

Matrix Expression of Pan Boolean Algebra PID Control

Chen Jin^{1, a}, Xu Yubiao^{1, b}, Hu Yeheng^{1, c}

¹ College of Engineering, Shanghai Second Polytechnic University, Shanghai, 201209, China

^aemail: chenjin@sspu.edu.cn, ^bemail:601435027@qq.com, ^cemail: yhhu@sspu.edu.cn

Keywords: Pan Boolean algebra; PID control; semi-tensor product of matrices

Abstract. By using the semi-tensor product of matrices theory method, this paper studies the Pan Boolean algebra PID control algorithm's conversion problem. It was found that the power computation of n-ary logical structure matrix can be converted into counting its (n+1)-ary structure matrix by its truth table. The Pan Boolean algebra PID control algorithm consisted of nine Pan Boolean logic equations is expressed by a set of structure matrix equations which have only three structure matrices. So the Pan Boolean logic algorithm can be converted into ordinary algebraic control algorithm.

Introduction

Pan Boolean algebra control derived from the Pan Boolean algebra in the early 80's^[1], After several decades of development, the Pan Boolean algebra control have theory and application of the systematic research achievements in the single variable control system design and control, the nine point control theory systematically discusses the Pan Boolean algebra controller design, dynamic analysis, static characteristics analysis and system stability analysis^[2]. The literature describes the parameters analysis of spatial Pan Boolean algebra of PID control, the 27 control parameters of Pan Boolean algebra of PID controller are described in this coordinate system^[3], It is used for the analysis of indicators of the performance of this controller. However, so far, because the classical mathematical tools of calculus and matrix theory can't be used for logic control research, Pan Boolean algebra is the main tool of Pan Boolean algebra control theory, available mathematical tools is very little, so it is difficult to achieve a breakthrough. In recent years, the semi tensor product of matrices theory method proposed by Cheng Daizhan professor and other scholars, who have achieved fruitful results in the logical problem analysis and control research field^{[4][6][7]}, and provides a good research tools for the Pan Boolean algebra control.

It is described in this paper that the Pan Boolean algebra PID control algorithm can be converted into an ordinary algebra PID control algorithm by the semi-tensor product theory and method. The main contents of this paper are as follows:

- (1) The power computation of Boolean logic structure matrix can be converted into counting its structure matrix which can be easily calculated by its truth table.
- (2) A set of Pan Boolean algebra PID control algorithm is expressed by a set of structure matrix algebra equation.
- (3) A logical structure matrix depended on the logical structure of its logical formula.

Matrix expression of Boolean logic operation

Definition 1^[6] The Boolean logic domain \mathcal{D}_t is

$$\mathcal{D}_t = \{T = 1, F = 0\}.$$

Definition 2^[6] A Boolean logic variable P is a variable which can take value from \mathcal{D}_t , that is $P \in \mathcal{D}_t$.

To obtain a matrix expression, we identify "T" and "F", respectively, with the vectors

$$T := 1 \equiv \begin{bmatrix} 1 \\ 0 \end{bmatrix}; F := 0 \equiv \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad (2)$$

Definition 3^[1] A Pan Boolean logic variable x_i^j is a variable which can take value from \mathcal{D}_i , that is $x_i^j \in \mathcal{D}_i$, $i = 1, 2, \dots, n, j = 1, 2, \dots, m$. Different from the classical Boolean logic variables, Pan Boolean logic variables also satisfy the following conditions:

$$x_i^1 \vee x_i^2 \vee \dots \vee x_i^m = 1. \quad (3)$$

$$x_i^1 \wedge x_i^2 \wedge \dots \wedge x_i^m = 0. \quad (4)$$

Definition 4^[6] Given a matrix $A \in M_{p \times q}$ such that $p \% q = 0$ or $q \% p = 0$, we define a A^n , $n > 0$, inductively as

$$\begin{cases} A^1 = A \\ A^{k+1} = A^k \otimes A, k = 1, 2, \dots \end{cases} \quad (5)$$

