类选择排序的可逆逻辑综合算法

万四爽 陈汉武 曹如进
(东南大学计算机科学与工程学院 南京 210096)

摘 要 可逆逻辑综合是指对给定的可逆函数自动构造对应的可逆逻辑电路.由于搜索空间随电路规模增长成指数增长,现有的可逆逻辑综合算法虽然能够得到近似最优的解,但是都存在计算时间过长的问题.文中提出了一种类似选择排序的可逆逻辑综合算法,其实质为基于变换规则的合成法.它采用一个无向无权图表示所有可以进行变换的路径,在综合的过程中,采用选择排序思想每次从小到大的选择需要交换的输出项,然后从路径选择图中找到最优的路径进行变换,最终使得函数的输出序列有序即完成综合.此外,文中还对得到的量子电路进行了优化.实验表明,相比其它综合算法,该算法不仅总能获得最优解或近似最优解,而且效率高、易于实现.

关键词 量子计算;可逆逻辑综合;Toffoli 门;选择排序;量子电路
中图法分类号 TP301 **DOI 号**: 10.3724/SP.J.1016.2010.02343

An Analogic Selection Sorting Algorithm for Synthesis of Reversible Logic Circuits

WAN Si-Shuang CHEN Han-Wu CAO Ru-Jin
(*School of Computer Science and Engineering, Southeast University, Nanjing* 210096)

Abstract Reversible logic studies have promising potential on energy lossless circuit design, quantum computation, nanotechnology, etc. Though existing synthesis methods can provide optimal solutions, yet they may suffer from long computation time, due to the fact that the search space is likely to grow exponentially as the circuit size increases. Therefore, in this paper, the authors propose an analogic selection sorting algorithm essentially based on the transformation-based algorithm. An unweighted, undirected graph is used for the representation of all transformable paths. During the synthesis process, a sequence of transformation is built to make all the output patterns appeared in the right place. The whole process can be implemented by a sequence of Toffoli gates. In addition, the authors propose a simplification algorithm to further optimize the generated circuit. The experimental results show that the algorithm, compared with other exact methods, can achieve optimal or very close to optimal solutions with less computation time. Furthermore, the algorithm is more easily understood and implemented.

Keywords quantum computing; reversible logic synthesis; Toffoli gate; selection sorting; quantum circuit

1 引 言

量子计算机可等效为一个量子图灵机,而理论上已证明,量子图灵机又可等价一个量子逻辑电路[1],因此可以通过量子逻辑门的级联与组合构成量子计算机.可逆逻辑综合源于可逆计算机的研究,现已广泛应用在量子计算、低功耗电路、纳米技术、

收稿日期:2009-04-08;最终修改稿收到日期:2010-10-18.本课题得到国家自然科学基金(60873101)、江苏省自然科学基金(BK2007104、BK2008209)资助.**万四爽**,男,1986 年生,硕士研究生,主要研究方向为量子计算、可逆电路综合. E-mail: wansishuang@seu.edu.cn. **陈汉武**,男,1955 年生,博士,教授,博士生导师,主要研究领域为量子计算、信息论. E-mail: hw_chen@seu.edu.cn. **曹如进**,男,1986 年生,硕士研究生,主要研究方向为语义网、算法分析与设计.

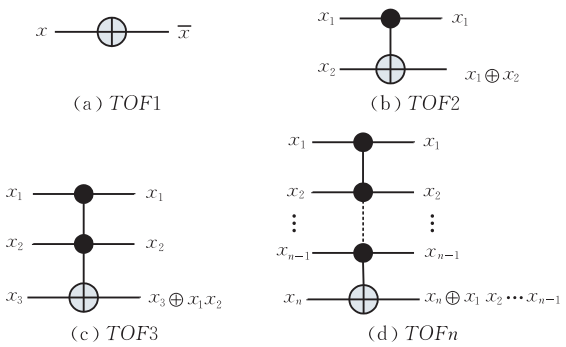
光计算和信息加密等领域中，因此可逆逻辑的研究已变得越来越重要.

可逆逻辑综合是指对给定的可逆函数自动构造对应的可逆逻辑电路. 与经典电路不同，可逆逻辑电路不能有回路，不能出现扇入、扇出操作，正是这些不同决定了经典的电路综合方法不能适用于量子可逆逻辑电路综合. 目前量子可逆逻辑电路大部分采用 Toffoli 门根据函数的真值表进行构造，主要可以分为两大类. 一类像采用布尔可满足性(Boolean Satisfiability，SAT)方法[2]、定量的布尔公式(Quantified Boolean Formula，QBF)方法[3]、动态规划(Dynamic Programming，DP)方法[4]和符号的可达性(Symbolic Reachability)[5]等近似穷举的搜索方法，虽然能够产生最优解，但是由于搜索的范围过大，往往需要很长的计算时间，而且随着电路规模的增大，搜索空间会成指数增长，导致运行时间无法忍受. 另一类采用启发式的搜索方法，如 Rademacher-Walsh 谱均数方法[6]、二进制共享决策图方法[7]、基于模板的变换法(Template-Based Transformation)[8]和基于 PPRM(Positive-Polarity Read-Muller)表达式的变换法[9]. 这类方法是穷举法的改进，在搜索的过程中增加了启发式规则，减少了搜索的范围，缩短了运行时间，但是这类方法合成的电路往往代价不是最小，而且还不能保证合成算法的收敛性. 此外，还可以将真值表翻译成酉矩阵，通过选取特定的通用门库[10]进行分解，其一般的解析表达式最近也已给出[11]. 此外，最近还有新的研究方向，对偶量子计算和对偶量子计算机[12]，其变换不再满足酉性，而且可以是不可逆的[12-13]，不过上述经典的可逆逻辑综合算法依然可能适用.

本文在真值表的基础上采用一种新的基于变换规则的综合算法，提出了路径选择图的概念，其实质是用一个无向无权图表示变换规则，通过一系列的变换，把输出系列变成输入序列. 在综合的过程中，采用选择排序的思想，按照递增顺序每次选择相应的输出项，从路径选择图中找到适合的路径进行交换，直到所有输出项有序即把输出变成了输入，此时综合完成. 这样得到的变化序列对应的量子线路的逆序即为综合的结果. 为了减少合成线路的代价，本文还提出了一系列的优化规则化简得到的量子线路.

