Disturbance Decoupling of Boolean Control Networks

Daizhan Cheng, *Fellow, IEEE*

***Abstract*—Disturbance decoupling problem (DDP) of Boolean control networks is considered. Using semi-tensor product of matrices and the matrix expression of logical functions, a working procedure is proposed to solve the problem. This procedure consists of two key design steps. First, how to convert a system into an output-friendly coordinate frame. An algorithm is provided to calculate the output-friendly subspaces. Secondly, it was shown how to find proper controllers to solve the problem if it is solvable. A state variable separation form is introduced to guide the design of controllers. Based on the design technique, necessary and sufficient conditions are obtained for the solvability of DDP.**

***Index Terms*—Boolean control network, coordinate transformation, output-friendly subspace, disturbance decoupling, canalyzing Boolean mapping.**

I. Introduction

THE Boolean network, introduced firstly by Kauffman [1], has been proved to be quite useful in modeling and quantitative description of cellular regulators [2]–[4]. Kauffman mentioned that [5] "Switching Boolean networks are of central importance to the construction of a statistical mechanics over ensembles of systems and to an adequate theory of complex but ordered systems."

As for Boolean control networks, it was pointed out by [6] that "Gene-regulatory networks are defined by trans and cis logic. $\cdots$ Both of these types of regulatory networks have input and output." From here one sees easily that a Boolean network with input(s) and output(s), called a Boolean control network, is a proper way to describe the dynamics of gene-regulatory networks. "One of the major goals of systems biology is to develop a control theory for complex biological systems." [7] We refer to [7] and the references therein for the importance of the control in Boolean network (particularly in systems biology).

Recently, using semi-tensor product of matrices and the matrix expression of logical functions, a new systematic approach to analysis and control of Boolean (control) networks has been proposed. In [8], the algebraic form of a Boolean network is introduced, which provides a framework for this new approach. Using it, some formulas are obtained to calculate the fixed points, cycles, basins of the attractors, and transient periods. In [9], using the input-state analysis, the structure of attractors of a Boolean network is investigated, and so called "rolling gears" structure is proposed, which gives an explanation why "tiny attractors" can decide the "vast, vast order" as described in [10]. The controllability and observability of Boolean control systems are discussed in [11]. The canonical forms and realization problem of Boolean control networks are investigated in [12]. This series of works showed that the semi-tensor product is a powerful tool in analyzing the structure of Boolean networks and the synthesis of Boolean control networks.

To give a brief introduction to this new framework, we first introduce some notations:

- $\mathbf{1}_k := (\underbrace{1\ 1\ \cdots\ 1}_{k})^T$.
- $\mathcal{D} := \{0,1\}$, where $1 \sim T$ means "true" and $0 \sim F$ means "false". A logical variable A will take value from $\mathcal{D}$, which is expressed as $A \in \mathcal{D}$.
- δ_n^i: the i-th column of the identity matrix I_n.
- $\Delta_n := \{\delta_n^i | i = 1, \cdots, n\}$, $\Delta_2 := \Delta$.
- A matrix $B \in M_{m\times n}$ is called a Boolean matrix, if all its entries are either 0 or 1. The set of $m \times n$ Boolean matrices is denoted by $\mathcal{B}_{m\times n}$.
- A matrix $L \in M_{n\times r}$ is called a logical matrix if the columns of L, denoted by $\mathrm{Col}(L)$, are of the form of δ_n^k. That is,
$$\mathrm{Col}(L) \subset \Delta_n.$$
Denote by $\mathcal{L}_{n\times r}$ the set of $n \times r$ logical matrices.
- If $L \in \mathcal{L}_{n\times r}$, by definition it can be expressed as $L = [\delta_n^{i_1}, \delta_n^{i_2}, \cdots, \delta_n^{i_r}]$. For the sake of compactness, it is briefly denoted as
$$L = \delta_n[i_1, i_2, \cdots, i_r].$$
- $\mathrm{F}_\ell\{x_1, \cdots, x_n\}$ denotes the set of logical functions with logical arguments $x_1, \cdots, x_n$.

Throughout this paper, we assume the product of two matrices $A \in M_{m\times n}$ and $B \in M_{p\times q}$ is semi-tensor product, $\ltimes$, which is a generalization of the conventional matrix product to the case when $n \neq p$. We refer to [13] or [8] for the definition and basic properties of this product. It is worth noting that the main properties of the conventional matrix product remain true. In most cases, the symbol, $\ltimes$ is omitted.

Disturbance decoupling problem (DDP) is one of the fundamental problems in control theory either for linear systems [14] or for nonlinear systems [15]. This paper considers the DDP for Boolean control networks.

The basic idea for solving DDP of logical dynamic control systems (i.e., the Boolean control networks), proposed in this paper, is dividing the problem into two steps: Step 1, finding a coordinate transformation, such that in the new coordinate

This work was supported in part by the National Natural Science Foundation (NNSF) of China under Grants 60674022, 60736022, and 60821091.

The author is with the Key Laboratory of Systems and Control, Academy of Mathematics and Systems Sciences, Chinese Academy of Sciences, Beijing 100190, P.R.China. (E-mail: dcheng@iss.ac.cn)

frame the outputs are only involved in a minimum set of coordinates; Step 2, isolating the dynamical equations of the coordinate variables, which are output related, and under the decomposed form try to find a proper (open-loop or state feedback) control such that this part of dynamic equations are disturbance independent.

The rest of the paper is organized as follows: Section 2 gives some preliminaries, including the algebraic expression of the dynamics of a Boolean network, and the coordinate transformation of Boolean dynamics. The DDP formulation is given in Section 3. Section 4 presents a method to find the output-friendly subspaces, which provides a tool to convert the dynamics of a Boolean network into an output localized form. Based on the output localized form obtained in Section 4, a design technique for controls to solve DDP is proposed in Section 5. Combining these two step works yields a necessary and sufficient condition for the solvability of DDP. Section 6 presents an illustrative example to describe the proposed technique. Section 7 is a brief conclusion.

II. Preliminary

A logical variable, $x \in \mathcal{D}$ means x can be either 0 or 1. A logical function, $f(x_i, \cdots, x_n)$ is a mapping: $\mathcal{D}^n \to \mathcal{D}$. A logical mapping $F : \mathcal{D}^n \to \mathcal{D}^k$ is defined by k logical functions

$$y_i = f_i(x_1, \cdots, x_n), \quad i = 1, \cdots, k. \tag{1}$$

Denote $X = (x_1, \cdots, x_n)^T$ and $Y = (y_1, \cdots, y_k)^T$. Then we simply denote (1) by

$$Y = F(X), \quad X \in \mathcal{D}^m,\ Y \in \mathcal{D}^k. \tag{2}$$

To use matrix expression in logic, we identify $1 \sim \delta_2^1$ and $0 \sim \delta_2^2$, equivalently, $\mathcal{D} \sim \Delta$. Then we can equivalently consider the mapping $F : \mathcal{D}^n \to \mathcal{D}^k$ as a mapping $F : \Delta^n \to \Delta^k$. Using vector form, we denote $x = \ltimes_{i=1}^n x_i \in \Delta_{2^n}$ and $y = \ltimes_{i=1}^k y_i \in \Delta_{2^k}$. Then we have the following result [9], [11].

