

基于粒矩阵的多输入多输出真值表快速并行约简算法

陈泽华* 马贺

(太原理工大学信息工程学院 太原 030024)

摘要: 真值表是表征逻辑输入与输出之间因果关系的重要工具, 真值表约简在数字逻辑电路的分析与设计具有重要意义。该文将真值表看作逻辑信息系统, 将真值表约简转化为逻辑信息系统的约简规则获取。采用粒计算分层粒化的思想, 在不同粒度下, 利用粒矩阵的知识表示形式、粒矩阵中的启发式知识以及粒矩阵运算, 设计了多输入多输出真值表快速并行约简算法。以发光二极管七段数字显示器为例进行了算法说明, 通过数学证明和算法复杂性分析证明了算法的正确性和有效性。

关键词: 数字逻辑电路; 真值表; 粒度; 粒矩阵; 并行约简; 粒计算

中图分类号: TN79; TP181

文献标识码: A

文章编号: 1009-5896(2015)05-1260-06

DOI: 10.11999/JEIT141129

Granular Matrix Based Rapid Parallel Reduction Algorithm for MIMO Truth Table

Chen Ze-hua Ma He

(College of Information Engineering, Taiyuan University of Technology, Taiyuan 030024, China)

Abstract: Truth table is an important tool to represent the logic causal relationships between inputs and outputs. The reduction of the truth table is of great significance in analysis and design of digital logic circuit. In this paper, the MIMO truth table is considered as a Logical Information System (LIS), and the traditional truth table reduction issue is converted into the minimal rule discovery of LIS. Granular Computing (GrC) method is then introduced. Firstly, the logical information system is hierarchically granulated. Secondly, the Granular Matrix (GrM) is defined and operated to represent the knowledge in different granularity, together with heuristic information hidden in the matrix, the rapid parallel reduction algorithm for the MIMO truth table is proposed. Light-Emitting Diode (LED) digital display is applied to illustrate the computing process. The mathematical proof and the complexity analysis proves the efficiency and validity of the proposed algorithm.

Key words: Digital logic circuit; Truth table; Granularity; Granular Matrix (GrM); Parallel reduction; Granular Computing (GrC)

1 引言

真值表约简在数字逻辑电路分析与设计中具有重要作用。经典方法有卡诺图法、公式法、Q-M及其改进算法以及立方体算法等^[1-7]。当输入变量增大时, 前3种方法算法复杂度比较高, 不利于程序实现, 而且不能对多个输出进行并行处理。立方体算法^[2]相对简单, 可以处理多输出, 本质依然依靠逻辑代数运算规律。目前, 真值表约简算法被内嵌在国外的电子设计自动化软件中, 工程师们不再需要自行设计约简算法。因此, 国内外对真值表的约简问题研究成果仍然停留在十几年前。

粒计算^[8]求解复杂问题的思想和方法受到关注, 基于粒计算的信息系统知识处理方法得到快速发展^[9-15]。文献[16]将真值表看作逻辑信息系统, 从知识工程角度分析真值表, 将真值表看作等价关系的粒计算模型, 通过分层粒化、矩阵运算, 将传统的真值表化简过程转化为基于粒矩阵运算的逻辑信息系统的约简规则提取过程。

本文进一步优化了文献[16]中的粒矩阵运算, 通过定理1减少了矩阵的存储规模、提高了算法的搜索效率, 并将其扩展为多输入多输出(MIMO)真值表快速并行约简算法。本文以发光二极管七段数字显示器为例, 详述了算法步骤。最后通过数学推导, 证明了本文算法与传统卡诺图约简算法的等价性。通过理论分析, 探讨了算法的复杂性和有效性。

2014-09-01 收到, 2015-01-16 改回

国家自然科学基金(61402319)和山西省回国留学人员科研资助项目(2013-031)资助课题

*通信作者: 陈泽华 zehuachen@163.com

2 预备知识

定义 1 真值表、最小项表达式、最简逻辑函数表达式^[1-3]：真值表是表征逻辑事件输入和输出之间全部可能状态的表格，是逻辑因果关系的表格表示。当一个真值表同时有多个输出时，称为多输出真值表。真值表中全部 m 个输入变量的乘积项(每个变量只以原变量或反变量的形式出现一次)称为最小项。真值表中所有逻辑函数值为 1 的最小项之和称为最小项表达式。简化后的真值表中所有输出值为 1 的输入变量乘积项的和称为最简逻辑函数表达式。