Theorem 1^[6] If $A \in M_{m \times n}$, $B \in M_{n \times t}$, then

$$A \otimes B = W_{[s,m]} \otimes B \otimes W_{[m,t]} \otimes A = (I_m \otimes B) \otimes A.$$

Theorem 2^[6], Given a logical function $\sigma(P_1, \dots, P_r)$ with logical variables P_1, \dots, P_r , there exists a unique 2×2^r matrix M_σ , called the structure matrix of σ , such that

$$\sigma(P_1, \dots, P_r) = M_\sigma \otimes P_1 \otimes P_2 \dots \otimes P_r \quad (6)$$

where $M_\sigma \in M_{2 \times 2^r}$.

Theorem 3^[6], if r-ary operator σ 's truth table is $[s_1, s_2, \dots, s_{2^r}]^T$, then its structure matrix is

$$M_\sigma = \begin{bmatrix} s_1 & s_2 & \dots & s_{2^r} \\ 1 - s_1 & 1 - s_2 & \dots & 1 - s_{2^r} \end{bmatrix} \quad (7)$$

Where $M_\sigma \in M_{2 \times 2^r}$

Application of Theorem 2 and theorem 3, we can get the r-ary Boolean logic "conjunction" operation's matrix expression and "disjunction" operation's matrix expressions are as follows:

If $y = P_1 \wedge P_2 \wedge \dots \wedge P_r$, then

$$y = M_{\wedge r} \otimes P_1 \otimes P_2 \dots \otimes P_r \quad (8)$$

$$M_{\wedge r} = \begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 1 \end{bmatrix} \quad (9)$$

Where, $M_{\wedge r} \in M_{2 \times 2^r}$.

If $y = P_1 \vee P_2 \vee \dots \vee P_r$, then

$$y = M_{\vee r} \otimes P_1 \otimes P_2 \dots \otimes P_r \quad (10)$$

$$M_{\vee r} = \begin{bmatrix} 1 & 1 & \dots & 1 \\ 0 & 0 & \dots & 0 \end{bmatrix} \quad (11)$$

Where, $M_{\vee r} \in M_{2 \times 2^r}$.

Especially, if $r=2$, then

$$M_{\vee 2} = M_{\vee} = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (12)$$

$$M_{\wedge 2} = M_{\wedge} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix} \quad (13)$$

Theorem 4 Power operation laws of Boolean logic structure matrix M_{\wedge}

$$M_{\wedge}^n = M_{\wedge} \otimes \dots \otimes M_{\wedge} = M_{\wedge}^{(n+1)} \quad (14)$$

Theorem 5 Power operation laws of Boolean logic structure matrix M_{\vee}

$$M_{\vee}^n = M_{\vee} \otimes \dots \otimes M_{\vee} = M_{\vee}^{(n+1)} \quad (15)$$

Proof (Theorem 4):

When $n=2$, there is

$$M_{\wedge}^2 = M_{\wedge} \otimes M_{\wedge} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix} \otimes \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix}$$

$$M_{\wedge}^2 = \begin{bmatrix} 1 & \dots & 2^{1+2} \\ \downarrow & & \downarrow \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} = M_{\wedge 3}$$

Assume when $n = k$, theorem 4 is right, there are

$$M_{\wedge}^k = M_{\wedge} \otimes \dots \otimes M_{\wedge} = M_{\wedge}^{(k+1)}$$

$$M_{\wedge}^{k+1} = M_{\wedge}^k \otimes M_{\wedge} = \begin{bmatrix} 1 & \dots & 2^{k_{\downarrow}} \\ \downarrow & 1 & 0 & \dots & 0 & 0 \\ 0 & 1 & \dots & 1 & 1 \end{bmatrix} \otimes \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & \dots & 2^{k+2_{\downarrow}} \\ \downarrow & 1 & 0 & \dots & 0 & 0 \\ 0 & 1 & \dots & 1 & 1 \end{bmatrix} = M_{\wedge}^{(k+2)}$$

Proof finished.

