正规式布尔函数 NPN 等价匹配算法

张菊玲[1]，郭文强[1]，杨晓梅[1]，朱义鑫[1]，杨国武[2,3*]

(1. 新疆财经大学信息管理学院 乌鲁木齐 830012；2. 电子科技大学计算机科学与工程学院 成都 611731；
3. 电子科技大学大数据研究中心 成都 611731)

【摘要】通过对香农分解代数余子式的运算研究，发现了对称变量和独立变量在 NP 等价变换中的 6 个属性，充分利用变量的对称性和独立性 NP 变换后的不变性、独立变量相位不确定性、在 NP 匹配中独立变量识别其他变量和其他变量识别独立变量的不可用性，提出了一种基于正规式的布尔函数 NPN 等价匹配算法。通过对大量 MCNC 标准电路库中电路和随机生成电路的 7-22 变量布尔函数的匹配实验，在两个实验电路集上本文算法与基于高阶通用特征匹配算法相比，匹配过程中的搜索空间平均减少了 58.8%、布尔匹配的速度提高了 45.6%，能够为电路优化和电路映射提供更加快速和有效的布尔匹配。

关 键 词 布尔差分； 正规式； NPN 等价； 独立变量； 香农分解
中图分类号 TP302.2 **文献标志码** A doi:10.12178/1001-0548.2022064

A Boolean Function NPN Equivalent Matching Algorithm Based on Canonical Form

ZHANG Juling[1], GUO Wenqiang[1], YANG Xiaomei[1], ZHU Yixin[1], and YANG Guowu[2,3*]

(1. School of Information Management, Xinjiang University of Finance and Economics Urumqi 830012;
2. School of Computer Science and Engineering, University of Electronic Science and Technology of China Chengdu 611731;
3. Big Data Research Center, University of Electronic Science and Technology of China Chengdu 611731)

Abstract By studying the operation of the cofactor of Shannon decomposition, six attributes of symmetric variables and independent variables in NP equivalent transformation are found. By making full use of the invariance of the symmetry and independence of variables after NP transformation, the phase uncertainty of independent variables, the unavailability of both independent variables to identify other variables and other variables to identify independent variables in the process of matching, we propose an NPN equivalent matching algorithm based on canonical form. We performed matching experiments on the 7-22 variable Boolean functions of a large number of MCNC benchmark circuits and randomly generated circuits.. The experimental results show that the algorithm of this paper reduces the search space in the matching process by 58.8% and increases matching speed by 45.6% on the two experimental circuit sets compared with the algorithm based on higher order general signature. This indicates that the algorithm of this paper can provide faster and more effective Boolean matching for circuit optimization and circuit mapping.

Key words Boolean difference; canonical form; NPN equivalence; independent variable; Shannon decomposition

自十九世纪 30 年代开始就有学者发现并意识到布尔匹配和布尔分类在开关电路中扮演着重要角色，人们开始从各个方面研究电路的设计与优化[1-3]。布尔等价分类和等价匹配作为电路设计和电路优化中的重要技术，逐渐被更多的学者研究[4-5]。

对布尔函数输入或输出的置换运算称为 P 操作，对输入或输出的非运算称为 N 操作。根据对布尔函数输入和输出执行 P 操作和 N 操作的组合，可产生 N 变换、P 变换、NP 变换和 NPN 变换等，也由此形成了布尔函数的 P 等价匹配、NP 等

收稿日期：2022－03－07；修回日期：2022－08－31
基金项目：新疆维吾尔自治区自然科学基金 (2019D01A27)
作者简介：张菊玲 (1977－)，女，博士生，副教授，主要从事逻辑综合、信息安全风险评估方面的研究.
*通信作者：杨国武，E-mail：guowu@uestc.edu.cn

价匹配和 NPN 等价匹配。其中 NPN 等价匹配研究较多，第一个 N 表示输入非，P 表示输入置换，第二个 N 表示输出非。给定一个n输入布尔函数，其 NPN 变换共有$n!2^{n+1}$个，采用穷尽法进行匹配，其复杂度是$O(n!2^{n+1})$。因此，布尔函数的 NPN 等价分类和匹配是 NP 难问题。当前，NPN 等价分类中 n 的最大值是 10[6]。

在数字电路的技术映射和工艺库绑定中，布尔函数 NPN 等价匹配是一个必要环节，其目的是为当前设计的电路找到一个最优的替代电路[7]。现有的布尔函数 NPN 等价匹配方法主要集中在成对比较法、基于正规式和基于 SAT 的方法[8-12]。除此之外，还存在一些基于 Walsh 谱特征和学习的方法[13-15]。

