all the polynomials in x with coefficients in $\mathbb{F}_2$ is denoted $\mathbb{F}_2[x]$). Such a polynomial vector can be easily determined by using the *Lagrange interpolation formula* (Menini and Tornambe, 2013a) (see Figure 2):

$$p(x) = \sum_{w \in \mathbb{F}_2^n} f(w) \prod_{i=1}^{n} (1 + w_i + x_i), \qquad (2)$$

$$f(x_1, x_2) = \begin{bmatrix} x_1 \wedge \neg x_2 \\ \neg x_1 \wedge x_2 \end{bmatrix}$$

Lagrange interpolation formula

$$
\begin{aligned}
p(x_1, x_2) &= \begin{bmatrix} 0 \wedge \neg 0 \\ \neg 0 \wedge 0 \end{bmatrix}(1 + 0 + x_1)(1 + 0 + x_2) + \begin{bmatrix} 0 \wedge \neg 1 \\ \neg 0 \wedge 1 \end{bmatrix}(1 + 0 + x_1)(1 + 1 + x_2) \\
&\quad + \begin{bmatrix} 1 \wedge \neg 0 \\ \neg 1 \wedge 0 \end{bmatrix}(1 + 1 + x_1)(1 + 0 + x_2) + \begin{bmatrix} 1 \wedge \neg 1 \\ \neg 1 \wedge 1 \end{bmatrix}(1 + 1 + x_1)(1 + 1 + x_2) \\
&= \begin{bmatrix} 0 \\ 0 \end{bmatrix}(x_1 x_2 + x_2 + x_1 + 1) + \begin{bmatrix} 0 \\ 1 \end{bmatrix}(x_1 x_2 + x_2) \\
&\quad + \begin{bmatrix} 1 \\ 0 \end{bmatrix}(x_1 x_2 + x_1) + \begin{bmatrix} 0 \\ 0 \end{bmatrix}(x_1 x_2) \\
&= \begin{bmatrix} x_1 x_2 + x_1 \\ x_1 x_2 + x_2 \end{bmatrix}
\end{aligned}
$$

Figure 2: **Lagrange interpolation formula**. Example of application of the expression given in (2) for the logical map $f : \mathbb{F}_2^2 \to \mathbb{F}_2^2$ representing the reduced dynamics of the Th-lymphocyte differentiation network (Veliz-Cuba, 2011), where x_1 and x_2 denote the presence/absence of T-bet and GATA-3, respectively.

where sums and products of the coefficients of the monomials in x appearing in (2) are carried out by using (1). Thus, in view of (2), each $f : \mathbb{F}_2^n \to \mathbb{F}_2^k$ can be represented by the corresponding polynomial vector $p(x)$ with coefficients being either 0 or 1 and whose degree in each variable x_i is lower than 2. In particular, for any polynomial vector $q \in \mathbb{F}_2^k[x]$ define its *canonical form* as follows:

$$q(x)_{\sharp \mathbb{F}_2} := \sum_{w \in \mathbb{F}_2^n} q(w) \prod_{i=1}^{n}(1 + w_i + x_i).$$

In the following, the ring of all such canonical forms is denoted $\overline{\mathbb{F}}_2[x]$. It is worth noticing that $\overline{\mathbb{F}}_2[x]$ has the structure of a *quotient ring* over $\mathbb{F}_2$, $\overline{\mathbb{F}}_2[x] = \mathbb{F}_2[x]/\langle x_1^2 + x_1, \ldots, x_n^2 + x_n \rangle$, since each $g \in \mathbb{F}_2[x]$ that can be written as $g(x) = g_1(x)(x_1^2 + x_1) + \cdots + g_n(x)(x_n^2 + x_n)$ for some $g_1, \ldots, g_n \in \mathbb{F}_2[x]$ vanishes identically over $\mathbb{F}_2$. Thus, in view of the Lagrange interpolation formula (2), each Boolean network can be represented through a discrete-time polynomial system of the form

$$x(t + 1) = p(x(t)), \tag{3}$$

where $p \in \overline{\mathbb{F}}_2^n[x]$. Such a polynomial representation has many advantages. As a matter of fact, it allows one to use all the tools arising from computational algebraic geometry over finite fields (Cox et al., 2015), which are efficiently implemented in several computer algebra systems as, *e.g.*, `Macaulay2` (Grayson
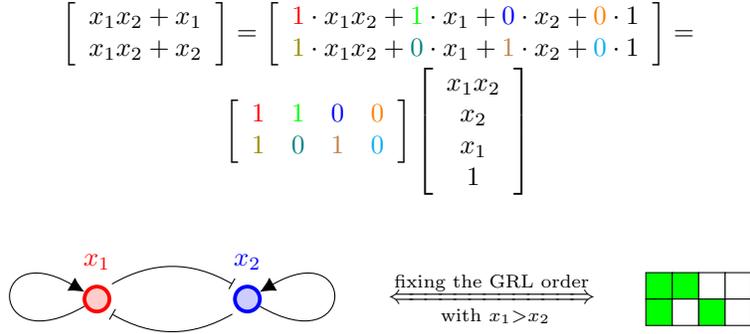
$$\left[\begin{array}{c} x_1x_2 + x_1 \\ x_1x_2 + x_2 \end{array}\right] = \left[\begin{array}{c} 1 \cdot x_1x_2 + 1 \cdot x_1 + 0 \cdot x_2 + 0 \cdot 1 \\ 1 \cdot x_1x_2 + 0 \cdot x_1 + 1 \cdot x_2 + 0 \cdot 1 \end{array}\right] =$$

$$\left[\begin{array}{cccc} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \end{array}\right] \left[\begin{array}{c} x_1x_2 \\ x_2 \\ x_1 \\ 1 \end{array}\right]$$

x_1 x_2 $\xleftrightarrow[\text{with } x_1 > x_2]{\text{fixing the GRL order}}$

Figure 3: **Representation of polynomial vectors through Boolean matrices**. Linear matrix in $\mathbb{F}_2^{2 \times 4}$ (where green and white cells represent entries equal to one and to zero, respectively) that represents the Boolean network modeling the reduced dynamics of the Th-lymphocyte differentiation.

and Stillman, 2018), which is especially suited for computations with polynomial dynamical systems (Menini and Tornambe, 2013). It is worth noticing that a wholly similar approach has been proposed in Kobayashi and Hiraishi (2017); Menini et al. (2017) to find a polynomial representation of the Boolean dynamics and to design a control law for the network using polynomial and integer programming tools.

