Partition-Based Solutions of Static Logical Networks With Applications

Yupeng Qiao, Hongsheng Qi, *Member, IEEE*, and Daizhan Cheng, *Fellow, IEEE*

***Abstract*— Given a static logical network, partition-based solutions are investigated. Easily verifiable necessary and sufficient conditions are obtained, and the corresponding formulas are presented to provide all types of the partition-based solutions. Then, the results are extended to mix-valued logical networks. Finally, two applications are presented: 1) an implicit function (IF) theorem of logical equations, which provides necessary and sufficient condition for the existence of IF and 2) converting the difference-algebraic network into a standard difference network.**

***Index Terms*— Difference-algebraic network, implicit function (IF) theorem, partition-based solutions, semitensor product (STP) of matrices, static logical network.**

I. Introduction

THE pioneer work of neural networks was done by McCulloch and Pitts in their famous paper "A logical calculus of the ideas immanent in influential" [12]. It was pointed out in [13] that "Note only was the McCulloch-Pitts model the first example of what would now be called a neural network, it was the first attempt to understand mental activity as a form of information processing—an insight that provides the inspiration for artificial intelligence and cognitive psychology alike."

When introducing Kauffman's work, which formulated genetic networks as Boolean (or logical) networks, [13] mentioned that "the genetic networks and neural networks were fundamentally the same thing." From this, one sees easily that Boolean network (or logical network) and neural network are two closely related tools, used to formulate and simulate genetic networks. But they may from different perspectives: Boolean network from logical aspect and neural network from numerical aspect.

This paper considers the static logical network Σ, and its application to dynamic-algebraic logical networks. As a logical type of neural network, Σ is described as follows:

$$\Sigma: \begin{cases} \varphi_1(\xi_1, \ldots, \xi_n) = c_1 \\ \qquad \vdots \\ \varphi_s(\xi_1, \ldots, \xi_n) = c_s \end{cases} \tag{1}$$

where $\xi_j \in \mathcal{D} := \{0, 1\}$, $j = 1, \ldots, n$ are logical variables, $c_i \in \mathcal{D}$, $i = 1, \ldots, s$ are logical constants, $\varphi_i : \mathcal{D}^n \to \mathcal{D}$, $i = 1, \ldots, s$ are logical functions.

In this paper, we will first propose a Boolean matrix, called the truth matrix of Σ and denoted by T_Σ, instead of the truth table used usually to characterize the network. Using it, the following fundamental problems will be investigated.

A. Partition-Based Solutions

Consider a logical network Σ, as described in (1). A logical relation A is called an antecedence solution of Σ, if $A \to \Sigma$ is a tautology. In other words, $A \Rightarrow \Sigma$ (for the notations and terminologies of logic, we follow [8]). A logical relation C is called a consequence solution of Σ, if $\Sigma \to C$ is a tautology. In other words, $\Sigma \Rightarrow C$. A logical relation E is called an antecedence-consequence solution of Σ, if E is both antecedence and consequence solution of Σ. In other words, $\Sigma \Leftrightarrow E$.

The problem of finding antecedence and consequence solutions was first proposed by Ledley [11], which introduced a solution-seeking method for static logical equations through automorphism of Boolean algebra. Sufficient conditions of the existence of solutions were proposed. The problem has also been addressed in [10]. In fact, Ledley was looking for a particular form of solutions. We first briefly describe Ledley's solutions [11]:

First, Ledley constructed a partition over the set of logical variables as

$$\{\xi_1, \ldots, \xi_n\} = \{x_1, \ldots, x_p\} \cup \{y_1, \ldots, y_q\} \tag{2}$$

where $p + q = n$. Then, he sought for the solutions of the form

$$y_\ell = f_\ell(x_1, \ldots, x_p), \quad \ell = 1, \ldots, q. \tag{3}$$

To emphasize that such solutions are partition depending, we call them the partition-based solutions.

In this paper, using truth matrix and semitensor product (STP) of matrices, we represent logical networks to algebraic form. Necessary and sufficient conditions are obtained for the existence of antecedence partition-based solution (APBS), of consequence partition-based solution (CPBS)

Manuscript received November 20, 2015; revised October 4, 2016; accepted February 9, 2017. Date of publication March 6, 2017; date of current version March 15, 2018. This work was supported by the National Natural Science Foundation of China under Grant 61074114, Grant 61273013, Grant 61333001, Grant 61104065, and Grant 61374168.

Y. Qiao is with the College of Automation Science and Engineering, South China University of Technology, Guangzhou 510640, China (e-mail: ypqiao@scut.edu.cn).

H. Qi is with the Key Laboratory of Systems and Control, Academy of Mathematics and Systems Sciences, Chinese Academy of Sciences, Beijing 100190, China, and also with the School of Mathematical Sciences, University of Chinese Academy of Sciences, Beijing 100049, China (e-mail: qihongsh@amss.ac.cn).

D. Cheng is with the Key Laboratory of Systems and Control, Academy of Mathematics and Systems Sciences, Chinese Academy of Sciences, Beijing 100190, China (e-mail: dcheng@iss.ac.cn).

Digital Object Identifier 10.1109/TNNLS.2017.2669972

2162-237X © 2017 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

and of antecedence-consequence partition-based solution (ACPBS), respectively. Corresponding formulas are presented to find out all types of partition-based solutions.

Then, the results are naturally extended to mix-valued logical networks.

B. Applications

Two applications are considered.

1) Using the ACPBS of a logical network, we consider the implicit function theorem (IFT) for the network. Easily verifiable necessary and sufficient conditions are obtained.
2) Using the partition-based solution of a logical network, we consider the difference-algebraic logical networks. We show that even when the conditions of IFT do not meet, we still can use APBS substitution to convert the network into a standard difference form.

In addition to the above applications for theoretical problems, partition-based solutions of logical networks have wide range of potential applications in various fields, such as diagnosis of disease, symbolic logic, and electrical circuit design [10].

The rest of this paper is organized as follows. Section II gives some useful notations, definitions, and propositions of STP. How to express a logical dynamic network into an algebraic state-space form is also introduced. Section III introduces the truth matrix of a static logical network with respect to a partition of the argument set. Necessary and sufficient conditions for APBS, CPBS, and ACPBS are, respectively, presented in Section IV. Formulas are presented to provide all these types of the solutions. In Section V, the results are extended to mix-valued logical equations. Then, two applications are discussed in Section VI, and certain related properties are studied. Section VII is the conclusion.

II. Semitensor Product of Matrices

This section gives a brief review for STP of matrices, which is the fundamental tool in our analysis. The readers are referred to [4] or [5] for details.

First, we give some notations.

1) $\mathbf{1}_n = [\underbrace{1, \ldots, 1}_{n}]^T$.
2) $\mathcal{M}_{m\times n}$: the set of $m \times n$ real matrices.
3) $\text{Col}(M)$ ($\text{Row}(M)$) is the set of columns (rows) of M. $\text{Col}_i(M)$ ($\text{Row}_i(M)$) is the i-th column (row) of M.
4) $\mathcal{D} := \{0, 1\}$.
5) $\mathcal{B}_{m\times n}$: the set of $m \times n$ Boolean matrices.
6) δ_n^i: the ith column of the identity matrix I_n.
7) $\Delta_n := \{\delta_n^i | i = 1, \ldots, n\}$, $\Delta := \Delta_2$.
8) A matrix $L \in \mathcal{M}_{m\times n}$ is called a logical matrix if the columns of L are of the form δ_m^k, $1 \le k \le m$. That is
$$\text{Col}(L) \subset \Delta_m.$$
Denote by $\mathcal{L}_{m\times n}$ the set of $m \times n$ logical matrices.
9) If $L \in \mathcal{L}_{n\times r}$, by definition it can be expressed as $L = [\delta_n^{i_1}, \delta_n^{i_2}, \ldots, \delta_n^{i_r}]$. For the sake of compactness, it is briefly denoted as
$$L = \delta_n[i_1, i_2, \ldots, i_r].$$

Definition 1: Let $M \in \mathcal{M}_{m\times n}$ and $N \in \mathcal{M}_{p\times q}$, and $t = \text{lcm}\{n, p\}$ be the least common multiple of n and p. The STP of M and N, denoted by $M \ltimes N$, is defined as
$$M \ltimes N := (M \otimes I_{t/n})(N \otimes I_{t/p}) \in \mathcal{M}_{mt/n\times qt/p} \quad (4)$$
where $\otimes$ is the Kronecker product.