本文基于对香农扩展定理代数余子式运算的研究发现：1) 布尔函数的变量在 NP 变换中其对称性和独立性不变；2) 独立变量具有相位不确定性；3) 利用独立变量区分其他变量的不可用性。利用这些特性首先能更早地判定两个布尔函数的不等价性，其次能有效地减少匹配中产生的候选正规式分支数量，从而减少匹配算法的空间复杂度，提高匹配速度。

1　基本概念与问题陈述

1.1　基本概念

令 $\boldsymbol{X}$ 是一个n维布尔向量$(x_0,x_1,\cdots,x_{n-1})$，布尔函数$f(\boldsymbol{X})$表示为$f(\boldsymbol{X}):B^n\to B$，$f(\boldsymbol{X})$是一个不含无关项的布尔函数。

定义 1　N 变换：将一个非操作φ作用在n维布尔向量$\boldsymbol{X}=(x_0,x_1,\cdots,x_{n-1})$上，$\boldsymbol{X}^{\varphi}(x_0,x_1,\cdots,x_{n-1})=(x_0^{\varphi(0)},x_1^{\varphi(1)},\cdots,x_{n-1}^{\varphi(n-1)})$。当$\varphi(i)=1$时，$x_i^{\varphi(i)}=x_i$；当$\varphi(i)=0$时，$x_i^{\varphi(i)}=\overline{x}_i$。

若$\boldsymbol{X}=(0,1,1,0)$，$\boldsymbol{\varphi}=(0,0,1,1)$，那么$\boldsymbol{X}^{\boldsymbol{\varphi}}=(1,0,1,0)$。

定义 2　P 变换：将一个置换操作π作用在n维布尔向量$\boldsymbol{X}=(x_0,x_1,\cdots,x_{n-1})$上，$\boldsymbol{X}_{\boldsymbol{\pi}}(x_0,x_1,\cdots,x_{n-1})=(x_{\pi(0)},x_{\pi(1)},\cdots,x_{\pi(n-1)})$，其中$\pi(i)=j$且$i,j\in\{0,1,\cdots,n-1\}$。

若$\boldsymbol{X}=(x_0,x_1,x_2,x_3)=(0,1,0,1)$，$\boldsymbol{\pi}=(1,3,2,0)$，那么$\boldsymbol{X}_{\boldsymbol{\pi}}(x_0,x_1,x_2,x_3)=(x_1,x_3,x_2,x_0)=(1,1,0,0)$。

定义 3　NP 变换：将一个 N 变换$\boldsymbol{\varphi}$和 P 变换$\boldsymbol{\pi}$先后作用在向量$\boldsymbol{X}=(x_0,x_1,\cdots,x_{n-1})$上，作用的变换记为$T$，变换$T$作用到布尔向量$\boldsymbol{X}$上有$T\boldsymbol{X}=\boldsymbol{X}_{\boldsymbol{\pi}}^{\boldsymbol{\varphi}}(x_0,x_1,\cdots,x_{n-1})=(x_{\pi(0)}^{\varphi(0)},x_{\pi(1)}^{\varphi(1)},\cdots,x_{\pi(n-1)}^{\varphi(n-1)})$。

若$\boldsymbol{X}=(x_0,x_1,x_2,x_3)=(0,1,1,1)$，$\boldsymbol{\varphi}=(0,0,1,1)$，$\boldsymbol{\pi}=(3,0,2,1)$，$T\boldsymbol{X}=(1,1,1,0)$。

定义 4　NPN 等价：给定两个布尔函数$f(\boldsymbol{X})$和$g(\boldsymbol{X})$，当且仅当存在一个 NP 变换T使得条件$f(T\boldsymbol{X})=g(\boldsymbol{X})$或$f(T\boldsymbol{X})=\overline{g(\boldsymbol{X})}$成立，那么$f(\boldsymbol{X})$与$g(\boldsymbol{X})$是 NPN 等价的，记为$f(\boldsymbol{X})\cong g(\boldsymbol{X})$[9]。

香农扩展定理：等式$f(x)=x_if_{x_i}+\overline{x}_if_{\overline{x}_i}$。

该等式也称为香农分解，其中$f_{x_i}=f[x_i\leftarrow 1]$和$f_{\overline{x}_i}=f[x_i\leftarrow 0]$称为香农代数余子式，其中$f_{x_i}$简记为$f_1$，$f_{\overline{x}_i}$简记为$f_0$。