2.3. Linear representation of Boolean networks

An interesting byproduct of the use of polynomials to represent Boolean dynamics is that such a representation allows to immerse the Boolean network into a linear discrete-time system. In fact, letting $N_n = \sum_{r=0}^{n} \binom{n}{r} = 2^n$ be the number of different monomials in $\overline{\mathbb{F}}_2[x]$ and letting $q(x) = \left[\begin{array}{ccc} q_1(x) & \cdots & q_{N_n}(x) \end{array}\right]^\top$ be the vector of all such monomials, for each $p = \left[\begin{array}{ccc} p_1 & \cdots & p_k \end{array}\right] \in \overline{\mathbb{F}}_2^k[x]$, there exists a matrix $P \in \mathbb{F}_2^{k \times N_n}$ such that

$$p(x)_{\sharp \mathbb{F}_2} = Pq(x). \tag{4}$$

In particular, the (i, j)-th entry of the matrix P is the coefficient of the monomial $q_j(x)$ in the i-th entry $p_i(x)$ of $p(x)$ (see Figure 3). Thus, each Boolean network can be efficiently stored by saving the matrix $P \in \mathbb{F}_2^{n \times N_n}$ corresponding to the function $p \in \mathbb{F}_2^n[x]$, which is composed by $n2^n$ bits, once that an order on the monomials $q_1, \ldots, q_{N_n}$ has been fixed. Moreover, the expression given in (4) allows one to easily obtain a map that immerse the polynomial system (3) into linear form. In fact, by taking the entries of q as new state variables, $z = q(x)$, in the z-coordinates the system exhibits linear dynamics:

$$z(t+1) = Az(t), \qquad\qquad x(t) = Bz(t), \tag{5}$$

$$A = \qquad\qquad B = \qquad\qquad q(x) = \begin{bmatrix} x_1 x_2 \\ x_1 \\ x_2 \\ 1 \end{bmatrix}$$

Figure 4: **Immersion of Boolean dynamics into a linear system**. Immersion of the reduced dynamics of the Th-lymphocyte differentiation network into a linear system, where matrices with entries in $\mathbb{F}_2$ have been depicted as tables in which green and white cells represent entries equal to one and to zero, respectively.

where $A \in \mathbb{F}_2^{N_n \times N_n}$ is such that $(q \circ p(x))_{\sharp \mathbb{F}_2} = Aq(x)$ (namely, the matrix A can be determined by inspection of the coefficients of $(q \circ p(x))_{\sharp \mathbb{F}_2}$) and $B \in \mathbb{F}_2^{n \times N_n}$, $B = (\frac{\partial q(x)}{\partial x})_{\# \mathbb{F}_2}\big|_{x=0}$ (see Figure 4). Note that the number of entries of $z(t)$ is the same as the number of states of the linear immersion obtained by using the semi-tensor product approach (Cheng and Qi, 2010).

The linear representation given in (5) has several desirable properties: first, it allows to determine a closed-form solution of the Boolean network (3) as

$$x(t) = BA^t q(x(0));$$

secondly, it allows to determine cycles by using linear algebra. In fact, $x \in \mathbb{F}_2^n$ belongs to a cycle of length ℓ if and only if

$$(A^\ell + I)q(x) = 0, \qquad (A^j + I)q(x) \neq 0, \ j = 1, \ldots, \ell - 1,$$

i.e., the vector $q(x)$ is an eigenvector relative to the eigenvalue 1 for A^ℓ, but not for A^j, $j = 1, \ldots, \ell - 1$, using $\mathbb{F}_2$ as ground field.

2.4. Reduced linear representation of Boolean networks

Even tough the representation through linear immersion of Boolean networks given in (5) has several advantages (as, *e.g.*, the possibility of computing the closed-form solution of the Boolean network and the availability of efficient tools to determine cycles), it has a major drawback: the size of the state. As a matter of fact, system (5) has 2^n states, whence the dimension of system (5) grows exponentially with the number of Boolean variables involved in the network. In order to overcome this issue, it is possible to use algebraic geometry to obtain a reduced linear representation of the Boolean network. In particular, assume that the interest is just to analyze the dynamical behavior of the output vector $y(t) \in \mathbb{F}_2^m$ which is related to the current state of the Boolean network through a logic function $h \in \overline{\mathbb{F}}_2^m[x]$ (possibly, $m = n$ and $h(x) = x$), *i.e.* $y(t) = h(x(t))$. By the same reasoning given in the previous section, there exists a matrix $C \in \mathbb{F}_2^{m \times N_n}$ such that

$$y = h(x) = Cq(x).$$

Thus, letting $A \in \mathbb{F}_2^{N_n \times N_n}$ be the matrix such that (5) holds, by the Cayley-Hamilton theorem (Meyer, 2000), there exists a monic polynomial $\sigma(s) = s^{N_n} +$