Theorem II.1. *Consider a logical mapping $F : \mathcal{D}^n \to \mathcal{D}^k$ defined by (1). There is a unique matrix $M_F \in \mathcal{L}_{2^k \times 2^n}$, called the structure matrix of the mapping F, such that in vector form F can be expressed as*

$$y = M_F x. \tag{3}$$

(3) is called the algebraic form of the logical mapping F.

It has also been proved that for the mapping F, its logical form (1) (or (2)) is equivalent to its algebraic form (3) and some easily computable formulas have been provides to convert one form to the other one.

A Boolean network consists of n nodes. Each node can take values from $\mathcal{D}$ at a time instance of $0, 1, 2, \cdots$ according to certain logical rules. The network dynamics can be described by a set of discrete time logical dynamic equations as

$$\begin{cases} x_1(t+1) = f_1(x_1(t), \cdots, x_n(t)) \\ \vdots \\ x_n(t+1) = f_n(x_1(t), \cdots, x_n(t)), \end{cases} \tag{4}$$

where f_i, $i = 1, \cdots, n$ are logical functions. Briefly, we can express it as

$$X(t+1) = F(X(t)), \tag{5}$$

where $F : \mathcal{D}^n \to \mathcal{D}^n$ is a logical mapping.

Set $x = \ltimes_{i=1}^n x_i$. Using Theorem II.1, we denote by $L \in \mathcal{L}_{2^n \times 2^n}$ the structure matrix of F. Then the algebraic form of (4) (or (5)) becomes

$$x(t+1) = Lx(t). \tag{6}$$

If in addition to the structure of a Boolean network, there are some inputs and outputs adding to the network, the Boolean network becomes a Boolean control network. In general, its dynamics can be expressed as

$$\begin{cases} x_1(t+1) = f_1(x_1(t), \cdots, x_n(t), u_1(t), \cdots, u_m(t)) \\ \vdots \\ x_n(t+1) = f_n(x_1(t), \cdots, x_n(t), u_1(t), \cdots, u_m(t)), \\ y_j(t) = h_j(x_1(t), \cdots, x_n(t)), \quad j = 1, \cdots, p. \end{cases} \tag{7}$$

where $u_i(t) \in \mathcal{D}$, $i = 1, \cdots, m$ are controls, $y_j(t) \in \mathcal{D}$, $j = 1, \cdots, p$ are outputs. Similar to the Boolean network, there are a unique $L \in \mathcal{L}_{2^n \times 2^{n+m}}$ and a unique $H \in \mathcal{L}_{2^p \times 2^n}$, such that the algebraic form of (7) is

$$\begin{cases} x(t+1) = Lu(t)x(t) \\ y = Hx(t), \end{cases} \tag{8}$$

where $x = \ltimes_{i=1}^n x_i$, $u = \ltimes_{i=1}^m u_i$, and $y = \ltimes_{i=1}^p y_i$.

We give some examples to illustrate these.

Example II.2. (i) *A Boolean network consists of four nodes A, B, C, D, with the dynamics as*

$$\begin{cases} A(t+1) = B(t) \\ B(t+1) = C(t) \\ C(t+1) = D(t) \wedge B(t) \\ D(t+1) = \neg C(t). \end{cases} \tag{9}$$

Setting $x = A \ltimes B \ltimes C \ltimes D$, its algebraic form is

$$x(t+1) = Lx(t), \tag{10}$$

where

$$L = \delta_{16}[2, 4, 5, 7, 12, 12, 15, 15, 2, 4, 5, 7, 12, 12, 15, 15].$$

(ii) *Adding two inputs and one output to (9), we have a Boolean control network as*

$$\begin{cases} A(t+1) = B(t) \\ B(t+1) = C(t) \vee u_1(t) \\ C(t+1) = D(t) \wedge [B(t) \vee u_1(t)] \\ D(t+1) = \neg C(t) \vee u_2(t), \\ y(t) = C(t) \wedge D(t). \end{cases} \tag{11}$$

Its algebraic form is

$$\begin{cases} x(t+1) = Lu(t)x(t) \\ y(t) = Hx(t), \end{cases} \tag{12}$$

where

$$\begin{aligned} L=\delta_{16}[&1,3,1,3,9,11,9,11,1,3,1,3,9,11,9,11,\\ &2,4,1,3,10,12,9,11,2,4,1,3,10,12,9,11,\\ &1,3,5,7,11,11,15,15,1,3,5,7,11,11,15,15,\\ &2,4,5,7,12,12,15,15,2,4,5,7,12,12,15,15]; \end{aligned}$$

and

$$H=\delta_2[1,2,2,2,1,2,2,2,1,2,2,2,1,2,2,2].$$

Details about the related calculations were given in [8].[1]

The coordinate transformation (or coordinate change) of a logical dynamics was firstly introduced in [12].

Definition II.3. *Let $(x_1,\cdots,x_n)$ be the state variables of a Boolean (control) network. A mapping $F:\mathcal{D}^n\to\mathcal{D}^n$ is said to be a (logical) coordinate transformation, if F is a bijective mapping.*

The following theorem shows how to construct a coordinate transformation.

Theorem II.4 ([12]). *The mapping $F:\mathcal{D}^n\to\mathcal{D}^n$, is a coordinate transformation, iff the structure matrix of F, $M_F\in\mathcal{L}_{2^n\times 2^n}$ is nosingular.*

Definition II.5. *Let $y_1,\cdots,y_k\in\mathcal{X}:=\mathrm{F}_\ell\{x_1,\cdots,x_n\}$. $\mathcal{V}=\{y_1,\cdots,y_k\}$ is said to be a k-dimensional regular subspace with regular sub-basis $\{y_1,\cdots,y_k\}$ if there exist $y_{k+1},\cdots,y_n\in\mathcal{X}$ such that $\{x_i|i=1,\cdots,n\}\to\{y_i|i=1,\cdots,n\}$ is a coordinate transformation.*

Assume

$$y_i=f_i(x_1,\cdots,x_n),\quad i=1,\cdots,k \tag{13}$$

is a set of logical functions. Denote $x=\ltimes_{i=1}^n x_i$ and $y=\ltimes_{i=1}^k y_i$. Then (13) has unique algebraic expression as

$$y=T_0x, \tag{14}$$

where $T_0\in\mathcal{L}_{2^k\times 2^n}$. Denote by $T_0=(t_{i,j})$, i.e., $t_{i,j}$ is the (i,j)-th element of T_0. Then we have