定义 2 逻辑信息系统^[16](Logic Information System, LIS)：逻辑信息系统 $LIS = (U, R, V, f)$ 是一个四元组，其中 U 为论域，由真值表中所有最小项组成。 $R = A \cup Y$ 表示属性集， A 表示逻辑输入属性集， Y 表示逻辑输出属性集，其中 $A = \{A_{m-1}, A_{m-2}, \dots, A_0\}$, $Y = \{Y_1, Y_2, \dots, Y_n\}$, m 为输入变量个数， n 为输出变量个数。 V 是属性值集合， $V = \{0, 1\}$ 。 $f: U \times R \rightarrow V$ 是一个信息函数，它指定 U 中每一个对象的属性值。逻辑信息系统中的每一条记录可以看作是一条逻辑规则。

注：在下文中用黑体 \mathbf{X}_i 表示矩阵或向量，用 X_i 表示等价关系或等价类。

定义 3 划分与粒化^[15,16]：在 LIS 中， $U = \{u_1, u_2, \dots, u_k, \dots, u_l\}$, $1 \leq k \leq l = 2^m$ ，输入变量组合 $A' \subseteq A$ 在 U 上产生划分：

$$U / \text{IND}(A') = \{X_1 \cdots X_i \cdots X_s\}, 1 \leq i \leq s \leq l \quad (1)$$

其中， $X_i \cap X_s = \phi$, $\cup X_i = U$ 且有

$$\mathbf{X}_i = (a_{i1} \cdots a_{ik} \cdots a_{il}) \quad (2)$$

$$a_{ik} = \begin{cases} 1, & u_k \in X_i \\ 0, & u_k \notin X_i \end{cases}, 1 \leq k \leq l \quad (3)$$

在 LIS 中，假设多个输出为 $Y = \{Y_1 \cdots Y_j \cdots Y_n\}$, $1 \leq j \leq n$, Y_j 对 U 产生划分：

$$U / Y_j = \{Y_{j0}, Y_{j1}\} \quad (4)$$

$$\mathbf{Y}_{j1} = (b_{j1} \cdots b_{jk} \cdots b_{jl}) \quad (5)$$

$$b_{jk} = \begin{cases} 1, & u_k \in Y_{j1} \\ 0, & u_k \notin Y_{j1} \end{cases}, 1 \leq k \leq l \quad (6)$$

等价类 X_i, Y_{j1}, Y_{j0} 为信息粒，均用长度为 l 的二进制向量表示。 \mathbf{Y}_{j1} 表示第 j 个输出中属性值为 1 的元素构成的信息粒，称为“1”输出向量； \mathbf{Y}_{j0} 表示第 j 个输出中属性值为 0 的元素构成的信息粒，称为“0”输出向量，LIS 是二值逻辑系统，显然有：

$$\mathbf{Y}_{j0} = (b'_{j1} \cdots b'_{jk} \cdots b'_{jl}), \text{ 其中: } b'_{jk} = 1 - b_{jk}.$$

定义 4 二进制粒矩阵：在逻辑信息系统中，

定义二进制粒矩阵 $\mathbf{BGrM} = \{\mathbf{X}_{s \times l}, \mathbf{Y}_{n \times l}, \mathbf{C}_{s \times n}\}$ ，其中

$$\mathbf{X}_{s \times l} = \mathbf{X} = \begin{pmatrix} \mathbf{X}_1 \\ \vdots \\ \mathbf{X}_i \\ \vdots \\ \mathbf{X}_s \end{pmatrix} = \begin{bmatrix} a_{11} & \cdots & a_{1k} & \cdots & a_{1l} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ a_{i1} & \cdots & a_{ik} & \cdots & a_{il} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ a_{s1} & \cdots & a_{sk} & \cdots & a_{sl} \end{bmatrix} \quad (7)$$

$$\mathbf{Y}_{n \times l} = \mathbf{Y}_0 = \begin{pmatrix} \mathbf{Y}_{10} \\ \vdots \\ \mathbf{Y}_{j0} \\ \vdots \\ \mathbf{Y}_{n0} \end{pmatrix} = \begin{bmatrix} b'_{11} & \cdots & b'_{1k} & \cdots & b'_{1l} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ b'_{j1} & \cdots & b'_{jk} & \cdots & b'_{jl} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ b'_{n1} & \cdots & b'_{nk} & \cdots & b'_{nl} \end{bmatrix} \quad (8)$$

$$\mathbf{C}_{s \times n} = \mathbf{X} \mathbf{Y}_0^T = (c_{ij})_{s \times n} \quad (9)$$

$$c_{ij} = \sum_{k=1}^l (a_{ik} b'_{kj}) = |X_i \cap Y_{j0}| \quad (10)$$

矩阵 \mathbf{X} 表示输入变量组合 A' 导出的等价类集合，矩阵 \mathbf{Y}_0 由全部 n 个输出的“0”输出向量构成。矩阵 $\mathbf{C}_{s \times n}$ (下文简称 \mathbf{C}) 反映了所有等价类 X_i 与 Y_{j1} 之间的包含关系，称为矩阵 \mathbf{X} 与 \mathbf{Y} 的粒逻辑关系矩阵。 \sqcup 表示集合的势。

