基于矩阵初等变换的量子可逆逻辑电路双向综合算法

王 冬[1,2] 刘 强[1] 李善治[1]
(河南大学软件学院 开封 475001)[1] (武汉大学软件工程国家重点实验室 武汉 430072)[2]

摘 要 基于矩阵初等变换,提出了量子可逆逻辑电路双向综合算法。该算法依据两数字间的汉明距离,通过交换矩阵行号或矩阵元素对量子可逆逻辑电路的矩阵进行初等行变换。在变换的过程中,利用邻接矩阵的电路转化规则,生成任意给定置换的量子可逆逻辑电路。与其它同类算法相比,由于不需要穷尽搜索,该算法的时空复杂度有大幅降低;又由于采用任意 n 量子扩展通用 Toffoli 门,该算法可综合任一置换(奇或偶置换)的量子可逆逻辑电路,并且电路中门的数量有所减少。

关键词 量子可逆逻辑电路,量子计算,Toffoli 门
中图法分类号 TP387,TN911.73 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2014.09.002

Bidirectional Reversible Logic Circuit Synthesis Algorithm Based on Matrix Elementary Transformation

WANG Dong[1,2] LIU Qiang[1] LI Shan-zhi[1]
(Software College, Henan University, Kaifeng 475001, China)[1]
(State Key Laboratory of Software Engineering, Wuhan University, Wuhan 430072, China)[2]

Abstract Quantum reversible logic circuit synthesis is one of the key techniques to construct the quantum computer. Bidirectional reversible logic circuit synthesis algorithm based on matrix elementary transformation was proposed in this paper. The algorithm can construct reversible logic circuit of any given permutation by using the circuit transformation rules of the adjacent matrix and the method of exchanging either the row number or the elements of the matrix in terms of hamming distance of two numbers. Compared with the other algorithms, the time-space complexity of our algorithm has been decreased greatly because of without exhaustive searching. Moreover, the types of the quantum reversible logic circuits synthesized by our algorithm are even and odd permutation because using n-qubit extended general Toffoli gates, and the number of quantum gates in the circuit is cut down.

Keywords Quantum reversible logic circuit, Quantum computation, Toffoli gate

1 引言

量子可逆逻辑电路综合技术是构建量子计算机的关键技术之一[1]。量子可逆逻辑电路综合通常采用量子门级联的方式,量子门均可逆,Bennett 证明采用可逆门构建可逆电路可有效降低计算能耗(理论上可以达到最低能耗[2])。

近 30 年来,人们已提出了多种量子门,如通用 Toffoli 门(General Toffoli gate, GT 门)、受控交换门(Controlled swap gate, Fredkin 门)等;Barenco 等人提出用一位量子门和控制非门(CNOT gate)可构造用于量子计算的通用门库[3];Liu 等人依据此通用门库,给出了建立一般的 n-qubit 受控酉门的两个解析表达式[4];Song 等人给出了量子门的代数特征[5]。如何使用指定量子门库中的量子门自动生成量子代价较小的量子可逆逻辑电路,其本质在于量子可逆逻辑综合技巧问题。Iwama 等人提出了特定 CNOT 电路的转化规则[6];Miller 应用谱函数实现近似最优的可逆逻辑电路化简[7];Maslov 等人提出利用真值表和模板技术构造并优化量子可逆逻辑电路的方法[8,9];Shende 将可逆逻辑电路综合转化为置换问题,提出了三量子可逆逻辑电路综合最优算法[10];Yang 在此基础上利用 GAP 软件实现了三量子最小长度和最小代价可逆逻辑电路综合[11]。在综合四量子电路时,Yang 等人利用广度优先搜索和双向综合技术,使用 CNP 量子门库可综合最长为 12 的四量子偶置换最优电路[12,13],这已是较好结果;Li 等人使用 CNP 量子门库,在广度优先搜索的基础上,巧妙构造哈希函数并利用线置换和向变换进行无损压缩来快速生成最大长度为 16 的最优四量子偶置换电路[14]。基于广度优先搜索的量子可逆逻辑电路综合是一个组合优化问题,搜索空间随着量子比特数的增加呈指数级增加。所以,以此为基础综合 $n(n\geqslant 4)$ 量子可逆逻辑电路所需的内存空间将远远超过经典计算机能够承受的范围,综合出的电路相对于整个 $n(n\geqslant 4)$ 量子可逆逻辑电路空间,所占比重将急剧下降。