Theorem II.6 ([12]). *The set of logical functions $\{y_1,\cdots,y_k\}\in\mathcal{X}$, defined by (13), is a regular sub-basis, iff the elements of T_0 in (14) satisfies*

$$\sum_{j=1}^{2^n}t_{i,j}=2^{n-k},\quad i=1,\cdots,2^k. \tag{15}$$

A sub-basis becomes a basis when $k=n$. For constructing a basis or a sub-basis, (15) may not be very convenient. In the following we will provide a new equivalent form. For this purpose, we need

Theorem II.7 ([12]). *Assume $y=\ltimes_{i=1}^p y_i$ and $z=\ltimes_{j=1}^q z_j$, where y_i and z_j are all logical functions of $\{x_1,\cdots,x_n\}$. Moreover, the algebraic forms of y and z are expressed respectively as*

$$y=Mx,\quad z=Nx,$$

where $M\in\mathcal{L}_{2^p\times 2^n}$ and $N\in\mathcal{L}_{2^q\times 2^n}$. Assume their product $w=yz$ has its algebraic form as $w=Wx$. Then $W\in\mathcal{L}_{2^{p+q}\times 2^n}$, satisfies

$$W_i=M_iN_i,\quad i=1,\cdots,2^n, \tag{16}$$

where W_i, M_i, and N_i are the i-th columns of W, M, and N respectively.

Consider $\{y_1,\cdots,y_k\}$ in (13) again. Assume their algebraic forms are

$$y_i=\delta_2[\alpha_1^i,\alpha_2^i,\cdots,\alpha_{2^n}^i]x,\quad i=1,\cdots,k. \tag{17}$$

Then we construct a Boolean matrix as

$$B_y=\begin{bmatrix} a_1^1 & a_2^1 & \cdots & a_{2^n}^1\\ a_1^2 & a_2^2 & \cdots & a_{2^n}^2\\ \vdots & & & \\ a_1^k & a_2^k & \cdots & a_{2^n}^k \end{bmatrix}\in\mathcal{B}_{k\times 2^n}, \tag{18}$$

where $a_j^i=\alpha_j^i \pmod{2}$.

Note that $\mathrm{Col}(B_y)\subset\mathcal{B}_{k\times 1}:=\mathcal{B}_k$. In fact, $\mathcal{B}_k$ consists of 2^k elements, which are

$$\beta_1=\begin{bmatrix}0\\ \vdots\\ 0\\ 0\end{bmatrix},\quad \beta_2=\begin{bmatrix}0\\ \vdots\\ 0\\ 1\end{bmatrix},\quad \beta_3=\begin{bmatrix}0\\ \vdots\\ 1\\ 0\end{bmatrix},\quad \cdots\quad \beta_{2^k}=\begin{bmatrix}1\\ \vdots\\ 1\\ 1\end{bmatrix}.$$

The following result is very convenient in constructing a basis or regular sub-basis.

Theorem II.8. *The set $\{y_1,\cdots,y_k\}$, defined by (13), forms a regular sub-basis, iff the numbers of each possible type of columns of B_y, defined in (18), are the same, which is 2^{n-k}. That is, there are 2^{n-k} columns which are equal to β_s, $s=1,\cdots,2^k$.*

Proof: Using Theorem II.7, it is easy to see that the i-th column of T_0, denoted by T_0^i, satisfies

$$T_0^i=\ltimes_{j=1}^k\delta_2^{\alpha_i^j},\quad i=1,\cdots,2^n.$$

Define a mapping from k-dimensional Boolean vector $\mathcal{B}_k$ to Δ_{2^k} by $\Phi:(a_1,\cdots,a_k)\mapsto\ltimes_{j=1}^k\delta_2^{\alpha_j}$, where

$$\alpha_j=\begin{cases}1, & a_j=1,\\ 2, & a_j=0.\end{cases}$$

Then it is easy to check that Φ is a one-to-one and onto mapping. Now note that (15) implies that there are 2^{n-k} columns, which are equal to $\delta_{2^k}^i$, $i=1,\cdots,2^k$. The conclusion follows. ∎

An immediate consequence is the following, which is very convenient in constructing logical coordinate transformation:

Corollary II.9. *Let*

$$y_i=\delta_2[\alpha_1^i,\alpha_2^i,\cdots,\alpha_{2^n}^i]x,\quad i=1,\cdots,n.$$

Then $F:\{x_1,\cdots,x_n\}\mapsto\{y_1,\cdots,y_n\}$ is a coordinate transformation, iff its Boolean matrix (18) consists of all different columns.

[1] A toolbox for all the related computations is available at http://lsc.amss.ac.cn/~dcheng/

III. Problem Formulation

Assume that in a Boolean control network there are some disturbance inputs, then we have a disturbed Boolean control network. In general, its dynamics is described as

$$\begin{cases} x_1(t+1) = f_1(x_1(t), \cdots, x_n(t), u_1(t), \cdots, u_m(t), \\ \qquad \xi_1(t), \cdots, \xi_q(t)) \\ \vdots \\ x_n(t+1) = f_n(x_1(t), \cdots, x_n(t), u_1(t), \cdots, u_m(t), \\ \qquad \xi_1(t), \cdots, \xi_q(t)), \\ y_j(t) = h_j(x(t)), \quad j = 1, \cdots, p, \end{cases} \tag{19}$$

where $\xi_i(t)$, $i = 1, \cdots, q$ are disturbances. Let $x(t) = \ltimes_{i=1}^n x_i(t)$, $u(t) = \ltimes_{i=1}^m u_i(t)$, $\xi(t) = \ltimes_{i=1}^q \xi_i(t)$, and $y(t) = \ltimes_{i=1}^p y_i(t)$. Then the algebraic form of (19) is expressed as

$$\begin{cases} x(t+1) = Lu(t)\xi(t)x(t), \\ y(t) = Hx(t), \end{cases} \tag{20}$$

where $L \in \mathcal{L}_{2^n \times 2^{n+m+q}}$, $H \in \mathcal{L}_{2^p \times 2^n}$.

We consider the following example.

Example III.1. *A disturbed Boolean control network is defined by the following equation:*

$$\begin{cases} A(t+1) = B(t) \wedge \xi(t) \\ B(t+1) = C(t) \vee u_1(t) \\ C(t+1) = D(t) \wedge [(B(t) \to \xi(t)) \vee u_1(t)] \\ D(t+1) = \neg C(t) \vee [\xi(t) \wedge u_2(t)], \\ \\ y(t) = C(t) \wedge D(t). \end{cases} \tag{21}$$

Roughly speaking, the disturbance decoupling problem is to find suitable controls such that for the closed-loop system the outputs are not affected by the disturbances.