Similarly, we can prove Theorem 5.

Now we give two illustrative examples.

The truth table for binary logic operator conjunction and disjunction as shown in table 1, Boolean logic expressions of binary Boolean logic conjunction and disjunction logic operations are as follows.

$$y_1^1 = x_1^1 \wedge x_1^1 \tag{16}$$

$$y_1^2 = x_1^1 \vee x_1^2 \tag{17}$$

Table 1 truth table for binary logic operator “ \wedge ” “ \vee ”

x_1^1	x_1^2	$y_1^1(\wedge)$	$y_1^2(\vee)$
1	1	1	1
1	0	0	1
0	1	0	1
0	0	0	0

By applying the semi-tensor product theory, binary Boolean logic conjunction and disjunction operation can be expressed as:

$$y_1^1 = x_1^1 \wedge x_1^1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix} \otimes x_1^1 \otimes x_1^1$$

$$= M_{\wedge} \otimes x_1^1 \otimes x_1^1 \tag{18}$$

$$y_1^2 = x_1^1 \vee x_1^1 = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \otimes x_1^1 \otimes x_1^1$$

$$= M_{\vee} \otimes x_1^1 \otimes x_1^1 \tag{19}$$

Table 2 truth table for treble logic operator “ \wedge ” “ \vee ”

x_1^1	x_1^2	x_1^3	$y_1^1(\wedge)$	$y_1^2(\vee)$
1	1	1	1	1
1	1	0	0	1
1	1	1	0	1
1	0	0	0	1
0	1	1	0	1
0	1	0	0	1
0	0	1	0	1
0	0	0	0	0

The truth table for treble logic operator conjunction and disjunction as shown in table 2, Boolean logic expressions of treble Boolean logic conjunction and disjunction logic operations are as follows.

$$y_1^1 = x_1^1 \wedge x_1^2 \wedge x_1^3 \tag{20}$$

$$y_1^2 = x_1^1 \vee x_1^2 \vee x_1^3 \tag{21}$$

By applying the semi-tensor product theory, treble Boolean logic conjunction and disjunction operation can be expressed as:

$$y_1^1 = x_1^1 \wedge x_1^2 \wedge x_1^3$$

$$= \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \otimes x_1^1 \otimes x_1^2 \otimes x_1^3$$

$$= M_{\wedge 3} \otimes x_1^1 \otimes x_1^2 \otimes x_1^3 \tag{22}$$

$$y_1^2 = x_1^1 \vee x_1^2 \vee x_1^3$$

$$= \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \times x_1^1 \times x_1^2 \times x_1^3$$

$$= M_{V_3} \times x_1^1 \times x_1^2 \times x_1^3 \tag{23}$$

$$M_{A_3} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \tag{24}$$

Called M_{A_3} as the treble logic conjunction operator's structure matrix.

$$M_{V_3} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{25}$$

Called M_{V_3} as the treble logic disjunction operator's structure matrix.

Boolean logic algorithm of Pan Boolean algebra PID control

As shown the Fig.1, given value (R (t))=Expected value when the system operates; Deviation (E (t))= Given value - Output value (C (t)); Deviation integral: $\int E(t)dt$; Deviation differential: $dE(t)/dt$; The controller outputs Y(Control action). Control action is divided into 3 kinds. Each kind can further subdivide into grades. In this paper, control action is divided into nine grades: Strengthen (Add a lot, add some, add weakly, add a little,), Keep, weaken (reduce a little, reduce weakly, reduce some, reduce a lot).