如一个n变量布尔函数$f(\boldsymbol{X})$通过x_ix_j香农分解后变为 4 个$n-2$变量布尔函数的或，也就是$f(\boldsymbol{X})=x_ix_jf_{x_ix_j}+x_i\overline{x}_jf_{x_i\overline{x}_j}+\overline{x}_ix_jf_{\overline{x}_ix_j}+\overline{x}_i\overline{x}_jf_{\overline{x}_i\overline{x}_j}$。

因此，若两个布尔函数是 NP 等价的，那么利用 NP 变换中的变量进行香农分解，依次分解后的布尔函数也一定是 NP 等价的。

定义 5　布尔差分：布尔函数$f(\boldsymbol{X})$关于变量x_i的布尔差分是$\dfrac{\partial f}{\partial x_i}=f_{x_i}\oplus f_{\overline{x}_i}$，记为$f'_{x_i}$[10]。

定义 6　k 阶通用特征：给定一个n变量布尔函数$f(\boldsymbol{X})$，令$b=x_{i_1}^{\varphi(i_1)}x_{i_2}^{\varphi(i_2)}\cdots x_{i_k}^{\varphi(i_k)}$，有$n-k$变量香农代数余子式$f_b$，$|f_b|$是布尔函数$f_b$的最小项个数，$|f_b|$称为布尔函数$f(\boldsymbol{X})$的$k$阶通用特征[1]。

布尔函数$f(\boldsymbol{X})$真值表中最小项个数称为 0 阶特征，记为$|f|$。若$b=x_0$，$|f_{x_0}|$是$f(\boldsymbol{X})$的 1 阶通用特征；若$b=x_0\overline{x}_1$，则$|f_{x_0\overline{x}_1}|$是$f(\boldsymbol{X})$的 2 阶通用特征。因此，n变量布尔函数$f(\boldsymbol{X})$具有$k(1\leqslant k\leqslant n)$阶特征$2^kC_n^k$个。

定义 7　布尔差分特征：布尔函数$f(\boldsymbol{X})$关于变量x_i的布尔差分特征是布尔函数f'_{x_i}的最小项个数，记为$|f'_{x_i}|$[7]。

定义 8　变量对称：布尔函数$f(\boldsymbol{X})$的两个变量x_i和$x_j/\overline{x}_j$是对称的，当x_i和$x_j/\overline{x}_j$交换后$f(\boldsymbol{X})$不变，即$f(\cdots,x_i,\cdots,x_j,\cdots)=f(\cdots,x_j/\overline{x}_j,\cdots,x_i,\cdots)$[1]。

根据香农代数余子式可以判定变量对称，当$f_{01}\oplus f_{10}=0$时，变量x_i与x_j对称；当$f_{00}\oplus f_{11}=0$时，变量x_i与$\overline{x}_j$对称[16]。

定义 9　独立变量：布尔函数$f(\boldsymbol{X})$的变量x_i是独立变量当且仅当$|f'_{x_i}|=0$[7]。

1.2　正规式

NPN 等价分类将n变量布尔函数划分为多个等价类，每个等价类中的所有布尔函数相互是 NPN 等价的，即它们之间均可相互转换。在上述等价类中选择一个布尔函数作为该类的代表，该代表称为正规式。基于正规式的布尔匹配更多应用于工艺库

绑定。对某个电路函数$f(\boldsymbol{X})$进行工艺库绑定，即在工艺库中寻找合适的基元实现该电路，通过计算$f(\boldsymbol{X})$的正规式并使用哈希查找快速实现[1]。

令由 m 个 NPN 等价布尔函数构成的等价类$E=\{f_0(\boldsymbol{X}),f_1(\boldsymbol{X}),\cdots,f_{m-1}(\boldsymbol{X})\}$，对$\forall i,j\in\{0,1,\cdots,m-1\}$都有$f_i(\boldsymbol{X})\cong f_j(\boldsymbol{X})$，从$E$中选择一个布尔函数作为该等价类的正规式，记为$F(\boldsymbol{X})$。

基于正规式的布尔函数 NPN 等价匹配可描述为：给定两个布尔函数$f(\boldsymbol{X})$和$g(\boldsymbol{X})$，它们的正规式分别为$F(\boldsymbol{X})$和$G(\boldsymbol{X})$，若有$F(\boldsymbol{X})=G(\boldsymbol{X})$，则有$f(\boldsymbol{X})\cong g(\boldsymbol{X})$。