2 一些基本概念

定义 1. Toffoli 门. 一个 n 比特的 Toffoli 门表示为 $TOFn(x_1,x_2,\cdots,x_n)$，保持前 $n-1$ 个输入不变，只有当前 $n-1$ 个输入比特同时为 1 时才改变第 n 个输入比特. $TOF1(x_1)$ 为非门(NOT)；$TOF2(x_1,x_2)$ 为控制非门(CNOT)；$TOF3(x_1,x_2,x_3)$ 为标准 Toffoli 门，如图 1 所示.



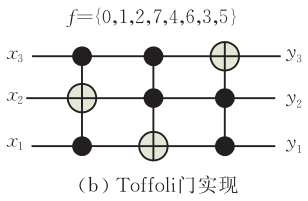
(a) $TOF1$ (b) $TOF2$ (c) $TOF3$ (d) $TOFn$

图 1 Toffoli 门

定义 2. n 个输入与 n 个输出变量的布尔函数 $f:(x_1,x_2,\cdots,x_n)\rightarrow\{y_1,y_2,\cdots,y_n\}$ 是可逆函数当且仅当 f 是双射的. n 变量的可逆函数可以用真值表描述，也可以用整数集合 $\{0,1,\cdots,2^{n-1}-1\}$ 的置换表示. 例如函数 $f=\{0,1,2,7,4,6,3,5\}$ 表示的 3 变量可逆函数真值表和对应的电路如图 2 所示.

x_3	x_2	x_1	y_3	y_2	y_1
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	1	0
0	1	1	1	1	1
1	0	0	1	0	0
1	0	1	1	1	0
1	1	0	0	1	1
1	1	1	1	0	1

(a) 真值表



(b) Toffoli门实现

图 2 函数 $f=\{0,1,2,7,4,6,3,5\}$

定义 3. 等长字符串 p 和 q 之间的 Hamming 距离 $H(p,q)$ 是 p 和 q 对应位置的不同字符的个数.

特别地，可逆函数 f 的任一输出项与对应输入项的 Hamming 距离简称为输出项的 Hamming 距离.

定义 4. 函数 f 的复杂度 $C(f)$ 是其所有输出项的 Hamming 距离之和. 例如图 2 中函数 f 的复杂度为 $C(f)=0+0+0+1+0+2+2+1=6$.

3 初步知识

在这个章节我们给出类选择排序可逆逻辑综合算法的一些定义和定理.

定义 5. 对 n 比特的可逆函数 f，如果存在两个只有 1 位不同的输出项 A、B，则可将 A、B 交换而得到一个新的可逆函数 f'，这个交换操作我们定义

为 SWAP[A,B]. 每个 SWAP 操作都可以用一个 Toffoli 门实现.

例 1.　考虑图 2 函数 $f=\{0,1,2,7,4,6,3,5\}$,其中输出项[000]与[001]只有一位不同,则可以做 SWAP[000,001]操作,从而得到一个新的函数 $f'=\{1,0,2,7,4,6,3,5\}$,对应的 Toffoli 门为 $TOF3(\bar{x}_3,\bar{x}_2,x_1)$,如图 3 所示.

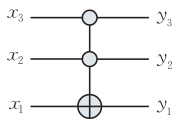
x_3	x_2	x_1	y_3	y_2	y_1
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	1	0
0	1	1	1	1	1
1	0	0	1	0	0
1	0	1	1	1	0
1	1	0	0	1	1
1	1	1	1	0	1

(a) f的真值表

SWAP[0,1]

x_3	x_2	x_1	y_3	y_2	y_1
0	0	0	0	0	1
0	0	1	0	0	0
0	1	0	0	1	0
0	1	1	1	1	1
1	0	0	1	0	0
1	0	1	1	1	0
1	1	0	0	1	1
1	1	1	1	0	1

(b) f'的真值表



(c) SWAP[0,1]对应的Toffoli门

图　3

定理 1.　每个 SWAP 操作改变输出项的 Hamming 距离为 1.

证明.　SWAP 操作只改变了输出项的一位,故 Hamming 距离改变为 1. 例如,将输入 000 的输出 000 改为 001,其 Hamming 距离减少 1.

定义 6.　n 比特可逆函数的路径选择图 $G=(V,E)$是一个无向无权图,点集 $V=\{0,1,\cdots,2^n-1\}$表示 2^n个输入或者输出,边集 $E=\{e_{ij}|\exists SWAP(i,j),i,j\in V\}$,即每条边表示连接的 2 个顶点能够进行 SWAP 操作. G 中两顶点 A、B 之间存在一条路径,则表明 A 经过一系列的 SWAP 操作能够到达 B,路径长度为此路径经过的边数.

例 2.　对于 3 比特的路径选择图如图 4 所示,其中点$(0)_2=000$ 与$(1)_2=001$ 之间有一条边,表示 0,1 之间可进行 SWAP[000,001]. 如果需要把 000 和 111 交换,可以从图中找出其中一条 0～7 的路径 000—>001—>101—>111,则经过 SWAP[000,001],SWAP[001,101],SWAP[101,111]可以实现 0 和 7 的交换.

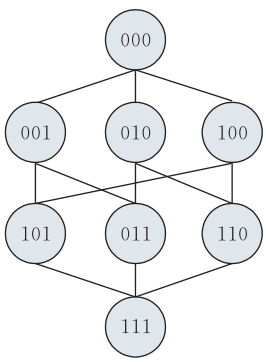


图 4　3 比特可逆函数的路径选择图

定理 2.　可逆函数输出项 A 和 B 的 Hamming 距离为 k,则在路径选择图中 A 到 B 必然存在一条长度至少为 k 的路径,即交换 A 和 B 要进行 k 步 SWAP 操作.

证明.　路径选择图是连通的,故 A 到 B 必然存在一条路径. 由定理 1,SWAP 操作每次仅改变 A 或者 B 的 1 位,而 A 和 B 的 Hamming 距离为 k,故交换 A、B 至少需要 k 次 SWAP 操作,即 A 到 B 肯定存在一条长度至少为 k 的路径.

根据定理 2,很容易得出如下引理.

引理 1.　对于 n 比特可逆函数,假设需要交换的两个输出项分别为 A 和 B,则 A 与 B 的 Hamming 距离 $1\leq H(A,B)\leq n$.

(1)当 $H(A,B)=1$,则 A、B 之间有一条直接路径(长度为 1). A 和 B 可以直接交换,即可以直接进行 SWAP[A,B]操作.

(2)当 $H(A,B)=2$,则 A、B 间有长度为 2 的路径,即该路径上有一个中间结点 C,则 A、B 交换可以通过 C 实现 SWAP[A,C],SWAP[C,B],而中间结点 C 有两种选择,故路径条数为 2.