下面通过定理 1 说明粒逻辑关系矩阵所表达的信息与含义：

定理 1 $X_i \subseteq Y_{j1}$ 的充分必要条件是 $c_{ij} = 0$ 。

证明 (充分性) 设 $\mathbf{X}_i = (a_{i1} \cdots a_{ik} \cdots a_{il})$, $\mathbf{Y}_{j0} = (b'_{j1} \cdots b'_{jk} \cdots b'_{jl})$, 其中 $a_{ik}, b_{jk} \in \{0, 1\}$, $1 \leq i, j \leq l$, 令 $c_{ij} = \mathbf{X}_i \mathbf{Y}_{j0}^T = (a_{i1} a_{i2} \cdots a_{il})(b'_{j1} b'_{j2} \cdots b'_{jl})^T = \sum_{k=1}^l a_{ik} b'_{kj}$, 若 $c_{ij} = 0$, 说明 \mathbf{X}_i 中的 1 均与 \mathbf{Y}_{j0} 中的 0 对应相乘，即 X_i 中的 1 均与 Y_{j1} 中的 1 相对应，故 $X_i \subseteq Y_{j1}$ 。

(必要性) 若 $X_i \subseteq Y_{j1}$, 则与 X_i 中全部 1 的位置对应的 Y_{j1} 中的值也为 1, 即 \mathbf{Y}_{j0} 中对应位置为 0, $c_{ij} = 0$ 。证毕

定理 2 若 $X_i \subseteq Y_{j1}$, 则信息粒 X_i 能正确区分 Y_{j1} 中的部分 1, 可形成一条以 Y_j 为输出的逻辑规则。

定义 5 矩阵 \mathbf{X} 的度：粒度为 ω 时，定义 $\text{He} = \{\sum_{i=1}^s |\mathbf{X}_i| |c_{ij} = 0\}$ 为矩阵 \mathbf{X} 的度。它反映了在粒度 ω 下，矩阵 \mathbf{X} 表示的 s 个等价类能够正确区分 Y_j 中属性值为 1 的元素个数之和。

3 多输入多输出真值表快速并行约简算法

3.1 符号定义

$\omega: \forall A' \subseteq A, \omega$ 表示 A' 中逻辑输入变量的个数，用来表征真值表输入空间的粒度， $1 \leq \omega \leq m$ 。

G_ω : 粒度 ω 下输入属性的所有组合, $G_\omega \subseteq P(A), G_\omega \neq \phi, P(A):A$ 的幂集, $|G_\omega| = C_m^\omega$ 。

X_ω : 粒度 ω 下 G_ω 产生的粒矩阵, $|X_\omega| = C_m^\omega$ 。

Y_0 : 由所有 n 个输出逻辑变量的“0”输出向量组成, $Y_0^T: Y_0$ 的转置。

$C_\omega = X_\omega Y_0^T, C_\omega$ 表示粒度 ω 下, 所有 X_ω 与输出之间的粒逻辑关系矩阵, $|C_\omega| = C_m^\omega$ 。

3.2 算法描述

区别于传统的任何一种算法, 基于粒计算的多输入多输出(m 输入 n 输出)真值表快速并行约简算法的主要思想是: 从知识工程的角度, 从粗到细, 在不同粒度空间下对逻辑信息系统进行约简, 约简的本质是借用代数规则和统计规律为启发式信息, 通过粒矩阵运算, 用最少的输入变量和变量值快速辨识出所有输出属性中所有的属性值 1。算法描述如表 1 所示。

3.3 计算实例

例 1 数字显示译码器^[3]

发光二极管七段数字显示器通常采用七段(a, b, c, d, e, f, g)字形显示, 如图 1 所示。表 2 给出一种七段译码器的功能表, 它接收 8421BCD 码, 输出逻辑值为 1 时, 对应的字段点亮, 输出为 0 时, 对应的字段熄灭。显示的字形图如图 2 所示。