Consider system (21). If we choose controllers as

$$u_1(t) = B(t), \quad u_2(t) = 0,$$

Then the closed-loop system becomes

$$\begin{cases} A(t+1) = B(t) \wedge \xi(t) \\ B(t+1) = C(t) \vee B(t) \\ C(t+1) = D(t) \\ D(t+1) = \neg C(t), \\ \\ y(t) = C(t) \wedge D(t). \end{cases} \tag{22}$$

It is obvious that the disturbance will not affect the output.

We give a rigorous definition:

Definition III.2. *Consider system (19). The DDP is solvable, if we can find a feedback control*

$$u(t) = \phi(x(t)), \tag{23}$$

a coordinate transformation $z = T(x)$, such that under z coordinate frame the closed-loop system becomes

$$\begin{cases} z^1(t+1) = F^1(z(t), \phi(x(t)), \xi(t)) \\ z^2(t+1) = F^2(z^2(t)) \\ y(t) = G(z^2(t)). \end{cases} \tag{24}$$

From Definition III.2 one sees that to solve the DDP problem there are two key issues: (i) finding a regular coordinate subspace z^2, which contains outputs; (ii) designing a control, such that the complement coordinate sub-basis z^1 and the disturbances ξ can be deleted from the dynamics of z^2. In the following two sections they will be investigated one by one.

IV. Y-Friendly Subspace

Definition IV.1. *Let $\mathcal{X} = \mathrm{F}_\ell\{x_1, \cdots, x_n\}$ be the state space, and $Y = \{y_1, \cdots, y_p\} \subset \mathcal{X}$. A regular subspace $\mathcal{Z} \subset \mathcal{X}$ is called a Y-friendly (or output-friendly) subspace, if $y_i \in \mathcal{Z}$, $i = 1, \cdots, p$. A Y-friendly subspace of minimum dimension is called a minimum Y-friendly subspace.*

This section devotes to finding output-friendly subspaces.

First, we consider one variable y. Since $y \in \mathcal{X}$, we have its algebraic expression as

$$y = \delta_2[i_1, i_2, \cdots, i_{2^n}]x := Hx. \tag{25}$$

Denote

$$n_j = |\{k | i_k = j, 1 \le k \le 2^n\}|, \quad j = 1, 2,$$

where $|\cdot|$ is the cardinal number of the set. Then we have the following.

Lemma IV.2. *Assume $Y = \{y\}$ has its algebraic form (25). There is a Y-friendly subspace of dimension r, iff n_1 and n_2 have a common factor 2^{n-r}.*

Proof: (Necessary) Assume there is a Y-friendly subspace $\mathcal{Z} = \mathrm{F}_\ell\{z_1, \cdots, z_r\}$ with $\{z_1, \cdots, z_r\}$ as its regular sub-basis. Denote $z = \ltimes_{i=1}^r z_i$. Then

$$z = T_0 x = (t_{i,j})x,$$

where $T_0 \in \mathcal{L}_{2^r \times 2^n}$. Since $y \in \mathcal{Z}$, we have

$$y = Gz = GT_0 x,$$

where $G \in \mathcal{L}_{2 \times 2^r}$. So G can be expressed as

$$G = \delta_2[j_1, \cdots, j_{2^r}].$$

Hence

$$H = \delta_2[i_1, i_2, \cdots, i_{2^n}] = \delta_2[j_1, \cdots, j_{2^r}]T_0.$$

Denote by $m_s = |\{k | j_k = s, 1 \le k \le 2^r\}|$, $s = 1, 2$. Using Theorem II.6, a straightforward computation shows that h has $2^{n-r}m_1$ columns, which are equal to δ_2^1 and $2^{n-r}m_2$ columns, which are equal to δ_2^2. That is, $n_1 = 2^{n-r}m_1$ and $n_2 = 2^{n-r}m_2$. The conclusion follows.

(Sufficiency) Let $y = Hx$ be as in (25), where $n_1 = 2^{n-r}m_1$ columns of H equal to δ_2^1 and $n_2 = 2^{n-r}m_2$ columns equal to δ_2^2. It suffices to construct a Y-friendly subspace, which is of dimension r. We construct a logical matrix $T_0 \in \mathcal{L}_{2^r \times 2^n}$ as follows. Let $J_1 = \{k | H_k = \delta_2^1\}$ and $J_2 = \{k | H_k = \delta_2^2\}$, where H_k is the k-th column of H. Simply letting $I_1 = \{1, \cdots, m_1\}$, and $I_2 = \{m_1+1, \cdots, 2^r\}$, we can split T_0 into 2×2 minors as: $T_0^{i,j} = \{t_{r,s} | r \in I_i$ and $s\in J_j\}$, $i,j=1,2$. We set them to be

$$\begin{array}{l} T_0^{1,1} = I_{m_1}\otimes \mathbf{1}^T_{2^{n-r}}; \quad T_0^{2,2} = I_{m_2}\otimes \mathbf{1}^T_{2^{n-r}};\\ T_0^{1,2} = 0; \quad T_0^{2,1} = 0. \end{array}$$

Now it is ready to verify that the T_0, constructed in this way, satisfies (15). According to Theorem II.6, $z = T_0 x$ forms a regular sub-basis.

Next, we define G as

$$G = \delta_2[\underbrace{1,\cdots,1}_{m_1},\underbrace{2,\cdots,2}_{m_2}].$$

A straightforward computation shows that $GT_0 = H$, which means

$$GT_0x = Hx = y.$$

■

For statement ease, we call a factor of the form 2^s the 2-type factor. In sub-basis construction, only 2-type factors are concerned.

From the proof of the Lemma IV.2 the following result is obvious.

Corollary IV.3. *Assume 2^{n-r} is the largest common 2-type factor of n_1 and n_2. Then the minimum Y-friendly subspace is of dimension r.*

Next, we consider the multi-output case. Let $Y = \{y_1,\cdots,y_p\}\subset \mathcal{X}$ be p logical functions, and denote $y = \ltimes_{i=1}^p y_i$. Then y can be expressed in its algebraic form as

$$y = \delta_{2^p}[i_1, i_2, \cdots, i_{2^n}]x := Hx. \tag{26}$$

Denote by

$$n_j = |\{k | i_k = j, 1\le k\le 2^n\}|, \quad j = 1,\cdots,2^p.$$

Using the same argument as for the single function case, it is easy to prove the following result. (In fact, the following Algorithm IV.5 could be considered as a constructive proof.)

Theorem IV.4. *Assume $y = \ltimes_{i=1}^p y_i$ has its algebraic form (26).*

1) *There is a Y-friendly subspace of dimension r, iff n_j, $j=1,\cdots,2^p$ have a common factor 2^{n-r}.*
2) *Assume 2^{n-r} is the largest common 2-type factor of n_j, $j=1,\cdots,2^p$. Then the minimum Y-friendly subspace is of dimension r.*

We give an algorithm for constructing a Y-friendly subspace. Assume 2^{n-r} is a common factor of n_i, denote by $n_i = m_i\cdot 2^{n-r}$, $i=1,\cdots,2^p$. We split the set of $\mathrm{Col}(H)$ into 2^p subsets as J_j, $j=1,\cdots,2^p$. $k\in J_j$, iff the k-th column of H satisfies $H_k = \delta_{2^p}^j$. To construct the required Y-friendly subspace is equivalent to construct a logical matrix $T_0\in \mathcal{L}_{2^r\times 2^n}$, such that we can find a logical matrix $G\in\mathcal{L}_{2^p\times 2^r}$, satisfying

$$GT_0 = H.$$

Algorithm IV.5.

