From STP to Logical Dynamic Systems

Daizhan Cheng

Institute of Systems Science
Academy of Mathematics and Systems Science
Chinese Academy of Sciences

Seminar at Dept. of Mathematics
Shanghai Jiao Tong University
Shanghai, April 29, 2015

Outline of Presentation

I. Semi-tensor Product of Matrices

☞ Tensor (Kronecker) Product

$$A_{m\times n} \otimes B_{p\times q} :=$$

$$\begin{bmatrix} a_{11}B & a_{12}B & \cdots & a_{1m}B \\ a_{11}B & a_{12}B & \cdots & a_{1m}B \\ \vdots & & & \\ a_{11}B & a_{12}B & \cdots & a_{1m}B \end{bmatrix} \in \mathcal{M}_{mp\times nq}.$$

☞ An Example

Example 1.1

$$A = \begin{bmatrix} a & b & c \\ d & e & f \end{bmatrix}, \quad B = I_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

$$A \otimes B = \begin{bmatrix} a & 0 & b & 0 & c & 0 \\ 0 & a & 0 & b & 0 & c \\ d & 0 & e & 0 & f & 0 \\ 0 & d & 0 & e & 0 & f \end{bmatrix}.$$

☞ Semi-tensor Product of Matrices

$A_{m\times n} \times B_{p\times q} =?$

Definition 1.2

Let $A \in \mathcal{M}_{m\times n}$ and $B \in \mathcal{M}_{p\times q}$. Denote

$$t := \operatorname{lcm}(n, p).$$

Then we define the semi-tensor product (STP) of A and B as

$$A \ltimes B := \left(A \otimes I_{t/n}\right)\left(B \otimes I_{t/p}\right) \in \mathcal{M}_{(mt/n)\times(qt/p)}. \tag{1}$$

☞ Some Basic Comments

- When $n = p$, A n $B = AB$. So the STP is a generalization of conventional matrix product.
- When $n = rp$, denote it by $A \succ_r B$;
  when $rn = p$, denote it by $A \prec_r B$.
  These two cases are called the multi-dimensional case, which is particularly important in applications.
- STP keeps almost all the major properties of the conventional matrix product unchanged.

☞ Examples

Example 1.3

1. Let $X = \begin{bmatrix} 1 & 2 & 3 & -1 \end{bmatrix}$ and $Y = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$. Then

$$X \textcolor{red}{\text{n}}\ Y = \begin{bmatrix} 1 & 2 \end{bmatrix} \cdot 1 + \begin{bmatrix} 3 & -1 \end{bmatrix} \cdot 2 = \begin{bmatrix} 7 & 0 \end{bmatrix}.$$

2. Let $X = \begin{bmatrix} -1 & 2 & 1 & -1 & 2 & 3 \end{bmatrix}^T$ and $Y = \begin{bmatrix} 1 & 2 & -2 \end{bmatrix}$. Then

$$X \textcolor{red}{\text{n}}\ Y = \begin{bmatrix} -1 \\ 2 \end{bmatrix} \cdot 1 + \begin{bmatrix} 1 \\ -1 \end{bmatrix} \cdot 2 + \begin{bmatrix} 2 \\ 3 \end{bmatrix} \cdot (-2) = \begin{bmatrix} -3 \\ -6 \end{bmatrix}.$$

Example 1.3 (Continued)

3. Let

$$A = \begin{bmatrix} 1 & 2 & 1 & 1 \\ 2 & 3 & 1 & 2 \\ 3 & 2 & 1 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 1 & -2 \\ 2 & -1 \end{bmatrix}.$$

Then

$$A \text{ n } B = \begin{bmatrix} \begin{bmatrix} 1 & 2 & 1 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \end{bmatrix} & \begin{bmatrix} 1 & 2 & 1 & 1 \end{bmatrix} \begin{bmatrix} -2 \\ -1 \end{bmatrix} \\ \begin{bmatrix} 2 & 3 & 1 & 2 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \end{bmatrix} & \begin{bmatrix} 2 & 3 & 1 & 2 \end{bmatrix} \begin{bmatrix} -2 \\ -1 \end{bmatrix} \\ \begin{bmatrix} 3 & 2 & 1 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \end{bmatrix} & \begin{bmatrix} 3 & 2 & 1 & 0 \end{bmatrix} \begin{bmatrix} -2 \\ -1 \end{bmatrix} \end{bmatrix}$$

$$= \begin{bmatrix} 3 & 4 & -3 & -5 \\ 4 & 7 & -5 & -8 \\ 5 & 2 & -7 & -4 \end{bmatrix}.$$