$\lambda_{N_n-1}s^{N_n-1} + \cdots + \lambda_0 \in \mathbb{F}_2[s]$, $\lambda_0, \ldots, \lambda_{N_n-1} \in \mathbb{F}_2$, such that

$$A^{N_n} + \lambda_{N_n-1}A^{N_n-1} + \cdots + \lambda_0 I_{N_n} = 0,$$

where I_{N_n} denotes the N_n-dimensional identity matrix over $\mathbb{F}_2$. In particular, $\sigma(s)$ is the characteristic polynomial of the matrix A using $\mathbb{F}_2$ as ground field,

$$\sigma(s) = \det(sI_{N_n} - A).$$

Note that $\sigma(s)$ is a polynomial in $\mathbb{F}_2[s]$, but $\sigma_{\sharp\mathbb{F}_2}(A)$ need not be equal to zero since A is a matrix in $\mathbb{F}_2^{N_n \times N_n}$ and not a scalar value in $\mathbb{F}_2$. Thus, by pre-multiplying the expression above by C and post-multiplying it by $q(x)$, since $y(t+i) = CA^i q(x(t))$ for all $(t,i) \in \mathbb{Z}_{\geq 0} \times \mathbb{Z}_{\geq 0}$, one obtains that

$$y(t + N_n) = \lambda_{N_n-1}y(t + N_n - 1) + \cdots + \lambda_0 y(t). \qquad (6)$$

However, the number N_n need not be the smallest integer $K \in \mathbb{Z}_{\geq 0}$ such that $y(t + K)$ can be expressed as a linear combination over $\mathbb{F}_2$ of $y(t), y(t + 1), \ldots, y(t + K - 1)$. In order to obtain such an integer K, it is possible to use some tools borrowed from algebraic geometry (Cox et al., 2006). Namely, define the operator $\Delta_p h(x) = (h \circ p(x))_{\sharp\mathbb{F}_2}$ and its iteration $\Delta_p^{k+1}h(x) = \Delta_p \Delta_p^k h(x)$, $k \in \mathbb{Z}_{\geq 0}$. Thus, given $k \in \mathbb{Z}_{\geq 0}$, consider the matrix M_k of $\overline{\mathbb{F}}_2^{m \times (k+1)}[x_1, \ldots, x_n]$ defined as follows:

$$M_k := \begin{bmatrix} \Delta_p^k h(x) & \Delta_p^{k-1}h(x) & \cdots & \Delta_p h(x) & h(x) \end{bmatrix}.$$

By (6), the vector $\begin{bmatrix} 1 & \lambda_{N_n-1} & \cdots & \lambda_0 \end{bmatrix}^\top \in \mathbb{F}_2^{N_n+1}$ is in the syzygy module of M_{N_n} (Cox et al., 2006). Thus, in order to determine whether $y(t + k)$ can be expressed as a linear combination over $\mathbb{F}_2$ of $y(t), y(t + 1), \ldots, y(t + k - 1)$ it suffices to check if there exists a vector $\gamma = \begin{bmatrix} \gamma_{k+1} & \gamma_k & \cdots & \gamma_0 \end{bmatrix} \in \mathbb{F}_2^{k+1}$, with $\gamma_{k+1} = 1$, in the syzygy module of M_k. If such a vector exists, then

$$y(t + k) = \gamma_{k-1}y(t + k - 1) + \cdots + \gamma_0 y(t),$$

otherwise $y(t + k)$ cannot be expressed as a linear combination over $\mathbb{F}_2$ of $y(t), y(t + 1), \ldots, y(t + k - 1)$. Therefore, the integer K can be determined by defining the matrices M_k, $k = 1, 2, \ldots, N_n$, and letting K be the smallest integer such that there exists a constant vector, with the first entry being equal to 1, in the corresponding syzygy module. It is worth noticing that such computations can be carried out by simply equating to 0 modulo 2 all the coefficients of the monomials in x appearing in the vector

$$\Delta_p^k h(x) + \gamma_{k-1}\Delta_p^{k-1}h(x) + \cdots + \gamma_1 \Delta_p h(x) + \gamma_0 h(x),$$

i.e., γ and K can be determined by solving systems of linear equations modulo 2. It is worth stressing that all these computations do not employ the matrices A, B, and C, which may have large dimensions. Thus, once that the integer K has been found, together with the corresponding vector $\gamma = \begin{bmatrix} 1 & \gamma_{K-1} & \cdots & \gamma_0 \end{bmatrix} \in$

$$p(x) = \begin{bmatrix} x_3 + 1 \\ x_1 + 1 \\ x_2 + 1 \end{bmatrix}$$

$$h(x) = x_1$$

$$\begin{aligned}
h(x) &= x_1 \\
\Delta_p h(x) &= x_3 + 1 \\
\Delta_p^2 h(x) &= x_2 + 1 + 1 = x_2 \\
\Delta_p^3 h(x) &= x_1 + 1 \\
\Delta_p^4 h(x) &= x_3 + 1 + 1 = x_3
\end{aligned}$$

c   d

$$\Delta_p^4 h(x) + \underbrace{1}_{\gamma_3} \Delta_p^3 h(x) + \underbrace{0}_{\gamma_2} \Delta_p^2 h(x) + \underbrace{1}_{\gamma_1} \Delta_p^1 h(x) + \underbrace{1}_{\gamma_0} h(x) = 0$$

$$E = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 1 \end{bmatrix}$$

Figure 5: **Reduced linear representations of a 3-repressilator.** (a) Polynomial representation, $x(t+1) = p(x(t))$, $y(t) = h(x(t))$. (b) Directional increments of the output. (c) Constant syzygy of the matrix M_4. (d) Dynamical matrix of the reduced linear representation.

