A Maximum Principle for Single–Input Boolean Control Networks

Dmitriy Laschov and Michael Margaliot

***Abstract*—Boolean networks are recently attracting considerable interest as computational models for genetic and cellular networks. We consider a Mayer–type optimal control problem for a single–input Boolean network, and derive a necessary condition for a control to be optimal. This provides an analog of Pontryagin's maximum principle for single–input Boolean networks.**

***Index Terms*—Semi–tensor product, logical functions, sum of products representation, systems biology, variational analysis, necessary condition for optimality.**

I. INTRODUCTION

A Boolean network consists of a set of Boolean variables whose state is determined by other variables in the network. Cellular automata, with two possible states per cell, are a particular case of Boolean networks. Here the state of each variable at time $k+1$ is determined by the state of its spatial neighbors at time k [21].

Boolean networks have been studied extensively as models for simple artificial neural networks (see, e.g. [13]). More recently, such networks gained renewed interest as models for biological systems. S. A. Kauffman pioneered the modeling and analysis of gene regulation networks using random Boolean networks [14], [15].

A Boolean network with n variables has 2^n possible states and therefore the dynamics for any initial condition must fall into an attractor. One possible way to enrich the dynamics is to consider Boolean networks with exogenous (binary) inputs. This leads to the concept of Boolean control networks (BCNs).

Daizhan Cheng and his colleagues developed an algebraic state–space representation of BCNs using the *semi–tensor product* of matrices. This representation is quite useful for studying BCNs in a control–theoretic framework. Examples include the analysis of disturbance decoupling [5], controllability and observability [9], realization theory [8], and more [10], [11], [6].

Here we use this state–space representation to analyze a Mayer–type optimal control problem for BCNs with a single input. Our main result is a necessary condition for a control to be optimal. This provides a kind of Pontryagin maximum principle (PMP) for BCNs. The proof of our main result is motivated by the simple proof of a special case of the PMP used in the variational analysis of switched systems [18] (see also [19], [23], [20]).

Some related work includes the numerical solution of optimal control problems for *Probabilistic Boolean Networks* using dynamic programming [24], [25], and the work of Akutsu et al. [1] demonstrating that control problems for BCNs are in general NP–hard.

Corresponding author: Prof. Michael Margaliot, School of Elec. Eng.-Systems, Tel Aviv University, Israel 69978. Homepage: `www.eng.tau.ac.il/~michaelm` Email: `michaelm@eng.tau.ac.il`

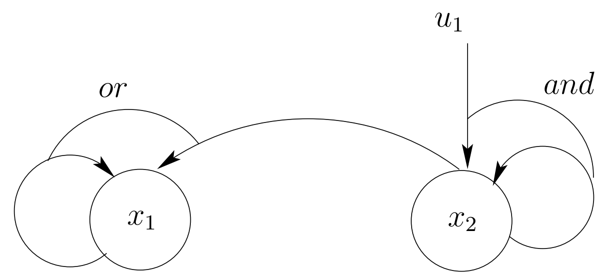

Fig. 1. Graphical representation of the BCN in Example 1.

The remainder of this note is organized as follows. Section II reviews BCNs. Section III describes Cheng's algebraic state–space representation of BCNs using the semi–tensor product of matrices. Section IV details our main result which is a new maximum principle (MP) for single–input BCNs. It also provides an analysis of the so–called singular case inwhich the MP itself does not provide direct information on the optimal control. The proof is given in Section V. Section VI details two examples demonstrating the application of the new MP.

II. BOOLEAN CONTROL NETWORKS

A Boolean control network is a discrete–time logical dynamic control system in the form

$$\begin{aligned} x_1(k+1) &= f_1(x_1(k), \dots, x_n(k), u_1(k), \dots, u_m(k)), \quad (1)\\ &\vdots \\ x_n(k+1) &= f_n(x_1(k), \dots, x_n(k), u_1(k), \dots, u_m(k)), \end{aligned}$$

where $x_i, u_i \in \{\text{True}, \text{False}\}$ and each f_i is a Boolean function.

A BCN may be represented graphically as a network with n nodes, representing the x_is, and m inputs. A directed edge from node i (input u_i) to node j implies that $x_j(k+1)$ depends on $x_i(k)$ $(u_i(k))$.

Example 1 *Fig. 1 depicts the graphical representation of the single–input BCN*

$$\begin{aligned} x_1(k+1) &= x_1(k) \vee x_2(k), \\ x_2(k+1) &= x_2(k) \wedge u_1(k). \end{aligned} \quad (2)$$

It is worth noting that a BCN with m inputs may be interpreted as a Boolean switched system switching between 2^m possible subsystems, with the value of the control determining which subsystem is active at every time step. For example, we may view (2) as a Boolean switched system switching between the two subsystems:

$$\begin{aligned} x_1(k+1) &= x_1(k) \vee x_2(k), \\ x_2(k+1) &= x_2(k) \wedge \text{True} = x_2(k), \end{aligned}$$

and

$$\begin{aligned} x_1(k+1) &= x_1(k) \vee x_2(k), \\ x_2(k+1) &= x_2(k) \wedge \text{False} = \text{False}. \end{aligned}$$

III. Algebraic state–space representation of BCNs

Daizhan Cheng [7] and his colleagues introduced the concept of a semi–tensor product and used it to represent BCNs in an algebraic state–space form. This representation is useful for studying BCNs in a control–theoretic framework. We briefly review this approach.