☞ Insight Meaning

Let $A \in \mathcal{M}_{m\times n}$. Consider a bilinear form

$$P(x, y) = x^T Ay. \tag{2}$$

Row Stacking Form:

$$V_r(A) = (a_{11}, a_{12}, \cdots, a_{1n}, \cdots, a_{m1}, \cdots, a_{mn}).$$

Column Stacking Form

$$V_c(A) = (a_{11}, a_{21}, \cdots, a_{m1}, \cdots, a_{1n}, \cdots, a_{mn}).$$

Then (using Row Stacking Form:)

$$P(x, y) = V_r(A) \text{ n } x \text{ n } y. \tag{3}$$

n can search pointer mechanically!

☞ Multi-linear Mapping

$$P : \mathrm{R}^m \times \mathrm{R}^n \times \mathrm{R}^s \to \mathrm{R}.$$

Cubic Matrix?

$$\begin{matrix} d_{111} & d_{112} & \cdots & d_{11n} \\ d_{121} & d_{122} & \cdots & d_{12n} \\ \vdots & \vdots & & \vdots \\ d_{1m1} & d_{1m2} & \cdots & d_{1mn} \end{matrix}$$

$$\begin{matrix} d_{k11} & d_{k12} & \cdots & d_{k1n} \\ d_{k21} & d_{k22} & \cdots & d_{k2n} \\ \vdots & \vdots & & \vdots \\ d_{km1} & d_{km2} & \cdots & d_{kmn} \end{matrix}$$

kth layer

$$\begin{matrix} d_{p11} & d_{p12} & \cdots & d_{p1n} \\ d_{p21} & d_{p22} & \cdots & d_{p2n} \\ \vdots & \vdots & & \vdots \\ d_{pm1} & d_{pm2} & \cdots & d_{pmn} \end{matrix}$$

$$
\begin{aligned}
& P(\delta_m^i, \delta_n^j, \delta_s^k) := d_{i,j,k}, \\
& \quad i = 1, \cdots, m; j = 1, \cdots, n; k = 1, \cdots, s.
\end{aligned}
$$

Define

$$
M_P = [d_{111}, \cdots, d_{11s}, \cdots, d_{mn1}, \cdots, d_{mns}].
$$

Then

$$
P(x, y, z) = M_P \;\mathrm{n}\; x \;\mathrm{n}\; y \;\mathrm{n}\; z. \tag{4}
$$

It is available for general multi-linear mappings.

☞ A Syntheses

$$
\begin{array}{rcl}
STP & : & A_{m\times n} \text{ n } B_{p\times q} \\
\\
n = p & \to & AB = A \text{ n } B \text{ (Conventional)} \\
A_i := \mathrm{Col}_i(A) & \to & A \otimes B = [A_1 \text{ n } B, \cdots, A_n \text{ n } B] \text{ (Kronecker)} \\
n = q & \to & A * B = [A_1 \text{ n } B_1, \cdots, A_n \text{ n } B_n] \text{ (Khatri-Lao)}
\end{array}
$$

- a syntheses of multi-products;
- with multi-functions of several products.

☞ Properties

Proposition 1.4

- (Distributive rule)

$$\begin{aligned} &A \text{ n } (\alpha B + \beta C) = \alpha A \text{ n } B + \beta A \text{ n } C; \\ &(\alpha B + \beta C) \text{ n } A = \alpha B \text{ n } A + \beta C \text{ n } A, \quad \alpha, \beta \in \mathrm{R}. \end{aligned} \tag{5}$$

- (Associative rule)

$$A \text{ n } (B \text{ n } C) = (A \text{ n } B) \text{ n } C. \tag{6}$$

Proposition 1.5

- $$(A \text{ n } B)^T = B^T \text{ n } A^T. \tag{7}$$

- Assume both A and B are invertible. Then

$$(A \text{ n } B)^{-1} = B^{-1} \text{ n } A^{-1}. \tag{8}$$

Proposition 1.6 (Pseudo-Commutativity)

Assume $A \in \mathcal{M}_{m\times n}$ is given.