When $n = p$, the STP coincides with the conventional matrix product. So the STP is a generalization of conventional matrix product. Fortunately, it keeps almost all the properties of the conventional matrix product available. So in this paper, all the matrix products are STP unless otherwise specified.

In addition, STP has some new properties. The following properties are frequently used in the sequel.

Proposition 2: Let $X \in \mathbb{R}^m$ be a column and M is any matrix. Then
$$X \ltimes M = (I_m \otimes M) \ltimes X. \quad (5)$$

Next, we consider the algebraic state-space representation of logical dynamic networks.

Definition 3: 1) A function $f : \mathcal{D}^n \to \mathcal{D}$ is called a Boolean function. It can be expressed as
$$y = f(x_1, x_2, \ldots, x_n), \quad y, x_1, \ldots, x_n \in \mathcal{D}. \quad (6)$$
2) A mapping $F : \mathcal{D}^n \to \mathcal{D}^m$ is called a Boolean mapping, where F is composed of m Boolean functions, as
$$F : \begin{cases} y_1 = f_1(x_1, \ldots, x_n) \\ \quad\vdots \\ y_m = f_m(x_1, \ldots, x_n). \end{cases} \quad (7)$$

Identifying
$$1 \sim \delta_2^1, \quad 0 \sim \delta_2^2$$
which is called the vector form of Boolean variable, then the Boolean function f becomes $f : \Delta^n \to \Delta$ and the Boolean mapping F becomes $F : \Delta^n \to \Delta^m$. In the vector form, we have the following algebraic state-space representation.

Theorem 4 [5]: Let $f : \mathcal{D}^n \to \mathcal{D}$ be a Boolean function. Then, there exists a unique logical matrix $M_f \in \mathcal{L}_{2\times 2^n}$, such that in the vector form, (6) can be expressed as
$$f(x_1, \ldots, x_n) = M_f \ltimes_{i=1}^n x_i. \quad (8)$$
M_f is called the structure matrix of f.

Consider the Boolean mapping (7). According to Theorem 4, there exist M_i, $i = 1, \ldots, m$ which are the structure matrices of the corresponding component functions. Then, we have the following result.

Theorem 5 [5]: Consider the Boolean mapping (7). In the vector form, let $x = \ltimes_{i=1}^n x_i$, $y = \ltimes_{i=1}^m y_i$. Then, there exists a unique logical matrix $M_F \in \mathcal{L}_{2^m\times 2^n}$, such that (7) can be expressed as
$$y = M_F x \quad (9)$$

where

$$M_F = M_1 * M_2 * \cdots * M_m \tag{10}$$

is called the structure matrix of F (the "*" in (10) is the Khatri–Rao product [9]).

The following proposition is straightforwardly verifiable.

Proposition 6: Let $X \in \Delta_m$ and $Y \in \Delta_n$. Then

$$\left(I_m \otimes \mathbf{1}_n^T\right) XY = X \tag{11}$$

and

$$\left(\mathbf{1}_m^T \otimes I_n\right) XY = Y. \tag{12}$$

III. Truth Matrix

Consider network (1) (or Σ). Denote the argument set as

$$\Xi := \{\xi_1, \xi_2, \ldots, \xi_n\}.$$

Let

$$\Xi = X \cup Y \tag{13}$$

be a partition of Ξ, that is, $X \cap Y = \emptyset$. Rename the elements in X and Y as: $X = \{x_1, \ldots, x_p\} \subset \Xi$ and $Y = \{y_1, \ldots, y_q\} \subset \Xi$, then $p + q = n$.

Definition 7: Consider network (1) (or Σ). A network S, expressed as

$$S : \begin{cases} y_1 = f_1(x_1, \ldots, x_p) \\ \quad\vdots \\ y_q = f_q(x_1, \ldots, x_p) \end{cases} \tag{14}$$

is called an APBS to Σ with respect to the partition (X, Y) if $S \Rightarrow \Sigma$; S is called a CPBS to Σ with respect to the partition (X, Y) if $\Sigma \Rightarrow S$; S is called an ACPBS to Σ with respect to the partition (X, Y) if $\Sigma \Leftrightarrow S$.

Express $\{x_i\}$ and $\{y_j\}$ into vector form such that $x_i, y_j \in \Delta_2$, and set $x = \prod_{i=1}^{p} x_i \in \Delta_{2^p}$, $y = \prod_{i=1}^{q} y_i \in \Delta_{2^q}$.

Definition 8: Consider network (1) (or Σ). A matrix $T_{\Sigma}^{(X,Y)} = \left(t_{i,j}\right) \in \mathcal{B}_{2^q \times 2^p}$ is called the truth matrix of network (1) (or of Σ) with respect to the partition (X, Y), if

$$t_{i,j} = \begin{cases} 1, & x = \delta_{2^p}^{j},\ y = \delta_{2^q}^{i} \text{ assure (1) to be true} \\ 0, & \text{otherwise.} \end{cases} \tag{15}$$

We provide the following algorithm to convert the truth table into a truth matrix.

Algorithm 9: 1) *Step 1:* Convert (1) into the form as

$$\varphi(\xi_1, \ldots, \xi_n) = 1. \tag{16}$$

To this end, we first convert all c_i, $i = 1, \ldots, s$ to 1 by taking $\neg$ on both sides of φ_i if $c_i = 0$. Then set $\varphi = \wedge_{i=1}^{s} \varphi_i$.

2) *Step 2:* By possibly reordering the variables we can assume $X = (\xi_1, \ldots, \xi_p)$ and $Y = (\xi_{p+1}, \ldots, \xi_n)$. Assume the truth table for φ with respect to this variable order is $T \in \mathcal{D}^{2^n}$. Then split T into equal blocks as

$$T = \begin{bmatrix} T_1 \\ \vdots \\ T_{2^s} \end{bmatrix}$$

where $T_i \in \mathcal{D}^{2^{n-p}}$, $i = 1, \ldots, 2^p$.

3) *Step 3:*

$$T_{(1)}^{(X,Y)} = [T_1, T_2, \ldots, T_{2^p}]. \tag{17}$$

TABLE I

Truth Table of (18)

ξ_1	ξ_2	ξ_3	$\xi_1 \wedge \xi_2$	$\neg\xi_3$	$\varphi(\xi_1, \xi_2, \xi_3)$
1	1	1	1	0	1
1	1	0	1	1	1
1	0	1	0	0	0
1	0	0	0	1	1
0	1	1	0	0	0
0	1	0	0	1	1
0	0	1	0	0	0
0	0	0	0	1	1

TABLE II

Truth Matrix of (18)

$y \backslash x$	1,1	1,0	0,1	0,0
1	1	0	0	0
0	1	1	1	1

Example 10: Consider a network

$$\Sigma :\ \varphi(\xi_1, \xi_2, \xi_3) = (\xi_1 \wedge \xi_2 t) \vee (\neg\xi_3) = 1. \tag{18}$$

The truth table of φ is Table I.