$\mathbb{F}_2^{K+1}$, $\gamma \in \mathrm{syz}(M_K)$, the dynamics of the output of the Boolean network can be represented, through the change of variables

$$\begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_K \end{bmatrix} =: y_e = O_K(x) := \begin{bmatrix} h(x) \\ \Delta_p h(x) \\ \vdots \\ \Delta_p^{K-1} h(x) \end{bmatrix}, \tag{7}$$

by the following linear dynamics

$$y_e(t+1) = E y_e(t), \quad E = \begin{bmatrix} 0_m & I_m & 0_m & \cdots & 0_m \\ 0_m & 0_m & I_m & \cdots & 0_m \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0_m & 0_m & 0_m & \cdots & I_m \\ \gamma_0 I_m & \gamma_1 I_m & \gamma_2 I_m & \cdots & \gamma_{K-1} I_m \end{bmatrix}, \tag{8}$$

where 0_m and I_m denote the m–dimensional zero and identity matrices, respectively. Note that the linear representation given in (8) may be much smaller than the one given in (5) since $y_e(t) \in \mathbb{F}_2^{mK}$, whereas $z(t) \in \mathbb{F}_2^{N_n}$. Furthermore, if the map $O_K : \mathbb{F}_2^n \to \mathbb{F}_2^{Km}$ is injective (as, $e.g.$, if $h(x) = x$), then it configures as a state immersion (Isidori, 2013), $i.e.$, there exists $O_K^{-1} : \mathbb{F}_2^{Km} \to \mathbb{F}_2^n$ such that $O_K^{-1} \circ O_K(x) = x$ for all $x \in \mathbb{F}_2^n$, whence it is possible to recast the state evolution of the Boolean network as

$$x(t) = O_K^{-1}(y_e(t)).$$

For instance, if $m = n$ and $h(x) = x$, it results that

$$O_K^{-1}(y_e) = [\, I_n \quad 0_n \quad \cdots \quad 0_n \,] y_e.$$

Finally, it is worth noticing that the approach given in this section for determining a reduced linear representation of the Boolean dynamics has some similarities with the technique given in Kobayashi et al. (2010) to determine if a Boolean network is controllable. As a matter of fact, in Kobayashi et al. (2010), the adjacency matrix of the Boolean dynamics is used to determine whether there exists a time $T \in \mathbb{Z}_{\geqslant 0}$ such that the output of the network at time T depends just on the inputs at time $0, 1, \ldots, T$. On the other hand, by using different tools, the approach given in this section allows one to determine whether the outputs at time T depends just on the outputs at time $0, 1, \ldots, T-1$.

3. Application

In this section, the tools presented in Section 2 are used to study the dynamics of a Boolean network taken from the literature. Namely, it is shown how the linear representation given in (8) can be exploited to fully characterize cycles and equilibria of Boolean networks without resorting to the linear immersion given in (5). The proposed method seems to be particularly appealing since, differently from other tools given in the literature (as, *e.g.*, the ones given in Veliz-Cuba, 2011), the dynamics of the Boolean network are fully represented by its immersion, without neglecting any transition.

3.1. Th-lymphocyte differentiation

Th-lymphocytes (or Th-cells) are a subtype of white blood cells that play a crucial role in the adaptive immune system. They support the activity of other immune cells by secreting small proteins (called cytokines), which assist and regulate the active immune response. Such cells, when subject to proper stimulations, can differentiate into Th1 or Th2 cells, which enable cell mediated immunity and humoral responses, respectively, by secreting different cytokines. Experimental evidences showed that these cells are strongly related with autoimmune diseases and with allergic reactions (Murphy and Reiner, 2002; Agnello et al., 2003).
The main objective of this subsection is to analyze the Boolean network modeling the Th-lymphocyte differentiation (Mendoza, 2006; Remy et al., 2006) through algebraic geometry and linear algebra tools. Such a Boolean network involves $n = 12$ Boolean state variables which represent presence/absence of regulatory factors (T-bet, GATA-3) signaling transduction factors (STAT1, STAT4, STAT6, SOCS-1), lymphokines (IFN-γ, IL-4, IL-12) and receptors (IFN-γR, IL-4R, IL-12R). Figure 6 depicts the wiring diagram of the Th-lymphocyte differentiation network and the corresponding Boolean network written in the polynomial form given in (3).
Letting $m = n = 12$, $h(x) = x = [\ x_1 \quad x_2 \quad \cdots \quad x_{12}\ ]^\top$, the Boolean vector

$$\gamma = [\ 1 \quad 0 \quad 0 \quad 0 \quad 1 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0\ ]^\top \in \mathbb{F}_2^{13}$$

is in the syzygy module of $M_{12} = [\ \Delta_p^{12} h(x) \quad \Delta_p^{11} h(x) \quad \cdots \quad \Delta_p h(x) \quad h(x)\ ]$. Therefore, the Boolean network modeling the Th-lymphocyte differentiation

10

a

b

$$p(x) = \begin{bmatrix} x_9 x_{11} + x_9 + x_{11} \\ x_7 x_{12} + x_{12} \\ 0 \\ x_1 x_{10} + x_1 \\ x_2 x_{10} + x_2 \\ x_3 x_8 + x_3 \\ x_4 \\ x_5 \\ x_6 x_{12} + x_6 \\ x_7 x_{11} + x_7 + x_{11} \\ (x_7 + (x_7 + 1)x_{11})(x_{12} + 1) \\ x_8 x_{11} + x_8 \end{bmatrix}$$