2 基于正规式的布尔匹配算法

2.1 布尔函数 NP 变换中变量的属性

根据对香农分解代数余子式运算的研究，本文得出在布尔函数 NP 等价变换中具有以下 6 个属性。

1) 布尔函数$f(\boldsymbol{X})$中的对称变量x_i经过 NP 变换转换为变量$x_j/\bar{x}_j$，转换后变量仍为对称变量。

2) 若变量x_i与变量x_j对称，经 NP 变换后为$x_k^{\varphi(k)}$和$x_l^{\varphi(l)}$，$x_k^{\varphi(k)}$和$x_l^{\varphi(l)}$是对称的。

3) 若布尔函数$f(\boldsymbol{X})$有独立变量x_i，那么必有等式$|f_{x_i}|=|f_{\bar{x}_i}|$。

证明：因为x_i与$f(\boldsymbol{X})$无关，必有$f_{x_i}=f_{\bar{x}_i}$，所以$|f_{x_i}|=|f_{\bar{x}_i}|$。

4) 若布尔函数$f(\boldsymbol{X})$有独立变量x_i，其他非独立变量x_j和x_k，并有$(|f_{x_j}|,|f'_{x_j}|)=(|f_{x_k}|,|f'_{x_k}|)$，那么有$(|f_{x_ix_j}|,|f'_{x_ix_j}|)=(|f_{x_ix_k}|,|f'_{x_ix_k}|)$。

证明：因变量x_i与$f(\boldsymbol{X})$是独立的，x_i与布尔函数f_{x_j}和f_{x_k}也必然是独立的，如果有$(|f_{x_j}|,|f'_{x_j}|)=(|f_{x_k}|,|f'_{x_k}|)$，那么必有$(|f_{x_ix_j}|,|f'_{x_ix_j}|)=(|f_{x_ix_k}|,|f'_{x_ix_k}|)$。

5) 若布尔函数$f(\boldsymbol{X})$有独立变量x_i，有$b=x_{i_1}^{\varphi(i_1)}x_{i_2}^{\varphi(i_2)}\cdots x_{i_k}^{\varphi(i_k)}(i\neq i_1,i_2,\cdots,i_k)$，则有$|f_{bx_i}|=|f_{b\bar{x}_i}|$。

证明：因为x_i对$f(\boldsymbol{X})$是独立的，那么对f_b也必是独立的，那么必有$f_{bx_i}=f_{b\bar{x}_i}$，所以有$|f_{bx_i}|=|f_{b\bar{x}_i}|$。

6) 布尔函数$f(\boldsymbol{X})$中的独立变量x_i经过 NP 变换后，独立性不变。

如有 3 变量布尔函数$f(\boldsymbol{X})=x_1x_2+\bar{x}_1\bar{x}_2$，若有 N 变换$\boldsymbol{\varphi}=(0,0,1)$和 P 变换$\boldsymbol{\pi}=(2,0,1)$，$f(\boldsymbol{X})$中有独立变量$x_0$且经过变换$x_0$变换为$\bar{x}_2$，$f(T\boldsymbol{X})=\bar{x}_0x_1+x_0\bar{x}_1$，明显$f(T\boldsymbol{X})$中有独立变量$x_2$。

充分条件：根据属性 1)、属性 2) 和属性 6) 可知，若两个布尔函数$f(\boldsymbol{X})$和$g(\boldsymbol{X})$是 NP 等价的，那么这两个布尔函数的变量具有相同的对称变量结构和独立变量结构。

因此，可通过上述充分条件先比较两个布尔函数变量的结构，尽早确定不等价情况。

2.2 本文使用的正规式

文献 [1] 提出了基于高阶通用特征的匹配算法，0～n 阶特征构成高阶通用特征向量，且 NP 等价类中具有最大特征向量的布尔函数作为正规式，证明了每个布尔函数具有唯一的由 0～n 阶特征值组成的特征向量$\boldsymbol{V}^f=(0\text{阶特征},1\text{阶特征},\cdots,n\text{阶特征})$，其中 0 阶特征 1 个，$m$阶特征有$C_n^m$个。利用特征向量计算正规式：按照 0 阶特征、1 阶特征、…、n 阶特征顺序计算布尔函数的通用特征，每计算完一次特征后，根据特征值的大小对变量进行排序，从而找出在某个或某几个排序下使特征向量值最大的情况，最后所得排序就是一个能将布尔函数转换为对应正规式的 NP 变换，再根据该 NP 变换计算出相应的正规式。