A. Semi–tensor product

Recall that the *Kronecker product* (see, e.g. [3, Chapter 7]) of two matrices $A \in \mathbb{R}^{m\times n}$ and $B \in \mathbb{R}^{p\times q}$ is

$$A \otimes B = \begin{bmatrix} a_{11}B & \cdots & a_{1n}B \\ \vdots & \ddots & \vdots \\ a_{m1}B & \cdots & a_{mn}B \end{bmatrix}.$$

Note that $A \otimes B \in \mathbb{R}^{(mp)\times(nq)}$.

Given two positive integers a, b, let $\operatorname{lcm}(a, b)$ denote the *least common multiplier* of a and b. For example, $\operatorname{lcm}(6, 8) = 24$. Let I_n denote the $n \times n$ identity matrix.

Definition 2 *The* semi–tensor product *of two matrices $A \in \mathbb{R}^{m\times n}$ and $B \in \mathbb{R}^{p\times q}$ is*

$$A \ltimes B = (A \otimes I_{\alpha/n})(B \otimes I_{\alpha/p}),$$

where $\alpha = \operatorname{lcm}(n, p)$.

Remark 3 *Note that $(A \otimes I_{\alpha/n}) \in \mathbb{R}^{(m\alpha/n)\times\alpha}$ and $(B \otimes I_{\alpha/p}) \in \mathbb{R}^{\alpha\times(q\alpha/p)}$, so $(A \ltimes B) \in \mathbb{R}^{(m\alpha/n)\times(q\alpha/p)}$.*

Remark 4 *If $n = p$, then $A \ltimes B = (A \otimes I_1)(B \otimes I_1) = AB$, so we recover the standard matrix product. Thus, we may view the semi–tensor product as a generalization of the standard matrix product that allows multiplying two matrices of arbitrary dimensions.*

Example 5 *Consider $a \ltimes b$ where $a, b \in \mathbb{R}^2$. Here $m = p = 2$ and $n = q = 1$, so $\alpha = \operatorname{lcm}(n, p) = 2$, and*

$$\begin{aligned} a \ltimes b &= (a \otimes I_2)(b \otimes I_1) \\ &= \begin{bmatrix} a_1 & 0 \\ 0 & a_1 \\ a_2 & 0 \\ 0 & a_2 \end{bmatrix} b \\ &= \begin{bmatrix} a_1b_1 & a_1b_2 & a_2b_1 & a_2b_2 \end{bmatrix}^T. \end{aligned}$$

Example 6 *Consider the semi–tensor product of a row–vector $a^T = \begin{bmatrix} a_1 & \dots & a_n \end{bmatrix}$ and a column vector $b = \begin{bmatrix} b_1 & \dots & b_p \end{bmatrix}^T$. Suppose that $s = n/p$ is an integer. Then $\alpha = \operatorname{lcm}(n, p) = n$ and*

$$\begin{aligned} a^T \ltimes b &= (a^T \otimes I_1)(b \otimes I_s) \\ &= a^T \begin{bmatrix} b_1 I_s \\ \vdots \\ b_p I_s \end{bmatrix}. \end{aligned}$$

Various properties of the semi–tensor product are analyzed in [7]. For our purposes, it is sufficient to note that this product is associative:

$$A \ltimes (B \ltimes C) = (A \ltimes B) \ltimes C,$$

and distributive:

$$(A + B) \ltimes C = (A \ltimes C) + (B \ltimes C).$$

B. Algebraic representation of Boolean functions

The semi–tensor product allows representing Boolean functions in an algebraic form. Let e_n^i denote the ith column of the identity matrix I_n. Represent the Boolean values True and False by $e_2^1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$ and $e_2^2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$, respectively. Then any Boolean function of n variables $f : \{\text{False}, \text{True}\}^n \to \{\text{False}, \text{True}\}$ can be equivalently represented as a mapping $\bar{f} : \{e_2^1, e_2^2\}^n \to \{e_2^1, e_2^2\}$. With some abuse of notation, we identify $\bar{f}$ with f. In other words, from here on a Boolean variable x_i is always a vector in $\{e_2^1, e_2^2\}$.

