

基于粒计算的逻辑函数快速粒约简算法

马 贺 张 裕 陈泽华

(太原理工大学信息工程学院 太原 030024)

摘 要 逻辑函数是描述数字电路中输入变量与输出变量之间逻辑因果关系的重要工具,研究逻辑函数的约简具有重要的理论和实际意义。针对计算机化简逻辑函数普遍存在的算法复杂度高、运算速度慢的问题,将粒计算思想与启发式搜索相结合来约简逻辑函数。首先将逻辑函数转化为最小项之和的表达形式,按照粒度由粗到细的次序,在不同粒度下的知识空间中利用吸收律和最小项之间的统计信息求取信息粒,当所有信息粒对应的最小项覆盖论域时,算法结束。算法由 MATLAB 编程实现。通过计算实例和算法复杂度分析证明了算法的快速性和有效性。

关键词 粒计算,启发式搜索,逻辑函数,知识约简

中图分类号 TP393 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2015.6.017

Rapid Logic Function Reduction Algorithm Based on Granular Computing

MA He ZHANG Yu CHEN Ze-hua

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

Abstract Multivariable logic function is an important tool in the digital circuit that describes the causal relationship between input and output variables. The research on multivariable logic function reduction plays an important role in theoretical and practical significance. However there is no effective means with low complexity. In this work, we first decomposed the multivariable logic function into different knowledge spaces and found heuristic information. Then we used the implicit statistical information to reduce the information granules in different knowledge spaces. Finally we constructed the results. Thus we finally proposed an algorithm to the multivariable logic function (it is also available to the ones that contain the 'don't care term'), and then reduced and realized it in MATLAB. The effect is obvious.

Keywords Granular computing, Heuristic search, Multivariable logic function, Knowledge reduction

1 引言

在数字电路中,逻辑函数是描述输入变量与输出变量之间的逻辑因果关系的重要工具。逻辑函数的每一种表达方式都对应一种具体的电路结构,逻辑函数的约简能够减少电路元器件个数,降低电路复杂性和功耗,提高电路可靠性。

传统的逻辑函数化简方法主要有公式化简法、卡诺图化简法以及基于卡诺图的 Q-M 化简法等。然而,公式化简法要求灵活运用布尔代数的各种规则且化简方法没有一定的规律可循。卡诺图化简法简单明了,容易理解,但不适合化简输入变量数大于 5 的逻辑函数。以上两种方法都不利于程序的实现^[1]。Q-M 算法的指导思想是利用相邻项的合并化简逻辑函数,规律性强,可利用计算机进行编程,但很繁琐,尤其是变量数目多时,工作量很大^[2]。因此学者们提出了一些改进方法^[3-5]和基于人工智能的逻辑函数化简方法^[6,7],但这些方法没有从根本上解决算法复杂度高的问题。

粒计算是一种解决问题的方法与可计算的数学模型^[8,9],它借助于其他学科的哲学思想和方法论,并将它们抽

象成为与具体领域无关的方法和策略^[10],其核心思想是将要求解的问题分而治之,再将所得到的结果进行合并。粒计算分层粒化思想、多粒度空间的各种运算以及对问题的求解方式符合人类从各个角度分析问题、求解问题的认知规律。

本文提出了一种基于粒计算的快速粒约简算法,并用 MATLAB 实现,达到了快速约简多变量逻辑函数的目的。首先将逻辑函数转化为最小项表达式之和的形式,并写出其对应的数字最小项,然后按照粒度由粗到细的次序,在不同粒度下利用最小项间的统计信息求取信息粒,设置停止条件,算法结束时即可得到最终约简结果。

2 预备知识

定义 1(逻辑函数)^[1] 如果以逻辑变量作为输入,以运算结果作为输出,那么当输入变量的取值确定后,输出的取值便随之而定。因此输入与输出之间是一种函数关系,这种关系称为逻辑函数,记作:

$$Y=F(A,B,C,\dots)$$

由于变量和输出(函数)的取值只有 0 和 1 两个状态,因

到稿日期:2014-04-20 返修日期:2014-07-29 本文受山西省回国留学人员科研资助项目(2013-031),山西省留学回国人员科技活动择优资助项目(2011-172),国家自然科学基金(61402319)资助。

