SCIENCE CHINA
Information Sciences

• LETTER •

June 2020, Vol. 63 169202:1–169202:3
https://doi.org/10.1007/s11432-018-9536-3

Data set approach for solving logical equations

Sen WANG[1], Jun-E FENG[1*], Yongyuan YU[1] & Xinhong WANG[2]

[1]*School of Mathematics, Shandong University, Jinan 250100, China;*
[2]*School of Information Science and Engineering, University of Jinan, Jinan 250022, China*

Received 24 May 2018/Accepted 10 July 2018/Published online 12 September 2019

Citation Wang S, Feng J-E, Yu Y Y, et al. Data set approach for solving logical equations. Sci China Inf Sci, 2020, 63(6): 169202, https://doi.org/10.1007/s11432-018-9536-3

Dear editor,

Since Kauffman built a Boolean network model to simulate the interaction of genes in 1969 [1], Boolean networks have been used to describe relationships of cells, proteins, DNA and RNA in biological systems [2–5]. In virtue of semi-tensor product (STP), an algebraic framework was developed to analyze Boolean networks [6, 7].

Logical equations arise when some variables are unknown. Ref. [8] presents three kinds of solutions to logical equations, which can be applied in many aspects. First, based on them, a dynamic-algebraic Boolean network, can be transformed into a standard Boolean network. Second, implicit functions on logical operations can be studied thereby. Third, they are potentially useful in diagnosis of disease, symbolic logic, and electrical circuit design [9].

To provide an alternative method to derive all these three kinds of solutions to a logical equation, this study constructs a set and a corresponding matrix which are called data set and data matrix, respectively. The method of [8] involves constructing a truth matrix for every possible partition, based on which some logical matrices can be obtained. Different from [8], for the data set approach provided in this study, only one data matrix needs to be constructed.

Let δ_n^i denote the i-th column of identity matrix I_n. $\Delta_n := \{\delta_n^1, \delta_n^2, \ldots, \delta_n^n\}$ and $\mathcal{D}^n := \{(x_1, x_2, \ldots, x_n) | x_i \in \mathcal{D}, i = 1, 2, \ldots, n\}$. Simply use $\Delta = \Delta_2$ and $\mathcal{D} = \{0, 1\}$. The i-th row (column) and the set of rows (columns) of matrix M are denoted by $\mathrm{row}_i(M)$ ($\mathrm{col}_i(M)$) and $\mathrm{Row}(M)$ ($\mathrm{Col}(M)$), respectively. If $\mathrm{Col}(M) \subset \Delta_n$, then M is called a logical matrix. Set $\mathcal{L}_{n\times s}$ consists of all $n \times s$ logical matrices. Let $\delta_n[i_1, i_2, \ldots, i_s] := [\delta_n^{i_1}, \delta_n^{i_2}, \ldots, \delta_n^{i_s}]$. Symbols $*$, $\otimes$ and $\ltimes$ respectively represent Khatri-Rao product, Kronecker product and STP of matrices A and B [7]. Additionally, STP is a generalization of conventional matrix product. Therefore, symbol "$\ltimes$" will be omitted throughout this study when no confusion is generated.

Consider the logical equations

$$\begin{cases} f_1(x_1, x_2, \ldots, x_n) = 1, \\ f_2(x_1, x_2, \ldots, x_n) = 1, \\ \quad\vdots \\ f_n(x_1, x_2, \ldots, x_n) = 1, \end{cases} \tag{1}$$

where $x_i \in \mathcal{D}$, $i = 1, 2, \ldots, n$ are unknown variables. Define the equivalent form of each $x_i \in \mathcal{D}$ as $\overline{x}_i = \delta_2^{2-x_i}$ and call x_i the scalar form and $\overline{x}_i$ the vector form. Based on STP, set of (1) can be transformed into the following expression:

$$\begin{cases} M_1\overline{x} = \delta_2^1, \\ M_2\overline{x} = \delta_2^1, \\ \quad\vdots \\ M_n\overline{x} = \delta_2^1, \end{cases} \tag{2}$$

where $M_i \in \mathcal{L}_{2\times 2^n}$, $i = 1, 2, \ldots, n$, $\overline{x} = \ltimes_{j=1}^n \overline{x}_j$, and $\overline{x}_j \in \Delta$, $j = 1, 2, \ldots, n$. And Eq. (2) can be rewritten as

$$L\overline{x} = \delta_{2^n}^1, \tag{3}$$

* Corresponding author (email: fengjune@sdu.edu.cn)