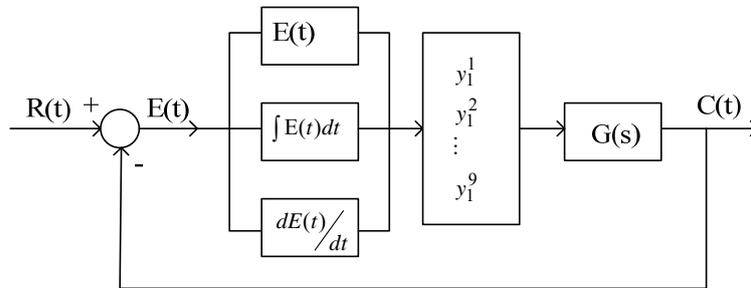


Fig.1: Pan Boolean algebra PID control diagram

With system deviation, rate of change of deviation (Differential), accumulated deviation (Integral) as input, controller actions according to some rules to keep system output steady around the given value.

Now the control rules of controller are represented using pan-Boolean algebra. Let X1 denote deviation, and x_1^1, x_1^2, x_1^3 denote three states respectively: deviation is greater than 0 (positive), deviation is in zero zone (zero), deviation is less than zero (negative); let X2 denote rate of change of deviation, and x_2^1, x_2^2, x_2^3 denote three states respectively: rate of change of deviation is greater than 0 (positive), rate of change of deviation is in zero zone (zero), rate of change of deviation is less than zero (negative); Let X3 denote deviation integral, and x_3^1, x_3^2, x_3^3 denote three states respectively: accumulated deviation is greater than 0 (positive), accumulated deviation is in zero zone (zero), accumulated deviation is less than zero (negative); Let y denotes control rules, and $y_1^1, y_1^2, y_1^3, y_1^4, y_1^5, y_1^6, y_1^7, y_1^8, y_1^9$ denotes 9 control rules: add a lot, add some, add weakly, add a little, keep, reduce a little, reduce weakly, reduce some, reduce a lot.

The Pan Boolean algebra PID control algorithm as formula (24) - (32) are shown [5].

$$y_1^1 = (x_1^1 \wedge x_2^2 \wedge x_3^3) \vee (x_1^1 \wedge x_2^3 \wedge x_3^3) \vee (x_1^1 \wedge x_2^1 \wedge x_3^2) \tag{26}$$

$$y_1^2 = (x_1^1 \wedge x_2^2 \wedge x_3^1) \vee (x_1^1 \wedge x_2^2 \wedge x_3^2) \vee (x_1^1 \wedge x_2^1 \wedge x_3^3) \tag{27}$$

$$y_1^3 = (x_1^1 \wedge x_2^1 \wedge x_3^2) \vee (x_1^1 \wedge x_2^2 \wedge x_3^1) \vee (x_1^1 \wedge x_2^3 \wedge x_3^2) \vee (x_1^2 \wedge x_2^3 \wedge x_3^3) \tag{28}$$

$$y_1^4 = (x_1^2 \wedge x_2^1 \wedge x_3^2) \vee (x_1^2 \wedge x_2^1 \wedge x_3^3) \vee (x_1^2 \wedge x_2^2 \wedge x_3^3) \tag{29}$$

$$y_1^5 = x_1^2 \wedge x_2^2 \wedge x_3^2 \tag{30}$$

$$y_1^6 = (x_1^2 \wedge x_2^2 \wedge x_3^1) \vee (x_1^2 \wedge x_2^3 \wedge x_3^1) \vee (x_1^2 \wedge x_2^3 \wedge x_3^2) \tag{31}$$

$$y_1^7 = (x_1^3 \wedge x_2^1 \wedge x_3^2) \vee (x_1^3 \wedge x_2^2 \wedge x_3^2) \vee (x_1^3 \wedge x_2^3 \wedge x_3^3) \vee (x_1^2 \wedge x_2^1 \wedge x_3^1) \tag{32}$$

$$y_1^8 = (x_1^3 \wedge x_2^1 \wedge x_3^3) \vee (x_1^3 \wedge x_2^2 \wedge x_3^2) \vee (x_1^3 \wedge x_2^3 \wedge x_3^1) \tag{33}$$

$$y_1^9 = (x_1^3 \wedge x_2^1 \wedge x_3^1) \vee (x_1^3 \wedge x_2^2 \wedge x_3^1) \vee (x_1^3 \wedge x_2^3 \wedge x_3^2) \tag{34}$$