- Let $Z \in \mathrm{R}^t$ be a row vector. Then

$$A \;\mathrm{n}\; Z = Z \;\mathrm{n}\; (I_t \otimes A); \tag{9}$$

- Let $Z \in \mathrm{R}^t$ be a column vector. Then

$$Z \;\mathrm{n}\; A = (I_t \otimes A) \;\mathrm{n}\; Z. \tag{10}$$

☞ Multi-dimensional Cases

- Let $\xi \in \mathrm{R}^n$ be a column (row). Then

$$\xi^k := \underbrace{\xi \,\mathrm{n} \cdots \mathrm{n}\, \xi}_{k}.$$

- Let $A \in \mathcal{M}_{m\times n}$ and $m|n$ or $n|m$. Then

$$A^k := \underbrace{A \,\mathrm{n} \cdots \mathrm{n}\, A}_{k}.$$

- In Boolean algebra, all matrices $A \in \mathcal{M}_{m\times n}$, where $m = 2^p$ and $n = 2^q$ (or for k-valued case: $m = k^p$ and $n = k^q$), which is the multiple-dimensional case.

☞ Swap Matrix

Definition 1.7

A swap matrix, $W_{[m,n]}$ is an $mn \times mn$ matrix constructed in the following way: label its columns by $(11, 12, \cdots, 1n, \cdots, m1, m2, \cdots, mn)$ and its rows by $(11, 21, \cdots, m1, \cdots, 1n, 2n, \cdots, mn)$. Then its element in the position $((I, J), (i, j))$ is assigned as

$$w_{(IJ),(ij)} = \delta_{i,j}^{I,J} = \begin{cases} 1, & I = i \text{ and } J = j, \\ 0, & \text{otherwise.} \end{cases} \tag{11}$$

When $m = n$ we briefly denote $W_{[n]} := W_{[n,n]}$.

☞ Example

Example 1.8

Let $m = 2$ and $n = 3$, the swap matrix $W_{[2,3]}$ is constructed as

$$W_{[2,3]} = \begin{array}{c} \begin{array}{cccccc} (11) & (12) & (13) & (21) & (22) & (23) \end{array} \\ \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \end{array} \begin{array}{c} \\ (11) \\ (21) \\ (12) \\ (22) \\ (13) \\ (23) \end{array}.$$

☞ Properties

Proposition 1.9

- Let $X \in \mathbb{R}^m$ and $Y \in \mathbb{R}^n$ be two columns. Then

$$W_{[m,n]} \ltimes X \ltimes Y = Y \ltimes X, \quad W_{[n,m]} \ltimes Y \ltimes X = X \ltimes Y. \tag{12}$$

- Let $A \in \mathcal{M}_{m\times n}$. Then

$$W_{[m,n]} V_r(A) = V_c(A), \quad W_{[n,m]} V_c(A) = V_r(A). \tag{13}$$

- Let $X_i \in \mathbb{R}^{n_i}$, $i = 1, \cdots, m$. Then

$$\begin{aligned} &\left(I_{n_1+\cdots+n_{k-1}} \otimes W_{[n_k,n_{k+1}]} \otimes I_{n_{k+2}+\cdots+n_m}\right) \\ &X_1 \ltimes \cdots \ltimes X_k \ltimes X_{k+1} \ltimes \cdots \ltimes X_m \\ &= X_1 \ltimes \cdots \ltimes X_{k+1} \ltimes X_k \ltimes \cdots \ltimes X_m. \end{aligned} \tag{14}$$

☞ Properties

Proposition 1.10

- The swap matrix is an orthogonal matrix as

$$W_{[m,n]}^{T} = W_{[m,n]}^{-1} = W_{[n,m]}. \tag{15}$$

-

$$W_{[m,n]} = \left[\delta_n^1 \text{ n } \delta_m^1 \quad \cdots \quad \delta_n^n \text{ n } \delta_m^1 \quad \cdots\cdots \quad \delta_n^n \text{ n } \delta_m^m\right], \tag{16}$$

where δ_n^i is the ith column of I_n.

☞ “$\times$” vs “n”

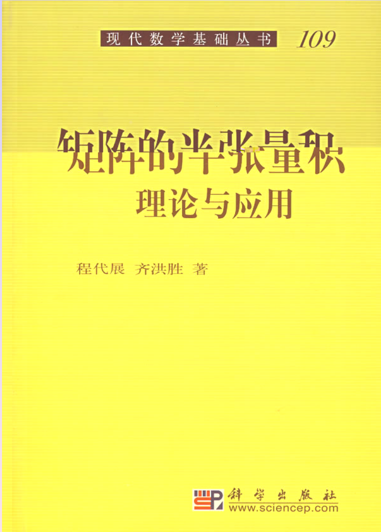
	CP $\times$	STP n
Domain	Equal Dimension	Arbitrary
Property	Similar	Similar
Applicability	linear, bilinear	multilinear
Commutativity	No	Pseudo-Commutative

Remark: Compare scalar product with matrix product:

- $a \times b$ is always defined $\Leftrightarrow$ $A \times B$ may not defined;
- $a \times b = b \times a$ $\Leftrightarrow$ in general $AB \neq BA$.

n overcomes these two obstacles!