- *Step 1. Split the rows of T_0 into 2^p blocks in such a way: I_1 consists of the first m_1 rows, I_2 consists of the following m_2 rows, and so on till I_{2^p} consists of the last m_{2^p} rows. (Note that $\sum_{i=1}^{2^p} m_i = 2^r$.) Partition T_0 into $2^p\times 2^p$ minors as*

$$T_0^{i,j} = \{t_{r,s} | r\in I_i, s\in J_j\}, \quad i,j = 1,\cdots,2^p.$$

- *Step 2. Note that $T_0^{i,j}$ is an $m_i\times(m_j 2^{n-r})$ minor. Set it as*

$$T_0^{i,j} = \begin{cases} I_{m_i}\otimes \mathbf{1}^T_{2^{n-r}}, & i=j\\ 0, & \text{otherwise}. \end{cases} \tag{27}$$

- *Step 3. Set*

$$z = \ltimes_{i=1}^r z_i := T_0 x.$$

Recover z_i, $i=1,\cdots,r$ from z. (We refer to [12] for recovering technique.)

Proposition IV.6. *Assume 2^{n-r} is a common factor of n_i. Then the z_i, $i=1,\cdots,r$, obtained from Algorithm IV.5 form a regular sub-basis of r dimensional Y-friendly subspace.*

Proof: Define a block diagonal matrix

$$G = \begin{bmatrix} \mathbf{1}^T_{m_1} & 0 & \cdots & 0\\ 0 & \mathbf{1}^T_{m_2} & \cdots & 0\\ \vdots & & & \\ 0 & 0 & \cdots & \mathbf{1}^T_{m_{2^p}} \end{bmatrix}. \tag{28}$$

By the construction of T_0, it is ready to check that

$$y = GT_0x = Gz.$$

■

We are particularly interested in constructing the minimum Y-friendly subspace. We give an example to describe how to construct it.

To present this example we need the following theorem from [12], which can recover a logical function from its structure matrix into its conjunctive normal form [16].

Theorem IV.7 ([12]). *Assume y is a logical function of $x_1,\cdots,x_n$, as*

$$y = f(x_1,\cdots,x_n), \tag{29}$$

and $M_f\in\mathcal{L}_{2\times 2^n}$ is the structure matrix of f. Split M_f into two equal parts as $M_f = [M_1, M_2]$. Then y can be expressed as

$$y = (x_1\wedge \phi_1(x_2,\cdots,x_n))\vee(\neg x_1\wedge\phi_2(x_2,\cdots,x_n)), \tag{30}$$

where ϕ_i has M_i as its structure matrix, $i=1,2$.

Note that using the decomposition formula (30) to ϕ_1 and ϕ_2, we can separate x_2 out. Continuing this procedure, we can finally recover the logical function f from its structure matrix.

Example IV.8. *Let $\mathcal{X} = \mathrm{F}_\ell\{x_1,x_2,x_3,x_4\}$.*

$$\begin{array}{l} y_1 = f_1(x_1,x_2,x_3,x_4) = (x_1\leftrightarrow x_3)\wedge(x_2\bar{\vee}x_4),\\ y_2 = f_2(x_1,x_2,x_3,x_4) = x_1\wedge x_3 \end{array} \tag{31}$$

We look for the minimum Y-friendly subspace. Setting $y = y_1 \ltimes y_2$ and $x = \ltimes_{i=1}^4 x_i$, it is easy to calculate that [8] $y = Mx$, where

$$M = \delta_4[3,1,4,4,1,3,4,4,4,4,4,2,4,4,2,4].$$

From M one sees easily that $n_1 = n_2 = n_3 = 2$ and $n_4 = 10$. Since the only common 2-type factor is $2 = 2^{n-r}$, we can have the minimum Y-friendly subspace of dimension $r = 3$. To construct T_0 we have:

$$J_1 = \{2,5\}; \quad J_2 = \{12,15\}; \quad J_3 = \{1,6\};$$
$$J_4 = \{3,4,7,8,9,10,11,13,14,16\}.$$

Now since $m_1 = m_2 = m_3 = 1$, and $m_4 = 5$, then $I_1 = \{1\}$, $I_2 = \{2\}$, $I_3 = \{3\}$, and $I_4 = \{4,5,6,7,8\}$. Setting $B^{1,1}$ as $\mathbf{1}_2^T$ yields that the 2nd and 5th columns of T_0 are equal to δ_8^1. Similarly, the 12th and 15th columns are equal to δ_8^2, etc. Finally, T_0 is obtained as

$$T_0 = \delta_8[3,1,4,4,1,3,5,5,6,6,7,2,7,8,2,8].$$

Correspondingly, we can construct G by formula (28) as

$$G = \delta_4[1,2,3,4,4,4,4,4]. \tag{32}$$

Finally, we construct the minimum Y-friendly subspace, say, it has a sub-basis as $\{z_1, z_2, z_3\}$. Setting $z = \ltimes_{i=1}^3 z_i$, we have

$$z = T_0 x.$$

Denote $z_i := E_i x$, $i = 1,2,3$. Then the structure matrices E_i can be uniquely calculated from T_0 as (We refer to [12] for the formulas.)

$$E_1 = \delta_2[1,1,1,1,1,1,2,2,2,2,2,1,2,2,1,2];$$
$$E_2 = \delta_2[2,1,2,2,1,2,1,1,1,1,1,2,1,2,2,1,2];$$
$$E_3 = \delta_2[1,1,2,2,1,1,1,1,2,2,1,2,1,2,2,2].$$

Then we can use Theorem IV.7 to find the logical expression of z_i from its structure matrix E_i. It is easy to calculate that

$$z_1 = \{x_1 \wedge [x_2 \vee (\neg x_2 \wedge x_3)]\} \vee \{\neg x_1 \wedge ([x_2 \wedge \neg(x_3 \vee x_4)] \vee [\neg x_2 \wedge (\neg x_3 \wedge x_4)])\};$$
$$z_2 = \{x_1 \wedge [(x_2 \wedge (x_3 \wedge \neg x_4)) \vee (\neg x_2 \wedge (x_3 \to x_4))]\} \vee \{\neg x_1 \wedge [(x_2 \wedge (x_3 \vee (\neg x_3 \wedge \neg x_4))) \vee (\neg x_2 \wedge (\neg x_3 \wedge x_4))]\};$$
$$z_3 = \{x_1 \wedge [(x_2 \wedge x_3) \vee \neg x_2]\} \vee \{\neg x_1 \wedge [(x_2 \wedge (\neg x_3 \wedge x_4)) \vee (\neg x_2 \wedge (x_3 \wedge x_4))]\}.$$