Structure Matrix of Pan Boolean Algebra PID Control Algorithm

By applying the theorem 1 and theorem 2, the formula (26) can be converted into semi-tensor product formula, the operation process are as follows.

$$\begin{aligned}
 y_1^1 &= (M_{A_1} \bowtie (x_1^1 \bowtie x_2^2) \wedge x_3^3) V(M_{A_1} \bowtie (x_1^1 \bowtie x_2^2) \wedge x_3^3) V(M_{A_1} \bowtie (x_1^1 \bowtie x_2^2) \wedge x_3^3) \\
 &= (M_{A_1} \bowtie M_{A_1} \bowtie (x_1^1 \bowtie x_2^2 \bowtie x_3^3)) V(M_{A_1} \bowtie M_{A_1} \bowtie (x_1^1 \bowtie x_2^2 \bowtie x_3^3)) V(M_{A_1} \bowtie M_{A_1} \bowtie (x_1^1 \bowtie x_2^2 \bowtie x_3^3)) \\
 &= (M_{A_1}^2 \bowtie (x_1^1 \bowtie x_2^2 \bowtie x_3^3)) V(M_{A_1}^2 \bowtie (x_1^1 \bowtie x_2^2 \bowtie x_3^3)) V(M_{A_1}^2 \bowtie (x_1^1 \bowtie x_2^2 \bowtie x_3^3)) \\
 &= M_{V_1} \bowtie ((M_{A_1}^2 \bowtie (x_1^1 \bowtie x_2^2 \bowtie x_3^3)) \bowtie (M_{A_1}^2 \bowtie (x_1^1 \bowtie x_2^2 \bowtie x_3^3))) V(M_{A_1}^2 \bowtie (x_1^1 \bowtie x_2^2 \bowtie x_3^3)) \\
 &= M_{V_1} \bowtie M_{V_1} \bowtie ((M_{A_1}^2 \bowtie (x_1^1 \bowtie x_2^2 \bowtie x_3^3)) \bowtie (M_{A_1}^2 \bowtie (x_1^1 \bowtie x_2^2 \bowtie x_3^3))) \bowtie (M_{A_1}^2 \bowtie (x_1^1 \bowtie x_2^2 \bowtie x_3^3)) \\
 &= M_{V_1}^2 \bowtie ((M_{A_1}^2 \bowtie (x_1^1 \bowtie x_2^2 \bowtie x_3^3)) \bowtie (M_{A_1}^2 \bowtie (x_1^1 \bowtie x_2^2 \bowtie x_3^3))) \bowtie (M_{A_1}^2 \bowtie (x_1^1 \bowtie x_2^2 \bowtie x_3^3)) \\
 &= M_{V_1}^2 \bowtie (M_{A_1}^2 \bowtie (x_1^1 \bowtie x_2^2 \bowtie x_3^3) \bowtie M_{A_1}^2 \bowtie (x_1^1 \bowtie x_2^2 \bowtie x_3^3) \bowtie M_{A_1}^2 \bowtie (x_1^1 \bowtie x_2^2 \bowtie x_3^3)) \\
 &= M_{V_1}^2 \bowtie M_{A_1}^2 \bowtie (x_1^1 \bowtie x_2^2 \bowtie x_3^3) \bowtie M_{A_1}^2 \bowtie (x_1^1 \bowtie x_2^2 \bowtie x_3^3) \bowtie M_{A_1}^2 \bowtie (x_1^1 \bowtie x_2^2 \bowtie x_3^3) \\
 &= M_{V_3} \bowtie M_{A_3} \bowtie (x_1^1 \bowtie x_2^2 \bowtie x_3^3) \bowtie M_{A_3} \bowtie (x_1^1 \bowtie x_2^2 \bowtie x_3^3) \bowtie M_{A_3} \bowtie (x_1^1 \bowtie x_2^2 \bowtie x_3^3) \\
 &= M_{V_3} \bowtie M_{A_3} \bowtie (x_1^1 \bowtie x_2^2 \bowtie x_3^3) \bowtie M_{A_3} \bowtie (I_9 \otimes M_{A_3}) \bowtie (x_1^1 \bowtie x_2^2 \bowtie x_3^3) \bowtie (x_1^1 \bowtie x_2^2 \bowtie x_3^3) \\
 &= M_{V_3} \bowtie M_{A_3} \bowtie (I_9 \otimes M_{A_3} \bowtie I_9 \otimes M_{A_3}) \bowtie (x_1^1 \bowtie x_2^2 \bowtie x_3^3) \bowtie (x_1^1 \bowtie x_2^2 \bowtie x_3^3) \bowtie (x_1^1 \bowtie x_2^2 \bowtie x_3^3) \\
 y_1^1 &= M_{\sigma_1} \bowtie (x_1^1 \bowtie x_2^2 \bowtie x_3^3) \bowtie (x_1^1 \bowtie x_2^2 \bowtie x_3^3) \bowtie (x_1^1 \bowtie x_2^2 \bowtie x_3^3) \tag{35} \\
 M_{\sigma_1} &= M_{V_3} \bowtie M_{A_3} \bowtie I_9 \otimes M_{A_3} \bowtie I_9 \otimes M_{A_3} \tag{36}
 \end{aligned}$$