(3)当 $H(A,B)=3$,则 A、B 间有长度为 3 的路径,即该路径上有两个中间结点 A_1,B_1. 则 A,B 的交换可以通过 A_1,B_1 实现,SWAP[A,A_1],SWAP[A_1,B_1], SWAP[B_1,B]. 因为 A_1 有 3 种选择,对特定的 A_1 而 B_1 有 2 种选择,故路径条数为 $3\times2=6$.

(4)当 $H(A,B)=k$,则 A 到 B 有长度为 k 的路径,且路径条数最多有 $k!$条.

定理 3.　函数 f 与经过 SWAP 操作后得到的新函数 f'相比复杂度或加 2 或减 2 或不变.

证明.　假设 f 的两个输出项为 A、B 可以进行 SWAP 操作,在 SWAP 之前 A、B 的 Hamming 距离分别记为 H_A、H_B. 不失一般性,假设 A、B 最后一位不同,则 A、B 可表示为

$$A=[x_1,x_2,\cdots,x_{n-1},x_n],$$
$$B=[x_1,x_2,\cdots,x_{n-1},\bar{x}_n],$$

SWAP 后,

$$A=[x_1,x_2,\cdots,x_{n-1},\bar{x}_n],$$
$$B=[x_1,x_2,\cdots,x_{n-1},x_n].$$

可以看到 H_A,H_B 的值或 $+1$,或 -1,而 H_A,H_B 取值互不影响,故当 H_A,H_B 同时取 1 时函数 f 的复杂度 $C(f)$增加 2;当同时取 -1 时 $C(f)$减 2;

当取不同值时时复杂度 $C(f)$不变.

引理 2. 假设 f 两个输出项 A 和 B 的 Hamming 距离为 k,交换 A、B 得到的新函数为 f',则用最少次数的 SWAP 操作后 f 的复杂度至多减少 $2k$,即 $\max\{C(f)-C(f')\}=2\times k$.

证明. 根据定理 2 可知,f 到 f'最少需要经过 k 步 SWAP 操作.而由定理 3 每次 SWAP 操作,最多能减少复杂度值为 2,故交换 A、B 后,f 复杂度至多减少 $2k$.

从引理 2 很容易得到如下推论.

推论 1. 对于 3 比特可逆函数,假设需要交换的 2 个输出项分别为 A、B,则 A 和 B 的 Hamming 距离为 $1\leq H(A,B)\leq 3$:

(1)当 $H(A,B)=1$,交换 A、B 最多能使函数复杂度减少 2.

(2)当 $H(A,B)=2$,交换 A、B 最多能使函数复杂度减少 4.

(3)当 $H(A,B)=3$,交换 A、B 最多能使函数复杂度减少 6.

4 算法描述

算法的目的是通过一系列的 SWAP 操作使得输出序列有序,而每一次 SWAP 就对应一个 Toffoli 门.当输出序列有序即输出经过一系列的 Toffoli 门作用变成了输入时,把变换对应的 Toffoli 门按逆序连接组成的线路即为综合结果.其过程类似选择排序:按照输入从 $0\sim 2^n-1$ 的顺序,通过路径选择图中的合适路径,每次把选择最小的输出项将其交换到正确的位置.其中路径的选择要使可逆函数 f 的复杂度逐渐减少直到为 0,当 f 的复杂度为 0 即输出项有序时算法终止.

算法由两部分组成:路径的选择和电路的生成.下面详细描述了这两步算法,然后给出了一个完整的例子来说明算法是如何进行的.

4.1 路径的选择

假设需交换输出项 A 和 B,其 Hamming 距离为 k.从路径选择图可以看到:(1)A、B 可能存在多条长度不同的路径,而根据定理 2 只需从路径选择图中找长度为 k 的 A 到 B 的路径;(2)A、B 可能存在多条长度为 k 的路径,根据定理 3 如果找到一条能使函数复杂度减少 $2k$ 的路径则算法终止,否则选择一条能使复杂度减少最多的路径.同时,因为已遍历过的输出项已经有序,故可将其对应路径选择图中的结点删除,从而减少路径选择的复杂度.由此可见,路径选择的过程是个终止条件很容易满足的深度优先搜索的过程.

算法 1. 从路径选择图中寻找交换 A,B 的路径.

输入:路径选择图 G,可逆函数 f,待交换的输出项 A、B

输出:交换 A、B 后能最大地减少 f 复杂度的 A 到 B 的路径 L

FIND-SWAP-PATH (G,A,B)

1. 计算 f 的复杂度 $C(f)$;
2. $h=H(A,B)$,$H(A,B)$为 A,B 的 Hamming 距离;
3. if $h=0$ return 0;
4. if $h=1$ return 路径 L: $A\rightarrow B$;
5. if $h>1$
6. 采用 DFS 在 G 中寻找 A 到 B 的路径,遍历层数为 h,如果找到一条路径 L,计算 f 按路径 L 交换后得到的 f'的复杂度 $C(f')$;
7. if $C(f)-C(f')=2*k$,停止遍历,return L: $A\rightarrow\cdots\rightarrow B$;
8. else
   遍历所有路径,找到 $\Delta e=C(f)-C(f')$最大的路径 L(如果 Δe 相等的话,任取一条路径 L);
9. return L: $A\rightarrow\cdots\rightarrow B$;
10. End.

算法 2. 从路径选择图中删除顶点 X.

输入:路径选择图 G,待删除顶点 X

输出:新的路径选择图

DELETE-VERTEX (X)

1. 在图 G 中找到顶点 X;
2. 删除所有连接到顶点 X 的边;
3. 删除顶点 X;
4. return G //G 为删除 X 后的新图;

4.2 线路生成

类似选择排序,按照输入从 $0\sim 2^n-1$ 的顺序,每次选择当前最小输出项,通过算法 1 得到的路径把其交换到正确的位置,直到整个输出序列有序.而每一次交换对应一系列 Toffoli 门,将所有 Toffoli 门逆序连接组成的线路即为综合结果.

算法 3. 根据可逆函数 f 生成 Toffoli 门表示的量子线路 Circuit.

输入:可逆函数 f:$(0,1,2,\cdots,2^n-1)\rightarrow\{y_1,y_2,\cdots,y_n\}$

输出:Toffoli 门表示的量子线路

BUILD-TOFFOLI-NETWORK (f)

1. for $i=0$ to 2^n-2
2. if $i=y[i]$ do nothing;
3. else
   L=FIND-SWAP-PATH (G,$y[i]$,i);
4. 按照路径 L 通过一系列 SWAP 操作把 i 交换至

$y[i]$的位置,得到 f';

5. $f=f'$ //用 f'更新 f;

6. End else

7. DELETE-VERTEX(i);

8. End for

9. 按照 SWAP 操作的逆序将每个 SWAP 对应的 Toffoli 门连接起来,得到量子电路 Circuit;

10. return Circuit; $\{t_1,t_2,\cdots,t_m\}$.