c

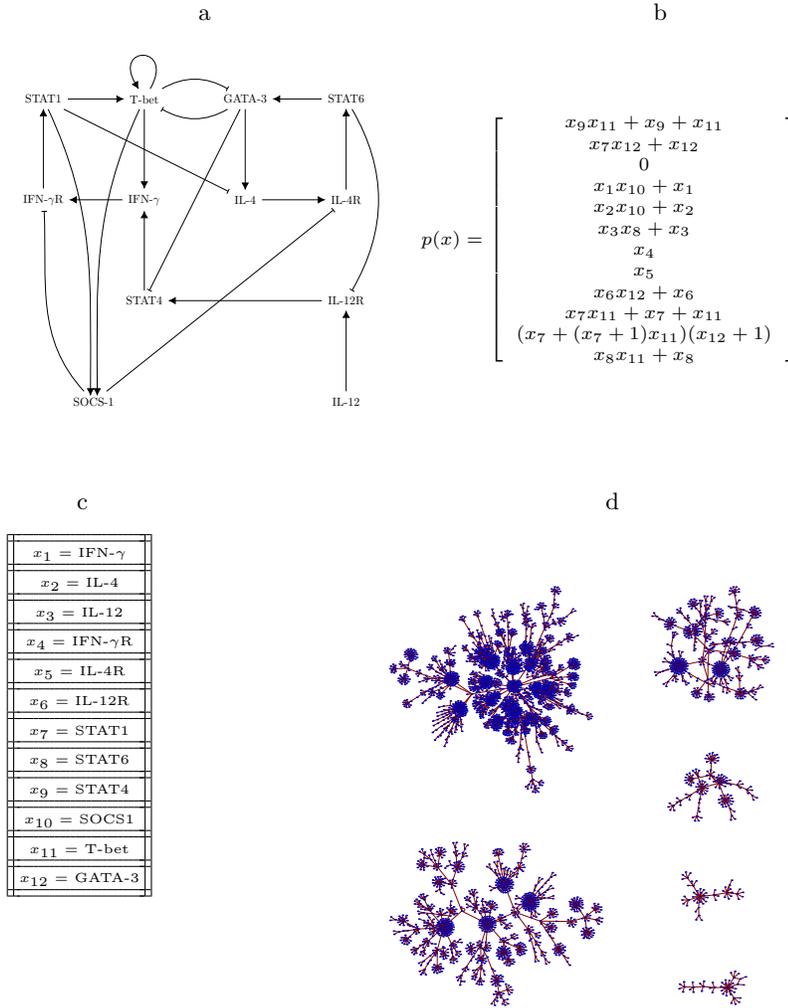| |
|---|
| x_1 = IFN-γ |
| x_2 = IL-4 |
| x_3 = IL-12 |
| x_4 = IFN-γR |
| x_5 = IL-4R |
| x_6 = IL-12R |
| x_7 = STAT1 |
| x_8 = STAT6 |
| x_9 = STAT4 |
| x_{10} = SOCS1 |
| x_{11} = T-bet |
| x_{12} = GATA-3 |

d

Figure 6: **Model of the Th-lymphocyte differentiation**. (**a**) Wiring diagram of the interactions. (**b**) Polynomial update rule of the Boolean network $x(t + 1) = p(x(t))$. (**c**) Physical meaning of the states of the Boolean network. (**d**) Extensive graphical representation of the Boolean network (see the Discussion section).

can be immersed into the linear system (8) with $E \in \mathbb{F}_2^{144 \times 144}$ (represented graphically in Figure 7). Note that the number of state variables of system (8) is smaller than the one of the linear immersion given in (5) or the one obtained by using the semi-tensor product (in fact, for such a Boolean model, one has $z(t) \in \mathbb{F}_2^{4096}$ whereas $y_e(t) \in \mathbb{F}_2^{144}$).

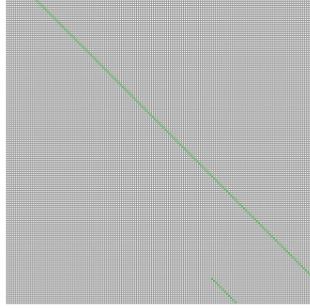By defining the vector $O_{12}(x)$ as in (7), the linear immersion of the Boolean

Figure 7: **Linear immersion of the Boolean network modeling the Th-lymphocyte differentiation**. Dynamical matrix of the linear immersion given in (8), where green and white cells represents entries being equal to one and zero, respectively.
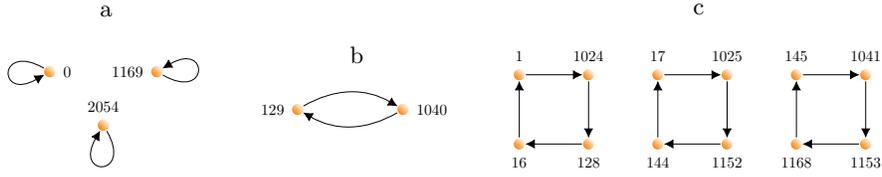

Figure 8: **Equilibria and cycles of the Boolean network modeling the Th-lymphocyte differentiation**. For compactness of representation, each Boolean state $x = [x_1 \quad \cdots \quad x_{12}]^\top \in \mathbb{F}_2^{12}$ has been uniquely identified by its Hamming distance (Hamming, 1950), $d = \sum_{i=1}^{12} 2^{12-i} x_i$, $d \in \mathbb{Z}_{\geqslant 0}$. (**a**) Equilibria. (**b**) Cycle of length 2. (**c**) Cycles of length 4.

network given in (8) can be used to characterize its cycles. In fact, a Boolean state $x \in \mathbb{F}_2^{12}$ belongs to a cycle of length ℓ if and only if

$$(E^\ell + I)O_{12}(x) = 0, \qquad (E^j + I)O_{12}(x) \neq 0, \ j = 1, \dots, \ell - 1.$$

By using this method, it can be derived that the Boolean network has (see Figure 8):

- three equilibria (*i.e.*, three cycles of length 1);

- a cycle of length 2;

- no cycle of length 3;

- three cycles of length 4;

- no cycle of length greater than or equal to 5.