在文献 [1] 的基础上，文献 [7] 提出了 DC(difference and cofactor) 特征向量增加了布尔差分特征，这样能够更加快速地找到布尔函数所在等价类中具有最大 DC 特征向量的正规式。

定义 10 DC 特征给定一个n变量布尔函数$f(\boldsymbol{X})$，令$b=x_{i_1}^{\varphi(i_1)}x_{i_2}^{\varphi(i_2)}\cdots x_{i_k}^{\varphi(i_k)}$，有$n-k$变量香农代数余子式$f_b$和布尔差分$f'_b$，$(|f_b|,|f'_b|)$称为布尔函数$f(\boldsymbol{X})$的$k$阶 $(k=1,2,\cdots,n)$DC 特征[7]。

本文将延续使用 DC 特征向量[7]，利用 NP 变换时变量变换前后对称性与独立性不变的属性，独立变量的属性 3)、4)、5) 和 6)，加快正规式的计算，从而提高布尔函数 NPN 等价匹配速度。

2.3 加快匹配的关键方法

本文将布尔函数的变量分为 3 类：独立变量、对称变量和非对称变量。在计算布尔函数正规式的过程中，根据变量的类型和 DC 特征值进行分组。具有相同 DC 特征值的变量为一组，每组根据变量类型分为独立变量类、对称变量类和非对称变量类。当所有的组都是“已解决”状态，产生一个候选 NP 变换。上述 3 类变量所在组处于“已解决”状态需满足的条件为：

1) 非对称变量，变量相位确定且所在组就一个非对称变量；

2) 对称变量，变量相位确定且所在组只有一个对称类，无其他对称类和非对称变量；

3) 独立变量，因独立变量的布尔差分特征值为 0，即使有其他对称变量与其具有相同的通用特征值，但布尔差分特征值一定不同，直接标记“已

解决”。

加快匹配的关键方法为：

1) 给定两个布尔函数$f(\boldsymbol{X})$和$g(\boldsymbol{X})$，首先计算他们的 0 阶特征、1 阶 DC 特征，判断所有变量的对称性和独立性，并根据以上结果对两个布尔函数的变量进行分组。

2) 根据上面的计算结果，比较两个布尔函数的变量是否具有相同的结构，即相同的 0 阶特征、1 阶 DC 特征、对称变量结构和独立变量结构。如果$f(\boldsymbol{X})$和$g(\boldsymbol{X})/\overline{g(\boldsymbol{X})}$具有相同结构再分别计算它们的正规式$F(\boldsymbol{X})$和$G(\boldsymbol{X})$，若等式$F(\boldsymbol{X})=G(\boldsymbol{X})$成立，则 NPN 等价，否则不等价。

3) 分别利用本文所提出的独立变量所具有的属性 3)、属性 4) 和属性 5)，能更好地减少计算正规式过程中的搜索空间，具体步骤为：

① 在计算特征值的过程中独立变量的相位始终无法确定。若布尔函数具有独立变量，必有一个对称类且只有独立变量。在计算候选正规式的过程中一定会产生两个分支[1,7]，分别尝试同相对称和反相对称，所需探测的正规 NP 变换数增加 1 倍。本文直接将独立变量标记为正相且“已解决”。

② 在文献 [1,7] 的算法中使用“已解决”的变量来计算“未解决”变量的高阶特征值，以期“未解决”变量能够获得不同的高阶特征值而变为“已解决”。

假设有已解决非独立变量x_j和独立变量组$\{x_{i_1},x_{i_2},\cdots,x_{i_k}\}$，因为独立变量组的相位不定，因此需要通过变量$x_j$计算$|f_{x_jx_{i_1}}|$和$|f_{x_j\bar{x}_{i_1}}|$来确定独立变量的相位。根据属性 5) 可知，无论计算多少次都无法确定独立变量相位，增加了不必要的计算，本算法直接将独立变量相位设置为正向，解决了该问题。

③ 同样，参考文献 [1,7] 的算法，利用独立变量去解决其他“未解决”的变量，以期用独立变量对“未解决”变量计算高阶特征值，从而使这些“未解决”的变量得以“解决”。