4.3 算法正确性证明

接下来我们给出算法的正确性证明.

定理 4. 路径选择图按顺序删除顶点后仍然是连通的.

证明. 假设当前删除的顶点为 $V(V\neq 2^n-1)$,因为是按顺序删除,则 V 中必有一位为 0,不失一般性假设 $V=(x_0,x_1,\cdots,0,\cdots,x_n)$,则必存在 $V'>V$,其中 V'相比 V 在刚刚假设为 0 的位上是 1,即 $V'=(x_0,x_1,\cdots,1,\cdots,x_n)$,故在路径选择图中存在一条直接路径连接 V'和 V. 则此命题得证. 证毕.

引理 3. 我们提出的类排序可逆逻辑综合算法肯定能得到可行解.

证明. 由算法 3 可以看到随着输入从 0 到 2^n-2,不断的通过交换使得输出序列变得有序. 当前面 2^n-1 个输出项有序的时,第 2^n 项即输入值为 2^n-1 时,对应的输出肯定为 2^n-1. 而由定理 4 可以得到从 $0\sim 2^n-2$ 的每步中路径选择图都是连通的,即存在交换的路径使得每步的输出与输入相同,当前面 2^n-1 个输出变得有序时,第 2^n 项必然也是有序的. 故类选择排序综合算法能得到可行解.

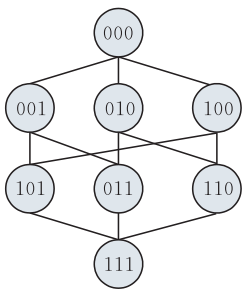
例 3. 以可逆函数 $f=[7,0,1,2,3,4,5,6]$为例,其真值表如图 5 所示,来说明类选择排序算法的流程.

$f=[7,0,1,2,3,4,5,6]$

x_3	x_2	x_1	y_3	y_2	y_1	H
0	0	0	1	1	1	3
0	0	1	0	0	0	1
0	1	0	0	0	1	2
0	1	1	0	1	0	1
1	0	0	0	1	1	3
1	0	1	1	0	0	1
1	1	0	1	0	1	2
1	1	1	1	1	0	1

$C=3+1+2+1+3+1+2+1=14$

(a) 带Hamming距离的真值表



(b) 路径选择图

图 5 $f=[7,0,1,2,3,4,5,6]$

1. 类似选择排序,从 000 开始,$y[000]=111\neq 000$ 即 000 对应的输出 111 不在正确位置上,故需交换[111,000],此时 $H(000,111)=3$,$C(f)=14$. 根据算法 1,通过遍历得到第 1 条路径(1) 111-101-001-000,按这条路径交换后得到函数的真值表如图 6(a)所示,复杂度减少 $\Delta C=14-10=4$,根据推论 1,距离为 3,复杂度可最多减少 6,即还可能存在比 4 更好的结果,故继续寻找新的路径;接着得到第 2 条路径(2) 111-101-100-000,按这条路径交换后得到函数的真值表如图 6(b)所示,函数复杂度减少了 4,故仍需继续寻找新的路径;接着得到第 3 条路径(3) 111-011-001-000,按这条路径交换后结果如图 6(c)所示,此时 $\Delta C=14-8=6$,函数复杂度减少 6,根据推论 1 可知这是最好的结果,故不需要继续寻找新的路径,按 111—>011—>001—>000 的顺序交换即 SWAP[111,011],SWAP[011,001],SWAP[001,000]可交换[111,000],得到新的函数 $f_1=[0,1,3,2,7,4,5,6]$. 然后删除路径选择图中的 000 结点,得到新的路径选择图如图 7(b)所示.

$f_1=[0,1,5,2,3,4,7,6]$

x_3	x_2	x_1	y_3	y_2	y_1	H
0	0	0	0	0	0	0
0	0	1	0	0	1	0
0	1	0	1	0	1	3
0	1	1	0	1	0	1
1	0	0	0	1	1	3
1	0	1	1	0	0	1
1	1	0	1	1	1	1
1	1	1	1	1	0	1

$C=0+0+3+1+3+1+1+1=10$

(a)

$f_1'=[0,4,1,2,3,5,7,6]$

x_3	x_2	x_1	y_3	y_2	y_1	H
0	0	0	0	0	0	0
0	0	1	1	0	0	2
0	1	0	0	0	1	2
0	1	1	0	1	0	1
1	0	0	0	1	1	3
1	0	1	1	0	1	0
1	1	0	1	1	1	1
1	1	1	1	1	0	1

$C=0+2+2+1+3+0+1+1=10$

(b)

$f_1''=[0,1,3,2,7,4,5,6]$

x_3	x_2	x_1	y_3	y_2	y_1	H
0	0	0	0	0	0	0
0	0	1	0	0	1	0
0	1	0	0	1	1	1
0	1	1	0	1	0	1
1	0	0	1	1	1	2
1	0	1	1	0	0	1
1	1	0	1	0	1	2
1	1	1	1	1	0	1

$C=0+0+1+1+2+1+2+1=8$

(c)

图 6 不同路径交换 000 和 111 后对应真值表

2. 接着看 001 项,从图 6(c)可以看到经过步 1 后 000 对应的输出项已经处于正确的位置,而 001 输入正好对应 001 输出,故这步不需要作交换操作,只需删除路径选择图上 001 的结点,得到新的路径选择图如图 7(c)所示. 此时可逆函数仍然是 $f_1=[0,1,3,2,7,4,5,6]$.

3. 当输入为 010 时,从图 6(c)中看到输出 $y[010]=011\neq 010$,故需交换[111,010],此时 $H(010,111)=1$,$C(f)=8$. 根据引理 1,可以直接交换[010,011]即SWAP[010,011],得到新的函数 $f_2=[0,1,2,3,7,4,5,6]$,其真值表如图 8(a)所示. 然后删除路径选择图中的 010 结点,得到新的路径选择图如图 7(d)所示.

4. 当输入为 011 时,从图 8(a)可以看出输出 $y[011]=011=011$,故不需进行交换操作,只需删除路径选择图上 011 的结点,得到新的路径选择图如图 7(e)所示. 此时,可逆函数依然为 $f_2=[0,1,2,3,7,4,5,6]$.