表 1 基于粒计算的多输入多输出真值表快速并行约简算法

- 输入: LIS = (U, R, V, f), $A = \{A_{m-1}, A_{m-2}, \dots, A_0\}$, $Y = \{Y_1, Y_2, \dots, Y_n\}$ 。
- 输出: n 个最简逻辑规则, $rl = \{rl_1, rl_2, \dots, rl_n\}$ 。
- 步骤 1 设置初始值 $\omega = 1$, 并计算 Y_0 。
 - 步骤 2 计算 X_ω 和 C_ω 。
 - 步骤 3 找到所有 $c_{ij} = 0$ 及与之对应的 X_ω , 并计算 He。
 - 步骤 4 按照 He 从大到小的顺序对 X_ω 进行排序。
 - 步骤 5 按照 X_ω 的顺序, 依次记录其中 X_i 区分出来的 1 的位置, 存入 $obj = \{obj_1, obj_2, \dots, obj_n\}$ 。
 - 步骤 6 判断 obj_i 中是否已经存在 X_i 区分出来的 1 的全部位置;
 - 步骤 7 若是, 不记录 X_i ;
 - 步骤 8 否则, 记录 X_i , 将其视为 rl_i 中的一条逻辑规则;
 - 步骤 9 判断 rl 中的规则是否覆盖输出中的全部 1;
 - 步骤 10 若是, 结束计算;
 - 步骤 11 否则, 继续计算。
 - 步骤 12 若 $\omega < m$, 则 $\omega = \omega + 1$, 返回步骤 2 继续计算, 否则转步骤 13。
 - 步骤 13 若有 Y_i 中的某些 1 仍未被区分出, 则将真值表中与之对应的记录视为一条逻辑规则并入 rl_i 中。
 - 步骤 14 输出 rl 中的逻辑规则。

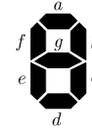


图 1 七段字形显示

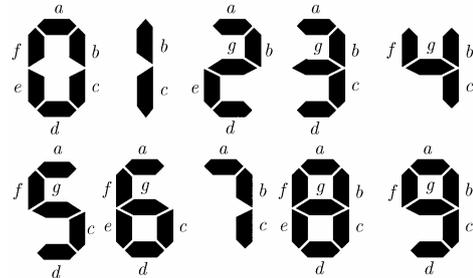


图 2 十进制数字

在这个典型的电路中, $m=4, n=7$ 。本例仅以 a 和 b 并行输出为例, 详细说明计算过程: 如表 2 所示, $U = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$, $A = \{A_0, A_1, A_2, A_3\}$, $Y = \{a, b\}$ 表示输出为 a 和 b 字段, $V = \{0, 1\}$ 。

(1) 设置初始值 $\omega = 1$, 此时粒度最粗。下文用 i 表示 A_i , 可以得到

$$|G_{\omega=1}| = C_4^1 = 4, G_{\omega=1} = \{\{A_0\}, \{A_1\}, \{A_2\}, \{A_3\}\}$$

$$U / Y_0 = \{\{a_0\}, \{b_0\}\} = \{\{2, 5\}, \{6, 7\}\}$$

$$Y_0 = \begin{pmatrix} 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \end{pmatrix}$$

$$U / A_0 = X_0 = \{\{2, 4, 6, 8, 10\}, \{1, 3, 5, 7, 9\}\}$$

$$U / A_1 = X_1 = \{\{3, 4, 7, 8\}, \{1, 2, 5, 6, 9, 10\}\}$$

$$U / A_2 = X_2 = \{\{5, 6, 7, 8\}, \{1, 2, 3, 4, 9, 10\}\}$$

$$U / A_3 = X_3 = \{\{9, 10\}, \{1, 2, 3, 4, 5, 6, 7, 8\}\}$$

表 2 七段译码器的逻辑信息系统

U	逻辑输入 A				逻辑输出 Y						
	A ₃	A ₂	A ₁	A ₀	a	b	c	d	e	f	g
1	0	0	0	0	1	1	1	1	1	1	0
2	0	0	0	1	0	1	1	0	0	0	0
3	0	0	1	0	1	1	0	1	1	0	1
4	0	0	1	1	1	1	1	1	0	0	1
5	0	1	0	0	0	1	1	0	0	1	1
6	0	1	0	1	1	0	1	1	0	1	1
7	0	1	1	0	1	0	1	1	1	1	1
8	0	1	1	1	1	1	1	0	0	0	0
9	1	0	0	0	1	1	1	1	1	1	1
10	1	0	0	1	1	1	1	1	0	1	1

$$\begin{aligned} \mathbf{X}_0 &= \begin{pmatrix} 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \end{pmatrix}, \quad \mathbf{X}_1 = \begin{pmatrix} 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \end{pmatrix} \\ \mathbf{X}_2 &= \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}, \quad \mathbf{X}_3 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \end{pmatrix} \\ \mathbf{C}_0 &= \mathbf{X}_0 \mathbf{Y}_0^T = \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}, \quad \mathbf{C}_1 = \mathbf{X}_1 \mathbf{Y}_0^T = \begin{pmatrix} 0 & 1 \\ 2 & 1 \end{pmatrix} \\ \mathbf{C}_2 &= \mathbf{X}_2 \mathbf{Y}_0^T = \begin{pmatrix} 1 & 2 \\ 1 & 0 \end{pmatrix}, \quad \mathbf{C}_3 = \mathbf{X}_3 \mathbf{Y}_0^T = \begin{pmatrix} 0 & 0 \\ 2 & 2 \end{pmatrix} \end{aligned}$$