本文依据量子可逆逻辑电路要实现的置换,给出其酉矩

到稿日期:2014-01-22 返修日期:2014-02-17 本文受国家自然科学基金项目(61004006),软件工程国家重点实验室(武汉大学)开放课题(SKLSE2012-09-41)资助。

王 冬(1977—),女,博士,副教授,主要研究方向为量子计算、量子电路综合,E-mail:juliawdd@qq.com;刘 强(1966—),女,副教授,主要研究方向为软件理论;李善志(1966—),男,实验师,主要研究方向为软件理论。

阵的构造方法，在此基础上提出基于矩阵初等变换的量子可逆逻辑电路双向综合算法(BMET)。理论分析表明，该算法综合 n 量子电路最多需要 $(n-1)\cdot 2^n+1$ 个扩展通用 Toffoli 门。与文献[12,13]相比，由于不需要穷尽搜索，该算法的时空复杂度呈指数级降低，与量子可逆逻辑电路构造性算法[15]相比，使用的扩展通用 Toffoli 门数量在最坏情况下减少 $(n+1)\cdot 2^n-1$。

2 预备知识

定义 1 设 $\Sigma=\{a_0,a_1,\cdots,a_{n-1}\}$ 是一个 n 元集合。集合 Σ 到 Σ 的 1－1 映射 f 叫做 Σ 上的一个置换，表示为：$f=\begin{pmatrix} a_0 & a_1 & \cdots & a_{n-1} \\ a_{g_0} & a_{g_1} & \cdots & a_{g_{n-1}} \end{pmatrix}$，其中 $(a_{g_0},a_{g_1},\cdots,a_{g_{n-1}})$ 是 $(a_0,a_1,\cdots,a_{n-1})$ 的一个新的排列[16]。

定义 2 Toffoli 量子门记为 $TOF(C,T)$，其中输入变量集合 $In=\{x_0,x_1,\cdots,x_{n-1}\}$，控制端集合 $C=\{x_{i_2},x_{i_3},\cdots,x_{i_l}\}$，受控端集合 $T=\{x_{i_1}\}$，$x_{i_m}\in In, m\in\{1,2,\cdots,l\}$，且 $C\cap T=\varnothing$，$C\cup T\subset In$。输出变量集合映射为 $\{x_0,x_1,\cdots,x_{i_1-1},x_{i_1}\oplus\prod_{m=2}^{l}x_{i_m},x_{i_1+1},\cdots,x_{n-1}\}$。若 $\exists m\in\{2,3,\cdots,l\}$，$x_{i_m}=0\Rightarrow\prod_{m=2}^{l}x_{i_m}=0$，受控端 x_{i_1} 的输出为 $x_{i_1}\oplus\prod_{m=2}^{l}x_{i_m}=x_{i_1}\oplus 0=x_{i_1}$；若 $\forall m\in\{2,3,\cdots,l\}$，$x_{i_m}=1\Rightarrow\prod_{m=2}^{l}x_{i_m}=1$，受控端 x_{i_1} 的输出为 $x_{i_1}\oplus\prod_{m=2}^{l}x_{i_m}=x_{i_1}\oplus 1=\bar{x}_{i_1}$。控制端的数量决定了不同的 Toffoli 门，如图 1 所示。