马 贺(1989-),女,硕士,主要研究方向为粒计算、智能信息处理;张 裕(1989-),男,硕士,主要研究方向为粒计算、粗糙集理论及应用;陈泽华(1974-),女,博士,副教授,CCF 会员,主要研究方向为粒计算、智能信息处理,E-mail:zehuachen@163.com(通信作者)。

此本文所讨论的都是二值函数。

常用的逻辑函数表示方法有逻辑真值表、逻辑函数式、逻辑图、波形图、卡诺图和硬件描述语言等。

定义 2(最小项)^[1] 在 n 变量逻辑函数中,若 ϕ 为包含 n 个因子的乘积项,而且这 n 个变量均以原变量或者反变量的形式在 ϕ 中出现一次,则称 ϕ 为该组变量的最小项。将原变量用‘1’表示,反变量用‘0’表示,这样的最小项称为数字最小项。

由于最小项与逻辑函数的几种表示方法之间可以相互转换,因此本文只讨论基于最小项表示的逻辑函数的约简,以下简称逻辑函数的约简。

例 1 对于最小项 $AB\bar{C}D, A\bar{B}\bar{C}D, A\bar{B}C\bar{D}, ABC\bar{D}$, 其对应的数字最小项形式为 1011, 1001, 1000, 1010。

定义 3(可约最小项组) 根据布尔逻辑规则,将若干可进行合并的最小项称为可约最小项组,用 RMG(Reductive Minterm Group)表示。

如例 1 中的最小项 $\{AB\bar{C}D, A\bar{B}\bar{C}D, A\bar{B}C\bar{D}, ABC\bar{D}\}$ 是一个可约最小项组,因为组内的最小项可以合并为 $A\bar{B}$ 。

若逻辑函数中的全部最小项用 U 表示,其对应的可约最小项组共有 p 个,即 $U = \bigcup_{i=1}^p RMG_i$,同时允许 $RMG_i \cap RMG_j \neq \emptyset (1 \leq i, j \leq p)$,即全部的可约最小项组对逻辑函数的最小项构成了一个覆盖。

定义 4(信息粒和信息粒度) 在可约最小项组中,所有最小项的公共因子称为信息粒,用 G_r 表示;公共因子所包含的输入变量的个数称为粒度,用 ω 表示。

例 1 中的信息粒为 $A\bar{B}$, $\omega=2$ 。

定义 5(信息粒集) 信息粒的集合称为信息粒集(Information Granule Set),用 $IGS = \{Gr_1, Gr_2, \dots, Gr_i, \dots\}$ 表示, $Gr_i \in IGS, i$ 表示第 i 个信息粒。

逻辑函数的约简过程中,会得到不同的信息粒,根据信息粒度大小的不同,找到最少的信息粒集对全部最小项构成覆盖,即可得到最简的逻辑函数。

定理 1 对于 n 输入的 m 个最小项之和,通过布尔代数运算可最终被约简为 1 的充要条件是 $m=2^n$ 。

证明:(充分性)由于每个输入变量的取值只有两种可能(要么为 1,要么为 0),因此 n 输入的最小项的取值组合最多有 2^n 种。

当 $m=2^n$ 时,说明这 m 个最小项遍历了 n 输入的所有取值组合。其对应的卡诺图中,所有的格均为 1,则其约简结果必为 1。

(必要性)对于 n 输入的 m 个最小项之和,若其通过布尔代数运算可最终被合并为 1,说明无论输入变量的取值组合如何,其输出结果必为 1,则其对应的卡诺图中的格全部为 1,此时最小项的个数 $m=2^n$ 。