Set a partition

$$\Xi = \{\xi_1, \xi_2, \xi_3\} = X \cup Y. \tag{19}$$

1) Assume $X_1 := \{\xi_1, \xi_2\}$ and $Y_1 = \{\xi_3\}$. Using Algorithm 9, it is easy to obtain Table II.
Then, we have the truth matrix of (18) under partition (X_1, Y_1) as

$$T_{(18)}^{(X_1,Y_1)} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}.$$

2) Next, if $X_2 = \{\xi_2, \xi_3\}$, $Y_2 = \{\xi_1\}$, by reordering the variables as $(\xi_2,\ \xi_3,\ \xi_1)$, from the corresponding truth table, we have

$$T_{(18)}^{(X_2,Y_2)} = \begin{bmatrix} 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 \end{bmatrix}.$$

Similarly, it is easy to see that

3) if $X_3 = \{\xi_1, \xi_3\}$, $Y_3 = \{\xi_2\}$, we have

$$T_{(18)}^{(X_3,Y_3)} = \begin{bmatrix} 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 \end{bmatrix}.$$

4) If $X_4 = \{\xi_1\}$, $Y_4 = \{\xi_2, \xi_3\}$, we have

$$T_{(18)}^{(X_4,Y_4)} = \begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 0 & 0 \\ 1 & 1 \end{bmatrix}.$$

TABLE III
TRUTH MATRIX OF (20) CORRESPONDING TO PARTITION (X, Y)

$\xi_3 \backslash \xi_1\xi_2$	δ_4^1	δ_4^2	δ_4^3	δ_4^4
δ_2^1	1	1	0	0
δ_2^2	0	1	1	1

5) If $X_5 = \{\xi_2\}$, $Y_5 = \{\xi_1, \xi_3\}$, we have

$$T_{(18)}^{(X_5,Y_5)} = \begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 0 & 0 \\ 1 & 1 \end{bmatrix}.$$

6) If $X_6 = \{\xi_3\}$, $Y_6 = \{\xi_1, \xi_2\}$, we have

$$T_{(18)}^{(X_6,Y_6)} = \begin{bmatrix} 1 & 1 \\ 0 & 1 \\ 0 & 1 \\ 0 & 1 \end{bmatrix}.$$

Remark 11:

1) Within each part of the partition (precisely, within X or Y), the elements are arranged according to its original order. This convention is to keep the corresponding truth matrix unique. Otherwise, the matrix will have some row and/or column permutations, which will not affect the following argument.
2) According to Algorithm 9, we know that truth matrix is a rearrangement of the truth table. And it contains complete information of a network corresponding to the preassigned partition. In this paper, it will be used to solve several problems.

IV. FINDING PBS VIA TRUTH MATRIX

A. Solving APBS

We start by analyzing an example.

Example 12 [10]: Consider a logical equation

$$\varphi(\xi_1, \xi_2, \xi_3) = (\neg\xi_1 \wedge \xi_3) \vee (\xi_1 \wedge \xi_2 \wedge \neg\xi_3) = 0. \quad (20)$$

Choose the partition as: $X = \{\xi_1, \xi_2\}$, $Y = \{\xi_3\}$. Then, it is easy to calculate the truth matrix corresponding to (X, Y) of (20) as in Table III.

Equivalently, we have

$$T_{(20)}^{(X,Y)} = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix}. \quad (21)$$

Then, we choose a logical matrix $M \in \mathcal{L}_{2\times 4}$ such that

$$M \le T_{(20)}^{(X,Y)}. \quad (22)$$

Note that for any $A = (a_{i,j})$, $B = (b_{i,j}) \in \mathcal{M}_{m\times n}$, $A \le B$ means

$$a_{i,j} \le b_{i,j}, \quad i = 1, \dots, m; \ j = 1, \dots, n.$$

Set

$$y = Mx. \quad (23)$$

Then, (23) is an APBS of (20). To see this, for any $x = \xi_1\xi_2$ and the corresponding $y = \xi_3$, because of (22), one sees easily that (20) is always true. That is, (23)$\Rightarrow$(20). Conversely, it is also clear that (22) is also necessary for (23) to imply (20). Otherwise, there exists at least one $x = (\alpha, \beta)$ and $y = M\alpha\beta$, which correspond to a position where $m_{i,j} > t_{i,j}$ (that is, where the entry of M is 1 and the entry of T is 0). Then, (20) does not hold with $\xi_1\xi_2 = \alpha\beta$ and $\xi_3 = M\alpha\beta$.

Now, it is obvious that there are two M values, which satisfy (22). They are

$$M_1 = \delta_2[1\ 1\ 2\ 2]; \quad M_2 = \delta_2[1\ 2\ 2\ 2].$$

According to the above argument, one can easily figure out that

$$\xi_3 = \delta_2[1\ 1\ 2\ 2]\xi_1\xi_2 \quad (24)$$

and

$$\xi_3 = \delta_2[1\ 2\ 2\ 2]\xi_1\xi_2 \quad (25)$$

are two APBSs of (20) with respect to this particular partition.

To investigate the general result, the Hamming weight of a Boolean vector $X = (x_1, \dots, x_n) \in \mathcal{B}_n$ is introduced as [2]

$$w_H(X) = \sum_{i=1}^{n} x_i.$$

From the discussion of Example 12, the following conclusion is obvious.

Theorem 13: Consider the logical network (1) (or Σ).

1) Let (X, Y) be a partition of $\Xi = \{\xi_1, \dots, \xi_n\}$. $|X| = p > 0$, $|Y| = q > 0$. Corresponding to (X, Y), the truth matrix is $T := T_\Sigma^{(X,Y)} \in \mathcal{B}_{2^q\times 2^p}$. Then, (1) (or Σ) has APBS, if and only if, the Hamming weight of each column of T is nonzero. That is
$$w_j := w_H(\mathrm{Col}_j(T)) > 0, \quad j = 1, \dots, 2^p. \quad (26)$$
2) Assume there exists a logical matrix $M \in \mathcal{L}_{2^q\times 2^p}$ satisfies $M \le T$, then $y = Mx$ is an APBS of (1) (or Σ).
3) There are
$$r = \prod_{j=1}^{2^p} w_j. \quad (27)$$
APBSs corresponding to this particular partition.
4) Assume (X_i, Y_i), $i = 1, \dots, \ell$ are all possible partitions of Ξ, $|X_i| = p_i$. Then, the total number of APBSs of (1), denoted by s_r, satisfies
$$s_r \le \sum_{i=1}^{\ell} \prod_{j=1}^{2^{p_i}} w_H(\mathrm{Col}_j(T_\Sigma^{(X_i,Y_i)})). \quad (28)$$

Remark 14:

1) Unlike the conventional case, in a partition (X, Y) of Ξ, the X and the Y are not commutative. This is because X and Y play different roles. Precisely, in an APBS, X is the set of arguments and Y is the set of functions.
2) The inequality appears in (28) because among solutions with respect to all the partitions there might be some duplicate ones (refer to Example 15).

Example 15: Recall Example 10. There are six different partitions.

1) For $T_{(18)}^{(X_1,Y_1)}$, there are two APBSs according to the partition (X_1, Y_1)

$$\xi_3 = \delta_2[1,2,2,2]\xi_1\xi_2 = \xi_1 \wedge \xi_2.$$
$$\xi_3 = \delta_2[2,2,2,2]\xi_1\xi_2 = \delta_2^2.$$

2) For $T_{(18)}^{(X_2,Y_2)}$, since $w_H\left(\text{Col}_3(T_{(18)}^{(X_2,Y_2)})\right) = 0$, there is no APBS according to the partition (X_2, Y_2).
3) For $T_{(18)}^{(X_3,Y_3)}$, since $w_H\left(\text{Col}_3(T_{(18)}^{(X_3,Y_3)})\right) = 0$, there is no APBS according to the partition (X_3, Y_3).
4) For $T_{(18)}^{(X_4,Y_4)}$, there are six APBSs according to the partition (X_4, Y_4).
   a) Case 1, $\xi_2\xi_3 = \delta_4[1,2]\xi_1$.
   Using Proposition 6, we have

$$\xi_2 = (I_2 \otimes \mathbf{1}_2^T)\delta_4[1,2]\xi_1 = \delta_2[1,1]\xi_1 = \delta_2^1$$
$$\xi_3 = (\mathbf{1}_2^T \otimes I_2)\delta_4[1,2]\xi_1 = \delta_2[1,2]\xi_1 = \xi_1.$$