5. 当输入为 100 时,从图 8(a)可以看出输出 $y[100]=111\neq 100$,需交换[111,100],此时 $H(100,111)=2$,$C(f)=6$. 根据算法 1 找到第 1 条路径(1)111-101-100,按照这条路径交换得到新函数的真值表如图 8(b)所示,函数复杂度减少$\Delta C=14-10=4$,根据推论 1,这条路径是最好选择,故不需要继续进行寻找新的路径,按 111—>101—>100 的顺序交换即进行 SWAP[111,101],SWAP[101,100]操作得到新函数 $f_3=[0,1,2,3,4,5,7,6]$. 然后删除路径选择图中的 100 结点,得到新的路径选择图如图 7(f)所示.

6. 当输入为 101 时,从图 8(b)可知输出 $y[101]=101=101$,故不需进行交换操作,只需删除路径选择图上 101 的结点,得到新的路径选择图如图 7(g)所示. 此时,$f_3=[0,1,2,$

3,4,5,7,6].

7. 当输入为 110 时,从图 8(b)可知输出 y[110]=111≠110,故需交换[111,110]. 此时 $H(110,111)=1$, $C(f)=2$. 根据引理 1,可以直接交换[010,011]即直接进行SWAP[110,111]操作,得到新的函数 f_4=[0,1,2,3,4,5,6,7],其真值表如图 8(c)所示. 然后删除路径选择图中的 110 结点,得到新的路径选择图如图 7(h)所示. 此时,原函数 f=[7,0,1,2,3,4,5,6]已经变成恒等函数 f_4=[0,1,2,3,4,5,6,7].

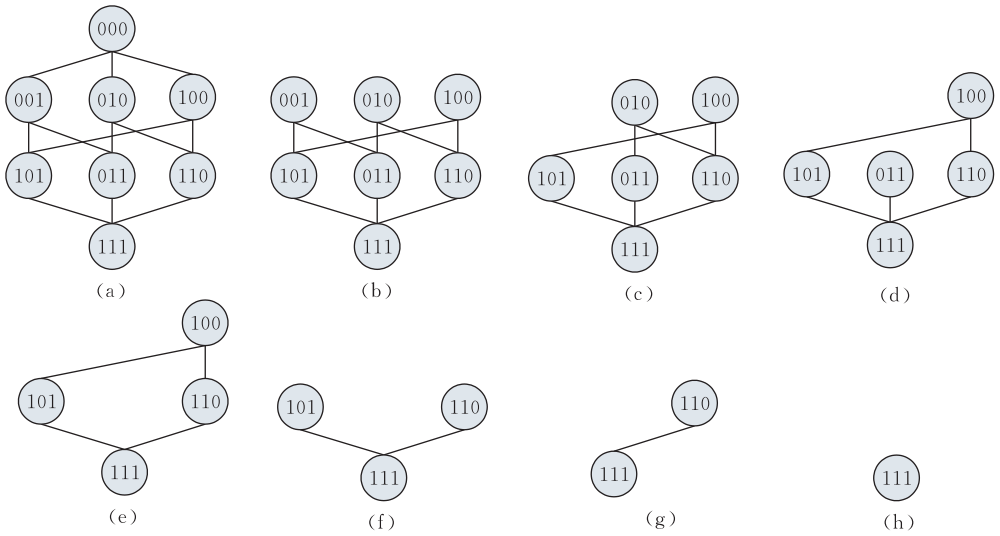


图 7 3 比特可逆函数路径选择图((a)～(h)对应步 1～8 分别删除顶点 000,001,010,011,100,101,110 后的路径图)

f_2=[0,1,2,3,7,4,5,6]

x_3	x_2	x_1	y_3	y_2	y_1	H
0	0	0	0	0	0	0
0	0	1	0	0	1	0
0	1	0	0	1	0	0
0	1	1	0	1	1	0
1	0	0	1	1	1	2
1	0	1	1	0	0	1
1	1	0	1	0	1	2
1	1	1	1	1	0	1

C=2+1+2+1=6

(a)

f_3=[0,1,2,3,4,5,7,6]

x_3	x_2	x_1	y_3	y_2	y_1	H
0	0	0	0	0	0	0
0	0	1	0	0	1	0
0	1	0	0	1	0	0
0	1	1	0	1	1	0
1	0	0	1	0	0	0
1	0	1	1	0	1	0
1	1	0	1	1	1	1
1	1	1	1	1	0	1

C=1+1=2

(b)

f_4=[0,1,2,3,4,5,6,7]

x_3	x_2	x_1	y_3	y_2	y_1	H
0	0	0	0	0	0	0
0	0	1	0	0	1	0
0	1	0	0	1	0	0
0	1	1	0	1	1	0
1	0	0	1	0	0	0
1	0	1	1	0	1	0
1	1	0	1	1	0	0
1	1	1	1	1	1	0

C=0

(c)

图 8

8. 根据步 1～7 得到的 SWAP 序列为 SWAP[111,011], SWAP[011,001], SWAP[001,000], SWAP[010,011], SWAP[111,101],SWAP[101,100],SWAP[110,111],对应的 Toffoli 门分别为 $TOF3(x_2,x_1,x_3)$, $TOF3(\bar{x}_3,x_1,x_2)$, $TOF3(\bar{x}_3,\bar{x}_2,x_1)$, $TOF3(\bar{x}_3,x_2,x_1)$, $TOF3(x_3,x_1,x_2)$, $TOF3(x_3,\bar{x}_2,x_1)$, $TOF3(x_3,x_2,x_1)$,按逆序连接起来对应的线路如图 9 即实现 f=[7,0,1,2,3,4,5,6].

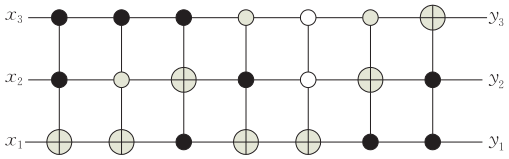


图 9 f=[7,0,1,2,3,4,5,6]对应的 Toffoli 门组成的电路图

4.4 算法复杂度分析

根据算法 2,可以看出外层循环次数为 2^n-2,在循环内部每次调用算法 1 FIND-SWAP-PATH 来找到交换路径,而其中路径个数最多为 $n!$,所以总的算法复杂度为 $2^n\times n!$. 但这是个非常宽泛的上界,因为我们在寻找路径的时候,搜索终止条件很容易满足. 此外,很明显看出随着算法的进行,可逆函数的输出趋向有序,路径图顶点数越来越少,每次选择的路径越来越短,路径条数越来越少,所以实际所需时间远小于 $2^n\times n!$. 这也体现在后面实验结果上.