证毕。

定理 2 对于一个 n 输入的 m 个最小项的可约最小项组,若粒度为 ω ,则该可约最小项组可最终被约简为信息粒的充要条件是 $m=2^{n-\omega}$ 。

证明:将最小项中的信息粒提取出来,则剩余的部分即为 m 个 $n-\omega$ 输入变量的全部最小项。根据定理 1,剩余部分可最终被约简为 1 的充要条件是 $m=2^{n-\omega}$,即 n 输入的 m 个最小项可最终被约简为信息粒的充要条件即为 $m=2^{n-\omega}$ 。

证毕。

例 1 中, $m=4, n=4$, 由于变量 A 取值恒为‘1’,变量 B 取值恒为‘0’,将 $A\bar{B}$ 视为一个信息粒, $\omega=2$, 由于 $2^{4-2}=4$, 因此这些最小项之和可被约简为信息粒 $A\bar{B}$ 。

定理 2 挖掘隐藏在最小项中的统计信息,说明了最小项能够最终被约简的统计条件,将最小项之和基于布尔逻辑关系的约简转化为基于统计关系的约简。

定义 6(无关项) 无关项^[1]指包含以下两种情况的逻辑函数式:

1) 由于每一组输入变量的取值都使一个而且仅有一个最小项的值为‘1’,因此当限制某些输入变量的取值不能出现时,可以用它们对应的最小项恒等于‘0’来表示。

2) 在输入变量的某些取值下,函数值取‘1’或‘0’皆可,并不影响电路的功能。

本文可以实现对包含无关项的逻辑函数的约简。

定理 3 将粗粒度下得到的信息粒视为启发信息粒,在较细粒度下不必计算包含启发信息粒的信息粒。

定理 2 对应于公式化简法中的吸收法则,即 $A+AB=A$ 。

例如,若在 $\omega=1$ 时, IGS 中包含了启发信息粒 A ,则在粒度 $\omega>1$ 的计算中,不必考虑含有 A 的信息粒 ($AB, AC, A\bar{B}\bar{C}, ABCD$ 等)。

3 逻辑函数的快速粒约简算法

3.1 符号定义

X : 系统的输入。

Y : 系统的输出。

G : 输入变量的所有组合, $G \subset P(X)$, 但 $G \neq \emptyset$ 。

U_m : 逻辑函数生成的最小项。

U_d : 所有无关项生成的最小项。

ω : 信息粒的变量个数,这里称作信息粒度, $1 \leq \omega \leq n$, n 为输入变量个数。

RMG_ω : 粒度 ω 下的所有 RMG , $RMG_\omega \subset RMG$ 。

G_ω : 在粒度 ω 下,输入变量的所有组合。 $G_{\omega i}$ 为粒度 ω 下第 i 个变量组合, $G_\omega \subset G, G_{\omega i} \subset G_\omega$ 。

V_ω : 在粒度 ω 下,信息粒所有取值组合。 $V_{\omega i}$ 为粒度 ω 下第 i 个取值组合, $V_\omega \subset V_\omega$ 。

$N_{G_{\omega i}/V_{\omega i}}$: 在粒度 ω 下,输入变量组合为 $G_{\omega i}$,取值组合为 $V_{\omega i}$ 时,最小项的个数。

N : n 输入变量下,粒度为 ω 时, $N=2^{n-\omega}$ 。

定理 2 可重新表述为,对于 m 个 n 输入的最小项之和,能够最终被约简为信息粒的充要条件是 $N_{G_{\omega i}/V_{\omega i}}=N$ 。

例如:输入变量为 $n=3, X=\{A, B, C\}, G(X)=\{\{A\}, \{B\}, \{C\}, \{A, B\}, \{A, C\}, \{B, C\}, \{A, B, C\}\}$ 。

$\omega=1$ 时,粒度最粗,只考虑信息粒的变量数为 1 的情况, $G_1=\{\{A\}, \{B\}, \{C\}\}, V_1=\{\{1\}, \{0\}\}, N=2^{3-1}=4$ 。