   Hence, we have an APBS

$$\begin{cases}\xi_2 = \delta_2^1\\ \xi_3 = \xi_1.\end{cases}$$

   Similarly, we have
   b) Case 2, $\xi_2\xi_3 = \delta_4[1,4]\xi_1$. The APBS is

$$\begin{cases}\xi_2 = \xi_1\\ \xi_3 = \xi_1.\end{cases}$$

   c) Case 3, $\xi_2\xi_3 = \delta_4[2,2]\xi_1$. The APBS is

$$\begin{cases}\xi_2 = \delta_2^1\\ \xi_3 = \delta_2^2.\end{cases}$$

   d) Case 4, $\xi_2\xi_3 = \delta_4[2,4]\xi_1$. The APBS is

$$\begin{cases}\xi_2 = \xi_1\\ \xi_3 = \delta_2^2.\end{cases}$$

   e) Case 5, $\xi_2\xi_3 = \delta_4[4,2]\xi_1$. The APBS is

$$\begin{cases}\xi_2 = \neg\xi_1\\ \xi_3 = \delta_2^2.\end{cases}$$

   f) Case 6, $\xi_2\xi_3 = \delta_4[4,4]\xi_1$. The APBS is

$$\begin{cases}\xi_2 = \delta_2^2\\ \xi_3 = \delta_2^2.\end{cases}$$

5) For $T_{(18)}^{(X_5,Y_5)}$, there are six APBSs according to the partition (X_5, Y_5).
   a) Case 1, $\xi_1\xi_3 = \delta_4[1,2]\xi_2$. The APBS is

$$\begin{cases}\xi_1 = \delta_2^1\\ \xi_3 = \xi_2.\end{cases}$$

   b) Case 2, $\xi_1\xi_3 = \delta_4[1,4]\xi_2$. The APBS is

$$\begin{cases}\xi_1 = \xi_2\\ \xi_3 = \xi_2.\end{cases}$$

   c) Case 3, $\xi_1\xi_3 = \delta_4[2,2]\xi_2$. The APBS is

$$\begin{cases}\xi_1 = \delta_2^1\\ \xi_3 = \delta_2^2.\end{cases}$$

   d) Case 4, $\xi_1\xi_3 = \delta_4[2,4]\xi_2$. The APBS is

$$\begin{cases}\xi_1 = \xi_2\\ \xi_3 = \delta_2^2.\end{cases}$$

   e) Case 5, $\xi_1\xi_3 = \delta_4[4,2]\xi_2$. The APBS is

$$\begin{cases}\xi_1 = \neg\xi_2\\ \xi_3 = \delta_2^2.\end{cases}$$

   f) Case 6, $\xi_1\xi_3 = \delta_4[4,4]\xi_2$. The APBS is

$$\begin{cases}\xi_1 = \delta_2^2\\ \xi_3 = \delta_2^2.\end{cases}$$

6) For $T_{(18)}^{(X_6,Y_6)}$, there are four APBSs according to the partition (X_6, Y_6).
   a) Case 1, $\xi_1\xi_2 = \delta_4[1,1]\xi_3$. The APBS is

$$\begin{cases}\xi_1 = \delta_2^1\\ \xi_2 = \delta_2^1.\end{cases}$$

   b) Case 2, $\xi_1\xi_2 = \delta_4[1,2]\xi_3$. The APBS is

$$\begin{cases}\xi_1 = \delta_2^1\\ \xi_2 = \xi_3.\end{cases}$$

   c) Case 3, $\xi_1\xi_2 = \delta_4[1,3]\xi_3$. The APBS is

$$\begin{cases}\xi_1 = \xi_3\\ \xi_2 = \delta_2^1.\end{cases}$$

   d) Case 4, $\xi_1\xi_2 = \delta_4[1,4]\xi_3$. The APBS is

$$\begin{cases}\xi_1 = \xi_3\\ \xi_2 = \xi_3.\end{cases}$$

It is worth noting that the solutions of case 2 in 4) and case 2 in 5) are the same, as we mentioned in Remark 14, they are duplicate solutions.

B. Solving CPBS

Consider network (1) (or Σ), we search for CPBS. Observing the truth matrix carefully, it is not difficult to find the following result.

Theorem 16: Consider the logical network (1) (or Σ).

1) Let (X, Y) be a partition of $\Xi = \{\xi_1, \ldots, \xi_n\}$. $|X| = p > 0$, $|Y| = q > 0$. Corresponding to (X, Y), the truth matrix is $T := T_\Sigma^{(X,Y)} \in \mathcal{B}_{2^q \times 2^p}$. Then, (1) has CPBS, if and only if, the Hamming weight of each column of T satisfies

$$w_j := w_H(\text{Col}_j(T)) \leq 1, \quad j = 1, \ldots, 2^p. \quad (29)$$

2) Assume a logical matrix $M \in \mathcal{L}_{2^q \times 2^p} \geq T$, then $y = Mx$ is a CPBS of (1) (or Σ).
3) There are

$$r = 2^{\mu p}. \quad (30)$$

CPBSs corresponding to this particular partition, where $\mu = |\mathrm{Col}^*(T)|$ and $\mathrm{Col}^*(T) = \{\mathrm{Col}_k(T) \in \mathrm{Col}(T) | \mathrm{Col}_k(T) = 0\}$.

4) Assume (X_i, Y_i), $i = 1, \ldots, \ell$ are all possible partitions of Ξ, $|X_i| = p_i$. Then, the total number of CPBSs of (1) (or Σ) is

$$s_r \leq \sum_{j=1}^{\ell} r_i \tag{31}$$

where r_i is the number of CPBSs of (1) (or Σ) corresponding to partition (X_i, Y_i).

We use the following examples to depict CPBSs.

Example 17: Recall Example 10 (also Example 15). According to Theorem 16, it does not have CPBS.

Example 18: Consider the network

$$\varphi(\xi_1, \xi_2, \xi_3) = [\xi_1 \wedge (\xi_2 \leftrightarrow \xi_3)] \vee [\neg\xi_1 \wedge (\neg\xi_2 \wedge \xi_3)] = 1. \tag{32}$$

1) When $X_1 = \{\xi_1, \xi_2\}$, $Y_1 = \{\xi_3\}$, the truth matrix of (32) according to partition (X_1, Y_1) is

$$T_{(32)}^{(X_1,Y_1)} = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{bmatrix}. \tag{33}$$

Using Theorem 16, there are two CPBSs

$$\xi_3 = \begin{bmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 \end{bmatrix} \xi_1\xi_2$$

and

$$\xi_3 = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix} \xi_1\xi_2.$$

Transfer them to logical form, we have

$$\xi_3 = \xi_1 \rightarrow \xi_2 \tag{34}$$

and

$$\xi_3 = \xi_1 \leftrightarrow \xi_2. \tag{35}$$

2) When $X_2 = \{\xi_1, \xi_3\}$, $Y_2 = \{\xi_2\}$, the truth matrix of (32) according to partition (X_2, Y_2) is

$$T_{(32)}^{(X_2,Y_2)} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \end{bmatrix}. \tag{36}$$

Using Theorem 16, there are two CPBSs

$$\xi_2 = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix} \xi_1\xi_3$$

and

$$\xi_2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix} \xi_1\xi_3.$$

Back to logical form, we have

$$\xi_2 = \xi_1 \leftrightarrow \xi_3 \tag{37}$$

and

$$\xi_2 = \xi_1 \wedge \xi_3. \tag{38}$$

3) When $X_3 = \{\xi_2, \xi_3\}$, $Y_3 = \{\xi_1\}$, the truth matrix of (32) according to partition (X_3, Y_3) is

$$T_{(32)}^{(X_3,Y_3)} = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}. \tag{39}$$

Using Theorem 16, there are two CPBSs

$$\xi_1 = \begin{bmatrix} 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \xi_2\xi_3$$

and

$$\xi_1 = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix} \xi_2\xi_3.$$

Transfer them into the logical form, we have

$$\xi_1 = \xi_2 \wedge \neg\xi_3 \tag{40}$$

and

$$\xi_1 = \xi_2 \leftrightarrow \xi_3. \tag{41}$$

Note that

$$T_{\varphi}^{(X,Y)} = (T_{\varphi}^{(Y,X)})^T \tag{42}$$

then for (X, Y), where $|X| = 1$, $|Y| = 2$, logical matrix M satisfying $M \leq T$ does not exist. That is, for (32), there is no CPBS with respect to the partition (X, Y) with $|X| = 1$ and $|Y| = 2$.

Remark 19: Taking into consideration of Algorithm 9, (42) follows immediately. So when we consider how many partitions do we have, (X, Y) and (Y, X) may be classified as one.

C. Solving ACPBS

About ACPBS of network (1) (or Σ), we have the following result.

Theorem 20: Consider network (1) (or Σ). With respect to a particular partition (X, Y), we have

1) The truth matrix $T := T_{\Sigma}^{(X,Y)} \in \mathcal{B}_{2^q \times 2^p}$ is a logical matrix, if and only if, (1) (or Σ) has an ACPBS

$$y = Tx \tag{43}$$

2) There is at most one ACPBS for (1) (or Σ) with respect to a particular partition (X, Y).
3) The ACPBS (43) is equivalent to network (1) (or Σ).