Similarly, from (32) we can easily calculate that

$$y_1 = \delta_2[1,1,2,2,2,2,2,2]z;$$
$$y_2 = \delta_2[1,2,1,2,2,2,2,2]z.$$

It ia easy to check that

$$\begin{aligned} y_1 &= z_1 \wedge z_2 \\ y_2 &= z_1 \wedge z_3. \end{aligned} \tag{33}$$

V. CONTROL DESIGN

In previous section the problem of finding a Y-friendly subspace was investigated. Assume a Y-friendly subspace is obtained as z^2. Then we can find z^1, such that $z = \{z^1, z^2\}$ form a new coordinate frame. Under z the system (19) can be expressed as

$$\begin{cases} z^1(t+1) = F^1(z(t), u(t), \xi(t)) \\ z^2(t+1) = F^2(z(t), u(t), \xi(t)) \\ y(t) = G(z^2(t)). \end{cases} \tag{34}$$

Comparing it with (24), one sees that solving DDP becomes finding $u(t) = u(z(t))$ such that

$$F^2(z(t), u(z(t)), \xi(t)) = \tilde{F}^2(z^2(t)). \tag{35}$$

Assume $z^2 = (z_1^2, \cdots, z_k^2)$ is of dimension k. We define a set of functions as

$$\begin{aligned} &e_1(z^2) = z_1^2 \wedge z_2^2 \wedge \cdots \wedge z_k^2; \\ &e_2(z^2) = z_1^2 \wedge z_2^2 \wedge \cdots \wedge \neg z_k^2; \\ &e_3(z^2) = z_1^2 \wedge \cdots \wedge \neg z_{k-1}^2 \wedge z_k^2; \\ &e_4(z^2) = z_1^2 \wedge \cdots \wedge \neg z_{k-1}^2 \wedge \neg z_k^2; \\ &\vdots \\ &e_{2^k}(z^2) = \neg z_1^2 \wedge \neg z_2^2 \wedge \cdots \wedge \neg z_k^2. \end{aligned}$$

Using Theorem IV.7, each equation of F^2, denoted by F_i^2, can be expressed as

$$F_j^2(z(t), u(t), \xi(t)) = \vee_{i=1}^{2^k}[e_i(z^2(t)) \wedge Q_j^i(z^1(t), u(t), \xi(t))], \quad j = 1, \cdots, k. \tag{36}$$

Proposition V.1. *$F^2(z(t), u(t), \xi(t)) = F^2(z^2(t))$, iff in the expression (36)*

$$Q_j^i(z^1(t), u(t), \xi(t)) = \text{const.}, \quad j = 1, \cdots, k; \; i = 1, \cdots, 2^p. \tag{37}$$

Proof: Sufficiency is trivial. As for the necessity, assume for a special pair i, j the Q_i^j is not constant. Consider the corresponding e_i. If its factor about z_s^2 is z_s^2, set $z_s^2 = 1$, and if its factor about z_s^2 is $\neg z_s^2$, set $z_s^2 = 0$, $s = 1, \cdots, k$. Then we have

$$e_i(z^2) = 1, \quad e_j(z^2) = 0, \; j \neq i.$$

Now since Q_i^j is not constant, when $Q_i^j = 1$, we have $F_i^2 = 1$, and when $Q_i^j = 0$, we have $F_i^2 = 0$. So for fixed z^2, F_i^2 can have different values, which means F_i^2 is not a function with arguments of z^2 only. ∎

Now we are ready to give the condition for the solvability of DDP. Summarizing the above argument, the following result is obvious.

Theorem V.2. *Consider system (19). The DDP is solvable, iff*

(i) there exists an output-friendly coordinate sub-basis, such that using this sub-basis the system is expressed into (34);

(ii) in (34) when F^2 is expressed as in (36), there exists feedback control $u(t) = u(z(t))$ such that (37) is satisfied.

Before ending this section, we consider the problem of solving DDP by constant controls.

Definition V.3. *A mapping $F:\mathcal{D}^n\to\mathcal{D}^p$ determined by*

$$y_j=f_j(x_1,\cdots,x_n),\quad j=1,\cdots,p$$

is called a canalizing Boolean mapping (CBM) if there exist a proper subset $\Lambda=\{\lambda_1,\cdots,\lambda_k\}\subset\{1,\cdots,n\}$ and $u_1,\cdots,u_k;v_1,\cdots,v_p\in\{0,1\}$ such that

$$f_j(x_1,\cdots,x_n)|_{x_{\lambda_i}=u_i,\ i=1,\cdots,k}=v_j,\quad j=1,\cdots,p. \tag{38}$$

If (38) holds, x_λ, $\lambda\in\Lambda$ are called the canalizing variables with canaling values $u=(u_1,\cdots,u_k)$ and canalized values $v=(v_1,\cdots,v_p)$. $F=(f_1,\cdots,f_p)$ is said to be a (u,v)-type CBM.

Note that when $k=1$ and $p=1$ the CBM becomes a standard canalizing Boolean function [17], which is important for genomic regulatory systems [5].

Define a mapping: $Q:\Delta_2^{n-k+m+q}\to\Delta_2^{p\times 2^k}$ as

$$Q(z^1(t),u(t),\xi(t))=[Q_1^1,\cdots,Q_1^{2^k},\cdots,Q_k^1,\cdots,Q_k^{2^k}]^T. \tag{39}$$

Then our purpose is to choose u such that Q_i^j, $i=1,\cdots,k$, $j=1,\cdots,2^k$ are constant. We have the following result.

Theorem V.4. *Consider system (19). The DDP is solvable by constant controls, iff*

(i) there exists an output-friendly coordinate sub-basis, and using this sub-basis the system is expressed into output-friendly form (34);

(ii) the mapping Q defined in (39) is a CBM with $u(t)$ as the canalizing variables.

We refer to [18] for verifying CBM and the technique of design of constant controllers.

VI. An Illustrative Example

Consider the following system

$$\begin{cases}x_1(t+1)=x_4(t)\bar{\vee}u_1(t)\\ x_2(t+1)=(x_2(t)\bar{\vee}x_3(t))\wedge\neg\xi(t)\\ x_3(t+1)=[(x_2(t)\leftrightarrow x_3(t))\vee\xi(t)]\bar{\vee}[(x_1\leftrightarrow x_5)\vee u_2(t)]\\ x_4(t+1)=[u_1(t)\to(\neg x_2(t)\vee\xi(t))]\wedge(x_2(t)\leftrightarrow x_3(t))\\ x_5(t+1)=(x_4(t)\bar{\vee}u_1(t))\leftrightarrow[(u_2(t)\wedge\neg x_2(t))\vee x_4(t)]\\ \\ y(t)=x_4(t)\wedge(x_1(t)\leftrightarrow x_5(t)),\end{cases} \tag{40}$$

where $u_1(t),u_2(t)$ are controls, $\xi(t)$ is a disturbance, $y(t)$ is the output.