Theorem 7 *[10] Let $f : \{e_2^1, e_2^2\}^n \to \{e_2^1, e_2^2\}$ be a Boolean function. There exists a unique binary matrix M_f of dimensions 2×2^n such that*

$$f(x_1, \dots, x_n) = M_f \ltimes x_1 \ltimes \cdots \ltimes x_n.$$

M_f is called the structure matrix *of f.*

Remark 8 *In order to provide some intuition on this representation, consider the case $n = 2$, i.e. $f = f(x_1, x_2)$. Recall that $x_i \in \{e_2^1, e_2^2\}$, so we may write $x_1 = \begin{bmatrix} v & \bar{v} \end{bmatrix}^T$ and $x_2 = \begin{bmatrix} w & \bar{w} \end{bmatrix}^T$, with $v, w \in \{0, 1\}$. Then*

$$x_1 \ltimes x_2 = \begin{bmatrix} vw & v\bar{w} & \bar{v}w & \bar{v}\bar{w} \end{bmatrix}^T, \tag{3}$$

i.e. $x_1 \ltimes x_2$ contains all the possible minterms *of v and w. Recall that any Boolean function may be represented as a sum of some minterms of its variables (see, e.g. [16]). This is known as the sum of products (SOP) representation. The multiplication $M_f \ltimes x_1 \ltimes x_2$ provides such a representation. Note that* (3) *implies that $x_1 \ltimes x_2 \in \{e_4^1, \dots, e_4^4\}$. Indeed, one and only one minterm is equal to* 1*.*

Example 9 *Consider the function $f(x) = \bar{x}$, i.e. f is defined by $f(e_2^1) = e_2^2$ and $f(e_2^2) = e_2^1$. It is easy to verify that $f(x) = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \ltimes x$. Consider the function $g(x_1, x_2) = x_1 \wedge x_2$. It is straightforward to verify that*

$$g(x_1, x_2) = M_g \ltimes x_1 \ltimes x_2,$$

with $M_g = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix}$. For example,

$$\begin{aligned} M_g \ltimes e_2^1 \ltimes e_2^2 &= M_g \ltimes \begin{bmatrix} 0 & 1 & 0 & 0 \end{bmatrix}^T \\ &= M_g \begin{bmatrix} 0 & 1 & 0 & 0 \end{bmatrix}^T \\ &= \begin{bmatrix} 0 & 1 \end{bmatrix}^T \\ &= e_2^2, \end{aligned}$$

corresponding to (*True* $\wedge$ *False*) $=$ *False.*

C. Algebraic representation of BCNs

Since the dynamics of BCNs is described by a set of Boolean functions, it is clear from the discussion above that the semi–tensor product can be used to provide an algebraic state–space representation of BCNs.

Theorem 10 *[11] Consider a BCN with state variables $x_1, \dots, x_n$, and inputs $u_1, \dots, u_m$, with $x_i, u_i \in \{e_2^1, e_2^2\}$. Let $x(k) = x_1(k) \ltimes \dots \ltimes x_n(k)$ and $u(k) = u_1(k) \ltimes \dots \ltimes u_m(k)$. There exists a unique matrix $L \in \mathbb{R}^{2^n \times 2^{n+m}}$ such that*

$$x(k+1) = L \ltimes u(k) \ltimes x(k). \tag{4}$$

The matrix L is called the transition matrix *of the BCN.*

Algorithms for converting a BCN in the form (1) to its algebraic representation (4), and vice versa, may be found in [10], [9].

Remark 11 *The intuition behind this representation is very similar to the algebraic representation of a single Boolean function. To demonstrate this, consider a BCN with $n = 2$ and $m = 1$. Then $x(k+1) = L \ltimes u_1(k) \ltimes x_1(k) \ltimes x_2(k)$. To simplify the notation, we omit from here on the dependence on k. Denote $x_1 = \begin{bmatrix} p & \bar{p} \end{bmatrix}^T$, $x_2 = \begin{bmatrix} q & \bar{q} \end{bmatrix}^T$, and $u_1 = \begin{bmatrix} v & \bar{v} \end{bmatrix}^T$. Then*

$$\begin{aligned} u_1 \ltimes x_1 \ltimes x_2 = \big[& vpq \quad vp\bar{q} \quad v\bar{p}q \quad v\bar{p}\bar{q} \\ & \bar{v}pq \quad \bar{v}p\bar{q} \quad \bar{v}\bar{p}q \quad \bar{v}\bar{p}\bar{q} \big]. \end{aligned}$$

Thus, $u \ltimes x$ includes all the possible minterms of the input and state variables. The equation $x(k+1) = L \ltimes u \ltimes x$ provides a description of (every minterm of) the next state in terms of the current state and inputs.