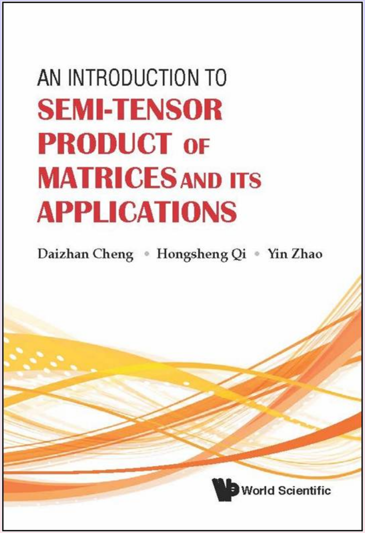
☞ My Book


现代数学基础丛书 109
矩阵的半张量积
理论与应用
程代展 齐洪胜 著
科学出版社
www.sciencep.com


系统与控制丛书
程代展 齐洪胜 著
矩阵的半张量积
——理论与应用
(第二版)
科学出版社

☞ My Book

Communications and Control Engineering
Daizhan Cheng
Hongsheng Qi
Zhiqiang Li
Analysis and Control of Boolean Networks
Semi-tensor Product Approach
Springer



AN INTRODUCTION TO
SEMI-TENSOR PRODUCT OF MATRICES AND ITS APPLICATIONS
Daizhan Cheng • Hongsheng Qi • Yin Zhao
World Scientific

II. Matrix Expression of Logic

☞ Logic

- $\mathcal{D} = \{0 \sim \text{False}, 1 \sim \text{True}\}$.

Logical Variables

$$x, y \cdots \in \mathcal{D}$$

Truth Table of Logical Functions

Table 1: Negation ($\neg x$)

x	$\neg x$
1	0
0	1

☞ Logic (continued)

Truth Table of Logical Functions (continued)

Table 2: Disjunction: $(x \vee y)$; Conjunction: $(x \wedge y)$; Conditional: $(x \rightarrow y)$; Biconditional: $(x \leftrightarrow y)$; Exclusive Or: $(x \bar{\nabla} y)$.

x	y	$x \vee y$	$x \wedge y$	$x \rightarrow y$	$x \leftrightarrow y$	$x \bar{\nabla} y$
1	1	1	1	1	1	0
1	0	1	0	0	0	1
0	1	1	0	1	0	1
0	0	0	0	1	1	0

☞ Vector Form of Logic

- δ_n^i: the ith column of I_n;
- $\Delta_n := \{\delta_n^i | i = 1, \cdots, n\}$, $\Delta := D_2$;

$$\text{True} \sim 1 \sim \delta_2^1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix};$$

$$\text{False} \sim 0 \sim \delta_2^2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}.$$

- A matrix $L \in \mathcal{M}_{n\times r}$ is called a logical matrix if

$$\mathrm{Col}(L) \subset \Delta_n.$$

Denote by $\mathcal{L}_{n\times r}$ the set of $n \times r$ logical matrices.

- Let $L = [\delta_n^{i_1}, \delta_n^{i_2}, \cdots, \delta_n^{i_r}] \in \mathcal{L}_{n\times r}$. Briefly,

$$L = \delta_n[i_1, i_2, \cdots, i_r].$$

Example 2.1

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix} = \delta_3[1,3,2,3].$$

☞ Vector Form of Logical Mapping

$$1 \sim \delta_2^1; \text{ and } 0 \sim \delta_2^2 \Rightarrow \mathcal{D} \sim \Delta.$$

Hence,

- Logical function:

$$f : \mathcal{D}^n \to \mathcal{D} \quad \Rightarrow \quad \Delta^n \to \Delta;$$

- Logical mapping:

$$F : \mathcal{D}^n \to \mathcal{D}^m \quad \Rightarrow \quad \Delta^n \to \Delta^m.$$

The later function (mapping) is called the vector form.