根据以上计算以及定义 5:He 的值为矩阵 \mathbf{C} 中 $c_{ij} = 0$ 所对应的矩阵 \mathbf{X} 中的向量 \mathbf{X}_i 中所包含的元素的个数之和即 $\text{He} = \left\{ \sum_{i=1}^s |\mathbf{X}_i|_{c_{ij}=0} \right\}$ 。在本例中, \mathbf{C}_0 中不存在 $c_{ij} = 0$ 的元素, 所以 $\text{He} = 0$ 。 \mathbf{C}_1 中只有 $c_{11} = 0$, 矩阵 \mathbf{X}_1 中的第 1 个等价类中元素个数为 4, 所以 $\text{He} = 4$ 。 \mathbf{C}_2 中 $c_{22} = 0$, 矩阵 \mathbf{X}_2 中的第 2 个等价类中元素个数为 6, 所以 $\text{He} = 6$ 。同理, \mathbf{C}_3 中 $c_{11} = c_{12} = 0$, 所以 $\text{He} = 2$ 。

根据 He 的大小对 $X_{\omega=1}$ 进行排序, 见表 3, 其中灰色规则表示重复识别的规则, 不做记录。此时 $\text{obj}_a = \{3, 4, 7, 8, 9, 10\}$, $\text{obj}_b = \{1, 2, 3, 4, 9, 10\}$, 均未完全区分输出中的 1, 继续计算。

(2)在细粒度 $\omega = 2$ 上继续计算:

$$\begin{aligned} |G_{\omega=2}| = C_4^2 = 6, \quad G_{\omega=2} = & \{ \{A_0 A_1\}, \{A_0 A_2\}, \{A_0 A_3\}, \\ & \{A_1 A_2\}, \{A_1 A_3\}, \{A_2 A_3\} \} \end{aligned}$$

可得

表 3 $\omega = 1$ 的计算过程

$X_{\omega=1}$	He	obj	rl
\mathbf{X}_2	6	$\text{obj}_b = \{1, 2, 3, 4, 9, 10\}$	$\text{rl}_b = \{A_2 = 0 \rightarrow b = 1\}$
\mathbf{X}_1	4	$\text{obj}_a = \{3, 4, 7, 8\}$	$\text{rl}_a = \{A_1 = 1 \rightarrow a = 1\}$
\mathbf{X}_3	2	$\text{obj}_a = \{9, 10\}$	$\text{rl}_a = \{A_3 = 1 \rightarrow a = 1\}$
	2	$\text{obj}_b = \{9, 10\}$	$\text{rl}_b = \{A_3 = 1 \rightarrow b = 1\}$

$$\begin{aligned} U / \text{IND}(A_0 A_1) = X_{01} &= \{ \{1, 5, 9\}, \{3, 7\}, \{4, 8\}, \{2, 6, 10\} \} \\ U / \text{IND}(A_0 A_2) = X_{02} &= \{ \{1, 3, 9\}, \{5, 7\}, \{6, 8\}, \{2, 4, 10\} \} \\ U / \text{IND}(A_0 A_3) = X_{03} &= \{ \{1, 3, 5, 7\}, \{9\}, \{10\}, \{2, 4, 6, 8\} \} \\ U / \text{IND}(A_1 A_2) = X_{12} &= \{ \{1, 2, 9, 10\}, \{5, 6\}, \{7, 8\}, \{3, 4\} \} \\ U / \text{IND}(A_1 A_3) = X_{13} &= \{ \{1, 2, 5, 6\}, \{9, 10\}, \{3, 4, 7, 8\} \} \\ U / \text{IND}(A_2 A_3) = X_{23} &= \{ \{1, 2, 3, 4\}, \{9, 10\}, \{5, 6, 7, 8\} \} \end{aligned}$$

$$\begin{aligned} \mathbf{X}_{01} &= \begin{pmatrix} 1000100010 \\ 0010001000 \\ 0001000100 \\ 0100010001 \end{pmatrix}, \quad \mathbf{X}_{02} = \begin{pmatrix} 1010000010 \\ 0000101000 \\ 0000010100 \\ 0101000001 \end{pmatrix} \\ \mathbf{X}_{03} &= \begin{pmatrix} 1010101000 \\ 0000000010 \\ 0000000001 \\ 0101010100 \end{pmatrix}, \quad \mathbf{X}_{12} = \begin{pmatrix} 1100000011 \\ 0000110000 \\ 0000001100 \\ 0011000000 \end{pmatrix} \\ \mathbf{X}_{13} &= \begin{pmatrix} 1100110000 \\ 0000000011 \\ 0011001100 \end{pmatrix}, \quad \mathbf{X}_{23} = \begin{pmatrix} 1111000000 \\ 0000000011 \\ 0000111100 \end{pmatrix} \\ \mathbf{C}_{01} &= \begin{pmatrix} 10 \\ 01 \\ 00 \\ 11 \end{pmatrix}, \quad \mathbf{C}_{02} = \begin{pmatrix} 00 \\ 11 \\ 01 \\ 10 \end{pmatrix}, \quad \mathbf{C}_{03} = \begin{pmatrix} 11 \\ 00 \\ 00 \\ 11 \end{pmatrix} \\ \mathbf{C}_{12} &= \begin{pmatrix} 10 \\ 11 \\ 01 \\ 00 \end{pmatrix}, \quad \mathbf{C}_{13} = \begin{pmatrix} 21 \\ 00 \\ 01 \end{pmatrix}, \quad \mathbf{C}_{23} = \begin{pmatrix} 10 \\ 00 \\ 12 \end{pmatrix} \end{aligned}$$