Example 12 *Consider the BCN in Example 1. Here $n = 2$ and $m = 1$, so $x(k) = x_1(k) \ltimes x_2(k)$ and $u(k) = u_1(k)$. Applying the algorithm described in [9], we find that the transition matrix is $L = \begin{bmatrix} 1&0&1&0&0&0&0&0 \\ 0&1&0&0&1&1&1&0 \\ 0&0&0&0&0&0&0&0 \\ 0&0&0&1&0&0&0&1 \end{bmatrix}$. To demonstrate the equivalence of the original dynamics and (4), consider for example the case $x_1(k) =$ False, $x_2(k) =$ True, and $u(k) =$ True. Then (2) yields*

$$x_1(k+1) = \textit{True}, \quad x_2(k+1) = \textit{True}. \tag{5}$$

In the algebraic framework, this corresponds to $x_1(k) = e_2^2$, $x_2(k) = u(k) = e_2^1$. Then

$$\begin{aligned} x(k+1) &= L \ltimes u(k) \ltimes x(k) \\ &= L \ltimes \begin{bmatrix} 1 \\ 0 \end{bmatrix} \ltimes \begin{bmatrix} 0 \\ 1 \end{bmatrix} \ltimes \begin{bmatrix} 1 \\ 0 \end{bmatrix} \\ &= L \ltimes \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\ &= L \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \\ &= \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}^T. \end{aligned}$$

Writing $x_1(k+1) = \begin{bmatrix} v & \bar{v} \end{bmatrix}^T$ and $x_2(k+1) = \begin{bmatrix} w & \bar{w} \end{bmatrix}^T$ yields $x(k+1) = \begin{bmatrix} vw & v\bar{w} & \bar{v}w & \bar{v}\bar{w} \end{bmatrix}^T$, so $v = w = 1$. Thus, $x_1(k+1) = x_2(k+1) = e_2^1$, and this agrees, of course, with (5).

IV. Main result

Consider a BCN in the algebraic state–space representation (4). From here on we consider the case of a single control (i.e., $m = 1$) and fix some (arbitrary) initial condition $x(0) = x_0 \in \{e_{2^n}^1, \dots, e_{2^n}^{2^n}\}$.

A. Optimal control problem

A fundamental problem for all dynamical control systems is to determine a control that is optimal in some sense. In other words, a control that maximizes (or minimizes) a given cost–functional.

Fix a final time $N > 0$. Let $\mathbb{U}$ denote the set of admissible controls, i.e. the set of all the sequences $\{u(0), \dots, u(N-1)\}$, with $u(i) \in \{e_2^1, e_2^2\}$. For a control $u \in \mathbb{U}$, let $x(k; u)$ denote the solution of (4), with $x(0) = x_0$, at time k. Fix a vector $r \in \mathbb{R}^{2^n}$, and consider the *cost–functional*

$$J(u) = r^T x(N; u). \tag{6}$$

We now pose a Mayer–type optimal control problem.

Problem 13 *Find a control $u^* \in \mathbb{U}$ that maximizes J.*

This problem clearly admits a solution, as $\mathbb{U}$ is a finite set. We refer to a control that maximizes J as an *optimal control*. In principle, Problem 13 may be solved numerically by simply calculating $x(N; u)$ for any $u \in \mathbb{U}$. However, this is clearly not practical for large values of N.

Remark 14 *Recall that $x(N)$ consists of all the minterms of the Boolean state variables at time N. Hence any Boolean function f of the state at time N may be represented in the form* (6)*, i.e. as $f = r_f^T x(N, u)$, where r_f is a* binary *vector. In this particular case, $J(u)$ can attain only two values, namely, zero and one. This yields a reachability problem that is quite relevant for BCNs that model biological networks, as here states can usually be divided into desirable and non–desirable states. For example, in a model of cell differentiation a non–desirable state corresponds to uncontrolled cell proliferation (see, e.g. [12], [17]). For a different approach for analyzing reachability in BCNs, see [9].*

Example 15 *Suppose that $n = 3$, and let $x_1(N) = \begin{bmatrix} v & \bar{v} \end{bmatrix}^T$, $x_2(N) = \begin{bmatrix} w & \bar{w} \end{bmatrix}^T$, and $x_3(N) = \begin{bmatrix} q & \bar{q} \end{bmatrix}^T$. Then*

$$\begin{aligned} x(N) &= x_1(N) \ltimes x_2(N) \ltimes x_3(N) \\ &= \big[vwq \quad vw\bar{q} \quad v\bar{w}q \quad v\bar{w}\bar{q} \quad \bar{v}wq \\ &\qquad \bar{v}w\bar{q} \quad \bar{v}\bar{w}q \quad \bar{v}\bar{w}\bar{q} \big]^T. \end{aligned}$$

Suppose that $r = \begin{bmatrix} 1 & 1 & 0 & 0 & \dots & 0 \end{bmatrix}^T$. Then $r^T x = vw$, so maximizing (6) *corresponds to trying to find a control u steering the BCN to $x_1(N) = x_2(N) = e_2^1$, if it exists.*

We are interested in developing an analytical characterization of optimal controls. Iterating (4) shows that for any two integers $k \geq j \geq 0$,

$$x(k;u) = C(k,j;u) \ltimes x(j;u), \tag{7}$$

where

$$C(k,j;u) = L \ltimes u(k-1) \ltimes L \ltimes u(k-2) \ltimes \cdots \ltimes L \ltimes u(j), \tag{8}$$

with $C(k,k;u) = I_{2^n}$. We refer to $C(k,j;u)$ as the *transition matrix* from time j to time k corresponding to the control u. Note that (8) implies that for any $k \geq l \geq j$,

$$C(k,j;u) = C(k,l;u) \ltimes C(l,j;u).$$

We can now state our main result.