☞ Structure Matrix (1)

Theorem 2.2

Let $y = f(x_1, \cdots, x_n) : \Delta^n \to \Delta$. Then there exists unique $M_f \in \mathcal{L}_{2\times 2^n}$ such that

$$y = M_f x, \quad \text{where } x = \text{n}_{i=1}^{n} x_i. \tag{17}$$

Definition 2.3

The M_f is called the **structure matrix** of f.

☞ Structure Matrix (2)

Theorem 2.4

Let $F : \Delta^n \to \Delta^k$ be defined by

$$y_i = f_i(x_1, \cdots, x_n).$$

Then there exists unique $M_F \in \mathcal{L}_{2^k \times 2^n}$ such that

$$y = M_F x, \tag{18}$$

where

$$x = \ltimes_{i=1}^n x_i; \qquad y = \ltimes_{i=1}^k y_i.$$

Definition 2.5

The M_F is called the **structure matrix** of F.

☞ Structure Matrices of Logical Operators

Table 3: Structure Matrices of Logical Operators

$\neg$	M_n	$\delta_2[2\ 1]$
$\vee$	M_d	$\delta_2[1\ 1\ 1\ 2]$
$\wedge$	M_c	$\delta_2[1\ 2\ 2\ 2]$
$\rightarrow$	M_i	$\delta_2[1\ 2\ 1\ 1]$
$\leftrightarrow$	M_e	$\delta_2[1\ 2\ 2\ 1]$
$\bar{\vee}$	M_p	$\delta_2[2\ 1\ 1\ 2]$

An Example

Example 2.6

There are three persons.

- A said: “B is a liar!”
- B said: “C is a liar!”
- C said: “A and B both are liars!”

Who is the liar?

Set P: A is honest; Q: B is honest; R: C is honest.
The logical expression is

$$(P \leftrightarrow \neg Q) \wedge (Q \leftrightarrow \neg R) \wedge (R \leftrightarrow \neg P \wedge \neg Q) = 1.$$

Its matrix form is

$$L(P, Q, R) = M_c M_c (M_e P M_n Q)(M_e Q M_n R)(M_e R M_c M_n P M_n Q)$$

We can calculate the canonical form of $L(P, Q, R)$ as

$$L(P, Q, R) = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 \end{bmatrix} PQR = \delta_2^1.$$

Only if $P = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$, $Q = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$, and $R = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$, then L is true, which means that only B is honest.

☞ Multi-valued Logic

- $\mathcal{D}_k = \{1, \frac{k-2}{k-1}, \cdots, \frac{1}{k-1}, 0\}$;
- $\Delta_k = \{\delta_k^1, \delta_k^2, \cdots, \delta_k^{k-1}, \delta_k^k\}$.

k-valued logical variables:

$$x, y \in \mathcal{D}_k$$

Using equivalence:

$$\delta_k^1 \sim 1, \quad \delta_k^2 \sim \frac{k-2}{k-1}, \quad , \cdots, \delta_k^k \sim 0,$$

we have

$$x, y \in \Delta_k.$$

Theorem 2.7

Let $y = f(x_1, \cdots, x_n) : \Delta_k^n \to \Delta_k$. Then there exists unique $M_f \in \mathcal{L}_{k \times k^n}$ such that

$$y = M_f x, \quad \text{where } x = \mathrm{n}_{i=1}^n x_i. \tag{19}$$

Example 2.8

A detective is investigating a murder case. He has the following clues:

1. 80% that A or B is the murderer;
2. If A is the murderer, the killing time is before midnight;
3. If B's confession is true, the light in the room of murder was on at the midnight;
4. If B's confession is a lie, it is very possible that the murder happened before midnight;
5. There is an evidence that the light in the room of murder at the midnight was off.

Example 2.8 (Continued)

Set $D_6 = \{T, \text{very likely}, 80\%, 1\text{-}80\%, \text{very unlikely}, F\}$.

- A: A is murderer;
- B: B is murderer;
- M: murder happened before midnight;
- S: B's confession is true;
- L: the light was on at midnight.

$$A \vee B = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}^T \tag{20}$$

$$A \to \neg M = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}^T \tag{21}$$

$$S \to L = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \tag{22}$$

$$\neg S \to M = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}^T \tag{23}$$

$$\neg L = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \tag{24}$$

Example 2.8 (Continued)

From (24) $\Rightarrow L = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}^T$
Then from (22), we have

$$M_i^6 SL = (M_i^6 W_{[6]} L)S = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T$$
$$\Rightarrow S = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}^T$$

Similarly, (23) $\Rightarrow M = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}^T$
Then from (21) $\Rightarrow A = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}^T$
Finally, from (20) $\Rightarrow B = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}^T$

We conclude that: A is **very unlikely** the murderer; B is **80%** the murderer.