$\omega=2$ 时,粒度变细,考虑信息粒的变量数为 2 的情况, $G_2=\{\{A, B\}, \{A, C\}, \{B, C\}\}, V_2=\{\{0, 1\}, \{0, 0\}, \{1, 0\}, \{1, 1\}\}, N=2^{3-2}=2$ 。

3.2 快速粒约简算法描述

本文针对最小项表示的逻辑函数进行约简,其本质是对输入的最小项进行分组,得到不同的可约最小项组,再对可约最小项组进行约简,得到信息粒。在对最小项分组时,按照粒

度从大到小的次序,先寻找粗粒度下的信息粒,再逐步寻找细粒度下的信息粒,这样可以避免冗余计算。同时,本文结合了可约最小项组的统计特性,将布尔逻辑运算转换为统计运算。具体计算步骤如下所示:

Step 1 将给定的逻辑函数转换为数字最小项

Step 2 生成 IGS,具体描述如下:

Procedure

for $\omega=1:n-1$

do 找到 RMG_ω 及对应的 G_r

if $\cup RMG=U_m$

break

end

do 是否有未被包含在 RMG 中的最小项,如果有,将其视为一个信息粒,并加入 IGS 中

do 删除全部由无关项组成的信息粒

do 输出最终结果,即信息粒集

3.3 计算实例

下面以例 2 对算法做详细说明。

例 2 $X(A, B, C, D) = \sum m(0, 1, 3, 5, 6, 7, 9, 10, 15) + d(2, 12)$

根据本文中的算法,具体计算步骤如下:

Step 1 根据给定的逻辑函数(含无关项),生成数字最小项,并将其填入表中。如表 1 所列,其中第二列为判定项,‘*’表示它对应的最小项为无关项,‘1’表示它所对应的最小项为逻辑函数最小项。

表 1 数字最小项

最小项	判定项	A	B	C	D
m_0	1	0	0	0	0
m_1	1	0	0	0	1
m_2	*	0	0	1	0
m_3	1	0	0	1	1
m_5	1	0	1	0	1
m_7	1	0	1	1	1
m_9	1	1	0	0	1
m_{10}	1	1	0	1	0
m_{12}	*	1	1	0	0
m_{15}	1	1	1	1	1

由表 1 可知, $U_m = \{m_0, m_1, m_3, m_5, m_7, m_9, m_{10}, m_{15}\}$, $U_d = \{m_2, m_{12}\}$ 。由输入可知,输入变量个数 $n=4$ 。

Step 2 生成 IGS。取粒度 $\omega=1$,此时 $N=2^{4-1}=8$, $G_1 = \{\{A\}, \{B\}, \{C\}, \{D\}\}$, $V_1 = \{\{0\}, \{1\}\}$ 。

$$N_{A/0}=6, N_{A/1}=4; N_{B/0}=6, N_{B/1}=4;$$

$$N_{C/0}=5, N_{C/1}=5; N_{D/0}=4, N_{D/1}=6。$$

由于任何一项的值均不等于 N ,因此在粒度 $\omega=1$ 时,没有产生任何信息粒。

取粒度 $\omega=2$, $N=2^{4-2}=4$,此时 $G_2 = \{\{A, B\}, \{A, C\}, \{A, D\}, \{B, C\}, \{B, D\}, \{C, D\}\}$, $V_2 = \{\{00\}, \{01\}, \{10\}, \{11\}\}$ 。系统的知识粒度变粗^[11]。依次计算该粒度下的 $N_{G_{\omega i}/V_{\omega i}}$ 。