5 优 化

此时算法得到的电路不是最优的,还可以继续优化,接下来给出我们的优化方法.

定理 5. 当 Toffoli 门 $A=TOFk(x_1,x_2,\cdots,x_{k-1},x_k)$ 与 $B=TOFl(y_1,y_2,\cdots,y_{l-1},y_l)$ 满足 (1) $x_k\notin\{y_1,y_2,\cdots,y_{l-1}\}$ 或者 (2) $\exists i\in[1,k-1]$, $j\in[1,l-1]$ 满足 $x_i=\bar{y}_j$,则 A 对 B 没有影响.

证明. 条件 1 说明了 B 的控制端前没有 A 的控制端,故无论 A 如果变化都不会影响 B. 条件 2 说明了 A 和 B 的控制端不可能同时满足条件,即输入

相同时,A 和 B 中只有一个能起作用,所以 A 可以对 B 不会有影响.

由定理 5 很容易得到性质 1.

性质 1. 交换规则. 若 Toffoli 门 A 与 B 互不影响,则可以交换 A、B 的位置.

例 4. 如图 10(a)中 B 受控端前没 A 的控制端,A 的受控端前没 B 的控制端,则 A、B 可交换. 图 10(b)中当 A 成立时 B 肯定不成立,同样 B 成立时 A 肯定不成立,则可将 A、B 交换.

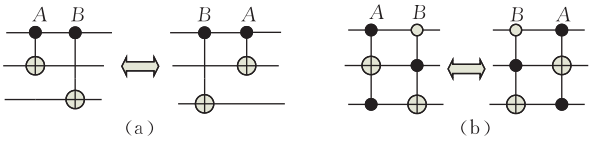


图 10　A、B 交换举例

定理 6. 合并规则. 相邻的 2 个 Toffoli 门,如果受控端相同而控制端只有一位不同,则可以把不同的控制端去掉,留下相同的控制端和受控端合并成 1 个 Toffoli 门.

证明. 不失一般性假设相邻的 2 个 Toffoli 门 A、B,前 $n-1$ 位为控制端,其中第 $n-1$ 位控制端不同,第 n 位为受控端,则 A 表示为 $TOFn(x_1,x_2,\cdots,x_{n-1},x_n)$,$B$ 为 $TOFn(x_1,x_2,\cdots,\bar{x}_{n-1},x_n)$,则经 A 和 B 作用后,受控端为

$$\begin{aligned}x_n''&=x_1x_2\cdots\bar{x}_{n-1}\oplus(x_1x_2\cdots x_{n-1}\oplus x_n)\\&=x_1x_2\cdots x_{n-2}(\bar{x}_{n-1}\oplus x_n)\oplus x_n\\&=x_1x_2\cdots x_{n-2}\oplus x_n\end{aligned}$$

而控制端不变,故得证. 证毕.

图 11 中给出了 Toffoli 门的一些合并例子.

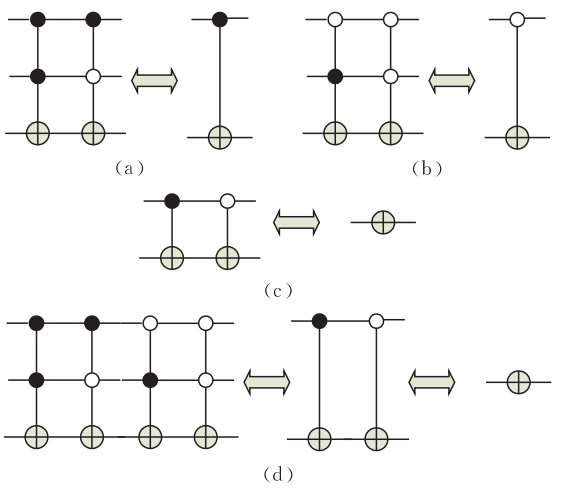


图 11　Toffoli 门的合并

定理 7. 消除规则. 如果存在连续的两个完全相同的 Toffoli 门,则可以将这 2 个 Toffoli 门去掉.

证明. 不失一般性,假设 2 个连续的 Toffoli 门为 A 和 B,A 和 B 相同,且前 $n-1$ 位为控制端,其中第 n 位为受控端,则 A 和 B 均可表示为 $TOFn(x_1,x_2,\cdots,x_{n-1},x_n)$,由于 A、B 完全相同,故控制端不变,受控端经过 A 和 B 后为

$$\begin{aligned}x_n''&=x_1x_2\cdots x_{n-1}\oplus(x_1x_2\cdots x_{n-1}\oplus x_n)\\&=(x_1x_2\cdots x_{n-1}\oplus x_1x_2\cdots x_{n-1})\oplus x_n=x_n.\end{aligned}$$

故经过 A、B 后受控端也没变化,则可将 A、B 从电路图中去掉.

根据定理 6 和定理 7 可得到如下优化算法.

算法 4. 对算法 3 中量子线路 $Circuit:\{t_1,t_2,\cdots,t_m\}$ 优化.

输入:算法 3 中量子线路 $Circuit:\{t_1,t_2,\cdots,t_m\}$

输出:优化后的量子线路 *Optimal-Circuit*

OPTIMAL-CIRCUIT (Circuit)

1. for $i=1$ to m
2. 根据推论 2 尽可能将与 $t[i]$ 有相同受控端的 Toffoli门交换在一起;
3. 根据定理 6 尽可能与 $t[i]$ 后面的 Toffoli 门进行合并;
4. $i=$ 不能与 $t[i]$ 合并的第一个 Toffoli 门下标;
5. 得到优化后的量子线路 *Optimal-Circuit*;
6. if *Circuit*=*Optimal-Circuit*

   //递归调用优化过程直到不能再优化为止

   return 优化后的量子线路 *Optimal-Circuit*;
7. else

   OPTIMAL-CIRCUIT (Optimal-Circuit).