Example 21: Assume that logical network (1) has already been expressed into the algebraic form as

$$\begin{aligned} &\varphi_1(\xi_1, \xi_2, \xi_3, \xi_4) \\ &= \delta_2[2, 1, 2, 2, 1, 2, 1, 2, 1, 1, 1, 2, 2, 1, 2, 1]\xi_1\xi_2\xi_3\xi_4 = \delta_2^1 \\ &\varphi_2(\xi_1, \xi_2, \xi_3, \xi_4) \\ &= \delta_2[2, 2, 2, 1, 2, 2, 1, 2, 1, 1, 2, 2, 1, 2, 2, 1]\xi_1\xi_2\xi_3\xi_4 = \delta_2^2. \end{aligned} \tag{44}$$

Because of the space of partitions crunch, here we consider only a few typical partitions in this example.

1) For partition $X_1 = \{\xi_1, \xi_2\}$ and $Y_1 = \{\xi_3, \xi_4\}$. Set $x = \xi_1\xi_2$, $y = \xi_3\xi_4$. It is easy to construct its truth matrix as in Table IV.

TABLE IV
TRUTH MATRIX OF (44) CORRESPONDING TO (X_1, Y_1)

$y\backslash x$	δ_4^1	δ_4^2	δ_4^3	δ_4^4
δ_4^1	0	1	0	0
δ_4^2	1	0	0	1
δ_4^3	0	0	1	0
δ_4^4	0	0	0	0

Then for (44), the truth matrix with respect to the above (X_1, Y_1) is a logical matrix. According to Theorem 20, (44) has an ACPBS with respect to (X_1, Y_1). Moreover, this solution is determined by

$$y = T_{(44)}^{(X_1,Y_1)} x = \delta_4[2, 1, 3, 2]x. \quad (45)$$

Using Proposition 6, one can easily calculate that

$$\begin{cases} \xi_3 = (I_2 \otimes \mathbf{1}_2^T) T_{(44)}^{(X_1,Y_1)} x = \delta_2[1, 1, 2, 1]x \\ \xi_4 = (\mathbf{1}_2^T \otimes I_2) T_{(44)}^{(X_1,Y_1)} x = \delta_2[2, 1, 1, 2]x. \end{cases} \quad (46)$$

Back to logical form, we have

$$\begin{cases} \xi_3 = \xi_1 \vee \neg\xi_2 \\ \xi_4 = \xi_1 \bar{\vee} \xi_2. \end{cases} \quad (47)$$

2) For partition $X_2 = \{\xi_1, \xi_3\}$ and $Y_2 = \{\xi_2, \xi_4\}$. Set $x = \xi_1\xi_3$, $y = \xi_2\xi_4$. It is easy to obtain the truth matrix as

$$T_2 := T_{(44)}^{(X_2,Y_2)} = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}. \quad (48)$$

From Theorems 13 and 16, (44) has neither APBS nor CPBS with respect to partition (X_2, Y_2).

3) For partition $X_3 = \{\xi_1\}$ and $Y_3 = \{\xi_2, \xi_3, \xi_4\}$. Set $x = \xi_1$, $y = \xi_2\xi_3\xi_4$. The truth matrix is

$$T_3 := T_{(44)}^{(X_3,Y_3)} = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}. \quad (49)$$

From Theorems 13 and 16, (44) has four APBSs and no CPBS with respect to partition (X_3, Y_3).

4) For $X_4 = \{\xi_1, \xi_2, \xi_3\}$ and $Y_4 = \{\xi_4\}$. Set $x = \xi_1\xi_2\xi_3$, $y = \xi_4$. The truth matrix is

$$T_4 := T_{(44)}^{(X_4,Y_4)} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}. \quad (50)$$

From Theorems 13 and 16, (44) has no APBS and 16 CPBSs with respect to partition (X_4, Y_4).

V. MIX-VALUED LOGICAL NETWORK

In Sections II–IV, we considered only Boolean functions. In fact, the technique developed and results obtained can easily be extended to mix-valued networks. Consider a mix-valued network as

$$\begin{cases} \varphi_1(\xi_1, \ldots, \xi_n) = c_1 \\ \quad\vdots \\ \varphi_s(\xi_1, \ldots, \xi_n) = c_s \end{cases} \quad (51)$$

where $\xi_i \in \mathcal{D}_{k_i}$, $i = 1, \ldots, n$, $\varphi_j : \prod_{i=1}^n \mathcal{D}_{k_i} \to \mathcal{D}_{\mu_j}$, $c_j \in \mathcal{D}_{\mu_j}$, $j = 1, \ldots, s$. Let (X, Y) be a partition of Ξ

$$\alpha := \prod_{\{i|\xi_i \in X\}} k_i; \quad \beta := \prod_{\{i|\xi_i \in Y\}} k_i. \quad (52)$$

Then, we define the truth matrix $T_\Sigma^{(X,Y)} \in \mathcal{B}_{\alpha\times\beta}$ in the same way as for Boolean case. Then, all the previous results remain available. Note that in mix-valued case, it is not convenient to use an explicit logical expression for (51). So we always express (51) in an algebraic form as

$$\begin{cases} M_1 \ltimes_{i=1}^n \xi_i = c_1 \\ \quad\vdots \\ M_s \ltimes_{i=1}^n \xi_i = c_s \end{cases} \quad (53)$$

where $\xi_i \in \Delta_{k_i}$, $i = 1, \ldots, n$, M_j is the structure matrix of φ_j, $j = 1, \ldots, s$.

We use an example to depict this.

Example 22: Consider the following network:

$$\begin{cases} \varphi(\xi_1, \xi_2, \xi_3, \xi_4) = M_1\xi_1\xi_2\xi_3\xi_4 = \delta_3^2 \\ \varphi(\xi_1, \xi_2, \xi_3, \xi_4) = M_2\xi_1\xi_2\xi_3\xi_4 = \delta_2^1 \end{cases} \quad (54)$$

where $\xi_1, \xi_4 \in \Delta_2$, $\xi_2, \xi_3 \in \Delta_3$

$$\begin{aligned} M_1 = \delta_3[&1, 1, 2, 1, 3, 3, 1, 2, 1, 2, 1, 3 \\ &3, 2, 1, 2, 2, 2, 3, 3, 2, 2, 3, 2 \\ &3, 2, 1, 3, 2, 2, 2, 3, 1, 2, 3, 1] \\ M_2 = \delta_2[&1, 2, 1, 1, 2, 1, 2, 2, 1, 1, 2, 1 \\ &2, 2, 1, 1, 2, 1, 1, 2, 1, 2, 1, 2 \\ &2, 1, 1, 1, 1, 2, 1, 1, 2, 2, 2, 1]. \end{aligned}$$

Consider the partition (X, Y) as

$$X = \{\xi_1, \xi_2\}, \quad Y = \{\xi_3, \xi_4\}$$

and let $x := \xi_1\xi_2$, $y := \xi_3\xi_4$.

It is easy to calculate that the truth matrix, which is shown in Table V.