Theorem 16 *Consider the BCN* (4) *with $m = 1$ and $x(0) = x_0$. Suppose that $u^* = \{u^*(0), \dots, u^*(N-1)\} \in \mathbb{U}$ is an optimal control for Problem 13, and let x^* denote the corresponding solution. Define the* adjoint $\lambda : \{1, \dots, N\} \to \mathbb{R}^{2^n}$ *as the solution of*

$$\begin{aligned} \lambda(k) &= (L \ltimes u^*(k))^T \ltimes \lambda(k+1), \\ \lambda(N) &= r, \end{aligned} \tag{9}$$

and the switching function $m : \{0, 1, \dots, N-1\} \to \mathbb{R}$ *by*

$$m(s) = \lambda^T(s+1) \ltimes L \ltimes \begin{bmatrix} 1 \\ -1 \end{bmatrix} \ltimes x^*(s). \tag{10}$$

Then for any $s \in \{0, 1, \dots, N-1\}$,

$$u^*(s) = \begin{cases} e_2^1, & \text{if } m(s) > 0, \\ e_2^2, & \text{if } m(s) < 0. \end{cases} \tag{11}$$

Theorem 16 provides a necessary condition for optimality in terms of the switching function m. Note that this is somewhat similar to the PMP for discrete–time dynamical systems (see, e.g. [22, Ch. 8]). In particular, it leads to a two–point boundary value problem in terms of (x, λ).

Remark 17 *It is instructive to verify that $m(\cdot)$ is indeed a scalar function. Since the dimensions of $\lambda^T(\cdot)$ are 1×2^n and those of L are $2^n \times 2^{n+1}$ (recall that we consider a BCN with $m = 1$ inputs), it follows from Remark 4 that*

$$\lambda^T(s+1) \ltimes L = \lambda^T(s+1) L \in \mathbb{R}^{1 \times 2^{n+1}}.$$

Since the dimensions of $x^*(\cdot)$ are $2^n \times 1$, Remark 3 implies that

$$\begin{bmatrix} 1 \\ -1 \end{bmatrix} \ltimes x^*(s) \in \mathbb{R}^{2^{n+1} \times 1}.$$

Thus,

$$\begin{aligned} m(s) &= \lambda^T(s+1) \ltimes L \ltimes \begin{bmatrix} 1 \\ -1 \end{bmatrix} \ltimes x^*(s) \\ &= \lambda^T(s+1) L (\begin{bmatrix} 1 \\ -1 \end{bmatrix} \ltimes x^*(s)) \end{aligned}$$

is a scalar.

When $m(s) = 0$, Eq. (11) does not provide any information on $u^*(s)$. The next result shows that this *singular case* can be easily handled.

Theorem 18 *Suppose that the conditions of Theorem 16 hold. If $m(k) = 0$ for some k, then there exists an optimal control u^* satisfying $u^*(k) = e_2^1$, and there exists an optimal control w^* satisfying $w^*(k) = e_2^2$.*

V. Proof of Main Result

Fix arbitrary time $p \in \{0, \dots, N-1\}$ and vector $v \in \{e_2^1, e_2^2\}$. Define a new control $u \in \mathbb{U}$ by

$$u(j) = \begin{cases} v, & \text{if } j = p, \\ u^*(j), & \text{otherwise.} \end{cases} \tag{12}$$

In other words, u is identical to the optimal control u^* except, perhaps, at a single time step. This provides an analog of the *needle variation* used in the proof of the PMP (see, e.g., [4], [2]).