Furthermore, by using classical linear algebra tools, the immersion (8) can be used to compute the solutions of the Boolean network in closed-form. As a matter of fact, there exist a nonsingular matrix $T \in \mathbb{F}_2^{144 \times 144}$ and a block diagonal
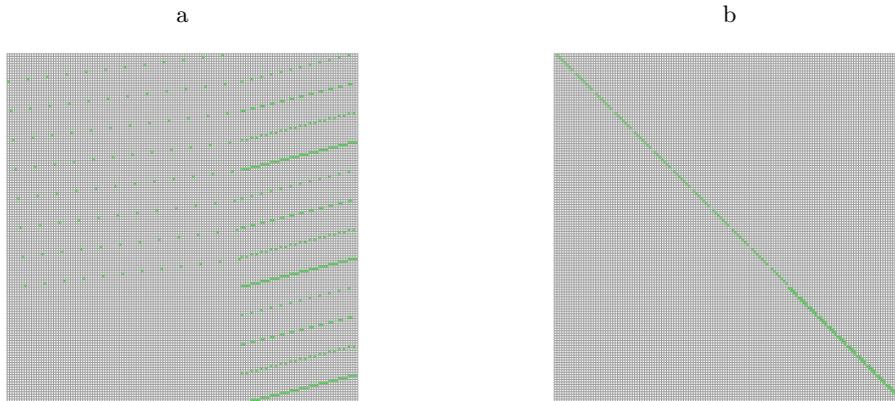
Figure 9: **Jordan normal form of the dynamical matrix of the immersion**. Matrices $T \in \mathbb{F}_2^{144 \times 144}$ and $J \in \mathbb{F}_2^{144 \times 144}$ such that $E = TJT^{-1}$. (**a**) Nonsingular matrix T. (**b**) Block diagonal matrix J.

matrix $J \in \mathbb{F}_2^{144 \times 144}$ in Jordan form such that $E = TJT^{-1}$ (see Figure 9 for a graphical representation of such matrices).

In particular, the matrix J can be expressed as

$$J = \text{blk diag} \left(\underbrace{J_{0,8}, \ldots, J_{0,8}}_{12 \text{ blocks}}, \underbrace{J_{1,4}, \ldots, J_{1,4}}_{12 \text{ blocks}} \right),$$

where $J_{i,j}$ denotes the Jordan block relative to eigenvalue $i \in \mathbb{F}_2$ of order $j \in \mathbb{Z}_{\geqslant 0}$. Thus, since $E^{\ell} = TJ^{\ell}T^{-1}$ and $x(t) = [\ I_{12} \quad 0_{12} \quad \cdots \quad 0_{12}\ ]y_e(t)$, the solutions of the Boolean network modeling the Th-lymphocyte differentiation are given by

$$x(t) = [\ I_{12} \quad 0_{12} \quad \cdots \quad 0_{12}\ ]TJ^tT^{-1}O_{12}(x(0)),$$

where

$$J^t = \text{blk diag} \left(\underbrace{J_{0,8}^t, \ldots, J_{0,8}^t}_{12 \text{ blocks}}, \underbrace{J_{1,4}^t, \ldots, J_{1,4}^t}_{12 \text{ blocks}} \right).$$

4. Discussion

It has been shown that tools from algebraic geometry and linear algebra can be suitably coupled in order to characterize and study the dynamics of Boolean networks. In particular, given the logic map representing the dynamics of the network, it is possible to find a polynomial that represents such a map. Such a representation and algebraic geometry tools can then be used to determine an

immersion that recasts the Boolean network in a linear, time-invariant, discrete-time system, here called the reduced linear representation. Such a system can be studied by using classical linear algebra tools in order to determine a closed-form expression for the solution of the Boolean network and to find its cycles.

The main advantage of the proposed tools is that the dimension of the linear system is generically much smaller than the one obtained by using other approaches taken from the literature, as, *e.g.*, the semi-tensor product or a monomial immersion. Therefore, in principle, the coupling of algebraic geometry and of linear algebra allows the analysis of networks involving a large number of binary state variables, in a simple and systematic way.

Since there always exists a constant vector in $\mathbb{F}_2^{N_n+1}$ in the syzygy module of M_{N_n}, the worst case computational complexity of the proposed method is 2^{N_n} (which corresponds to the brute force approach in which all the vectors in $\mathbb{F}_2^{N_n+1}$ are tested to determine whether they are in the syzygy module of M_{N_n}). Nonetheless, such a computational complexity can be highly reduced if efficient methods are used to compute Gröbner bases over finite fields such as the one given in Faugère and Ars (2004).

It is worth mentioning that, since even the singleton attractor detection problem is known to be an NP-hard problem (Akutsu et al., 2012; Melkman and Akutsu, 2013), the method proposed in this paper need not be able to construct a simplified linear model for all the Boolean networks. Nevertheless, in some cases of practical interest, as those given in this paper, it can be efficiently employed to characterize the dynamical behavior of Boolean networks. Furthermore, differently from the tools given in Akutsu et al. (2012) and Melkman and Akutsu (2013), the technique given in this paper can be used also if the logical function that governs the Boolean dynamics is not a nested canalyzing function. The drawback is that the computational complexity of the proposed method may be larger than the one of the techniques given in Akutsu et al. (2012) and Melkman and Akutsu (2013) when the Boolean network satisfies the nested canalyzing assumption.

The effectiveness of these tools has been shown through their application to the analysis of the Boolean network modeling the Th-lymphocyte differentiation. In particular, it has been shown that, while the classical semi-tensor product immerses the Boolean network into a linear system with state of dimension 4096, the proposed approach immerses the same Boolean network into a linear system with state dimension 144. This suggests that the reduced linear representation can be employed to find closed-form solutions of Boolean networks with a large number of Boolean states.