TABLE V
TRUTH MATRIX OF (54)

$y\backslash x$	δ_6^1	δ_6^2	δ_6^3	δ_6^4	δ_6^5	δ_6^6
δ_6^1	0	0	0	0	0	1
δ_6^2	0	0	0	0	1	0
δ_6^3	1	0	0	1	0	0
δ_6^4	0	1	1	0	0	0
δ_6^5	0	0	0	0	1	0
δ_6^6	0	0	1	0	0	0

According to Theorem 13, it is easy to figure out that there are four APBSs, which are

$$y = T_i x, \quad i = 1, 2, 3, 4$$

where

$$T_1 = \delta_6[3, 4, 4, 3, 2, 1]$$
$$T_2 = \delta_6[3, 4, 6, 3, 2, 1]$$
$$T_3 = \delta_6[3, 4, 4, 3, 5, 1]$$
$$T_4 = \delta_6[3, 4, 6, 3, 5, 1].$$

Consider the case of $i = 1$. Using Proposition 6, we have

$$\begin{cases} \xi_3 = M_1^1 \xi_1 \xi_2 = (I_3 \otimes \mathbf{1}_2^T) T_1 \xi_1 \xi_2 \\ \xi_4 = M_1^2 \xi_1 \xi_2 = (\mathbf{1}_3^T \otimes I_2) T_1 \xi_1 \xi_2. \end{cases}$$

It is easy to calculate that

$$M_1^1 = \delta_3[2, 2, 2, 2, 1, 1]; \quad M_1^2 = \delta_2[1, 2, 2, 1, 2, 1].$$

Similarly, we can calculate that when $i = 2$

$$M_2^1 = \delta_3[2, 2, 3, 2, 1, 1]; \quad M_2^2 = \delta_2[1, 2, 2, 1, 2, 1]$$

when $i = 3$

$$M_3^1 = \delta_3[2, 2, 2, 2, 3, 1]; \quad M_3^2 = \delta_2[1, 2, 2, 1, 1, 1]$$

when $i = 4$

$$M_4^1 = \delta_3[2, 2, 3, 2, 3, 1]; \quad M_4^2 = \delta_2[1, 2, 2, 1, 1, 1].$$

According to Theorem 16, there is no CPBS of (54) with respect to partition (X, Y).

All other partitions can be verified in a similar way. But we omit this.

VI. Implicit Function Theorem

A. Existence of Implicit Function Theory

To the best of our knowledge, the IFT of Boolean equations has been discussed first in [1], which provides a sufficient condition for local IFT. Necessary and sufficient condition for global IFT of k-valued logical networks was given in [6] by using k-types. In the following, an even more simple way is provided to verify and design IF for mix-valued logical networks.

First, we state the problem clearly.

Definition 23: Consider the mix-valued network (51). Let (X, Y) be a preassigned partition of $\Xi = \{\xi_1, \ldots, \xi_n\}$. $|X| = p > 0$, $|Y| = q > 0$, $p + q = n$. The problem of the existence of IF is: can we find a network

$$\begin{cases} y_1 = f_1(x_1, \ldots, x_p) \\ \quad \vdots \\ y_q = f_q(x_1, \ldots, x_p) \end{cases} \tag{55}$$

where $\{x_1, \ldots, x_p\} = X \subset \Xi$, $\{y_1, \ldots, y_q\} = Y \subset \Xi$, such that (55) is logically equivalent to (51).

According to the argument in Section IV, it is clear that an ACPBS provides the required IF. Since the necessary and sufficient condition for the existence of ACPBS is known, we have the following result.

Theorem 24 (Implicit Function Theorem): Consider the network of mix-valued logical equations (51) with a preassigned partition (X, Y). The following are equivalent.

1) The network of IFs exists.
2) The truth matrix $T_{(51)}^{(X,Y)}$ is a logical matrix.
3) Equation (51) has an ACPBS with respect to partition (X, Y).

Moreover, the ACPBS can be expressed as

$$y = T_{(51)}^{(X,Y)} x \tag{56}$$

where $x = \ltimes_{i=1}^{p} x_i$, $y = \ltimes_{j=1}^{q} y_j$.

It is worth noting that the existence of the network of IFs such as (55) depends on the partition. Different partitions may provide different networks of IFs.

We give an example to show this.

Example 25: Consider the following logical network:

$$\begin{cases} \varphi(\xi_1, \xi_2, \xi_3, \xi_4) = M_1 \xi_1 \xi_2 \xi_3 \xi_4 = \delta_2^1 \\ \varphi(\xi_1, \xi_2, \xi_3, \xi_4) = M_2 \xi_1 \xi_2 \xi_3 \xi_4 = \delta_2^2 \end{cases} \tag{57}$$

where $\xi_i \in \Delta_2$, $i = 1, 2, 3, 4$

$$M_1 = \delta_2[1, 2, 1, 2, 2, 1, 2, 1, 1, 2, 2, 1, 1, 2, 1, 1]$$
$$M_2 = \delta_2[2, 1, 1, 1, 2, 1, 1, 2, 1, 1, 2, 2, 2, 1, 1, 1].$$

1) Given a partition (X_1, Y_1), where $X_1 = \{\xi_1, \xi_2\}$, and $Y_1 = \{\xi_3, \xi_4\}$. Set $x = \xi_1 \xi_2$ and $y = \xi_3 \xi_4$, it is easy to calculate the truth matrix

$$T_{(57)}^{(X_1,Y_1)} = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \end{bmatrix}. \tag{58}$$

From Theorem 24, there exists an IF as

$$y = T_{(57)}^{(X_1,Y_1)} x. \tag{59}$$

Using Proposition 6, we have

$$\xi_3 = \delta_2[1, 2, 2, 1] \xi_1 \xi_2$$
$$\xi_4 = \delta_2[1, 2, 2, 1] \xi_1 \xi_2.$$

Back to the logical form, we have

$$\xi_3 = \xi_1 \leftrightarrow \xi_2$$
$$\xi_4 = \xi_1 \leftrightarrow \xi_2.$$

2) Given a partition (X_2, Y_2), where $X_2 = \{\xi_1, \xi_3\}$, and $Y_2 = \{\xi_2, \xi_4\}$. A similar argument shows that the IF is

$$\xi_2 = \xi_1 \leftrightarrow \xi_3$$
$$\xi_4 = \xi_3.$$

3) Given a partition (X_3, Y_3), where $X_3 = \{\xi_1, \xi_4\}$, and $Y_3 = \{\xi_2, \xi_3\}$. The corresponding IF is

$$\xi_2 = \xi_1 \leftrightarrow \xi_4$$
$$\xi_3 = \xi_4.$$

4) Given a partition (X_4, Y_4), where $X_4 = \{\xi_3, \xi_4\}$, and $Y_4 = \{\xi_1, \xi_2\}$. It is easy to verify that there does not exist an IF.

5) Given a partition (X_5, Y_5), where $X_5 = \{\xi_2, \xi_4\}$, and $Y_5 = \{\xi_1, \xi_3\}$. The IF is

$$\xi_1 = \xi_2 \leftrightarrow \xi_4$$
$$\xi_3 = \xi_4.$$

6) Given a partition (X_6, Y_6), where $X_6 = \{\xi_2, \xi_3\}$, and $Y_6 = \{\xi_1, \xi_4\}$. The IF is

$$\xi_1 = \xi_2 \leftrightarrow \xi_3$$
$$\xi_4 = \xi_3.$$

7) For partition (X, Y) with $|X| = 1$, $|Y| = 3$ or $|X| = 3$, $|Y| = 1$, it is easy to see that the corresponding truth matrix is not logical, that is, the conditions of IFT do not meet, and hence, the IF does not exist.

For network (57), we have that: 1) the existence of IF is related to the preassigned partition and 2) the IF with respect to different partitions is different.

B. Difference-Algebraic Networks

Consider an evolutionary game (EG), where players have different numbers of strategies (actions), then the EG becomes an mix-valued logical network [7]. Now if the profile has certain restriction, that is, the actions chosen by players have some restriction, and then the EG becomes a difference-algebraic network. In general, a mix-valued logical difference-algebraic network is defined as

$$\begin{cases} \xi_1(t+1) = g_1(\xi_1, \ldots, \xi_n) \\ \quad\vdots \\ \xi_p(t+1) = g_p(\xi_1, \ldots, \xi_n) \end{cases}$$
$$\begin{cases} \varphi_1(\xi_1, \ldots, \xi_n) = c_1 \\ \quad\vdots \\ \varphi_q(\xi_1, \ldots, \xi_n) = c_q \end{cases} \tag{60}$$

where $\xi_i \in \mathcal{D}_{k_i}$, $i = 1, \ldots, n$, $c_j \in \mathcal{D}_{\mu_j}$, $j = 1, \ldots, q$.

To investigate the difference-algebraic network, a natural and convenient way is to convert it to normal difference network.

Case 1: ACPBS Substitution.

Proposition 26: Consider the algebraic part of (60). Assume it has an ACPBS with respect to the partition (X, Y), where $X = \{\xi_1, \ldots, \xi_p\}$, $Y = \{\xi_{p+1}, \ldots, \xi_n\}$. Then, (60) can be converted into a classical logical dynamic network.