计算并根据 He 的大小对 $X_{\omega=2}$ 进行排序, 见表 4。表 4 中灰色的规则表示其分辨出来的行已经被包含在 obj 中, 不做记录。由于本次计算至 $X_{\omega=2}$ 中的前 3 个 \mathbf{X}_i 就已经完成了运算。因此表 4 只列出了这 3 个 \mathbf{X}_i 。

表 4 $\omega = 2$ 的计算过程

$X_{\omega=2}$	He	obj	rl
\mathbf{X}_{02}	3	$\text{obj}_a = \{1, 3, 9\}$	$\text{rl}_a = \{A_0 = 0 \wedge A_2 = 0 \rightarrow a = 1\}$
	2	$\text{obj}_a = \{6, 8\}$	$\text{rl}_a = \{A_0 = 1 \wedge A_2 = 1 \rightarrow a = 1\}$
\mathbf{X}_{12}	3	$\text{obj}_b = \{1, 3, 9\}$	$\text{rl}_b = \{A_0 = 0 \wedge A_2 = 0 \rightarrow b = 1\}$
	3	$\text{obj}_b = \{2, 4, 10\}$	$\text{rl}_b = \{A_0 = 1 \wedge A_2 = 0 \rightarrow b = 1\}$
	4	$\text{obj}_b = \{1, 2, 9, 10\}$	$\text{rl}_b = \{A_1 = 0 \wedge A_2 = 0 \rightarrow b = 1\}$
	2	$\text{obj}_b = \{3, 4\}$	$\text{rl}_b = \{A_1 = 1 \wedge A_2 = 1 \rightarrow b = 1\}$
\mathbf{X}_{01}	3	$\text{obj}_b = \{1, 5, 9\}$	$\text{rl}_b = \{A_0 = 0 \wedge A_1 = 0 \rightarrow b = 1\}$
	2	$\text{obj}_b = \{4, 8\}$	$\text{rl}_b = \{A_0 = 1 \wedge A_1 = 1 \rightarrow b = 1\}$

计算至 \mathbf{X}_{02} 后, 输出属性 $\{a\}$ 的计算满足终止条件, 以后计算不再考虑属性 $\{a\}$ 。此时, $rl_a = \{A_1 = 1 \rightarrow a = 1, A_3 = 1 \rightarrow a = 1, A_0 = 0 \wedge A_2 = 0 \rightarrow a = 1, A_0 = 1 \wedge A_2 = 1 \rightarrow a = 1\}$ 。计算至 \mathbf{X}_{01} 后, 属性 $\{b\}$ 中的 1 也已经全部被区分, 此时: $rl_b = \{A_2 = 0 \rightarrow b = 1, A_0 = 0 \wedge A_1 = 0 \rightarrow b = 1, A_0 = 1 \wedge A_1 = 1 \rightarrow b = 1\}$ 。结束计算。

输出逻辑规则分别为

$$Y_a = A_1 + A_3 + A_0 A_2 + \bar{A}_0 \bar{A}_2, Y_b = \bar{A}_2 + A_0 A_1 + \bar{A}_0 \bar{A}_1$$

同理, 当选择全部 7 个输出并行计算时, 可以并行得到与文献[3]相同的计算结果:

$$Y_a = A_1 + A_3 + A_0 A_2 + \bar{A}_0 \bar{A}_2, Y_b = \bar{A}_2 + A_0 A_1 + \bar{A}_0 \bar{A}_1,$$

$$Y_c = A_0 + A_2 + A_3 + \bar{A}_1, Y_d = A_3 + \bar{A}_0 A_1 + \bar{A}_0 \bar{A}_2$$

$$+ A_1 \bar{A}_2 + A_0 \bar{A}_1 A_2, Y_e = \bar{A}_0 A_1 + \bar{A}_0 \bar{A}_2,$$

$$Y_f = A_3 + \bar{A}_0 \bar{A}_1 + \bar{A}_0 A_2 + \bar{A}_1 A_2,$$

$$Y_g = A_3 + \bar{A}_0 A_2 + A_1 \bar{A}_2 + \bar{A}_1 A_2$$

4 算法分析

4.1 算法正确性证明

在单输出情况下, 本算法与卡诺图约简算法等价。证明如下:

令 LIS 具有 m 个逻辑输入, K 为 LIS 对应的卡诺图^[1,3]。 K 中任意一个依照规则得到的圈 Ω_i 由 2^k 个取值为 1 的最小项组成 ($0 \leq k \leq m$), $\Omega = \cup\{\Omega_i | \Omega_i \in K\}$ 。 $\Theta_i = \{\text{com}(\Omega_i)\}$, $\text{com}(\Omega_i)$ 表示 Ω_i 中的公共因子。根据卡诺图化简原则, Θ_i 是 Ω_i 的约简, $Y = \sum \Theta_i$ 构成卡诺图的化简结果。

令 $|\Omega_i|$ 为 Ω_i 中最小项个数, $|\Theta_i|$ 为 Ω_i 中公共因子的变量个数, 即约简后输入变量的个数。 k , $|\Omega_i|$ 和 $|\Theta_i|$ 存在如下关系: $|\Omega_i| = 2^k$, $|\Theta_i| = m - k$ 。

由式(10)以及定理 1、定理 2 可知: 在粒度 ω 下, 若存在 $c_{ij} = 0$, 则对应的信息粒 X_i 能正确区分 Y_j 中的部分属性值 1, 形成一条以 Y_j 为输出的逻辑规则, X_i 中 1 的位置表明了表中被约简的规则的位置。在粒度 ω 下所有的约简结果 R_ω 都包含 ω 个输入变量。因此有: $\omega = |\Theta_i|$, $R_\omega = \cup\{\Theta_i | |\Theta_i| = \omega\}$ 。对于 m 输入 LIS, $k = m - \omega$, 因此粒度 ω 下的一条规则约简, 必然对应卡诺图中一个包含 $|\Omega_i| = 2^{m-\omega}$ 个最小项的圈。粒度 ω 下的所有规则约简, 必然对应卡诺图中所有包含 $|\Omega_i| = 2^{m-\omega}$ 个最小项的圈。

等价性证明保证了本文算法的正确性。

4.2 算法复杂性分析

对 m 输入 n 输出真值表, 需要化简 n 个卡诺图, Q-M 算法复杂度为 $O(n \times 3^m)$, 立方体算法的复杂性

相对较低, 和每次循环中包含的立方体个数有关。假设第 i 次循环中立方体个数为 N_i , 其中 $1 \leq i \leq m$, 对 m 输入 n 输出真值表, 总复杂度为 $O(n \times m \times C_{N_i}^2)$ 。与文献[15]相比, 本文将判断 $NE(i) = 1$ 变为判断 $c_{ij} = 0$, 减少了内存占用空间和计算量。同时, 实现对 n 输出进行并行计算, 时间复杂度为 $O(2^m)$, 由于本文算法采用具有稀疏性质的粒矩阵的计算方式, 既利于编程实现, 也节约了存储空间。因此与其他算法相比, 本文算法具有显著优势。

4.3 无关项的处理

当真值表含有无关项时, 应用卡诺图法、Q-M 算法和立方体算法时, 无关项通常被假设为 1 参与运算, 最后需要验证无关项是否构成冗余项, 如果是则删除, 否则留下得到最终结果。将真值表当作 LIS 进行处理, 不需要对无关项的输出做任何假设, 通过设定粒度, 会自动按需将无关项放入卡诺图法对应的大圈中, 化简结果不受影响。

5 结束语

Q-M 算法及其改进算法本质是卡诺图方法的扩展。立方体方法本质依据逻辑运算规则^[2]。本文提出的基于粒矩阵的真值表快速并行约简算法, 本质上是从信息处理角度在不同粒度空间下实现逻辑规则的并行约简, 在思路与传统方法不同, 在算法上还有进一步优化的空间。

本文创新之处在于: (1) 将真值表当作逻辑信息系统, 在不同的粒度空间, 通过定义粒矩阵及其基本运算, 并行实现逻辑规则化简; (2) 利用二值逻辑特性, 将文献[16]中寻找矩阵中的 $NE(i) = 1$ 转变为寻找 $c_{ij} = 0$, 减小了相关的数值比较和搜索, 并将输出矩阵规模减半, 从而降低了算法的空间复杂度和时间复杂度; (3) 通过矩阵运算并设置启发式算子和终止条件, 并行得到所有输出的逻辑化简, 提高了计算效率; (4) 不需要单独考虑无关项; (5) 证明了本文算法与经典的卡诺约简方法是等价的。以上几点决定了本文算法的快速性和可靠性。

真值表化简是数字逻辑电路分析和设计中的重要问题, 本文将等价关系的粒计算模型应用于 m 输入 n 输出真值表化简, 一方面丰富了粒计算理论研究, 另一方面为数字逻辑电路的分析与设计提供了新思路。如何将二元关系的粒计算模型^[17]推广到时序逻辑电路中的状态转移表(图)的化简是我们正在进行的工作。

参考文献

- [1] 阎石. 数字电子技术基础[M]. 第 5 版, 北京: 高教出版社, 2006: 29-54.