Note that the proposed tool allows to efficiently determine the cycles of the Boolean network. As a matter of fact, by comparing the extensive graphical representation of the Boolean network modeling the Th-lymphocyte differentiation depicted in Figure 6d (that is a directed graph in which each node represents a Boolean state $x \in \mathbb{F}_2^{12}$ and where there is an arc between the node corresponding to $x \in \mathbb{F}_2^{12}$ and the node corresponding to $y \in \mathbb{F}_2^{12}$ if and only if $y = p(x)$, where $p \in \mathbb{F}_2^{12}[x]$ is the polynomial vector given in Figure 6b) and the cycles depicted in Figure 8, it can be easily noted that all the cycles of the Boolean network

have been determined by using the proposed approach. In particular, note that there is a cycle in each connected component of the graph depicted in Figure 6d and hence each connected component of such a graph constitutes the basin of attraction of the corresponding cycle. However, determining the cycles of the Boolean network by inspecting the corresponding extensive graphical representation may be rather challenging, due to the fact that this graph has 4096 nodes (that is the same number of states obtained using the semi-tensor product approach). Hence, the proposed tool seems to be particularly appealing, since it just requires the computation of the null-space of matrices in $\mathbb{F}_2^{144 \times 144}$.

Furthermore, it is to be noted that, differently from the tools proposed in Veliz-Cuba (2011), the technique given in this paper does not neglect the dynamics of any subsystem of the network, by allowing a comprehensive representation of all of its states. As a matter of fact, note that the reduced Boolean network modeling the Th-lymphocyte differentiation (depicted in Figures 2 and 3) has just 3 equilibria (*i.e.*, cycles of length 1 that corresponds to the steady states of the original system):

- $[\ 0 \quad 0\ ]^\top$ that corresponds to the equilibrium

$$[\ 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad \mathbf{0} \quad \mathbf{0}\ ]^\top$$

  of the original system;

- $[\ 0 \quad 1\ ]^\top$ that corresponds to the equilibrium

$$[\ 0 \quad 1 \quad 0 \quad 0 \quad 1 \quad 0 \quad 0 \quad 1 \quad 0 \quad 0 \quad \mathbf{0} \quad \mathbf{1}\ ]^\top$$

  of the original system;

- $[\ 1 \quad 0\ ]^\top$ that corresponds to the equilibrium

$$[\ 1 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 1 \quad \mathbf{1} \quad \mathbf{0}\ ]^\top$$

  of the original system;

and no cycle of length greater than or equal to 2. Therefore, such a network has reduced representation capabilities with respect to system (8), since some complex behaviors (as the cycles of length 2 and 4) are neglected in such a simplified model. Indeed, the main difference between the dimension reduction approach given in this paper and the one given in Veliz-Cuba (2011); Veliz-Cuba et al. (2014, 2015) is that the latter just preserves the steady states, whereas the former allows to take into account also the transient behavior of the network as well as cycles of length greater than or equal to 2. Nonetheless, since the dimension reduction approach given Veliz-Cuba et al. (2015) works directly on the wiring diagram of the network, it can be used also to deal with very large sparse AND-NOT networks.

References

Agnello, D., Lankford, C. S., Bream, J., Morinobu, A., Gadina, M., O'shea, J. J., and Frucht, D. M. (2003). Cytokines and transcription factors that regulate T helper cell differentiation: new players and new insights. *Journal of Clinical Immunology*, 23(3):147–161.

Akutsu, T., Kosub, S., Melkman, A. A., and Tamura, T. (2012). Finding a periodic attractor of a Boolean network. *IEEE/ACM transactions on computational biology and bioinformatics*, 9(5):1410–1421.

Albert, R. and Barabási, A.-L. (2000). Dynamics of complex systems: Scaling laws for the period of Boolean networks. *Physical Review Letters*, 84(24):5660.

Bansal, M., Belcastro, V., Ambesi-Impiombato, A., and Di Bernardo, D. (2007). How to infer gene networks from expression profiles. *Molecular systems biology*, 3(1):78.

Bansal, M., Gatta, G. D., and Di Bernardo, D. (2006). Inference of gene regulatory networks and compound mode of action from time course gene expression profiles. *BMC Bioinformatics*, 22(7):815–822.

Busetto, A. G. and Lygeros, J. (2014). Experimental design for system identification of Boolean control networks in biology. In *IEEE 53rd Annual Conference on Decision and Control (CDC)*, pages 5704–5709. IEEE.

Chaouiya, C., Naldi, A., and Thieffry, D. (2012). Logical modelling of gene regulatory networks with GINsim. In *Bacterial Molecular Networks*, pages 463–479. Springer.

Cheng, D. and Qi, H. (2010). A linear representation of dynamics of Boolean networks. *IEEE Transactions on Automatic Control*, 55(10):2251–2258.

Cheng, D., Qi, H., and Li, Z. (2010). *Analysis and Control of Boolean Networks: A Semi-tensor Product Approach*. Springer Science & Business Media.

Covert, M. W., Schilling, C. H., Famili, I., Edwards, J. S., Goryanin, I. I., Selkov, E., and Palsson, B. O. (2001). Metabolic modeling of microbial strains in silico. *Trends in biochemical sciences*, 26(3):179–186.

Cox, D. A., Little, J., and O'Shea, D. (2006). *Using algebraic geometry*. Springer Science & Business Media.

Cox, D. A., Little, J., and O'Shea, D. (2015). *Ideals, varieties, and algorithms*. Springer Science & Business Media.

Crick, F. H. (1958). On protein synthesis. In Sanders, F. K., editor, *Symposia of the Society for Experimental Biology: The Biological Replication of Macromolecules*, volume 12, pages 138–163.

Crick, F. H. (1970). Central dogma of molecular biology. *Nature*, 227(5258):561.

Davidich, M. and Bornholdt, S. (2008). The transition from differential equations to Boolean networks: a case study in simplifying a regulatory network model. *Journal of Theoretical Biology*, 255(3):269–277.