Proof: Substituting the ACPBS into the difference part of the network yields what we expected. ■

Case 1 is very natural. In fact, it is standard for solving conventional differential (or difference)—algebraic equations for the state $\xi \in \mathbb{R}^n$: solving the second group of variables as the functions of the first group of variables, and then substituting them into the differential (or difference) equations. But as shown in Sections IV and V that networks which have ACPBS are fewer. When a network does not have ACPBS, which means that the second group of variables is not solvable from the algebraic equations as the functions of first group of variables, we consider APBS substitution in the following. Note that in this case, the corresponding conventional differential (or difference)—algebraic equations in $\mathbb{R}^n$ are difficult to be solved. So APBS substitution is totally new and only suitable for logical networks.

Case 2: APBS Substitution.

Proposition 27: Consider the algebraic part of (60). Assume it has APBS with respect to the partition (X, Y), where $X = \{\xi_1, \ldots, \xi_p\}$, $Y = \{\xi_{p+1}, \ldots, \xi_n\}$. Then, (60) can be converted into several classical logical dynamic networks corresponding to different APBSs.

Example 28: Consider the following network:

$$\begin{cases} \xi_1(t+1) = G_1\xi \\ \xi_2(t+1) = G_2\xi \end{cases}$$
$$\begin{cases} \varphi_1(\xi_1, \xi_2, \xi_3) = M_1\xi = \delta_3^3 \\ \varphi_2(\xi_1, \xi_2, \xi_3) = M_2\xi = \delta_2^1 \end{cases} \tag{61}$$

where $\xi_1 \in \Delta_2$, $\xi_2,\ \xi_3 \in \Delta_3$, $\xi = \ltimes_{i=1}^3 \xi_i$, and

$$G_1 = \delta_2[1, 2, 2, 1, 1, 1, 2, 2, 1, 1, 2, 1, 2, 2, 1, 1, 2, 1]$$
$$G_2 = \delta_3[3, 2, 1, 3, 3, 2, 2, 1, 1, 1, 3, 3, 2, 2, 2, 1, 2, 3]$$
$$M_1 = \delta_3[1, 3, 3, 3, 1, 2, 2, 3, 1, 3, 1, 3, 3, 2, 1, 1, 2, 3]$$
$$M_2 = \delta_2[1, 1, 2, 1, 2, 1, 2, 1, 1, 2, 2, 1, 1, 1, 2, 2, 2, 1].$$

Our goal is to convert (61) into a standard dynamic logical network.

Set the partition as (X, Y), where

$$X = \{\xi_1, \xi_2\}, \quad Y = \{\xi_3\}. \tag{62}$$

Using M_1 and M_2, it is easy to calculate the truth matrix of φ_1-φ_2 as

$$T := T_\varphi^{(X,Y)} = \begin{bmatrix} 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \end{bmatrix}.$$

Since T is a logical matrix, according to Theorem 16, the ACBPS is

$$\xi_3 = T\xi_1\xi_2. \tag{63}$$

Plugging (63) into the difference part of (61) yields

$$\begin{aligned} \xi_1(t+1) &= G_1\xi_1(t)\xi_2(t)T\xi_1(t)\xi_2(t) \\ &= G_1[I_6 \otimes T]\xi_1(t)\xi_2(t)\xi_1(t)\xi_2(t) \\ &= G_1[I_6 \otimes T]Mr_6\xi_1(t)\xi_2(t) \\ &= \delta_2[2, 1, 2, 1, 2, 1]\xi_1(t)\xi_2(t) \end{aligned}$$

and

$$\begin{aligned} \xi_2(t+1) &= G_2\xi_1(t)\xi_2(t)T\xi_1(t)\xi_2(t) \\ &= G_2[I_6 \otimes T]\xi_1(t)\xi_2(t)\xi_1(t)\xi_2(t) \\ &= G_2[I_6 \otimes T]Mr_6\xi_1(t)\xi_2(t) \\ &= \delta_3[2, 3, 3, 3, 3, 3]\xi_1(t)\xi_2(t) \end{aligned}$$

where $Mr_6 := \text{diag}(\delta_6^1, \delta_6^2, \ldots, \delta_6^6)$ is the power-reducing matrix.

That is, we convert (61) into a standard difference equation

$$\begin{cases} \xi_1(t+1) = \delta_2[2, 1, 2, 1, 2, 1]\xi_1(t)\xi_2(t) \\ \xi_2(t+1) = \delta_3[2, 3, 3, 3, 3, 3]\xi_1(t)\xi_2(t). \end{cases} \tag{64}$$

Example 29: Consider Example 28 again. Assume now the φ_2 is removed. That is

$$\begin{cases}\xi_1(t+1) = G_1\xi \\ \xi_2(t+1) = G_2\xi\end{cases}$$
$$\varphi(\xi_1, \xi_2, \xi_3) = M\xi = \delta_3^3. \quad (65)$$

Then, the corresponding truth matrix of φ with respect to partition (62) becomes

$$T := T_\varphi^{(X,Y)} = \begin{bmatrix} 0 & 1 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 \end{bmatrix}.$$

Then, we have four logical matrices, denoted by H_i, $i = 1, 2, 3, 4$, satisfying $H_i \leq T$. Hence, for φ, there are four APBSs with respect to the partition (62) as

1)

$$\xi_3 = H_1\xi_1\xi_2 = \begin{bmatrix} 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 \end{bmatrix}\xi_1\xi_2.$$

Plugging it into the difference part of (65) yields

$$\begin{aligned}\xi_1(t+1) &= G_1\xi_1(t)\xi_2(t)H_1\xi_1(t)\xi_2(t) \\ &= G_1[I_6 \otimes H_1]\xi_1(t)\xi_2(t)\xi_1(t)\xi_2(t) \\ &= G_1[I_6 \otimes H_1]Mr_6\xi_1(t)\xi_2(t) \\ &= \delta_2[2, 1, 2, 1, 2, 1]\xi_1(t)\xi_2(t)\end{aligned}$$

and

$$\begin{aligned}\xi_2(t+1) &= G_2\xi_1(t)\xi_2(t)H_1\xi_1(t)\xi_2(t) \\ &= G_2[I_6 \otimes H_1]\xi_1(t)\xi_2(t)\xi_1(t)\xi_2(t) \\ &= G_2[I_6 \otimes H_1]Mr_6\xi_1(t)\xi_2(t) \\ &= \delta_3[1, 3, 1, 3, 2, 3]\xi_1(t)\xi_2(t).\end{aligned}$$

Hence, in this case, we reduce the difference-algebraic network (65) into a difference one

$$\begin{cases}\xi_1(t+1) = \delta_2[2, 1, 2, 1, 2, 1]\xi_1(t)\xi_2(t) \\ \xi_2(t+1) = \delta_3[1, 3, 1, 3, 2, 3]\xi_1(t)\xi_2(t).\end{cases} \quad (66)$$

2)

$$\xi_3 = H_2\xi_1\xi_2 = \begin{bmatrix} 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}\xi_1\xi_2.$$

Plugging it into the difference part of (61) yields

$$\begin{cases}\xi_1(t+1) = \delta_2[2, 1, 2, 1, 2, 1]\xi_1(t)\xi_2(t) \\ \xi_2(t+1) = \delta_3[1, 3, 1, 1, 2, 3]\xi_1(t)\xi_2(t).\end{cases} \quad (67)$$

3)

$$\xi_3 = H_3\xi_1\xi_2 = \begin{bmatrix} 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \end{bmatrix}\xi_1\xi_2.$$

Plugging it into the difference part of (61) yields

$$\begin{cases}\xi_1(t+1) = \delta_2[2, 1, 2, 1, 2, 1]\xi_1(t)\xi_2(t) \\ \xi_2(t+1) = \delta_3[2, 3, 1, 3, 2, 3]\xi_1(t)\xi_2(t).\end{cases} \quad (68)$$

4)

$$\xi_3 = H_4\xi_1\xi_2 = \begin{bmatrix} 0 & 1 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}\xi_1\xi_2.$$

Plugging it into the difference part of (61) yields

$$\begin{cases}\xi_1(t+1) = \delta_2[2, 1, 2, 1, 2, 1]\xi_1(t)\xi_2(t) \\ \xi_2(t+1) = \delta_3[2, 3, 1, 1, 2, 3]\xi_1(t)\xi_2(t).\end{cases} \quad (69)$$

Remark 30: Note that when using ACPBS, the reduced difference equation is equivalent to the original difference-algebraic network. So they have the same solutions. For example, the solution of (61) and the solution of (64) are exactly the same. This is because of the logical equivalence between the two logical expressions (i.e., the two static sets of equations) are equivalent. As for APBS substitution, we only know the solution(s) of the reduced difference equation is(are) also solution(s) of the original difference-algebraic network, but not vice versa. As for Example 29, we only know that all the solutions of (66)–(69) are also the solutions of (65), but the reverse is not true. The following example shows this.