M_{σ_1} is called the structure matrix of Pan Boolean algebra PID control algorithm y_1^1 . Similarly, the formula from (27) to (34) can be converted into semi-tensor product expression, as follows.

$$\begin{aligned}
 y_1^2 &= M_{V_3} \bowtie M_{A_3} \bowtie (x_1^1 \bowtie x_2^2 \bowtie x_3^3) \bowtie M_{A_3} \bowtie (x_1^1 \bowtie x_2^2 \bowtie x_3^3) \bowtie M_{A_3} \bowtie (x_1^1 \bowtie x_2^2 \bowtie x_3^3) \\
 &= M_{\sigma_2} \bowtie (x_1^1 \bowtie x_2^2 \bowtie x_3^3) \bowtie (x_1^1 \bowtie x_2^2 \bowtie x_3^3) \bowtie (x_1^1 \bowtie x_2^2 \bowtie x_3^3) \tag{37}
 \end{aligned}$$

$$M_{\sigma_2} = M_{V_3} \bowtie M_{A_3} \bowtie I_9 \otimes M_{A_3} \bowtie I_9 \otimes M_{A_3} \tag{38}$$

M_{σ_2} is called structure matrix of Pan Boolean algebra PID control algorithm y_1^2 .

$$\begin{aligned}
 y_1^3 &= M_{V_3} \bowtie M_{A_3} \bowtie (x_1^1 \bowtie x_2^2 \bowtie x_3^3) \\
 &= M_{\sigma_3} \bowtie (x_1^1 \bowtie x_2^2 \bowtie x_3^3) \tag{39}
 \end{aligned}$$

$$M_{\sigma_3} = M_{V_3} \bowtie M_{A_3} \bowtie I_2 \otimes M_{A_3} \bowtie I_2 \otimes M_{A_3} \bowtie I_2 \otimes M_{A_3} \tag{40}$$

M_{σ_3} is called structure matrix of Pan Boolean algebra PID control algorithm y_1^3 .