III. Boolean Network

Kaffman: for cellular networks, gene regulatory networks, etc.

☞ Network Graph

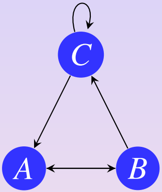


Figure 2: A Boolean network

☞ Network Dynamics

$$\begin{cases} A(t+1) = B(t) \wedge C(t) \\ B(t+1) = \neg A(t) \\ C(t+1) = B(t) \vee C(t) \end{cases} \tag{25}$$

Boolean Control Network

☞ Network Graph

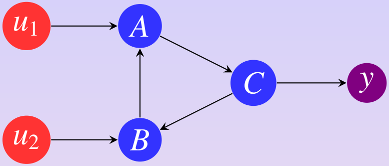


Figure 3: A Boolean control network

☞ Network Dynamics

Its logical equation is

$$\begin{cases} A(t+1) = B(t) \wedge u_1(t) \\ B(t+1) = C(t) \vee u_2(t) \\ C(t+1) = A(t) \end{cases} \qquad (26)$$
$$y(t) = \neg C(t)$$

☞ Dynamics of Boolean Network

$$
\begin{cases}
x_1(t+1) = f_1(x_1(t), \cdots, x_n(t)) \\
\vdots \\
x_n(t+1) = f_n(x_1(t), \cdots, x_n(t)), \quad x_i \in \mathcal{D},
\end{cases}
\tag{27}
$$

where

$$
\mathcal{D} := \{0, 1\}.
$$

☞ Dynamics of Boolean Control Network

$$\begin{cases} x_1(t+1) = f_1(x_1(t), \cdots, x_n(t), u_1(t), \cdots, u_m(t)) \\ \vdots \\ x_n(t+1) = f_n(x_1(t), \cdots, x_n(t), u_1(t), \cdots, u_m(t)), \end{cases}$$
$$y_j(t) = h_j(x(t)), \quad j = 1, \cdots, p, \tag{28}$$

where $x_i, u_i, y_i \in \mathcal{D}$.

☞ Matrix Expression of Subspace

- State Space: $\mathcal{X} = F_\ell(x_1, \cdots, x_n)$
- Subspace: $\mathcal{V} = F_\ell(y_1, \cdots, y_k)$, $y_i \in \mathcal{X}$ is described by

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

- Algebraic Form:

$$y = F_v x,$$

  where

$$x = \mathrm{n}_{i=1}^{n} x_i, \ y = \mathrm{n}_{i=1}^{k} y_i, \ F_v \in \mathcal{L}_{2^k \times 2^n}.$$

- Conclusion: Each $F_v \in \mathcal{L}_{2^k \times 2^n}$ uniquely determines a subspace $\mathcal{V}$.

☞ Algebraic Form of BN (27)

$$x(t+1) = Lx(t), \tag{29}$$

where $L \in \mathcal{L}_{2^n \times 2^n}$.

☞ Algebraic Form of BCN (28)

$$\begin{cases} x(t+1) = Lu(t)x(t) \\ y(t) = Hx(t), \end{cases} \tag{30}$$

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

Examples

Example 3.5

- Consider Boolean network (25) in Fig. 2. We have

$$L = \delta_8[3\ 7\ 7\ 8\ 1\ 5\ 5\ 6].$$

- Consider Boolean control network (26) in Fig. 3. We have

$$\begin{aligned} L &= \delta_8[1\ 1\ 5\ 5\ 2\ 2\ 6\ 6\ 1\ 3\ 5\ 7\ 2\ 4\ 6\ 8 \\ &\quad\ 5\ 5\ 5\ 5\ 6\ 6\ 6\ 6\ 5\ 7\ 5\ 7\ 6\ 8\ 6\ 8]; \\ H &= \delta_2[2\ 1\ 2\ 1\ 2\ 1\ 2\ 1]. \end{aligned}$$

☞ Topological Structure

- Find “fixed points”, “cycles”;
- Find “basin of attraction” ,“transient time”;
- “Rolling Gear” structure, which explains why “tiny attractors” decide “vast order”.

References:

D. Cheng, H. Qi, A linear representation of dynamics of Boolean networks, *IEEE Trans. Aut. Contr.*, vol. 55, no. 10, pp. 2251-2258, 2010. (**Regular Paper**)

D. Cheng, Input-state approach to Boolean networks, *IEEE Trans. Neural Networks*, vol. 20, no. 3, pp. 512-521, 2009. (**Regular Paper**)

Basic Control Properties

- Controllability under open-loop or closed-loop controls;
- Observability;
- Algebraic description of input-output transfer graph.