Example 31: Consider Example 29 again. It is easy to verify that the following trajectory is a solution of (65):

$$\begin{aligned}\xi_0(t) &= (\delta_2^1, \delta_3^3, \delta_3^2) \to (\delta_2^2, \delta_3^1, \delta_3^1) \\ &\to (\delta_2^1, \delta_3^3, \delta_3^2) \to (\delta_2^2, \delta_3^1, \delta_3^3) \to \cdots\end{aligned}$$

where the rest part "$\cdots$" is assumed to follow (66) [or any fixed one from (66)–(69)]. To see that ξ_0 is not the solution of any one of (66)–(69), it is clear that no one of them consists with both $(\delta_2^2, \delta_3^1, \delta_3^1)$ and $(\delta_2^2, \delta_3^1, \delta_3^3)$.

VII. Conclusion

In this paper, the truth matrix of a static logical network is introduced. Using it the partition-based solutions of a static logical network, including APBS, CPBS, and ACPBS, have been solved completely by providing necessary and sufficient conditions and the corresponding formulas for each case.

Compared with the existing results in the literature, our method presents very easily verifiable necessary and sufficient conditions for the existence of all partition-based solutions. More concise algorithms are given to provide explicit expressions of the solutions. Furthermore, using truth matrix and STP, the obtained results can be extended to mix-valued logical equations immediately.

Finally, the results obtained have been used to address two problems.

1) Find necessary and sufficient condition for the existence of IF. To the best of our knowledge, this is the first IF theorem for mix-valued logical networks.
2) Convert a difference-algebraic logical network into the classical difference form. As the ACPBS substitution

exists, the resulting difference network is equivalent to the original difference-algebraic network. But when the APBS substitution is used, we can only assure the solution(s) of the reduced difference network is(are) the solution(s) of the original difference-algebraic network. A problem remains for further study is: can we find all the solutions by using all possible APBSs? If there are some missing solutions, are we able to find them?

An obvious disadvantage of this approach is the computational complexity. Since the number of different partitions (ignoring symmetric ones) is $2^{n-1} - 1$, when n is not small finding all partitions is a heavy job.

The approach proposed in this paper is very useful for investigating general difference-algebraic logical control networks. Some further results will be delivered in coming papers. As mentioned Section I, logical network and neural network are two different approaches to the genetic network. The combination of logical and numerical approaches may be the further growing direction of neural networks.

References

[1] F. Bazso, "Derivation of vector-valued Boolean functions," *Acta Math. Hung.*, vol. 87, no. 3, pp. 197–203, 2000.

[2] C. Carlet, "Boolean functions for cryptography and error correcting codes," in *Boolean Models and Methods in Mathematics, Computer Science, and Engineering*, Y. Crama and P. L. Hammer, Eds. Cambridge, U.K.: Cambridge Univ. Press, 2010.

[3] D. Cheng and H. Qi, "State–space analysis of Boolean networks," *IEEE Trans. Neural Netw.*, vol. 21, no. 4, pp. 584–594, Apr. 2010.

[4] D. Cheng, H. Qi, and Z. Li, *Analysis and Control of Boolean Networks: A Semi-Tensor Product Approach*. London, U.K.: Springer, 2011.

[5] D. Cheng, H. Qi, and Y. Zhao, *An Introduction to Semi-Tensor Product of Matrices and Its Applications*. Singapore: World Scientific, 2012.

[6] D. Cheng and X. Xu, "Bi-decomposition of multi-valued logical functions and its applications," *Automatica*, vol. 49, no. 7, pp. 1979–1985, 2013.

[7] D. Cheng, F. He, H. Qi, and T. Xu, "Modeling, analysis and control of networked evolutionary games," *IEEE Trans. Autom. Control*, vol. 60, no. 9, pp. 2402–2415, Sep. 2015.

[8] A. G. Hamilton, *Logic for Mathematicians*. Cambridge, U.K.: Cambridge Univ. Press, 1988.

[9] C. G. Khatri and C. R. Rao, "Solutions to some functional equations and their applications to characterization of probability distributions," *Sankhyā, Indian J. Statist. A*, vol. 30, no. 2, pp. 167–180, 1968.

[10] K. H. Kim, *Boolean Matrix Theory and Applications*. New York, NY, USA: Marcel Dekker, 1982.

[11] R. S. Ledley, "Digital computational methods in symbolic logic, with examples in biochemistry," *Proc. Nat. Acad. Sci. USA*, vol. 41, no. 7, pp. 498–511, 1955.

[12] W. S. McCulloch and W. Pitts, "A logical calculus of the ideas immanent in nervous activity," *Bull. Math. Biophys.*, vol. 5, no. 4, pp. 115–133, 1943.

[13] M. M. Waldrop, *Complexity: The Emerging Science at the Edge of Order and Chaos*. New York, NY, USA: Simon & Schuster, 1992.

[14] G.-Q. Zhang, "Automata, Boolean matrices, and ultimate periodicity," *Inf. Comput.*, vol. 152, no. 1, pp. 138–154, 1999.

Yupeng Qiao received the Ph.D. degree in systems theory from the Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing, China, in 2008.

From 2008 to 2010, she was a Post-Doctoral Fellow with the College of Automation Science and Engineering, South China University of Technology, Guangzhou, China, where she is currently an Associate Professor. Her current research interests include nonlinear control and game theory.

Hongsheng Qi (M'10) received the Ph.D. degree in systems theory from the Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing, China, in 2008.

From 2008 to 2010, he was a Post-Doctoral Fellow with the Key Laboratory of Systems Control, Chinese Academy of Sciences, where he is currently an Associate Professor with the Academy of Mathematics and Systems Science. His current research interests include nonlinear control, logical dynamic systems, and game theory.

Dr. Qi was a recipient of *Automatica* 2008–2010 Theory/Methodology Best Paper Prize in 2011 and the second recipient of the second National Natural Science Award of China in 2014.

Daizhan Cheng (SM'01–F'06) received the B.S. degree from the Department of Mechanics, Tsinghua University, Beijing, China, in 1970, the M.S. degree from the Graduate School of Chinese Academy of Sciences, Beijing, in 1981, and the Ph.D. degree from Washington University, St. Louis, MO, USA, in 1985.

Since 1990, he has been a Professor with the Institute of Systems Science, Academy of Mathematics and Systems Science, Chinese Academy of Sciences. He has authored or co-authored over 200 journal papers, 11 books, and over 100 conference papers. His current research interests include nonlinear control systems, switched systems, Hamiltonian systems, logical dynamic systems, and numerical realization for control design.

Dr. Cheng has been an IFAC Fellow since 2008. He was a recipient of the Second National Natural Science Award of China in 2008 and 2014 and a recipient of the *Automatica* 2008–2010 Theory/Methodology Best Paper Prize in 2011. He was an Associate Editor of the *International Journal of Mathematical Systems, Estimation and Control* from 1990 to 1993, the *Automatica* from 1999 to 2002, and the *Asian Journal of Control* from 2001 to 2004, a Subject Editor of the *International Journal of Robust and Nonlinear Control* from 2005 to 2008, an Associate Editor of the *Journal of Systems Science and Complexity*, the *Journal of Control Theory and Applications*, and the *Journal Systems Science and Mathematics*, and the Editor-in-Chief of the *Journal of Control Theory and Applications*. He was the Chairman of the IEEE CSS Beijing Chapter from 2006 to 2008, the Chairman of the Technical Committee on Control Theory of Chinese Association of Automation from 2003 to 2009. He is currently the Deputy Editor-in-Chief of the *Control and Decision*.