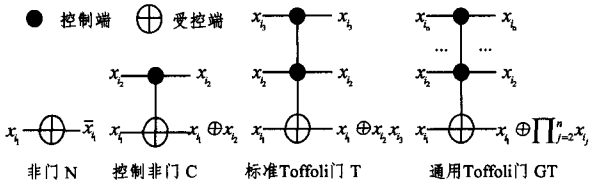


图 1 Toffoli 量子门

定义 3 通用 Toffoli 门只有置 1 控制条件。然而 1 并没有任何特殊之处，控制线置 0 控制受控线信号反转是合理的[1]。增加了置 0 控制条件的通用 Toffoli 门称为扩展通用 Toffoli 门(Extended General Toffoli Gate)，记为 $EGT(C,T)$，如图 2 所示。

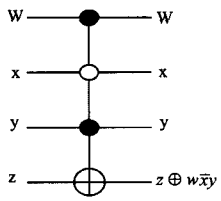


图 2 EGT 门

量子可逆逻辑电路是可逆函数的线路模型，由若干量子门级联构成，n 量子可逆逻辑电路实现的可逆函数为：$f(x_0,x_1,\cdots,x_{2^n-1})=(x_{g_0},x_{g_1},\cdots,x_{g_{2^n-1}})$。其中 $(x_{g_0},x_{g_1},\cdots,x_{g_{2^n-1}})$ 是 $(x_0,x_1,\cdots,x_{2^n-1})$ 的重新排列，所以 f 是一个置换，记作：$f=\begin{pmatrix} x_0 & x_1 & \cdots & x_{2^n-1} \\ x_{g_0} & x_{g_1} & \cdots & x_{g_{2^n-1}} \end{pmatrix}$。

定理 1 任何 $n\times n$ 规模的不含 $n\times n$ 门的可逆逻辑电路均实现偶置换[17]。

证明：因为偶置换的乘积还是偶置换，所以只需证明只有一个门的电路的情况即可。设 G 是 $n\times n$ 电路中的一个门，根据题意可知，此门不是 $n\times n$ 的，所以至少有一条线不受 G 的影响。不失一般性，假设 x_n 为最高位线，则 G 为 $(n-1)$ 位量子门，以此实现 2^{n-1} 元集合 $(0,1,\cdots,2^{n-1}-1)$ 上的一个置换。由于最高位线不受 G 门影响，因此有：$x_n\cdot 2^{n-1}\oplus G(k)=G(x_n\cdot 2^{n-1}\oplus k)$，$(x_n\in(0,1),k<2^{n-1},G(k)$ 为 k 在 G 门置换下的对应元素，$G(k)<2^{n-1})$。因此，在 G 门和最高位线所组成的 n 位可逆电路所实现的置换中，G 置换必然成对出现：其中一个是最高位为 0，另一个是最高位为 1。两种情况下，其余线上均实现 G 置换，则两种情况下所实现的置换的奇偶性必然相同。所以 G 门和最高位线所组成的 n 位可逆逻辑电路所实现的置换是这两个不邻接的置换的乘积。根据置换群的基本原理：奇偶性相同的两个置换的乘积必为偶置换[16]，则有 G 门和最高位线所组成的 n 位量子可逆逻辑电路仅实现偶置换。

由此可知，在 $n(n\geqslant 4)$ 量子电路中仅使用三量子门只能实现偶置换[12,13]。为能够综合任意 $n(n\geqslant 4)$ 量子奇偶置换电路，我们采用 $n(n\geqslant 4)$ 量子 EGT 门。

定义 4 设有两个不同的二进制数 s 和 t。连接 s 和 t 的邻接矩阵定义为：以 s 开头以 t 结束的一组二进制数组成的矩阵，并且矩阵中相邻的两个数恰好仅有一位不同。

例 1 $s=(1001)_B$，$t=(0111)_B$，连接 s 和 t 的邻接矩阵为：
$$\begin{pmatrix} 1 & 0 & 0 & 1 & \cdots & \cdots & s \\ 1 & 0 & 1 & 1 & \cdots & \cdots & d_1 \\ 1 & 1 & 1 & 1 & \cdots & \cdots & d_2 \\ 0 & 1 & 1 & 1 & \cdots & \cdots & t \end{pmatrix}$$