$$N_{AB/00}=4, N_{AB/01}=2, N_{AB/11}=2, N_{AB/10}=2;$$

$$N_{AC/00}=3, N_{AC/01}=3, N_{AC/11}=2, N_{AC/10}=2;$$

$$N_{AD/00}=2, N_{AD/01}=4, N_{AD/11}=2, N_{AD/10}=2;$$

$$N_{BC/00}=3, N_{BC/01}=3, N_{BC/11}=2, N_{BC/10}=2;$$

$$N_{BD/00}=3, N_{BD/01}=3, N_{BD/11}=3, N_{BD/10}=1;$$

$$N_{CD/00}=2, N_{CD/01}=3, N_{CD/11}=3, N_{CD/10}=2。$$

由于 $N_{AB/00}=N_{AD/01}=N$,因此 $Gr_1=\bar{A}\bar{B}$, $Gr_2=\bar{A}D$,对应的可约最小项组分别为 $RMG_1=\{m_0, m_1, m_2, m_3\}$, $RMG_2=\{m_1, m_3, m_5, m_7\}$ 。由于最小项组包含的最小项没有覆盖 U_m ,因此继续计算。

取粒度 $\omega=3$, $N=2^{4-3}=2$,此时 $G_3 = \{\{A, B, C\}, \{A, B, D\}, \{A, C, D\}, \{B, C, D\}\}$, $V_1 = \{\{000\}, \{001\}, \{010\}, \{011\}, \{100\}, \{101\}, \{110\}, \{111\}\}$ 。

$\omega=2$ 时,结果中已经存在 $\bar{A}\bar{B}$ 和 $\bar{A}D$,由定理 3 可知,以上两个信息粒可以作为启发信息,因此在本次计算中,不必计算 $N_{ABC/000}$, $N_{ABC/001}$, $N_{ABD/000}$, $N_{ABD/001}$, $N_{ABD/011}$, $N_{ACD/001}$, $N_{ACD/011}$,搜索空间减少。依次计算其他 $N_{G_{\omega i}/V_{\omega i}}$,计算方法同上,不再一一列出。

由于 $N_{BCD/001}=N_{BCD/010}=N_{BCD/111}=N$, $Gr_3=\bar{B}\bar{C}D$, $Gr_4=\bar{B}CD$, $Gr_5=BCD$,因此对应的可约最小项组为: $RMG_3=\{m_1, m_3\}$, $RMG_4=\{m_2, m_{10}\}$, $RMG_5=\{m_7, m_{16}\}$ 。此时,可约最小项组包含的最小项已经覆盖 U_m ,计算停止。因此有:

$$IGS = \{Gr_1, Gr_2, Gr_3, Gr_4, Gr_5\}$$

$$= \{\bar{A}\bar{B}, \bar{A}D, \bar{B}\bar{C}D, \bar{B}CD, BCD\}$$

故最终结果为 $Y = \bar{A}\bar{B} + \bar{A}D + \bar{B}\bar{C}D + \bar{B}CD + BCD$ 。

3.4 仿真实验与算法复杂性分析

本文在 MATLAB 平台对该算法进行编程实现,并通过大量的计算实例进行验证,所得结果均与传统计算方法的计算结果相同,从而说明了本算法的正确性。

在 MATLAB 编程环境,输入为逻辑函数数字最小项形式,其具体形式见例 2。粒约简时,将信息粒所对应的列取出,转换为十进制数,再求取统计信息进行判断。所以,本算法的时间复杂度主要体现在信息粒由粗到细的判断,而该过程的时间复杂度为 $O(2^n)$,故本算法的时间复杂度为 $O(2^n)$ 。由于启发式信息存在,因此实际的时间复杂度视具体情况有所降低。与 Q-M 算法及文献[12]中的算法相比,由于本算法按照粒度从粗到细的次序计算信息粒,相当于先画出卡诺图中包含较多最小项的大圈,再画出较小的圈,避免了 Q-M 算法中先画出所有小圈再将其合并为大圈而产生的冗余计算,因此利用启发式算子能够避免文献[12]中大圈覆盖小圈的情况。在时间复杂度相同的情况下,本文中得到的结果比文献[13]中得到的结果更为简洁。由于文献[14]在对逻辑函数中的项进行十进制表示时,其基数取为 3,与本文中的基数 2 相比,空间复杂度较大。而文献[15]利用矩阵计算进行约简,空间复杂度也相对较大。因此,本算法在简洁性、时间复杂度和空间复杂度方面都有所改善。