例 5. 考虑例 4 中得到的量子电路,对其运用算法4,优化过程如图12所示,其中图(a)为例子中

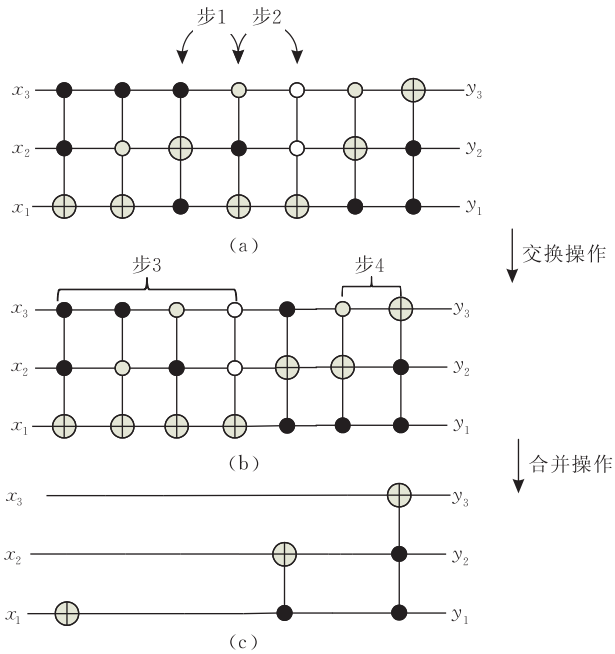


图 12　$f=[7,0,1,2,3,4,5,6]$得到 Toffoli 门的合并过程

得到的量子电路，步 1 和步 2 表示进行交换操作，交换后得到的图如(b)所示，步 3 和步 4 表示图(b)中电路前四项和后两项可以分别进行合并，从而可以得到图(c)中所示的最优电路.

6 实验结果和分析

为了验证本文提出的算法的正确性，我们用 C++ 编写了算法的实现程序，对任意可逆函数均可得到对应的电路图，实验环境是 Intel Core™2 CPU 2.13GHz，2GB 内存，操作系统为 Windows XP SP2. 考虑到 $16!=2.09227899\times10^{13}$ 对计算时间和存储的要求难以实现，故采用流行的 3×3 量子电路的所有可逆函数的实验来进行统计分析，不进行优化总共用时 0.219s，优化后总共用时 0.410s，其结果如表 1 所示.

表 1 中的每一行表示的是使用相同门数量对应的可逆函数的个数，每一列为所有可逆函数的和为 8!=40320. 最优情况是来自文献[4]中的实验结果，使用的门库含有 NOT、CNOT、*TOF*3；POT 列是文献[14]的实验结果，只使用了 *TOF*3，是目前快速综合算法中比较好的结果；优化前是指采用本文提出的算法在优化前只使用了 *TOF* 门的结果；优化后的结果使用了 NOT、CNOT、*TOF*3 门.

从表 1 中还可以明显看出，优化后我们的算法使用的门数量平均值为 6.14，相比文献[10]中的 6.83 更接近最优值的 5.87[4]，意味着我们的算法能更最近最优解.

表 1 所有 3 比特可逆函数的综合结果

门的数量	电路的数量			
	最优情况	*POT*	优化前	优化后
14			2	
13			57	2
12		25	394	21
11		258	1542	158
10		1265	3895	699
9		3788	6938	2241
8	577	7820	8906	4981
7	10253	10630	8310	8095
6	17049	9126	5723	9731
5	8921	4996	2953	8112
4	2780	1833	1161	4471
3	625	476	348	1499
2	102	90	78	282
1	12	12	12	27
0	1	1	1	1
门数量均值	5.87	6.83	7.65	6.14

接下来采用文献[9,14-16]中用到的函数综合例子(包括 9 个 3 比特函数和 3 个 4 比特函数)，把我们的算法与其它目前结果较好的算法进行比较，采用的包括 PPRM 表达式方法[9]、Non-search 方法[16]和 POT 方法[14]. 其结果如表 2 所示，从表中可以看出，虽然本文得到的结果和 POT 方法相同，接近 PPRM 方法，好于 Non-search 方法，但是从平均时间上来说，在实验环境不优于文献[10](Intel Xeon 3GHz，2Gn 内存的工作站)的情况下，本文综合每个电路的平均时间为 0.01ms，远好于 POT 方法的 0.2ms 和 PPRM 方法的 0.5s[10]. 由此可见，在得到近似最优电路的前提下，我们算法的效率是其它算法的几十到几百倍.

表 2 和其它启发式算法的比较结果

函数	Toffoli 门数量					本文得到的综合电路 ($x_4=d, x_3=c, x_2=b, x_1=a$)
	PPRM	Non-search	POT	优化前	优化后	
[1,0,3,2,5,7,4,6]	4	6	4	5	4	*TOF*3(*c*,*b*,*a*) *TOF*3(*c*,*a*,*b*) *TOF*3(*c*,*b*′,*a*) *TOF*3(*c*′,*a*)
[7,0,1,2,3,4,5,6]	3	3	3	7	3	*TOF*1(*a*)*TOF*2(*c*,*b*)*TOF*(*c*,*b*,*a*)
[0,1,2,3,4,6,5,7]	3	3	3	3	3	*TOF*3(*c*,*b*,*a*)T0F3(*c*,*a*,*b*)*TOF*3(*c*,*b*,*a*)
[0,1,2,4,3,5,6,7]	5	7	5	5	5	*TOF*3(*c*,*b*′,*a*)*TOF*3(*c*,*a*,*b*)*TOF*3(*b*,*a*,*c*) *TOF*3(*c*,*a*,*b*) *TOF*3(*b*′,*c*,*a*)
[0,1,2,3,4,5,6,8,7,9,10,11,12,13,14,15]	7	15	7	7	7	*TOF*4(*d*,*c*′,*b*′,*a*)*TOF*4(*d*,*c*′,*a*,*b*) *TOF*4(*d*,*b*,*a*,*c*)*TOF*4(*b*,*c*,*a*,*d*) *TOF*4(*d*,*b*,*a*,*c*)*TOF*4(*d*,*c*′,*a*,*b*) *TOF*4(*d*,*c*′,*b*′,*a*)
[1,2,3,4,5,6,7,0]	3	3	3	5	5	*TOF*3(*b*,*a*,*c*)*TOF*2(*c*,*a*)*TOF*2(*c*,*b*) *TOF*2(*a*,*b*)*TOF*3(*c*′,*a*)
[1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,0]	4	4	4	14	4	*TOF*4(*d*′,*c*′,*b*′,*a*)*TOF*3(*d*,*c*,*a*) *TOF*2(*d*′,*c*)*TOF*1(*d*)
[0,7,6,9,4,11,10,13,8,15,14,1,12,3,2,5]	4	3	6	14	3	*TOF*2(*d*,*a*)*TOF*2(*c*,*b*)*TOF*2(*d*,*c*)
[3,6,2,5,7,1,0,4]	7	8	8	9	6	*TOF*3(*c*,*b*,*a*)*TOF*2(*a*,*c*)*TOF*3(*c*′,*b*,*a*) *TOF*3(*c*,*a*,*b*)*TOF*2(*a*′,*b*)*TOF*2(*b*,*a*)