$$\begin{aligned}
 y_1^4 &= M_{V_3} \bowtie M_{A_3} \bowtie (x_1^2 \bowtie x_2^1 \bowtie x_3^2) \bowtie M_{A_3} \bowtie (x_1^2 \bowtie x_2^1 \bowtie x_3^2) \bowtie M_{A_3} \bowtie (x_1^2 \bowtie x_2^1 \bowtie x_3^2) \\
 &= M_{\sigma_4} \bowtie (x_1^2 \bowtie x_2^1 \bowtie x_3^2) \bowtie (x_1^2 \bowtie x_2^1 \bowtie x_3^2) \bowtie (x_1^2 \bowtie x_2^1 \bowtie x_3^2) \tag{41}
 \end{aligned}$$

$$M_{\sigma_4} = M_{V_3} \bowtie M_{A_3} \bowtie I_9 \otimes M_{A_3} \bowtie I_9 \otimes M_{A_3} \tag{42}$$

M_{σ_4} is called structure matrix of Pan Boolean algebra PID control algorithm y_1^4 .

$$\begin{aligned}
 y_1^5 &= M_{V_3} \bowtie (x_1^2 \bowtie x_2^2 \bowtie x_3^2) \\
 &= M_{\sigma_5} \bowtie x_1^2 \bowtie x_2^2 \bowtie x_3^2 \tag{43}
 \end{aligned}$$

$$M_{\sigma_5} = M_{V_3} \tag{44}$$

M_{σ_5} is called structure matrix of Pan Boolean algebra PID control algorithm y_1^5 .

$$\begin{aligned}
 y_1^6 &= M_{V_1}^2 \bowtie M_{A_1}^2 \bowtie (x_1^2 \bowtie x_2^2 \bowtie x_3^2) \bowtie M_{A_1}^2 \bowtie (x_1^2 \bowtie x_2^2 \bowtie x_3^2) \bowtie M_{A_1}^2 \bowtie (x_1^2 \bowtie x_2^2 \bowtie x_3^2) \\
 &= M_{\sigma_6} \bowtie (x_1^2 \bowtie x_2^2 \bowtie x_3^2) \bowtie (x_1^2 \bowtie x_2^2 \bowtie x_3^2) \bowtie (x_1^2 \bowtie x_2^2 \bowtie x_3^2) \tag{45}
 \end{aligned}$$

$$M_{\sigma_6} = M_{V_3} \bowtie M_{A_3} \bowtie I_9 \otimes M_{A_3} \bowtie I_9 \otimes M_{A_3} \tag{46}$$

M_{σ_6} is called structure matrix of Pan Boolean algebra PID control algorithm y_1^6 .

$$\begin{aligned}
 y_1^7 &= M_{V_1}^2 \bowtie M_{A_1}^2 \bowtie (x_1^3 \bowtie x_2^1 \bowtie x_3^2) \\
 &= M_{\sigma_7} \bowtie (x_1^3 \bowtie x_2^1 \bowtie x_3^2) \tag{47}
 \end{aligned}$$

$$M_{\sigma_7} = M_{V_3} \bowtie M_{A_3} \bowtie I_2 \otimes M_{A_3} \bowtie I_2 \otimes M_{A_3} \bowtie I_2 \otimes M_{A_3} \tag{48}$$

M_{σ_7} is called structure matrix of Pan Boolean algebra PID control algorithm y_1^7 .

$$y_1^8 = M_{\sigma_8} \bowtie (x_1^3 \bowtie x_2^2 \bowtie x_3^2) \bowtie (x_1^3 \bowtie x_2^2 \bowtie x_3^2) \bowtie (x_1^3 \bowtie x_2^2 \bowtie x_3^2) \tag{49}$$

$$M_{\sigma_8} = M_{V_3} \bowtie M_{A_3} \bowtie I_9 \otimes M_{A_3} \bowtie I_9 \otimes M_{A_3} \tag{50}$$

M_{σ_8} is called structure matrix of Pan Boolean algebra PID control algorithm y_1^8 .