It follows from the definition of the transition matrix that

$$\begin{aligned} x(N;u^*) &= C(N,p+1;u^*) \ltimes C(p+1,p;u^*) \\ &\quad \ltimes C(p,0;u^*) \ltimes x_0 \\ &= C(N,p+1;u^*) \ltimes L \ltimes u^*(p) \ltimes x^*(p), \end{aligned}$$

and similarly

$$\begin{aligned} x(N;u) &= C(N,p+1;u) \ltimes L \ltimes u(p) \ltimes x(p;u) \\ &= C(N,p+1;u^*) \ltimes L \ltimes v \ltimes x^*(p), \end{aligned}$$

where the second equation follows from the definition of u. Thus,

$$\begin{aligned} & x^*(N) - x(N;u) \\ &\quad = C(N,p+1;u^*) \ltimes L \ltimes (u^*(p) - v) \ltimes x^*(p). \end{aligned}$$

Hence,

$$\begin{aligned} & J(u^*) - J(u) \\ &\quad = r^T (C(N,p+1;u^*) \ltimes L \ltimes (u^*(p) - v) \ltimes x^*(p)). \end{aligned} \tag{13}$$

To simplify this expression, let $w^T(p+1) = r^T C(N, p+1; u^*)$. Then $w^T(N) = r^T C(N,N;u^*) = r^T$, and

$$\begin{aligned} w(p) &= C^T(N,p;u^*) r \\ &= (C(N,p+1;u^*) \ltimes C(p+1,p;u^*))^T r \\ &= (C(N,p+1;u^*) C(p+1,p;u^*))^T r \\ &= C^T(p+1,p;u^*) C^T(N,p+1;u^*) r \\ &= C^T(p+1,p;u^*) \ltimes C^T(N,p+1;u^*) r \\ &= (L \ltimes u^*(p))^T \ltimes w(p+1). \end{aligned}$$

Comparing this with (9), we find that $w(p) = \lambda(p)$ for all p, and thus (13) yields

$$J(u^*) - J(u) = \lambda^T(p+1) \ltimes L \ltimes (u^*(p) - v) \ltimes x^*(p). \tag{14}$$

Suppose that $u^*(p) = e_2^1$. Choose v in the definition of u (12) as $v = e_2^2$. Then

$$J(u^*) - J(u) = m(p). \tag{15}$$

If $m(p) < 0$, then this contradicts the fact that u^* is optimal. We conclude that if $m(p) < 0$, then $u^*(p) \neq e_2^1$, so $u^*(p) = e_2^2$. A similar argument shows that if $m(p) > 0$, then $u^*(p) = e_2^1$. Since p is arbitrary, this completes the proof of Theorem 16. $\square$

To prove Theorem 18, suppose that $m(p) = 0$. Define a new control u by

$$u(k) = \begin{cases} \begin{bmatrix} 1 & 1 \end{bmatrix}^T - u^*(k), & k = p, \\ u^*(k), & \text{otherwise.} \end{cases}$$

Note that $u \in \mathbb{U}$. Then (15) implies that $J(u) = J(u^*)$, so u is also an optimal control. $\square$

VI. Examples

To demonstrate the application of the MP, we consider two simple examples.

Example 19 *Consider the BCN*

$$\begin{aligned} x(k+1) &= x(k) \wedge u(k), \\ x(0) &= \textit{True}. \end{aligned} \tag{16}$$

Here $n = m = 1$ and the algebraic state–space form is

$$\begin{aligned} x(k+1) &= L \ltimes u(k) \ltimes x(k), \\ x(0) &= e_2^1, \end{aligned} \tag{17}$$

with $L = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix}$. Fix some final time $N > 0$ and consider Problem 13 for $r = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$. Letting $x^*(N) = \begin{bmatrix} w & \bar{w} \end{bmatrix}^T$, this means that we are trying to maximize w, i.e. to find a control u^* steering the system to $x^*(N) = e_2^1$, if it exists.

To analyze this problem using the MP, consider the value of the switching function at time $N-1$:

$$\begin{aligned} m(N-1) &= \lambda^T(N) \ltimes L \ltimes \begin{bmatrix} 1 \\ -1 \end{bmatrix} \ltimes x^*(N-1) \\ &= r^T \ltimes L \ltimes \begin{bmatrix} 1 \\ -1 \end{bmatrix} \ltimes x^*(N-1) \\ &= \begin{bmatrix} 1 & 0 \end{bmatrix} \ltimes \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix} \ltimes \begin{bmatrix} 1 \\ -1 \end{bmatrix} \ltimes x^*(N-1) \\ &= \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix} \ltimes \begin{bmatrix} 1 \\ -1 \end{bmatrix} \ltimes x^*(N-1) \\ &= \begin{bmatrix} 1 & 0 \end{bmatrix} \ltimes x^*(N-1). \end{aligned} \tag{18}$$

We consider two cases.
Case 1. *Suppose that $x^*(N-1) = e_2^2$. It then follows from* (17) *that*

$$\begin{aligned} x^*(N) &= L \ltimes u^*(N-1) \ltimes e_2^2 \\ &= L \ltimes \begin{bmatrix} v \\ \bar{v} \end{bmatrix} \ltimes e_2^2 \\ &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} 0 & v & 0 & \bar{v} \end{bmatrix}^T \\ &= e_2^2, \end{aligned}$$

so $r^T x^*(N) = 0$.
Case 2. *Suppose that $x^*(N-1) = e_2^1$. Then $m(N-1) = 1$, so* (11) *implies that $u^*(N-1) = e_2^1$, and* (17) *yields $x^*(N) = e_2^1$. Using* (9) *yields*

$$\begin{aligned} \lambda(N-1) &= (L \ltimes u^*(N-1))^T \ltimes \lambda(N) \\ &= (L \ltimes e_2^1)^T \ltimes e_2^1 \\ &= e_2^1. \end{aligned}$$