where $L = M_1 * M_2 * \cdots * M_n \in \mathcal{L}_{2^n \times 2^n}$. The solution sets of (3) and (1) are $\overline{\mathcal{X}}_a = \{\delta_{2^n}^i | \mathrm{col}_i(L) = \delta_{2^n}^1\}$ and

$$\mathcal{X}_a = \{(x_1, x_2, \ldots, x_n) | \ltimes_{i=1}^n \overline{x}_i \in \overline{\mathcal{X}}_a,\ \overline{x}_i = \delta_2^{2-x_i},\ i = 1, 2, \ldots, n\}.$$

Ref. [7] provides a procedure to derive every $\overline{x}_i$ from $\overline{x} = \ltimes_{i=1}^n \overline{x}_i$.

Three types of solutions to a logical equation were defined in [8] and restated in the following.

Definition 1. (1) A logical function is called an antecedence partition-based solution (APBS) of a logical equation if we can derive the equation from the function.

(2) A logical function is called a consequence partition-based solution (CPBS) of a logical equation if we can derive the function from the equation.

(3) A logical function is called an antecedence-consequence partition-based solution (ACPBS) of a logical equation if it is both APBS and CPBS of the logical equation.

An alternative approach to find these solutions is provided in this study based on the following two kinds of sets called data set and data space.

Definition 2. (1) A subset of $\mathcal{D}^n$ is called a data set. A matrix M is called a data matrix of $\mathcal{X}$ if M takes all the elements of a data set $\mathcal{X}$ as its rows in any order. That is, $\mathrm{Row}(M) = \mathcal{X}$.

(2) A data set $\mathcal{X}$ is called a data space if there exist a partition $\{\Gamma, \Lambda\}$ of $\{1, 2, \ldots, n\}$ with $\Gamma = \{p_1, p_2, \ldots, p_t\}, \Lambda = \{q_1, q_2, \ldots, q_s\}$, and a function $f : \mathcal{D}^t \to \mathcal{D}^s$ such that $\mathcal{X}$ can be expressed as

$$\begin{aligned} \mathcal{X} = & \{(x_1, x_2, \ldots, x_n) | (x_{q_1}, x_{q_2}, \ldots, x_{q_s}) \\ & = f(x_{p_1}, x_{p_2}, \ldots, x_{p_t}),\ x_{p_i} \in \mathcal{D}, \\ & i = 1, 2, \ldots, t\}. \end{aligned} \quad (4)$$

Then $\{x_{p_i}, i = 1, 2, \ldots, t\}$ is called a basis of $\mathcal{X}$ and the corresponding data matrix is called data space matrix.

Some aspects need to be noticed.

Remark 1. (1) Given a data set $\mathcal{X}$, the corresponding data matrix may be not unique. But it does not affect the subsequent results.

(2) For a data space $\mathcal{X}$, the basis may be not unique.

When $x = (x_1, x_2, \ldots, x_n)$ takes values from data set $\mathcal{X}$, we can also say that x takes values from the corresponding data matrix M.

Consider part of variables $\{x_{p_1}, x_{p_2}, \ldots, x_{p_t}\} \subset \{x_1, x_2, \ldots, x_n\}$. Construct a new matrix M' as

$$M' = [\mathrm{col}_{p_1}(M), \mathrm{col}_{p_2}(M), \ldots, \mathrm{col}_{p_t}(M)].$$

Note that there may be same rows in M'. In this study, when we say that $(x_{p_1}, x_{p_2}, \ldots, x_{p_t})$ takes values from M, it also means that $(x_{p_1}, x_{p_2}, \ldots, x_{p_t})$ takes values from $\mathrm{Row}(M')$. Moreover, we say that each value is taken only once if all the rows in M' are different. According to Definition 2, for a data set $\mathcal{X}$ and the corresponding data matrix M, if $\mathcal{X}$ is a data space, the vector of the basis $(x_{p_1}, x_{p_2}, \ldots, x_{p_t})$ must take all values in $\mathcal{D}^t$ and each value is taken only once. Conversely, if a vector of variables $(x_{p_1}, x_{p_2}, \ldots, x_{p_t})$ $(t < n)$ takes all values in $\mathcal{D}^t$ and each value is taken only once, then $\mathcal{X}$ is a data space. The corresponding function $(x_{q_1}, x_{q_2}, \ldots, x_{q_s}) = f(x_{p_1}, x_{p_2}, \ldots, x_{p_t})$ is directly obtained in the data space matrix M. Naturally, the following result is obvious.