Dilão, R. (2014). The regulation of gene expression in eukaryotes: bistability and oscillations in repressilator models. *Journal of Theoretical Biology*, 340:199–208.

Faugère, J.-C. and Ars, G. (2004). *Comparison of XL and Gröbner basis algorithms over Finite Fields*. PhD thesis, INRIA.

Fornasini, E. and Valcher, M. E. (2013). Observability, reconstructibility and state observers of Boolean control networks. *IEEE Transactions on Automatic Control*, 58(6):1390–1401.

Grayson, D. R. and Stillman, M. E. (2018). Macaulay2, a software system for research in algebraic geometry. Available at `https://faculty.math.illinois.edu/Macaulay2/`.

Hamming, R. W. (1950). Error detecting and error correcting codes. *Bell Labs Technical Journal*, 29(2):147–160.

Hinkelmann, F., Brandon, M., Guang, B., McNeill, R., Blekherman, G., Veliz-Cuba, A., and Laubenbacher, R. (2011a). ADAM: analysis of discrete models of biological systems using computer algebra. *BMC Bioinformatics*, 12(1):295.

Hinkelmann, F., Murrugarra, D., Jarrah, A. S., and Laubenbacher, R. (2011b). A mathematical framework for agent based models of complex biological networks. *Bulletin of mathematical biology*, 73(7):1583–1602.

Isidori, A. (2013). *Nonlinear control systems*. Springer Science & Business Media.

Kauffman, S. A. (1969). Metabolic stability and epigenesis in randomly constructed genetic nets. *Journal of theoretical biology*, 22(3):437–467.

Kobayashi, K. and Hiraishi, K. (2017). Optimization-based approaches to control of probabilistic Boolean networks. *Algorithms*, 10(1):31.

Kobayashi, K., Imura, J.-I., and Hiraishi, K. (2010). Polynomial-time algorithm for controllability test of a class of Boolean biological networks. *EURASIP Journal on Bioinformatics and Systems Biology*, 2010(1):210685.

Melkman, A. A. and Akutsu, T. (2013). An improved satisfiability algorithm for nested canalyzing functions and its application to determining a singleton attractor of a Boolean network. *Journal of Computational Biology*, 20(12):958–969.

Mendoza, L. (2006). A network model for the control of the differentiation process in Th cells. *Biosystems*, 84(2):101–114.

17

Menini, L., Possieri, C., and Tornambe, A. (2017). Boolean network representation of a continuous-time system and finite-horizon optimal control: application to the single-gene regulatory system for the lac operon. *International Journal of Control*, 90(3):519–552.

Menini, L. and Tornambe, A. (2013). Deformations for linear periodic discrete-time systems: the adjoint normal form. *International Journal of Control*, 86(7):1248–1257.

Menini, L. and Tornambe, A. (2013a). Immersion and Darboux polynomials of Boolean networks with application to the pseudomonas syringae hrp regulon. In *IEEE 52nd Annual Conference on Decision and Control (CDC)*, pages 4092–4097. IEEE.

Menini, L. and Tornambe, A. (2013b). Observability and dead-beat observers for boolean networks modeled as polynomial discrete-time systems. In *IEEE 52nd Annual Conference on Decision and Control (CDC)*, pages 4428–4433. IEEE.

Meyer, C. D. (2000). *Matrix analysis and applied linear algebra*. Siam.

Murphy, K. M. and Reiner, S. L. (2002). Decision making in the immune system: the lineage decisions of helper T cells. *Nature Reviews Immunology*, 2(12):933.

Müssel, C., Hopfensitz, M., and Kestler, H. A. (2010). BoolNet—an R package for generation, reconstruction and analysis of Boolean networks. *Bioinformatics*, 26(10):1378–1380.

Papin, J. A., Price, N. D., Wiback, S. J., Fell, D. A., and Palsson, B. O. (2003). Metabolic pathways in the post-genome era. *Trends in biochemical sciences*, 28(5):250–258.

Possieri, C. and Teel, A. R. (2017). Asymptotic stability in probability for stochastic Boolean networks. *Automatica*, 83:1–9.

Remy, E., Ruet, P., Mendoza, L., Thieffry, D., and Chaouiya, C. (2006). From logical regulatory graphs to standard Petri nets: dynamical roles and functionality of feedback circuits. In *Transactions on Computational Systems Biology VII*, pages 56–72. Springer.

Rohr, C., Marwan, W., and Heiner, M. (2010). Snoopy—a unifying Petri net framework to investigate biomolecular networks. *Bioinformatics*, 26(7):974–975.

Shah, O. S., Chaudhary, M. F. A., Awan, H. A., Fatima, F., Arshad, Z., Amina, B., Ahmed, M., Hameed, H., Furqan, M., Khalid, S., et al. (2018). ATLANTIS-attractor landscape analysis toolbox for cell fate discovery and reprogramming. *Scientific Reports*, 8(1):3554.

Veliz-Cuba, A. (2011). Reduction of Boolean network models. *Journal of Theoretical Biology*, 289:167–172.

Veliz-Cuba, A., Aguilar, B., Hinkelmann, F., and Laubenbacher, R. (2014). Steady state analysis of Boolean molecular network models via model reduction and computational algebra. *BMC bioinformatics*, 15(1):221.

Veliz-Cuba, A., Aguilar, B., and Laubenbacher, R. (2015). Dimension reduction of large sparse AND-NOT network models. *Electronic Notes in Theoretical Computer Science*, 316:83–95.

Veliz-Cuba, A., Jarrah, A. S., and Laubenbacher, R. (2010). Polynomial algebra of discrete models in systems biology. *BMC Bioinformatics*, 26(13):1637–1643.

Wuensche, A. (2011). *Exploring discrete dynamics*. Luniver Press.