假设有独立变量组$\{x_{i_1},x_{i_2},\cdots,x_{i_k}\}$和未解决的一个变量组$\{x_{j_1},x_{j_2},\cdots,x_{j_m}\}$，且变量$x_{j_1},x_{j_2},\cdots,x_{j_m}$均为非对称变量。先根据独立变量组产生两个分支，一分支为同相对称，一个分支为反相对称[1,7]。即两个分支都分别利用独立变量x_{i_1}对变量$x_{j_1},x_{j_2},\cdots,x_{j_m}$计算其更高阶特征或 DC 特征。变量$x_{j_1},x_{j_2},\cdots,x_{j_m}$之所以没有“解决”是因为这些变量具有一样的 DC 特征值。以正相为例，文献 [7] 计算二阶特征值$(|f_{x_{i_1}x_{j_1}}|,|f'_{x_{i_1}x_{j_1}}|),(|f_{x_{i_1}x_{j_2}}|,|f'_{x_{i_1}x_{j_2}}|),\cdots,(|f_{x_{i_k}x_{j_k}}|,|f'_{x_{i_k}x_{j_k}}|)$，根据属性 4)，必然有$(|f_{x_{i_1}x_{j_1}}|,|f'_{x_{i_1}x_{j_1}}|)=(|f_{x_{i_1}x_{j_2}}|,|f'_{x_{i_1}x_{j_2}}|)=\cdots=(|f_{x_{i_k}x_{j_k}}|,|f'_{x_{i_k}x_{j_k}}|)$，增加了不必要的计算。根据属性 4)，本算法不使用独立变量计算后面组的高阶特征。

利用属性 3)、4) 或 5)，本文对于独立变量所在的组直接标记为“已解决”，针对具有独立变量的布尔函数 NPN 等价匹配，可降低空间复杂度 50%。

例 1：布尔函数$f=x_1x_3+\bar{x}_1\bar{x}_3+x_3x_4+\bar{x}_3\bar{x}_4$，计算一阶 DC 特征向量和对称变量检测，$\boldsymbol{V}_f=\{(10,10,0),(12,8,12),(10,10,0),(8,12,12),(8,12,12)\}$，对称检测得到两个对称类，$\{x_1,\bar{x}_3,\bar{x}_4\}$和$\{x_0,x_2\}$，其中$\{x_0,x_2\}$又是独立变量类。因此产生两组$G_1=\{x_1,\bar{x}_3,\bar{x}_4\}$，$G_2=\{x_0,x_2\}$，根据上述“已解决”的判断，这里$G_1$和$G_2$都“已解决”，获得排序$\{x_1,\bar{x}_3,\bar{x}_4,x_0,x_2\}$，直接得到正规式$F(\boldsymbol{X})=x_0\bar{x}_3+\bar{x}_0x_3+x_0x_2+\bar{x}_0\bar{x}_2$。

例 1 中如果不考虑本文方法，那么需要利用变量x_1计算x_0和x_2的二阶特征，其目的是获得这两个变量的相位，结果一定是相位不确定，因此产生两个排序$\{x_1,\bar{x}_3,\bar{x}_4,x_0,x_2\}$和$\{x_1,\bar{x}_3,\bar{x}_4,x_0,\bar{x}_2\}$。

2.4　匹配算法

本算法采用树形结构存储计算正规式过程中产生的候选正规式，利用树的深度优先搜索实现。当第 m 个组之前的所有组已经解决且都无法解决后续尚未“解决”的组；同时，第 m 个组需要分为 p 种情况，这时都将产生 p 个分支。本文通过上述的关键方法减少分支数，以提高匹配速度。

函数 Initial() 用来计算布尔函数 f 和 g 的 0 阶特征、1 阶 DC 特征值、对称性、独立性、相位信息和初始分组信息，伪代码如下。

```
Procedure 1 Initial Boolean Variable
Function Initial(f)
Input: f
Output: V_f
  Compute 0st signature |f| of f
  Compute 1st DC signature V_f of f
  Determine the phase of (x_0, x_1, ..., x_{n-1}) of f
  Check the independent variables of f
  Check the symmetric variables of f
  Update V_f
  k=group(V_f) //分组
  Return V_f
End function
```