Setting $x(t)=\ltimes_{i=1}^5x_i(t)$, $u=u_1(t)\ltimes u_2(t)$, we express (40) into it algebraic form as

$$\begin{cases}x(t+1)=Lu(t)\xi(t)x(t)\\ y(t)=Hx(t),\end{cases} \tag{41}$$

where

$$\begin{aligned}L=\delta_{32}[&30,30,14,14,32,32,16,16,32,32,15,15,30,30,13,13,\\ &30,30,14,14,32,32,16,16,32,32,15,15,30,30,13,13,\\ &32,32,16,16,20,20,4,4,20,20,3,3,30,30,13,13,\\ &32,32,16,16,20,20,4,4,20,20,3,3,30,30,13,13,\\ &30,26,14,10,32,28,16,12,32,28,16,12,30,26,14,10,\\ &26,30,10,14,28,32,12,16,28,32,12,16,26,30,10,14,\\ &32,28,16,12,20,24,4,8,20,24,4,8,30,26,14,10,\\ &28,32,12,16,24,20,8,4,24,20,8,4,26,30,10,14,\\ &13,13,29,29,15,15,31,31,15,15,32,32,13,13,30,30,\\ &13,13,29,29,15,15,31,31,15,15,32,32,13,13,30,30,\\ &13,13,29,29,3,3,19,19,3,3,20,20,13,13,30,30,\\ &13,13,29,29,3,3,19,19,3,3,20,20,13,13,30,30,\\ &13,9,29,25,15,11,31,27,15,11,31,27,13,9,29,25,\\ &9,13,25,29,11,15,27,31,11,15,27,31,9,13,25,29,\\ &13,9,29,25,3,7,19,23,3,7,19,23,13,9,29,25,\\ &9,13,25,29,7,3,23,19,7,3,23,19,9,13,25,29];\end{aligned}$$

$$\begin{aligned}H=\delta_2[&1,2,2,2,1,2,2,2,1,2,2,2,1,2,2,2\\ &2,1,2,2,2,1,2,2,2,1,2,2,2,1,2,2].\end{aligned}$$

First, we have to find the minimum output-friendly subspace. Observing h, we have $n_1=8$ and $n_2=24$. Then we have the largest 2-type common factor $2^s=2^3$, and $m_1=1$, $m_2=3$. Hence, we know that the minimum output-friendly subspace is of dimension $n-s=5-3=2$. Using Algorithm IV.5, we may choose

$$\begin{aligned}T_0=\delta_4[&1,2,3,4,1,2,3,4,1,2,3,4,1,2,3,4,\\ &2,1,4,3,2,1,4,3,2,1,4,3,2,1,4,3].\end{aligned}$$

and

$$G=\delta_2[1,2,2,2].$$

From T_0 we can find the output-friendly sub-basis, denote it by $\{z_4,z_5\}$, with $z_4=M_4x$ and $z_5=M_5x$. Using the method proposed in [12], we can easily calculate M_4 and M_5 from T_0. In fact, for two factor case, we simply have the following rule: For M_1 each column of δ_4^1 or δ_4^2 of T_0 yields a column δ_2^1 in the corresponding column of M_1; otherwise, we have δ_2^2; and for M_2 each column of δ_4^1 or δ_4^3 of T_0 yields a δ_2^1 in the corresponding column of M_2; otherwise, we have δ_2^2. Hence, we have

$$\begin{aligned}M_4=\delta_2[&1,1,0,0,1,1,0,0,1,1,0,0,1,1,0,0,\\ &1,1,0,0,1,1,0,0,1,1,0,0,1,1,0,0];\\ M_5=\delta_2[&1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,\\ &0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1].\end{aligned}$$

Using Corollary II.9, we simply set $z_i=M_ix$, $i=1,2,3$, where M_i are chosen as follows:

$$\begin{aligned}M_1=\delta_2[&1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,\\ &0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0];\\ M_2=\delta_2[&0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,\\ &0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1];\\ M_3=\delta_2[&1,1,1,1,2,2,2,2,2,2,2,2,1,1,1,1,\\ &1,1,1,1,2,2,2,2,2,2,2,2,1,1,1,1]\end{aligned}$$

It is easy to check that the Boolean matrix B_z of $\{z_1, z_2, z_3, z_4, z_5\}$ has no equal columns. So it is a coordinate change. From M_i, the z_i can be calculated as

$$\begin{cases} z_1 = x_1 \\ z_2 = \neg x_2 \\ z_3 = x_2 \leftrightarrow x_3 \\ z_4 = x_4 \\ z_5 = x_1 \leftrightarrow x_5. \end{cases} \tag{42}$$

Setting $z = \ltimes_{i=1}^5 z_i$ and $x = \ltimes_{i=1}^5 x_i$, the algebraic form of (42) is $z = Tx$ with

$$T = \delta_{32}[9, 10, 11, 12, 13, 14, 15, 16, 5, 6, 7, 8, 1, 2, 3, 4, 26, 25, 28, 27, 30, 29, 32, 31, 22, 21, 24, 23, 18, 17, 20, 19].$$

Conversely, we have $x = T^T z$, with

$$T^T = [13, 14, 15, 16, 9, 10, 11, 12, 1, 2, 3, 4, 5, 6, 7, 8, 30, 29, 32, 31, 26, 25, 28, 27, 18, 17, 20, 19, 22, 21, 24, 23].$$

The inverse mapping of the coordinate transformation (42) becomes

$$\begin{cases} x_1 = z_1 \\ x_2 = \neg z_2 \\ x_3 = z_2 \bar{\vee} z_3 \\ x_4 = z_4 \\ x_5 = z_1 \leftrightarrow z_5. \end{cases}$$