Corollary 1. If $\mathcal{X}$ is a data space and it has a basis $\{x_{p_1}, x_{p_2}, \ldots, x_{p_t}\}$, then the corresponding data matrix M has 2^t rows.

It is clear that if $\mathcal{X}$ is a data space, then the number of variables in any basis of $\mathcal{X}$ is unique. Thus we can define the dimension of a data space.

Definition 3. Given a data space $\mathcal{X}$, define the dimension of $\mathcal{X}$ as the number of variables in any basis of $\mathcal{X}$.

For any vector $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_t) \in \mathcal{D}^t$, define a function

$$d(\alpha) = \alpha [2^{t-1} \ \ 2^{t-2} \ \ \cdots \ \ 2^0]^{\mathrm{T}} + 1. \quad (5)$$

It is trivial to see that the range of $d(\cdot)$ is $\{1, 2, \ldots, 2^t\}$. For any $\beta \in \{1, 2, \ldots, 2^t\}$, we can identify α such that $d(\alpha) = \beta$ by the following method:

$$\beta_0 = \beta - 1,\ \alpha_i = \left[\frac{\beta_{i-1}}{2^{t-i}}\right],\ \beta_i = \beta_{i-1} - \alpha_i \times 2^{t-i},\ i = 1, 2, \ldots, t,$$

where in the second equation $[a]$ is the largest integer which is less than or equal to a [7]. So the function $d(\cdot)$ is one-to-one. Combining the discussion above, we can draw the following statement.

Theorem 1. Suppose there is a $2^t \times n$ $(t < n)$ data matrix M and a corresponding data set $\mathcal{X}$. Then $\mathcal{X}$ is a data space if and only if there exists $\{p_1, p_2, \ldots, p_t\} \subset \{1, 2, \ldots, n\}$ such that any two elements of the following vector are different:

$$M' [2^{t-1} \ \ 2^{t-2} \ \ \cdots \ \ 2^0]^{\mathrm{T}}, \quad (6)$$

where $M' = [\mathrm{col}_{p_1}(M), \mathrm{col}_{p_2}(M), \ldots, \mathrm{col}_{p_t}(M)]$. If so, $\{x_{p_1}, x_{p_2}, \ldots, x_{p_t}\}$ is a basis.

Theorem 2 demonstrates the feasibility of expanding the dimension of a data space.

Theorem 2. If a data space $\mathcal{X}$ is of dimension t $(t < n-1)$, then add 2^t rows to the corresponding data space matrix M to construct a new data matrix $\widetilde{M}$, such that the new corresponding data set $\widetilde{\mathcal{X}}$ is a data space with dimension $t+1$.

In [8], all possible partitions are listed and the corresponding truth matrices are established. For every truth matrix, the logical matrices are found whose entries have certain relationships with those of the truth matrix in corresponding positions. And from those matrices, all three types of solutions are derived.

Next, we will present a new method to derive these three kinds of solutions of a logical equation. To this end, we build a data matrix for the logical equation first. Then construct new data matrices which correspond to all three types of solutions.

For logical equation (3) and the solution set $\mathcal{X}_a$, it is obvious that $\mathcal{X}_a$ is a data set. If $\mathcal{X}_a$ is a data space, according to Definitions 1 and 2, the corresponding function

$$(x_{q_1}, x_{q_2}, \ldots, x_{q_s}) = f(x_{p_1}, x_{p_2}, \ldots, x_{p_t})$$

is an ACPBS of (3).

If the corresponding data matrix M_a is of dimensions $2^t \times n$ $(t < n)$, list all $\{p_1, p_2, \ldots, p_t\} \in \{1, 2, \ldots, n\}$ to test whether they satisfy the condition of Theorem 1. In this way, we can find all the ACPBSs of (3).

In the following, suppose that M_a is of dimensions $m \times n$. Let l be an integer satisfying $2^l < m < 2^{l+1}$. Choose t $(l+1 \leqslant t \leqslant n-1)$ variables $x_{p_1}, x_{p_2}, \ldots, x_{p_t}$ and verify whether $(x_{p_1}, x_{p_2}, \ldots, x_{p_t})$ takes different values for any two rows in M_a. That is, pick out corresponding t columns of M_a to construct a matrix M' and verify whether any two elements of the vector (6) are different. If so, add some rows to M_a such that $(x_{p_1}, x_{p_2}, \ldots, x_{p_t})$ takes all different values in $\mathcal{D}^t$ and the new matrix $\widetilde{M}_a$ is a data space matrix. Then derive function $(x_{q_1}, x_{q_2}, \ldots, x_{q_s}) = f(x_{p_1}, x_{p_2}, \ldots, x_{p_t})$ $(x_{p_i} \in \mathcal{D}, i = 1, 2, \ldots, t)$ according to Definition 2.