函数 Canonical() 用来计算布尔函数的最大正规变换，伪代码如下。

```
Procedure 2 Compute the Maximum Transformation
Function Canonical(f, V_f, k)
Input: f, V_f, k
Output: M_f
If (E_1) //如果获得一个候选变换
  If (Empty(M_f))
    M_f=T
  Else
    If (V^{M_f} < V^T)
      M_f=T
    End if
  End if
Else
  For all G_i, i ∈ {1,2,⋯,k} do
    If (E_2) //如果所有组都“已解决”
      break
    End if
  End for
  If (i==k+1)
    return M_f
  Else
    If (E_3) //如果G_i“已解决”
      Add the variable of G_i to CT_List
      Update V_f
      k=k+1
      Canonial(f, V_f, k)
    Else
      For all G_i, i ∈ {1,2,⋯,m} do
        Split G_i
        Create node_list
      End For
      For all node_j in node_list do
        Update (node_j, group)
        k=k+1
        If (E_1)then
          If (Empty(M_f))
            M_f=T
          Else
            If (V^{M_f} < V^T)
             M_f=T
            End if
          End if
        Else
          Add the variables in G_i to CT_list
          Update_signature(f)
          m=group(V_f)
          Canonical(f, CT_List, group, k)
        End if
      End for
    End if
  End if
End if
return M_f
End function
```

给定两个 n 变量布尔函数 f 和 g，算法先分别调用 Initial() 计算它们的初始特征向量和变量结构，如果变量结构相同再分别调用 Canonical() 函数计算它们两个的正规变换，最后计算正规式 F 和 G，如果 F=G 成立，那么布尔函数 f 和 g 是 NPN 等价的。其中，当出现 $|f| = |g| = 2^{n-1}$ 时，处理方法同文献 [1]。

3 实验结果

基于 MCNC 标准电路库中电路和随机生成电路的布尔函数，对本文提出的算法和文献 [1] 中的基于高阶通用特征的算法进行了测试、比较和验证。测试环境配置为 3.3 GHz CPU、8 GB RAM。

表 1 MCNC 标准库电路 NPN 等价匹配结果

n	AVG/s	A.T/个	AVG[1]/s	A.T[1]/个
7	0.000 24	1.7	0.002 15	66.7
8	0.005 91	20.0	0.009 89	27.1
9	0.001 08	2.1	0.001 63	3.0
10	0.000 67	1.5	0.000 98	2.1
11	0.001 38	1.7	0.002 83	3.2
12	0.001 12	1.6	0.001 77	2.7
13	0.019 32	5.0	0.042 42	7.8
14	0.005 07	1.5	0.007 13	2.1
15	0.018 34	1.7	0.027 30	2.4
16	0.016 01	1.8	0.023 80	2.9
17	0.125 48	1.9	0.188 72	2.8
18	0.141 11	1.8	0.217 07	4.4
19	0.681 86	1.5	1.139 60	3.4
20	1.345 87	2.6	1.901 59	3.8
21	4.345 12	1.3	7.124 78	2.0
22	9.308 26	2.1	13.954 70	4.5

表 1 为对 MCNC 标准电路库中电路的布尔函数 NPN 等价匹配的实验结果，表 2 为随机生成电路的布尔函数 NPN 等价匹配的结果。两表中的第 1 列是变量个数，第 2～3 列和 4～5 列是本文和文献 [1] 算法的匹配平均时间 (AVG) 和平均候选正规

变换数 (A.T)。

根据表 1 中的实验结果可以看出，本文算法对 MCNC 标准电路库中电路的布尔函数的匹配速度比文献 [1] 提升了 40.1%，搜索空间减少了 42.1%。根据表 2 可以看出本文算法对随机电路的布尔函数匹配速度比文献 [1] 提升了 51%，搜索空间减少 75.4%。可以 NPN 等价匹配对随机电路的布尔函数匹配速度提升更高，其原因是随机产生电路的布尔函数中具有独立变量的情况更多一些。

通过实验可以看出，本文算法能够大大减少计算正规式过程中的搜索空间，并大幅提高了布尔函数匹配速度。

表 2　随机生成电路 NPN 等价匹配结果

n	AVG/s	A.T/个	AVG[1]/s	A.T[1]/个
7	0.000 20	5.4	0.000 7	13.2
8	0.000 31	4.2	0.000 9	22.1
9	0.000 55	4.7	0.001 0	16.8
10	0.000 33	2.8	0.001 3	19.6
11	0.001 64	3.2	0.003 2	19.1
12	0.003 72	3.1	0.006 8	12.5
13	0.003 42	2.9	0.009 9	17.2
14	0.004 96	3.0	0.009 7	14.8
15	0.003 52	3.1	0.007 8	16.5
16	0.010 53	3.8	0.024 9	13.9
17	0.016 46	4.5	0.032 6	23.4
18	0.021 17	3.8	0.042 3	13.3
19	0.084 73	4.2	0.156 5	15.8
20	0.076 91	4.4	0.119 5	12.5
21	0.055 98	3.7	0.085 9	13.3
22	0.094 80	5.4	0.134 9	18.3