邻接矩阵中相邻两个二进制数的转换，可由一个 EGT 门实现，让其中变化的那一位作受控端，其余不变的位作控制端。则实现将 s 变换到 t 的电路如图 3 所示。

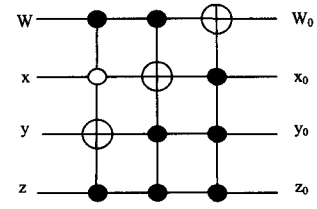


图 3 将 s 变换到 t 的电路

3 可逆函数的矩阵表示

设 $e_0,e_1,\cdots,e_{2^n-1}$ 为 2^n 维 Hilbert 空间的一组标准正交基，$e_h(h\in(0,1,\cdots,2^n-1))$ 表示第 h 位为 1、其余 2^n-1 位均为 0 的单位行向量，即 $e_h=\begin{pmatrix} 0 & \cdots & 1 & \cdots & 0 \\ \text{第 0 位} & & \text{第 } h \text{ 位} & & \text{第 } 2^n-1 \text{ 位} \end{pmatrix}$。$n$ 量子可逆逻辑电路实现的置换 $f=\begin{pmatrix} x_0 & x_1 & \cdots & x_{2^n-1} \\ x_{g_0} & x_{g_1} & \cdots & x_{g_{2^n-1}} \end{pmatrix}$ 在映射 σ：$\sigma(x_{g_j})=e_{x_{g_j}}$，$j\in(0,1,\cdots,2^n-1)$ 下被惟一表示成矩阵 $\begin{pmatrix} e_{x_{g_0}} \\ e_{x_{g_1}} \\ \vdots \\ e_{x_{g_{2^n-1}}} \end{pmatrix}$。

在置换 $f=\begin{pmatrix} x_0 & x_1 & \cdots & x_{2^n-1} \\ x_{g_0} & x_{g_1} & \cdots & x_{g_{2^n-1}} \end{pmatrix}$ 中，$(x_{g_0}, x_{g_1}, \cdots, x_{g_{2^n-1}})$是$(x_0, x_1, \cdots, x_{2^n-1})$的一个新排列，由 σ 的定义和矩阵乘法规则有：$\begin{pmatrix} e_{x_{g_0}} \\ e_{x_{g_1}} \\ \vdots \\ e_{x_{g_{2^n-1}}} \end{pmatrix} \times \begin{pmatrix} x_0 \\ x_1 \\ \cdots \\ x_{2^n-1} \end{pmatrix} = \begin{pmatrix} x_{g_0} \\ x_{g_1} \\ \vdots \\ x_{g_{2^n-1}} \end{pmatrix}$。则矩阵 $\begin{pmatrix} e_{x_{g_0}} \\ e_{x_{g_1}} \\ \vdots \\ e_{x_{g_{2^n-1}}} \end{pmatrix}$ 实现了置换 $f=\begin{pmatrix} x_0 & x_1 & \cdots & x_{2^n-1} \\ x_{g_0} & x_{g_1} & \cdots & x_{g_{2^n-1}} \end{pmatrix}$ 的函数功能。又因为$(x_0, x_1, \cdots, x_{2^n-1})$的全排列构成对称群 S_{2^n}，其中每一个置换都是惟一的，则在 $f=\begin{pmatrix} x_0 & x_1 & \cdots & x_{2^n-1} \\ x_{g_0} & x_{g_1} & \cdots & x_{g_{2^n-1}} \end{pmatrix}$ 中，x_j 惟一对应 x_{g_j}，$j\in(0,1,\cdots,2^n-1)$，又从 σ 的定义知：x_{g_j} 惟一对应 $e_{x_{g_j}}$，所以矩阵 $\begin{pmatrix} e_{x_{g_0}} \\ e_{x_{g_1}} \\ \vdots \\ e_{x_{g_{2^n-1}}} \end{pmatrix}$ 是 n 量子可逆逻辑电路实现置换 $f=\begin{pmatrix} x_0 & x_1 & \cdots & x_{2^n-1} \\ x_{g_0} & x_{g_1} & \cdots & x_{g_{2^n-1}} \end{pmatrix}$ 的惟一矩阵形式，在数学上，可用该矩阵表示 n 量子可逆逻辑电路实现的置换功能。