Hence,

$$\begin{aligned} m(N-2) &= \lambda^T(N-1) \ltimes L \ltimes \begin{bmatrix} 1 \\ -1 \end{bmatrix} \ltimes x^*(N-2) \\ &= \begin{bmatrix} 1 & 0 \end{bmatrix} \ltimes x^*(N-2). \end{aligned}$$

Comparing this with (18)*, we conclude that there are two possibilities. Either $x^*(N) = e_2^2$ (and then any control is optimal) or $x^*(N) = e_2^1$ and then the (unique) optimal control is $u^*(k) = e_2^1$, for any $k \in \{0, 1 \dots, N-1\}$.*

An examination of (16) *shows that the unique optimal control is indeed $u^*(k) = e_2^1$ for $k \in \{0, 1, \dots, N-1\}$. Thus, in this example the MP provides a complete characterization of the optimal control.*

The next example demonstrates the application of Theorems 16 and 18.

Example 20 *Consider the single–input BCN*

$$\begin{aligned} x_1(k+1) &= u(k)\bar{x}_2(k), \\ x_2(k+1) &= u(k) + \bar{x}_1(k) + \bar{x}_2(k). \end{aligned} \tag{19}$$

Here $n = 2$ and $m = 1$. Suppose that the initial condition is $x_1(0) = x_2(0) = False$, and consider Problem 13 with $N = 3$ and $r = e_4^1$. In other words, the problem is to determine a control that maximizes $J(u) = (e_4^1)^T x(3)$. Intuitively, this amounts to finding a control steering the state to $x_1(3) = x_2(3) = True$, if it exists.

The algebraic state–space form is

$$\begin{aligned} x(k+1) &= L \ltimes u(k) \ltimes x(k) \\ x(0) &= e_4^4, \end{aligned} \tag{20}$$

with $L = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$.

To characterize the optimal control, we begin by calculating

$$\begin{aligned} m(2) &= \lambda^T(3) \ltimes L \ltimes \begin{bmatrix} 1 \\ -1 \end{bmatrix} \ltimes x^*(2) \\ &= r^T \ltimes L \ltimes \begin{bmatrix} 1 \\ -1 \end{bmatrix} \ltimes x^*(2) \\ &= \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix} \ltimes L \ltimes \begin{bmatrix} 1 \\ -1 \end{bmatrix} \ltimes x^*(2) \\ &= \begin{bmatrix} 0 & 1 & 0 & 1 \end{bmatrix} \ltimes x^*(2). \end{aligned} \tag{21}$$

Thus, $m(2) \geq 0$. If $m(2) > 0$, then $u^*(2) = e_2^1$. If $m(2) = 0$, then by Theorem 18 there exists an optimal control u^* with $u^*(2) = e_2^1$. Hence,

$$\begin{aligned} \lambda(2) &= (L \ltimes u^*(2))^T \lambda(3) \\ &= (L \ltimes e_2^1)^T r \\ &= \begin{bmatrix} 0 & 1 & 0 & 1 \end{bmatrix}^T, \end{aligned} \tag{22}$$

and

$$\begin{aligned} m(1) &= \lambda^T(2) \ltimes L \ltimes \begin{bmatrix} 1 \\ -1 \end{bmatrix} \ltimes x^*(1) \\ &= \begin{bmatrix} -1 & 0 & 0 & 0 \end{bmatrix} \ltimes x^*(1). \end{aligned} \tag{23}$$

Thus, $m(1) \leq 0$ and we may assume that $u^*(1) = e_2^2$. Then

$$\begin{aligned} \lambda(1) &= (L \ltimes u^*(1))^T \lambda(2) \\ &= \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}^T, \end{aligned} \tag{24}$$

so

$$\begin{aligned} m(0) &= \lambda^T(1) \ltimes L \ltimes \begin{bmatrix} 1 \\ -1 \end{bmatrix} \ltimes x(0) \\ &= \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix} \ltimes L \ltimes \begin{bmatrix} 1 \\ -1 \end{bmatrix} \ltimes e_4^4 \\ &= 1. \end{aligned} \tag{25}$$

Since $m(0) > 0$, $u^*(0) = e_2^1$. Summarizing, $\{u^*(0), u^*(1), u^*(2)\} = \{e_2^1, e_2^2, e_2^1\}$ is an optimal control. A calculation shows that the corresponding trajectory is $x^*(1) = e_4^1$, $x^*(2) = e_4^4$, $x^*(3) = e_4^1$, so this control indeed steers the system to the desired location.