Now under the coordinate frame z, the equation (41) becomes

$$\begin{aligned} z(t+1) &= Tx(t+1) \\ &= TLu(t)\xi(t)x(t) \\ &= TLu(t)\xi(t)T^T z(t) \\ &= TL(I_8 \otimes T^T)u(t)\xi(t)z(t) \\ &:= \tilde{L}u(t)\xi(t)z(t); \end{aligned}$$

and

$$y(t) = Hx(t) = HT^T z(t) := \tilde{H}z(t),$$

where

$$\begin{aligned} \tilde{L} = \delta_{32}[&17, 17, 1, 1, 19, 19, 3, 3, 17, 17, 2, 2, 19, 19, 4, 4, \\ &17, 17, 1, 1, 19, 19, 3, 3, 17, 17, 2, 2, 19, 19, 4, 4, \\ &17, 17, 1, 1, 27, 27, 11, 11, 19, 19, 4, 4, 27, 27, 12, 12, \\ &17, 17, 1, 1, 27, 27, 11, 11, 19, 19, 4, 4, 27, 27, 12, 12, \\ &17, 21, 2, 6, 19, 23, 4, 8, 17, 21, 2, 6, 19, 23, 4, 8, \\ &17, 21, 2, 6, 19, 23, 4, 8, 17, 21, 2, 6, 19, 23, 4, 8, \\ &17, 21, 2, 6, 27, 31, 12, 16, 19, 23, 4, 8, 27, 31, 12, 16, \\ &17, 21, 2, 6, 27, 31, 12, 16, 19, 23, 4, 8, 27, 31, 12, 16, \\ &1, 1, 17, 17, 3, 3, 19, 19, 1, 1, 18, 18, 3, 3, 20, 20, \\ &1, 1, 17, 17, 3, 3, 19, 19, 1, 1, 18, 18, 3, 3, 20, 20, \\ &1, 1, 17, 17, 11, 11, 27, 27, 1, 1, 18, 18, 11, 11, 28, 28, \\ &1, 1, 17, 17, 11, 11, 27, 27, 1, 1, 18, 18, 11, 11, 28, 28, \\ &1, 5, 18, 22, 3, 7, 20, 24, 1, 5, 18, 22, 3, 7, 20, 24, \\ &1, 5, 18, 22, 3, 7, 20, 24, 1, 5, 18, 22, 3, 7, 20, 24, \\ &1, 5, 18, 22, 11, 15, 28, 32, 1, 5, 18, 22, 11, 15, 28, 32, \\ &1, 5, 18, 22, 11, 15, 28, 32, 1, 5, 18, 22, 11, 15, 28, 32]; \end{aligned}$$

$$\begin{aligned} \tilde{H} = \delta_2[&1, 2, 2, 2, 1, 2, 2, 2, 1, 2, 2, 2, 1, 2, 2, 2, \\ &1, 2, 2, 2, 1, 2, 2, 2, 1, 2, 2, 2, 1, 2, 2, 2]. \end{aligned}$$

Then a mechanical procedure can convert the original system into Y-friendly coordinate frame z as

$$\begin{cases} z_1(t+1) = z_4(t) \bar{\vee} u_1(t) \\ z_2(t+1) = z_3(t) \vee \xi(t) \\ z_3(t+1) = z_5(t) \vee u_2(t) \\ z_4(t+1) = [u_1(t) \rightarrow (z_2(t) \vee \xi(t))] \wedge z_3(t) \\ z_5(t+1) = (u_2(t) \wedge z_2(t)) \vee z_4(t) \\ y = z_4 \wedge z_5 \end{cases} \tag{43}$$

Now in the output-friendly subspace (z_4, z_5) we may choose

$$u_1(t) = z_2(t) = \neg x_2(t), \quad u_2(t) = 0.$$

Then the only unlimited variable out of this space is z_3. Enlarging the output-friendly subspace to including z_3, One sees that the closed-loop system is in such a form that the DDP is solved. Since in system (43) the controls which solve the DDP is obvious, we need not to use general formula.

VII. Conclusion

The DDP of Boolean control networks has been investigated. First, the output-friendly regular subspaces were considered and formulas were provided to construct them. Secondly, under an output-friendly coordinate frame the solvability of DDP has been converted to solving a set of algebraic equations, by putting the dynamics of output-related state variables into a variable-separated form. Putting them together, a necessary and sufficient condition has been obtained for the solvability of DDP. A detailed control design technique was presented. An illustrative example was included to depict the method.

References

[1] S. Kauffman, “Metabolic stability and epigenesis in randomly constructed genetic nets.” vol. 22, no. 3, p. 437, 1969.

[2] S. Huang and D. Ingber, “Shape-dependent control of cell growth, differentiation, and apoptosis: switching between attractors in cell regulatory networks,” *Experimental Cell Research*, vol. 261, no. 1, pp. 91–103, 2000.

[3] S. Huang, “Regulation of cellular states in mammalian cells from a genomewide view,” in *Gene Regulation and Metabolism: Post-Genomic Computational Approaches*, C.-V. Julio and R. Hofestadt, Eds. MIT Press, 2002, pp. 181–220.

[4] C. Farrow, J. Heidel, J. Maloney, and J. Rogers, “Scalar equations for synchronous Boolean networks with biological applications,” vol. 15, no. 2, pp. 348–354, 2004.

[5] S. Kauffman, *The Origins of Order: Self-organization and Selection in Evolution*. New York: Oxford University Press, 1993.

[6] T. Ideker, T. Galitski, and L. Hood, “A new approach to decoding life: systems biology,” *Annu. Rev. Genomics Hum. Genet.*, vol. 2, pp. 343–372, 2001.

[7] T. Akutsu, M. Hayashida, W. Ching, and M. Ng, “Control of Boolean networks: hardness results and algorithms for tree structured networks,” *Journal of Theoretical Biology*, vol. 244, no. 4, pp. 670–679, 2007.

[8] D. Cheng and H. Qi, “A lienar representation of dynamics of Boolean networks,” 2009, provisionally accepted by IEEE Trans. Auto. Contr. (Preprint: http://lsc.amss.ac.cn/∼dcheng/preprint).

[9] D. Cheng, “Input-state approach to Boolean networks,” vol. 20, no. 3, pp. 512–521, 2009.

[10] S. Kauffman, *At Home in the Universe*. Oxford Univ. Press, 1995.

[11] D. Cheng and H. Qi, "Controllability and observability of Boolean control networks," *Automatica*, vol. 45, no. 7, pp. 1659–1667, 2009.

[12] D. Cheng, Z. Li, and H. Qi, "Realization of Boolean control networks," *Automatica*, vol. 46, no. 1, 2010.

[13] D. Cheng, "Sime-tensor product of matrices and its applications — a survey," in *Proc. ICCM'07*. Hangzhou: Higher Edu. Press, Int. Press, 2007, pp. 641–668.

[14] W. Wonham, *Linear Multivariable Control: A Geometric Aproach*, 2nd ed. Springer, Berlin, 1979.

[15] A. Isidori, *Nonlinear Control Systems*, 3rd ed. Springer, 1995.

[16] A. Hamilton, *Logic for mathematicians*, revised ed. Cambridge: Cambridge Univ Pr, 1988.

[17] I. Shmulevich, H. Lähdesmäki, and K. Egiazarian, "Special methods for testing membership in certain post classes and class of forcing functions," *IEEE Signal Processing Letters*, vol. 11, no. 2, pp. 289–292, 2004.

[18] D. Cheng, Z. Li, and H. Qi, "Canalizing Boolean mapping and its application to disturbance decoupling of Boolean control networks," in *Proc. ICCA'09*, Christchurch, New Zealand, 2009.