例 2　置换 $f=\begin{pmatrix} 0&1&2&3&4&5&6&7&8&9&10&11&12&13&14&15 \\ 0&3&2&1&4&5&6&7&8&10&12&14&15&13&11&9 \end{pmatrix}$ 的矩阵如表 1 所列。

$\sigma(x_{g_0})=\sigma(0)=e_{x_{g_0}}=e_0$

$\sigma(x_{g_1})=\sigma(3)=e_{x_{g_1}}=e_3$

$\sigma(x_{g_2})=\sigma(2)=e_{x_{g_2}}=e_2$

$\sigma(x_{g_3})=\sigma(1)=e_{x_{g_3}}=e_1$

$\sigma(x_{g_4})=\sigma(4)=e_{x_{g_4}}=e_4$

$\sigma(x_{g_5})=\sigma(5)=e_{x_{g_5}}=e_5$

$\sigma(x_{g_6})=\sigma(6)=e_{x_{g_6}}=e_6$

$\sigma(x_{g_7})=\sigma(7)=e_{x_{g_7}}=e_7$

$\sigma(x_{g_8})=\sigma(8)=e_{x_{g_8}}=e_8$

$\sigma(x_{g_9})=\sigma(10)=e_{x_{g_9}}=e_{10}$

$\sigma(x_{g_{10}})=\sigma(12)=e_{x_{g_{10}}}=e_{12}$

$\sigma(x_{g_{11}})=\sigma(14)=e_{x_{g_{11}}}=e_{14}$

$\sigma(x_{g_{12}})=\sigma(15)=e_{x_{g_{12}}}=e_{15}$

$\sigma(x_{g_{13}})=\sigma(13)=e_{x_{g_{13}}}=e_{13}$

$\sigma(x_{g_{14}})=\sigma(11)=e_{x_{g_{14}}}=e_{11}$

$\sigma(x_{g_{15}})=\sigma(9)=e_{x_{g_{15}}}=e_9$

表 1　置换 f 的矩阵表示

1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0

4　BMET 算法

量子可逆逻辑电路综合就是找到一个量子门序列实现置换函数的功能。由于置换可以用酉矩阵表示，因此此问题转化为：找到将酉矩阵变换为单位矩阵的量子门序列。由第 3 节可知：量子可逆逻辑电路的酉矩阵为每一行每一列只有一个 1、其它位置全为 0 的稀疏矩阵。将这种矩阵变换成单位矩阵只需逐次有序调整矩阵的行，对矩阵实施初等行变换即可。其中的每次调整都对应着一个量子门序列，将这些量子门级联即可生成给定置换的量子可逆逻辑电路。

4.1　算法描述

第 1 步　依据第 3 节的方法将给定置换表示成矩阵。

随着量子比特数的增加，在计算机中存储 $2^n\times 2^n$ 规模的矩阵所需的内存空间呈指数增长，如再进行矩阵运算，算法的空间复杂度将更大。考虑到 n 量子可逆逻辑电路的矩阵的特点为：每一行每一列只有一个 1，其余位置全为 0。受稀疏矩阵各种存储方法[18]的启发，我们在算法中仅使用一个 2^n 规模的数组 *Matrix* 来记录矩阵每行中 1 的位置，这样只需 2^n 个内存单元，比存储整个矩阵少用 2^n 倍内存单元，降低了算法空间复杂度。

如何记录 1 的位置是关键，因为 1 的位置的计算方法与算法的效率息息相关。我们采用从矩阵第 0 行开始到第 2^n-1 行结束，每行自左起，从 0 开始计数来计算 1 的位置。这样就可以用 *Matrix* 数组的下标代表矩阵的行号，数组元素代表 1 在每一行中的位置。如将表 1 的矩阵存储在 *Matrix* 数组中，*Matrix*={0,3,2,1,4,5,6,7,8,10,12,14,15,13,11,9}，*Matrix*[1]=3 表示：矩阵第 1 行中的 1 在第 3 列。