续　表

函数	Toffoli 门数量					本文得到的综合电路 ($x_4=d,x_3=c,x_2=b,x_1=a$)
	PPRM	Non-search	POT	优化前	优化后	
[1,2,7,5,6,3,0,4]	6	8	7	10	8	$TOF3(c,b,a)\,TOF3(c,a,b)\,TOF3(b,a,c)$ $TOF2(c,b)\,TOF3(c',b,a)\,TOF3(b,a,c)$ $TOF3(c',a,b)\,TOF2(c',a)$
[4,3,0,2,7,5,6,1]	7	8	5	7	7	$TOF3(c,a',b)\,TOF3(b',a',c)\,TOF3(c,b,a)$ $TOF3(b,a',c)\,TOF2(c,b)\,TOF2(a,b)$ $TOF(b',c',a)$
[7,5,2,4,6,1,0,3]	7	6	6	9	6	$TOF3(c,b,a)\,TOF3(b,a,c)\,TOF2(b,c)$ $TOF2(a',b)\,TOF3(c',b,a)\,TOF2(a,c)$
平均值	5.00	6.17	5.08	7.92	5.08	

此外,相比其它算法,很明显可以看出我们的算法思路简单清晰,并且在任何实验环境条件下均可轻松模拟实现.而且,对于该算法得到的 $TOFn$ 门,可以很容易根据文献[11]中的方法继续化简,因此该算法具有一定的实用性.

7　结　论

本文提出了一种新的可逆逻辑综合算法,其实质为基于变换规则的合成法.它采用一个无向无权图表示所有可以进行变换的路径,并在综合的过程中,类似选择排序算法的思想从小到大地选择需要交换的输出项,接着从路径选择图中找到最优的路径进行变换,最终综合完成后使输出项序列有序,而每条变换路径即对应一系列的电路门级联.在此基础上,本文还提出了一个优化算法化简所得到的量子电路,使最后得到的电路代价减少.实验表明,相比其它综合算法,该算法不仅总能获得最优解或近似最优解,而且效率优势明显.

此外,本文提出的算法易于理解,对于有计算机专业基础的人容易实现,同时基于图论解决可逆逻辑综合问题的思想也有一定的借鉴意义.

参　考　文　献

[1] Deutsch D. Quantum theory, the Church-Turing principle and the universal quantum computer//Proceedings of the Royal Society. London, 1985, 400: 97-117

[2] Wille R, Groβe D. Fast exact Toffoli network synthesis of reversible logic//Proceedings of the Computer-Aided Design. San Jose, CA, 2007: 60-64

[3] Wille R, Le H M, Dueck G W, Groβe D. Quantified synthesis of reversible logic//Proceedings of the Design, Automation and Test. Europe, 2008: 1015-1020

[4] Shende V V, Prasad A K, Markov I L, Hayes J P. Synthesis of reversible logic circuits. IEEE Transactions on Circuits and Systems-I, 2003, 22(6): 723-729

[5] Hung W N N, Song X Y, Yang G W, Yang J, Perkowski M. Quantum logic synthesis by symbolic reachability analysis//Proceedings of the Design Automation Conference. San Diego, California, 2004: 838-841

[6] Miller D M, Dueck G W. Spectral techniques for reversible logic synthesis//Proceedings of the 6th International Symposium Representations Methodology of Future Computing Technologies. Trier, 2003: 56-62

[7] Kerntopf P. A new heuristic algorithm for reversible logic synthesis//Proceedings of the 41st Annual Design Automation Conference. San Diego, California, 2004: 834-837

[8] Maslov D, Dueck G W, Miller D M. Toffoli network synthesis with templates. IEEE Transactions on Circuits and Systems-I, 2005, 24(6): 807-817

[9] Gupta P, Agrawal A, Jha N K. An algorithm for synthesis of reversible logic circuits. IEEE Transactions on Circuits and Systems-I, 2006, 25(11): 807-817

[10] Barenco A, Bennett C H, Cleve R, DiVincenzo D P, Margolus N, Shor P, Sleator T, Smolin J A, Weinfurter H. Elementary gates for quantum computation. Physical Review A, 1995, 52(5): 3457-3467

[11] Liu Y, Long G L, Sun Y. Analytic one-bit and CNOT gate constructions of general n-qubit controlled gates. International Journal of Quantum Information, 2008, 6(3): 447-462

[12] Long G L. General quantum interference principle and duality computer. Communications in Theoretical Physics, 2006, 45(5): 825-844

[13] Gudder S. Mathematical theory of duality quantum computers. Quantum Information Processing, 2006, 6(1): 37-48

[14] Zheng Y, Huang C. A novel Toffoli network synthesis algorithm for reversible logic//Proceedings of the Design Automation Conference. Yokohama, 2009: 739-744

[15] Saeedi M, Zamani M S, Sedighi M. On the behavior of substitution-based reversible circuit synthesis algorithms: Investigation and improvement//Proceedings of the IEEE Computer Society Annual Symposium. Porto Alegre, 2007: 428-436

[16] Saeedi M, Sedighi M, Zamani M S. A novel synthesis algorithm for reversible circuits//Proceedings of the IEEE/ACM International Conference on Computer-Aided Design. San Jose, California, 2007: 65-68

WAN Si-Shuang, born in 1986, M. S. candidate. His current research interests include quantum computing, quantum reversible logic circuit synthesis.

CHEN Han-Wu, born in 1955, Ph. D., professor, Ph. D. supervisor. His current research interests include quantum computing, information theory.

CAO Ru-Jin, born in 1986, M. S. candidate. His current research interests include semantic Web, algorithm analysis and design.

Background

This work is supported by the National Natural Science Foundation of China under grant No. 60873101 and Natural Science Foundation of Jiangsu province Nos. BK2007104, BK2008209.

Reversible logic synthesis is considered as a rapidly developing research area. Interest in reversible logic is sparked by its necessity in quantum technologies. Reversible logic studies have promising potential on energy lossless circuit design, quantum computation, nanotechnology, etc.

Synthesis of reversible logic circuits means automatically constructing the desired quantum reversible logic circuits. Though existing synthesis methods can provide optimal solutions, yet they may suffer from long computation time, due to the fact that the search space is likely to grow exponentially as the circuit size increases.

Therefore, in this paper, the authors propose an analogic selection sorting algorithm essentially based on the transformation-based algorithm.