4　结 束 语

本文通过对布尔函数香农分解代数余子式运算的研究，得出 6 个有利于布尔匹配的属性。利用这些属性提出了更有效的基于正规式的 NPN 布尔匹配算法，有效减少了算法的搜索空间和提高了布尔函数 NPN 等价匹配的速度。该算法能够更好地应用到电路设计和优化中去，具有一定的应用价值。如何解决布尔函数在最坏情况下的 NPN 等价匹配难题，是下一步的研究难点。

参　考　文　献

[1] ABDOLLAHI A, PEDRAM M. Symmetry detection and Boolean matching utilizing a signature-based canonical form of Boolean functions[J]. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2008, 27(6): 1128-1137.

[2] CHEN K C, YANG C Y. Boolean matching algorithms[J]. International Symposium on VLSI Technology Systems and Applications Proceedings, 1993(1): 44-48.

[3] LUO Q B, LI X Y, YANG G W. Quantum circuit implementation of S-box for SM4 cryptographic algorithm[J]. Journal of University of Electronic Science and Technology of China, 2021, 50(6): 820-826.

[4] ZHANG Y, YANG G W, HUNG W, et al. Computing affine equivalence classes of Boolean functions by group isomorphism[J]. IEEE Transactions on Computers, 2016, 65(12): 3606-3616.

[5] JOHNSTON F E. The theory of group representations[M]. Maryland, Baltimore: The Johns Hopkins University Press, 1938.

[6] ZHANG J L, YANG G W, HUNG W, et al. A group algebraic approach to NPN classification of Boolean functions[J]. Theory of Computing Systems, 2019, 63: 1278-1297.

[7] ZHANG J L, YANG G W, HUNG W, et al. A canonical-based NPN Boolean matching algorithm utilizing Boolean difference and cofactor signature[J]. IEEE Access, 2017, 5: 27777-27785.

[8] ABDOLLAHI A. Signature based Boolean matching in the presence of don't cares[C]//Design Automation Conference Proceedings. Anaheim: [s. n.], 2008: 642-647.

[9] 张菊玲, 杨国武, 吴尽昭, 等. 基于对称及特征的 NPN 布尔匹配算法[J]. 电子科技大学学报, 2018, 47(6): 876-881.
ZHANG J L, YANG G W, WU J Z, et al. NPN Boolean matching algorithm based on symmetry and signature[J]. Journal of University of Electronic Science and Technology of China, 2018, 47(6): 876-881.

[10] ZHANG J L, YANG G W, HUNG W, et al. A new pairwise NPN Boolean matching algorithm based on structural difference signature[J]. Symmetry, 2019, 11(1): 27.

[11] SAFARPOUR S, VENERIS A, BAECKLER G, et al. Efficient SAT-based Boolean matching for FPGA technology mapping[C]//Design Automation Conference Proceedings. San Francisco: [s. n.], 2006: 466-471.

[12] WANG X Q, YANG Y. New approach of exploiting symmetry in SAT-based Boolean matching for FPGA technology mapping[C]//IEEE International Conference on Vehicular Electronics and Safety Proceedings. Dongguan: IEEE, 2013: 282-285.

[13] MUZIO J, MILLER D M, HURST S L. Number of spectral coefficients necessary to identify a class of Boolean functions[J]. Electronics Letters, 2007, 18(13): 577-578.

[14] THORNTON M A, DRECHSLER R, GUNTHER W. Logic circuit equivalence checking using Haar spectral coefficients and partial BDDs[J]. VLSI Design, 2014, 14(1): 53-64.

[15] LAI C F, JIANG J, WANG K H, et al. Boolean matching of function vectors with strengthened learning[C]// IEEE/ACM International Conference on Computer-Aided Design Proceedings. San Jose: IEEE, 2010: 596-601.

[16] ZHANG J S, CHRZANOWSKAJESKE M, MISHCHENKO A, et al. Linear cofactor relationships in Boolean functions[J]. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2006, 25(6): 1011-1023.

编　辑　叶　芳