第 2 步　构建电路。

从第 0 行开始依序检查矩阵第 i 行的 1 是否位于第 i 列（?*Matrix*[i]=i），若不是，通过调换矩阵的行，使得第 i 行中的 1 位于第 i 列（即 *Matrix*[i]=i），按此方法继续，直到矩阵变为单位矩阵（$\forall i, i\in(0\cdots 2^n-1)$，*Matrix*[$i$]=$i$）为止。在调整过程中，为尽量减少量子门的数量，我们依据数字间的汉明距离采用双向变换算法。

定义 5　两个数 x, y 之间的汉明距离 $Hd(x,y)$ 是指 x, y 的二进制表示中不同位的个数。由定义 4 可知，汉明距离可表示在将 x 变换到 y 的过程中需要使用的 EGT 门的数量。

考虑到调整矩阵元素与调整矩阵行号是相对操作，其结果相同，如图 4 所示。而汉明距离决定了在调整过程中所使

用的 EGT 门数量。为减少门数，在每次调整中（如果目前 $Matrix[i]=j$，则调整到 $Matrix[i]=i$），首先找到矩阵中 1 的位置为 i 的第 k 行（$Matrix[k]=i$），比较矩阵这两行中 1 的位置的汉明距离 $Hd(j,i)$ 和两行行号的汉明距离 $Hd(k,i)$，以较小者生成门电路，这样可有效减少电路中门的数量。其中，调整矩阵元素为反向生成电路，调整矩阵行号为正向生成电路。

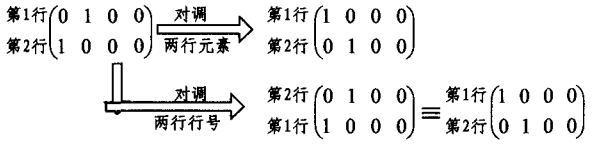


图 4　两种等价的矩阵调整方法示例

在将量子可逆逻辑电路的矩阵调整变换为单位矩阵的过程中，为使后期的变换不会影响到前期已经调整到位的行，在生成两个数的邻接矩阵时，总是先从右向左将较大数中的 0 全部变成 1，然后再从左向右将较大数中的 1 全部变成 0，直至变成较小数为止。

算法 1　构建量子可逆逻辑电路 GenerateCircuit。

输入：存放矩阵的数组 Matrix。

输出：存放量子电路的数组 Gates，存放电路中门数量的变量 GatesNumber。

```
1. for (i=0;i<2^n-1;i++)
2. {if (Matrix[i]!=i)
3. {for (k=i+1;Matrix[k]! =i;k++);
4. if (Hd(Matrix[i],i)<=Hd(k,i))
5.     {AdjaMtrix=AdjactMatrix(Matrix[i],i)
6.       for(con=0;con<Hd(Matrix[i],i);con++)
7.         {store Gate(AdjaMatrix[con],AdjaMatrix[con+1]) to
           RGates;
8.           RGatesNumber++;
9.           for (h=i+1;Matrix[h]!=AdjaMatrix[con+1];h++)
10.           swap(Matrix[i],Matrix[h]);}}
11.      else
12.       {AdjaMtrix=AdjactMatrix(k,i)
13.        for(con=0;con<Hd(k,i);con++)
14.         {store Gate(AdjaMatrix[con],AdjaMatrix[con+1]) to
            Gates;
15.          GatesNumber++;
16.          swap(Matrix[AdjaMatrix[con]],Matrix[AdjaMatrix
             [con+1]]);}}}
17. for (i=RGatesNumber-1;i>=0;i--)
18.   Gates[GatesNumber++]=RGates[i];
```