$$\begin{aligned}
 y_1^9 &= M_{V_1}^2 \bowtie M_{A_1}^2 \bowtie (x_1^3 \bowtie x_2^1 \bowtie x_3^2) \bowtie M_{A_1}^2 \bowtie (x_1^3 \bowtie x_2^1 \bowtie x_3^2) \bowtie M_{A_1}^2 \bowtie (x_1^3 \bowtie x_2^1 \bowtie x_3^2) \\
 &= M_{\sigma_9} \bowtie (x_1^3 \bowtie x_2^1 \bowtie x_3^2) \bowtie (x_1^3 \bowtie x_2^1 \bowtie x_3^2) \bowtie (x_1^3 \bowtie x_2^1 \bowtie x_3^2) \tag{51}
 \end{aligned}$$

$$M_{\sigma_9} = M_{V_3} \bowtie M_{A_3} \bowtie I_9 \otimes M_{A_3} \bowtie I_9 \otimes M_{A_3} \tag{52}$$

M_{σ_9} is called structure matrix of Pan Boolean algebra PID control algorithm y_1^9 .

Discussion and Results

In section 2, the theorem 4 and theorem 5 simplifies the power calculation of logical structure matrix. In fact, the theorem 4 can be extended to general Boolean logical structure matrix calculation, such as conditional and all binary operators.

As shown formula (26),(27),(29),(31),(33) and formula (34), it is found that there is the same logical structure in the several formula, which is consisted of three logical conjunction items and two logical disjunction operations. So their structure matrices, such that $M_{\sigma 1}$ 、 $M_{\sigma 2}$ 、 $M_{\sigma 4}$ 、 $M_{\sigma 6}$ 、 $M_{\sigma 8}$ 、 $M_{\sigma 9}$ are the same. Similarly, because the logical structure of the formula (28) and formula (32) are the same, their structure matrices $M_{\sigma 3}$ and $M_{\sigma 7}$ are the same. Consequently, nine structure matrix's calculation can be converted into three structure matrix's ($M_{\sigma 1}$, $M_{\sigma 3}$, $M_{\sigma 5}$) calculation. When the Pan Boolean algebra PID control algorithm expressed by the semi-tensor product, the logical control algorithm's calculation process becomes simple. Moreover, it is important that this is a conversion from Boolean logical algebra method into ordinary algebra method. We can use ordinary algebra tools to research the logical algebra control problems.

Acknowledgement

The research work was supported by Shanghai Second Polytechnic University key subject construction project (Project No. XXKPY1609).

The research work was supported by the Graduate Foundation of Shanghai Second Polytechnic University.

References

- [1] Zhang Nanguan, Xiao Xian, Zhu Wujia, Pan Boolean algebra axiom system [J]. Journal of Air Force Institute of meteorology, 1985 (1): 338 ~ 353
- [2] Zhang Nanguan, The new control principle [B], National Defence Industry Press, 2003.
- [3] Jin Chen, Guanlin Chen, Zhenhua Wang, haihui Song, Jin Zhou, PID Pan Boolean Algebra Controller Parameters Space Analysis[C], Proceedings of the 30th Chinese Control Conference: 3840-3844, July 22-24, 2011, Yantai, China.
- [4] CHENG Dai-Zhan¹, QI Hong-Sheng¹, ZHAO Yin, Analysis and Control of Boolean Networks: A Semi-tensor Product Approach, ACTA AUTOMATICA SINICA, 2011, Vol. 37 (5): 529-539.
- [5] Chen, J., Zhou, J., Hong, X., Chen, G., Wang, Z.: PID Pan-Boolean algebra control. Journal of Central South University(Science and Technology)2009, 40(Sppl.1): 154–157.
- [6] Cheng Dai-Zhan, Qi Hong-Sheng. Semi-tensor Product of Matrices: Theory and Applications. Beijing: Science Press,2007.
- [7] Daizhan Cheng, Hongsheng Qi, Zhiqiang Li, Analysis and Control of Boolean Networks[B], Springer London Ltd; 2011.
- [8] Cheng D Z, Qi H S. A linear representation of dynamics of Boolean networks. IEEE Transactions on Automatic Control, 2010, 55(10): 2251-2258.
- [9] Cheng D Z, Qi H S. Controllability and observability of Boolean control networks. Automatica, 2009, 45(7):1659-1667.