- Yan Shi. Fundamentals of Digital Electronics[M]. 5th Edition, Beijing: Higher Education Press, 2006: 29-54.
- [2] 边计年, 薛宏熙, 苏明, 等. 数字系统设计自动化[M]. 第2版 北京: 清华大学出版社, 2005: 221-226.
- Bian Ji-nian, Xue Hong-xi, Su Ming, *et al.* Design Automation for Digital Systems[M]. 2nd Edition, Beijing: Tsinghua University Press, 2005: 221-226.
- [3] 刘宝琴. 数字电路与系统[M]. 第2版, 北京: 清华大学出版社, 2007: 39-56.
- Liu Bao-qin. Digital Circuits and Systems[M]. 2nd Edition, Beijing: Tsinghua University Press, 2007: 39-56.
- [4] Cepek O, Kucera P, and Kurik S. Boolean functions with long prime implicants[J]. *Information Processing Letters*, 2013, 113(19-21): 698-703.
- [5] Dusa A. A mathematical approach to the boolean minimization problem[OL]. <http://www.compass.org/wpseries/Dusa2007a.pdf>, 2010.
- [6] Altun M and Marc D. Riedel: lattice-based computation of Boolean functions[OL]. http://www.mriedel.ece.umn.edu/wiki/images/7/7b/Altun_Riedel_Lattice-Based_Computation_of_Boolean_Functions.pdf, 2010.
- [7] Hemaspaandra E and Schnoor H. Minimization for generalized Boolean formulas[OL]. <http://arxiv.org/pdf/1104.2312.pdf>, 2011.
- [8] Zadeh L A. Some reflections on soft computing, granular computing and their roles in the conception, design and utilization of information/intelligent systems[J]. *Soft Computing*, 1998(2): 23-25.
- [9] 苗夺谦, 徐菲菲, 姚一豫. 粒计算的集合论描述[J]. 计算机学报, 2012, 35(2): 351-363.
- Miao Duo-qian, Xu Fei-fei, and Yao Yi-yu. Set-theoretic formulation of granular computing[J]. *Chinese Journal of Computers*, 2012, 35(2): 351-363.
- [10] 张清华, 幸禹可, 周玉华. 基于粒计算的增量式知识获取方法[J]. 电子与信息学报, 2011, 33(2): 435-441.
- Zhang Qing-hua, Xing Yu-ke, and Zhou Yu-hua. The incremental knowledge acquisition algorithm based on granular computing[J]. *Journal of Electronics & Information Technology*, 2011, 33(2): 435-441.
- [11] Qian Yu-hua, Hu Zhang, Sang Yan-li, *et al.* Multigranulation decision-theoretic rough sets[J]. *International Journal of Approximate Reasoning*, 2014, 55(1): 225-237.
- [12] Yang Xi-bei, Qian Yu-hua, and Yang Jing-yu. On characterizing hierarchies of granulation structures via distances[J]. *Fundamenta Informaticae*, 2013, 123(3): 365-380.
- [13] Min Fan, Hu Qing-hua, and Zhu W. Feature selection with test cost constraint[J]. *International Journal of Approximate Reasoning*, 2014, 55(1): 167-179.
- [14] Qian Jin, Miao Duo-qian, Zhang Ze-hua, *et al.* Parallel attribute reduction algorithms using MapReduce[J]. *Information Science*, 2014, 279(20): 671-690.
- [15] 陈泽华, 张裕, 谢刚. 基于粒计算的最简决策规则挖掘算法[J]. 控制与决策, 2015, 30(1): 143-148.
- Chen Ze-hua, Zhang Yu, and Xie Gang. A mining algorithm for concise decision rules based on granular computing[J]. *Control and Decision*, 2015, 30(1): 143-148.
- [16] 陈泽华, 曹长青, 谢刚. 基于粒矩阵的多变量真值表快速约简算法[J]. 模式识别与人工智能, 2013, 26(8): 745-750.
- Chen Ze-hua, Cao Chang-qing, and Xie Gang. Granular matrix based rapid reduction algorithm for multivariable truth table[J]. *Pattern Recognition and Artificial Intelligence*, 2013, 26(8): 745-750.
- [17] Chen Ze-hua, Lin T Y, and Xie Gang. Knowledge approximations in binary relation: granular computing approach[J]. *International Journal of Intelligent System*, 2013, 28(9): 843-864.
- 陈泽华: 女, 1974年生, 副教授, 主要研究方向为粒计算、智能信息处理与智能控制。
- 马贺: 女, 1989年生, 硕士生, 研究方向为粒计算、组合逻辑优化。