结束语 逻辑函数约简是数字电路设计中的重要环节,而最小项是逻辑函数的标准形式。本文提出了一种基于粒计算的逻辑函数快速粒约简算法,首先将逻辑函数转换为数字最小项形式,按照粒度从小到大的顺序,利用最小项中的统计信息求出不同粒度下的信息粒,设置停止搜索的条件,即可输出最终的约简结果。实验结果表明了该算法的正确性和有效性。

本算法具有如下特点:1)引入粒的概念,通过信息粒度从粗到细的变化,实现对信息粒的逐步求解;2)挖掘逻辑函数最小项统计特性,并将其应用到快速寻找信息粒的过程。本文工作为实现自动逻辑函数最小项的约简算法提供了新的方法,丰富了粒计算的应用,后续工作还在继续中。

(下转第 87 页)

- [7] Liu D, Li T R, Liang D C. A new discriminate analysis approach under decision-theoretic rough sets[C]//Proceedings of the 6th International Conference on Rough Sets and Knowledge Technology, Lecture Notes in Computer Science. Berlin Heidelberg: Springer-Verlag, 2011; 476-485
- [8] 韩虎, 党建武. 双隶属度模糊粗糙支持向量机[J/OL]. <http://www.cnki.net/kcms/doi/10.3778/j.jssn.1002-8331.1311-0260.html>
Han Hu, Dang Jian-wu. Fuzzy rough support vector machine with dual membership[J/OL]. <http://www.cnki.net/kcms/doi/10.3778/j.jssn.1002-8331.1311-0260.html>
- [9] Yao Y Y. Three-way decisions with probabilistic rough sets[J]. Information Sciences, 2010, 180(3): 341-353
- [10] 刘盾, 姚一豫, 李天瑞. 三支决策粗糙集[J]. 计算机科学, 2011, 38(1): 246-250
Liu Dun, Yan Yi-yu, Li Tian-rui. Three-way Decision-theoretic Rough Sets[J]. Computer Science, 2011, 38(1): 246-250
- [11] Yao Y Y. Decision-theoretic rough set models[C]//Proceedings of the 2th International Conference on Rough Sets and Knowledge Technology, Lecture Notes in Computer Science. Berlin Heidelberg: Springer-Verlag, 2007; 1-12
- [12] Yao Y Y. Three-way decision: an interpretation of rules in rough set theory[C]//Proceedings of the 4th International Conference on Rough Sets and Knowledge Technology, Lecture Notes in Computer Science. Berlin Heidelberg: Springer-Verlag, 2009;
- [13] Liu D, Li T R, Liang D C. Incorporating logistic regression to decision-theoretic rough sets for classifications[J]. International Journal of Approximate Reasoning, 2013; 55(1): 197-210
- [14] Yao Y Y. The superiority of three-way decisions in probabilistic rough set models[J]. Information Sciences, 2011, 181(6): 1080-1096
- [15] 焦李成, 张莉, 周伟达. 支撑向量预选取的中心距离比值法[J]. 电子学报, 2001, 29(3): 383-386
Jiao Li-cheng, Zhang Li, Zhou Wei-da. Pre-extracting Support Vectors for Support Vector Machine[J]. Acta Electronica Sinica, 2001, 29(3): 383-386
- [16] 杨静, 于旭, 谢志强. 改进向量投影的支持向量预选取方法[J]. 计算机学报, 2012, 35(5): 1002-1010
Yang Jing, Yu Xu, Xie Zhi-qiang. Support Vectors Pre-extracting Method Based on Improved Vector Projection[J]. Chinese Journal of Computers, 2012, 35(5): 1002-1010
- [17] 刘盾, 李天瑞, 李华雄. 粗糙集理论: 基于三枝决策视角[J]. 南京大学学报: 自然科学版, 2013, 49(5): 574-581
Liu Dun, Li Tian-rui, Li Hua-xiong. Rough set theory: A three-way decisions perspective[J]. Journal of Nanjing University: Natural Sciences, 2013, 49(5): 574-581
- [18] 李航. 统计学习方法[M]. 北京: 清华大学出版社, 2012; 95-130
Li Hang. Statistics learning method[M]. Beijing: Tsinghua University Press, 2012; 95-130