References:

D. Cheng, H. Qi, Controllability and observability of Boolean control networks, *Automatica*, vol. 45, no. 7, pp. 1659-1665, 2009. (**Regular Paper**)

Y. Zhao, H. Qi, D. Cheng, Input-state incidence matrix of Boolean control networks and its applications, *Sys. Contr. Lett.*, vol. 46, no. 12, pp. 767-774, 2010.

☞ System Realization

- State space expression;
- Input-output realization;
- Kalman decomposition, minimum realization.

References:

- D. Cheng, Z. Li, H. Qi, Realization of Boolean control networks, *Automatica*, vol. 46, no. 1, pp. 62-69, 2010. (**Regular Paper**)

- D. Cheng, H. Qi, State space analysis of Boolean network, *IEEE Trans. Neural Networks*, vol. 21, no. 4, pp. 584-594, 2010. (**Regular Paper**)

☞ Control Design

- Disturbance decoupling;
- Stability and stabilization;
- Canalizing mapping and its applications.

References:

D. Cheng, Disturbance Decoupling of Boolean control networks, *IEEE Trans. Aut. Contr.*, vol. 56, no. 1, pp. 2-10, 2011. (**Regular Paper**)

D. Cheng, H. Qi, Z. Li, J.B. Liu, Stability and stabilization of Boolean networks, *Int. J. Robust Nonlin. Contr.*, vol. 21, no. 2, pp. 134-156, 2001.

☞ Optimal Control

- Topological structure of Boolean control networks;
- Optimal control and its design.
- k- and Mix-valued and higher-order control networks.

References:

- Y. Zhao, Z. Li, D. Cheng, Optimal control of logical control networks *IEEE Trans. Aut. Contr.*, vol. 56, no. 8, pp. 1766-1776, (**Regular Paper**).
- Z. Li, D. Cheng, Algebraic approach to dynamics of multi-valued networks, *Int. J. Bifurcat. Chaos*, vol. 20, no. 3, pp. 561-582, 2010.

☞ Identification

- Identify the dynamic evolution;
- Identify via input-output data.

References:

D. Cheng, Y. Zhao, Identification of Boolean control networks, *Automatica*, vol. 47, no. 4, pp. 702-710, 2011.(**Regular Paper**)

D. Cheng, H. Qi, Z. Li, Model construction of Boolean network via observed data, *IEEE Trans. Neural Networks*, vol. 22, no. 4, pp. 525-536, 2011. (**Regular Paper**)

☞ My Book

Communications and Control Engineering
Daizhan Cheng
Hongsheng Qi
Zhiqiang Li
Analysis and
Control of Boolean
Networks
Semi-tensor Product Approach
Springer

IV. Dynamic Game

☞ Static Game

Definition 4.1

(1) A static game G consists of three ingredients: (i) n players, named $A_1, \cdots, A_n$; (ii) each player A_i has k_i possible actions, denoted by $x_i \in \mathcal{D}_{k_i}$, $i = 1, \cdot n$; (iii) n payoff functions for n players respectively as

$$c_j(x_1 = i_1, \cdots, x_n = i_n) = c^j_{i_1\ i_2\ \cdots\ i_n}, \quad j = 1, \cdots, n. \tag{31}$$

(2) A set of actions $s = (x_1, \cdots, x_n)$, is a strategy of G, denoted by S.

(3) A strategy $\{x_j^*\}$ is a Nash equilibrium if

$$c_j(x_1^*, \cdots, , x_j^*, \cdots, x_n^*) \geq c_j(x_1^*, \cdots, x_j, \cdots, x_n^*) \\ j = 1, \cdots, n. \tag{32}$$

Example 4.2

Prisoners' Dilemma

- Action 1: Confess
- Action 2: Deny

Table 4: Payoff bi-matrix

$P_1 \backslash P_2$	1	2
1	-3,-3	0,-5
2	-5,0	-1,-1

Nash Equilibrium is $(1, 1)$.