VII. Conclusions

We considered a Mayer–type optimal control problem for single–input BCNs. Using the algebraic state–space formulation developed by Cheng, we derived a necessary condition for optimality in the form of a maximum principle. We also analyzed the singular case where the switching function defined in the MP is zero.

We believe that it is possible to extend the MP described here to the case of a BCN with several inputs. This topic is currently under study. Another possible topic for further research is the application of the MP to BCNs that model biological networks.

Acknowledgments

We are grateful to the anonymous reviewers and the AE for their helpful comments.

References

[1] T. Akutsu, M. Hayashida, W.-K. Ching, and M. K. Ng, "Control of Boolean networks: Hardness results and algorithms for tree structured networks," *J. Theoretical Biology*, vol. 244, pp. 670–679, 2007.

[2] M. Athans and P. L. Falb, *Optimal Control*. McGrew Hill, 1966, reprinted by Dover Publications in 2006.

[3] D. S. Bernstein, *Matrix Mathematics*. Princeton University Press, 2005.

[4] B. Bonnard and M. Chyba, *Singular Trajectories and their Role in Control Theory*. Springer, 2003.

[5] D. Cheng, "Disturbance decoupling of Boolean control networks." [Online]. Available: http://lsc.amss.ac.cn/~dcheng/preprint

[6] D. Cheng, "Input-state approach to Boolean networks," *IEEE Trans. Neural Networks*, vol. 20, pp. 512–521, 2009.

[7] D. Cheng and Y. Dong, "Semi-tensor product of matrices and its some applications to physics," *Methods Appl. Anal.*, vol. 10, pp. 565–588, 2003.

[8] D. Cheng, Z. Li, and H. Qi, "Realization of Boolean control networks," *Automatica*, vol. 46, pp. 62–69, 2010.

[9] D. Cheng and H. Qi, "Controllability and observability of Boolean control networks," *Automatica*, vol. 45, pp. 1659–1667, 2009.

[10] D. Cheng and H. Qi, "A linear representation of dynamics of Boolean networks," *IEEE Trans. Automatic Control*, vol. 55, pp. 2251–2258, 2010.

[11] D. Cheng and H. Qi, "State-space analysis of Boolean networks," *IEEE Trans. Neural Networks*, vol. 21, pp. 584–594, 2010.

[12] A. Datta, R. Pal, A. Choudhary, and E. R. Dougherty, "Control approaches for probabilistic gene regulatory networks," *IEEE Signal Processing Magazine*, vol. 24, pp. 54–63, 2010.

[13] M. H. Hassoun, *Fundamentals of Artificial Neural Networks*. MIT Press, 1995.

[14] S. A. Kauffman, "Metabolic stability and epigenesis in randomly constructed genetic nets," *J. Theoretical Biology*, vol. 22, pp. 437–467, 1969.

[15] S. A. Kauffman, *Origins of Order: Self–Organization and Selection in Evolution*. Oxford University Press, 1993.

[16] G. Langholz, J. L. Mott, and A. Kandel, *Foundations of Digital Logic Design*. World Scientific, 1998.

[17] Q. Liu, X. Guo, and T. Zhou, "Optimal control for probabilistic Boolean networks," *IET Systems Biology*, vol. 4, pp. 99–107, 2010.

[18] M. Margaliot, "Stability analysis of switched systems using variational principles: An introduction," *Automatica*, vol. 42, pp. 2059–2077, 2006.

[19] M. Margaliot and M. S. Branicky, "Nice reachability for planar bilinear control systems with applications to planar linear switched systems," *IEEE Trans. Automatic Control*, vol. 54, pp. 1430–1435, 2009.

[20] M. Margaliot and D. Liberzon, "Lie–algebraic stability conditions for nonlinear switched systems and differential inclusions," *Systems Control Lett.*, vol. 55, pp. 8–16, 2006.

[21] J. L. Schiff, *Cellular Automata: A Discrete View of the World*. Wiley-Interscience, 2008.

[22] S. P. Sethi and G. L. Thompson, *Optimal Control Theory: Applications to Management Science and Economics*, 2nd ed. Kluwer Academic Publishers, 2000.

[23] Y. Sharon and M. Margaliot, "Third-order nilpotency, finite switchings and asymptotic stability," *J. Diff. Eqns.*, vol. 233, pp. 136–150, 2007.

[24] I. Shmulevich, E. R. Dougherty, S. Kim, and W. Zhang, "Probabilistic Boolean networks: a rule-based uncertainty model for gene regulatory networks," *Bioinformatics*, vol. 18, pp. 261–274, 2002.

[25] I. Shmulevich, E. R. Dougherty, and W. Zhang, "From Boolean to probabilistic Boolean networks as models of genetic regulatory networks," *Proc. of the IEEE*, vol. 90, pp. 1778–1792, 2002.