Proposition 1. Suppose that a data set $\mathcal{X}_a$ is the solution set of logical equation (3) and the corresponding data matrix is M_a. If we can add some rows to M_a to construct a data space matrix $\widetilde{M}_a$ and let the corresponding data space be $\widetilde{\mathcal{X}}_a$, then the function $(x_{q_1}, x_{q_2}, \ldots, x_{q_s}) = f(x_{p_1}, x_{p_2}, \ldots, x_{p_t})$, $x_{p_i} \in \mathcal{D}$, $i = 1, 2, \ldots, t$ of $\widetilde{\mathcal{X}}_a$ is a CPBS of logical equation (3).

As we can choose $t = l+1, l+2, \ldots, n-1$ and pick out any t variables in the process above, and in the added rows, $(x_{q_1}, x_{q_2}, \ldots, x_{q_s})$ can take whatever values we desire, it is easy to see that in this way we can obtain all CPBSs of logical equation (3).

Likewise for M_a, we can choose t $(l+1 \leqslant t \leqslant n-1)$ variables $x_{p_1}, x_{p_2}, \ldots, x_{p_t}$, pick out corresponding t columns of M_a to construct a matrix M', and calculate the vector (6). Verify whether $(x_{p_1}, x_{p_2}, \ldots, x_{p_t})$ takes all values in $\mathcal{D}^t$. If the answer is yes, then we see in which rows $(x_{p_1}, x_{p_2}, \ldots, x_{p_t})$ takes same value. Remove some of such rows to the extent that for any several same rows of M' there is only one remaining. Completing this process we obtain a new data matrix $\widetilde{M}_a$. Similar to Proposition 1, we have Proposition 2.

Proposition 2. Suppose that a data set $\mathcal{X}_a$ is derived from logical equation (3) and the corresponding data matrix is M_a. If we can remove some rows from M_a to construct a data space matrix $\widetilde{M}_a$ and let the corresponding data space be $\widetilde{\mathcal{X}}_a$, then function $(x_{q_1}, x_{q_2}, \ldots, x_{q_s}) = f(x_{p_1}, x_{p_2}, \ldots, x_{p_t})$, $x_{p_i} \in \mathcal{D}$, $i = 1, 2, \ldots, t$ of $\widetilde{\mathcal{X}}_a$ is an APBS of logical equation (3).

As we can take $t = 1, 2, \ldots, l$, pick out any t variables in the process above, and choose the removed rows, it is easy to see that in this way we can obtain all APBSs of logical equation (3).

Acknowledgements This work was supported by National Natural Science Foundation of China (Grant No. 61773371).

References

1 Kauffman S A. Metabolic stability and epigenesis in randomly constructed genetic nets. J Theory Biol, 1969, 22: 437–467

2 Akutsu T, Miyano S, Kuhara S. Inferring qualitative relations in genetic networks and metabolic pathways. Bioinformatics, 2000, 16: 727–734

3 Li F F. Feedback control design for the complete synchronisation of two coupled Boolean networks. Int J Syst Sci, 2016, 47: 2996–3003

4 Li H T, Zhao G D, Meng M, et al. A survey on applications of semi-tensor product method in engineering. Sci China Inf Sci, 2018, 61: 010202

5 Zhu Q X, Liu Y, Lu J Q, et al. Observability of Boolean control networks. Sci China Inf Sci, 2018, 61: 092201

6 Cheng D Z. Semi-tensor product of matrices and its application to Morgan's problem. Sci China Inf Sci, 2001, 44: 195–212

7 Cheng D Z, Qi H S. A linear representation of dynamics of Boolean networks. IEEE Trans Autom Control, 2010, 55: 2251–2258

8 Qiao Y P, Qi H S, Cheng D Z. Partition-based solutions of static logical networks with applications. IEEE Trans Neural Netw Learn Syst, 2018, 29: 1252–1262

9 Kim K. Boolean Matrix Theory and Applications. New York: Marcel Decker, 1982