☞ Dynamic Game

$$G \Rightarrow G_\infty$$

Payoff Functions

$$J_j = \overline{\lim_{T\to\infty}} \frac{1}{T} \sum_{t=1}^{T} c_j(x(t)), \quad j = 1, \cdots, n.$$

☞ Strategy with Finite Memory

Definition 4.3

A strategy for G

☞ Human-Machine Game

$$m(t+1) = f(m(t), h(t)), \tag{34}$$

$$J_h = \overline{\lim_{T\to\infty}} \frac{1}{T} \sum_{t=1}^{T} c_h(x(t)).$$

Theorem 4.4

(1) The best strategy is state-control periodic.

(2) The best strategy ($h^*(t)$) satisfies

$$h^*(t+1) = g(m(t), h(t)) = Lm(t)h(t). \tag{35}$$

☞ Human-Machine Game (continued)

Find best strategy:

(1) find cycles on state-control space;

(2) find optimal L, where

$$L \in \mathcal{L}_{q \times pq},$$

where p: Number of machine strategies; q: Number of human strategies;

References:

Y. Zhao, Z. Li, D. Cheng, Optimal control of logical control networks, *IEEE Trans. Aut. Contr.*, vol. 56, no. 8, pp. 1766-1776, 2011 (**Regular Paper**).

☞ Mixed Strategy

Consider player i:

$$S_i = \{1, 2, \cdots, k_i\}$$

$$x_i = j, \quad \text{with Probability } p_i(j),$$

where $\sum_{j=1}^{k_i} p_i(j) = 1$.

- Finite Horizon case:

$$J_h = E\left[\sum_{t=1}^{N} \lambda^t c_h(h(t), m(t)) \Big| m(0)\right].$$

Here $0 < \lambda < 1$ (discount factor).

Theorem 4.5

Let $J^*(m(0))$ be the optimal value of J_h. Then

$$J^*(x(0)) = J_0(x(0)), \tag{36}$$

where the function J_0 is given by the last step of a dynamic programming algorithm. Setting $c_t := \lambda^t c(h(t), m(t))$, the algorithm proceeds backward in time from time step N to time step 0 as follows.

$$J_N(m(N)) = \max_{h(N)\in\Delta_r} c_t(h(N), m(N)). \tag{37}$$

and for $t = N-1, N-2, \cdots, 1, 0$:

$$J_t(m(t)) = \max_{h(t)\in\Delta_r} E\left[c_t(h(t), m(t)) + J_{t+1}(m(t+1)) \middle| m(t), h(t)\right]. \tag{38}$$

- Infinite Horizon case:

$$J_h = E\left[\sum_{t=1}^{\infty} \lambda^t c_h(h(t), m(t)) \middle| m(0)\right].$$

Receding Horizon Based Feedback Control:

Denote

$$\min_{h \in \Delta_k} \min_{h_i \neq h_j \in \Delta_r} \left| c(m, h_i) - c(m, h_j) \right| := d.$$

$$M := \max_{h \in \Delta_r, m \in \Delta_k} |c(h, m)| < \infty.$$

Theorem 4.6

Assume $d > 0$. Then the optimal control sequence $u^*(0), u^*(1), \cdots$ obtained by receding horizon control is exactly the optimal control for the infinite horizon case, provided that the prediction horizon length ℓ satisfies

$$\ell > \log_\lambda \frac{(1-\lambda)d}{2M}. \tag{39}$$

D. Cheng, Y. Zhao, T. Xu. Receding horizon based feedback optimization for mix-valued logical networks, *IEEE Trans. Aut. Contr.*, In press, On line: `http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=7079492`, DOI: 10.1109/TAC.2015.2419874.

☞ Networked Evolutionary Games

Definition 4.7

A networked evolutionary game (NEG), denoted by $\mathcal{G} = ((N,E), G, \Pi)$, consists of three factors:

(i) a network graph: (N,E);

(ii) a fundamental network game (FNG): G with two players. Players i and j play this game provided $(i,j) \in E$.

(iii) a local information based strategy updating rule (SUR):

$$x_i(t+1) = f_i(x_j(t), c_j(t) \mid j \in U(i)), \quad i = 1, \cdots, n. \tag{40}$$

D. Cheng, F. He, H. Qi, T. Xu. Modeling, analysis and control of networked evolutionary games, *IEEE Trans. Aut. Contr.*, In press, On line: `http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=7042754`, DOI: 10.1109/TAC.2015.2404471. (**Regular Paper**)

V. Concluding Remarks

The algebraic state space representation of logical dynamic systems has various applications:

- (networked) evolutionary games;
- logical circuit design and related topics:
- cryptography:
- fuzzy control:
- graph theory and formation control:
- communication;
- control of power systems and engine transient control;

Thank you!

Question?