(上接第 78 页)

参考文献

- [1] 阎石. 数字电子技术基础(第五版)[M]. 北京: 高等教育出版社, 2006
Yan shi. Fundamentals of digital electronics(5th Edition)[M]. Beijing: Higher Education Press, 2006
- [2] 刘宝琴. 数字电路与系统(第二版)[M]. 北京: 清华大学出版社, 2007
Liu Bao-qin. Digital circuits and systems(2nd Edition)[M]. Beijing: Tsinghua University Press, 2007
- [3] 刘淑琴, 虞烈, 张颖红, 等. 化简逻辑函数的新方法[J]. 西安交通大学学报, 1998, 32(8): 48-51
Liu Shu-qin, Yu Lie, Zhang Ying-hong, et al. Method for simplifying logic functions[J]. Journal of Xi'an Jiaotong University, 1998, 32(8): 48-51
- [4] 朱幼莲. 计算机化简逻辑函数的算法研究[J]. 南京理工大学学报: 自然科学版, 2003, 27(4): 405-408
Zhu You-lian. Research on simplification algorithm of logical functions with computer[J]. Journal of Nanjing University of Science and Technology: natural science, 2003, 27(4): 405-408
- [5] 王波. 关于实质本源蕴涵项的识别问题[J]. 计算机研究与发展, 1995, 32(12): 40-44
Wang Bo. On the identification of essential prime implicants[J]. Computer Research and Development, 1995, 32(12): 40-44
- [6] 张义清, 管致锦, 吕彦明. 基于粗糙集的组合逻辑优化算法[J]. 兰州理工大学学报, 2007, 33(1): 88-91
Zhang Yi-qing, Guan Zhi-jun, Lv Yan-ming. An algorithm of combinatory logic optimization based on rough set[J]. Journal of Lanzhou University of Technology, 2007, 33(1): 88-91
- [7] 张义清, 吕彦明, 李洵. 基于粗糙集多输出逻辑函数优化算法的研究[J]. 南通大学学报: 自然科学版, 2006, 4(4): 59-61
Zhang Yi-qing, Lv Yan-ming, Li Xun. An optimal algorithm of multiple output logic function based on rough set[J]. Journal of Nantong University: natural science, 2006, 4(4): 59-61
- [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(1): 23-25
- [9] Lin T Y. Granular Computing: Practices, Theories, and Future Directions[M]//Encyclopedia of Complexity and Systems Science. 2009; 4339-4355
- [10] 苗夺谦, 王国胤, 刘清. 粒计算: 过去, 现在与展望[M]. 北京: 科学出版社, 2007
Miao Duo-qian, Wang Guo-yin, Liu Qing. Granular computing: past, present and future[M]. Beijing: Science Press, 2007
- [11] 刘清. 粗糙集及粗糙推理[M]. 北京: 科学出版社, 2001
Liu Qing. Rough set and rough reasoning[M]. Beijing: Science Press, 2001
- [12] Nosrati M, Hariri M. An Algorithm for Minimizing of Boolean Functions Based on Graph DS[J]. World Applied Programming, 2011, 1(3): 209-214
- [13] Eigen D. Minimizing Boolean Sum of Products Functions[C]//CCSIT 2012. 2012; 36-48
- [14] Duşa A. A mathematical approach to the boolean minimization problem[J]. Quality & Quantity, 2010, 44(1): 99-113
- [15] 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, Xie Gang. Granular matrix based rapid reduction algorithm for multivariable truth table[J]. Pattern Recognition and Artificial Intelligence, 2013, 26(8): 745-750