第 1－3 行，找到矩阵中的第 i 行，其中的 1 不在第 i 列，及找到第 k 行，其中的 1 在第 i 列。

第 4－10 行，以两行中 1 的位置生成邻接矩阵，再依据定义 4，以此邻接矩阵来生成电路。

第 12－16 行，以两行的行号生成邻接矩阵，再依据定义 4，以此邻接矩阵来生成电路。

第 17－18 行　将逆向电路与正向电路连接在一起。

第 3 步　优化电路，优化规则为：

1. 如果存在连续两个完全相同的 EGT 门，则可以将这两个门一起移除，如图 5 所示。

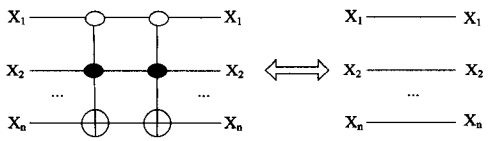


图 5　优化规则 1

证明：不失一般性，假设连续的两个相同的 EGT 门 A 和 B 的前 $n-1$ 条线为控制端，其中第一条线为置 0 条件控制，其余为置 1 条件控制。第 n 条线为受控端。根据定义 2 可知，经过 A 的输出为：$(x_1,x_2,\cdots,x_{n-1};x_n\oplus\overline{x_1}\oplus\prod_{k=2}^{n-1}x_k)$，因为 B 与 A 完全相同，所以再经过 B 的输出为 $(x_1,x_2,\cdots,x_{n-1};x_n\oplus\overline{x_1}\oplus\prod_{k=2}^{n-1}x_k\oplus\overline{x_1}\oplus\prod_{k=2}^{n-1}x_k)=(x_1,x_2,\cdots,x_{n-1};x_n\oplus 0)=(x_1,x_2,\cdots,x_{n-1};x_n)$，其结果等价于不经过任何门的输出结果，故移除连续的两个完全相同的 EGT 门不会影响电路的功能。

算法 2　按照优化规则 1 优化电路的算法 SimplyRule1。

输入：存放量子电路的数组 Gates。

输出：存放量子电路的数组 Gates，其中的门有删除。

```
1. char * p1=Gates, * p2=p1+4;
2. for (i=0;i<=3 && p1[i]==p2[i];i++);
3. if (i==4)
4. {for (i=0;i<=3;i++)
5.   {p1[i]='*';p2[i]='*';}
6. GatesNumber-=2;}
```

第 2 行，比较两个相邻的门是否相同。

第 3－5 行，删除两个相邻且相同的门，“*”代表删除。

2. 对于相邻的两个 EGT 门，如果受控端相同而控制端只有一位控制条件不同，则可以把这一位控制端去掉，留下相同的控制端和受控端合并成 1 个 EGT 门，如图 6 所示。

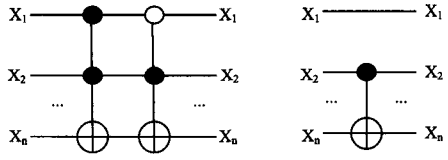


图 6　优化规则 2

证明：不失一般性，假设连续的两个 EGT 门 A 和 B 的前 $n-1$ 条线为控制端，第 n 条线为受控端，且 A 和 B 只有第一条线上的控制条件不同：A 为置 1 控制，B 为置 0 控制。经过 A 的输出为：$(x_1,x_2,\cdots,x_{n-1};x_n\oplus\prod_{k=1}^{n-1}x_k)$，再经过 B 的输出为：

$$\begin{aligned}&(x_1,x_2,\cdots,x_{n-1};x_n\oplus\prod_{k=1}^{n-1}x_k\oplus\overline{x_1}\cdot\prod_{k=2}^{n-1}x_k)\\&=(x_1,x_2,\cdots,x_{n-1};x_n\oplus(x_1\oplus\overline{x_1})\cdot\prod_{k=2}^{n-1}x_k)\\&=(x_1,x_2,\cdots,x_{n-1};x_n\oplus 1\cdot\prod_{k=2}^{n-1}x_k)\\&=(x_1,x_2,\cdots,x_{n-1};x_n\oplus\prod_{k=2}^{n-1}x_k)\end{aligned}$$

此结果正好等价于只经过一个 EGT 门的输出结果，此门在第一条线上没有控制，其余线上的控制条件与 A 和 B 保持一致，并且受控端与 A 和 B 在同一条线上，得证。

算法 3　按照优化规则 2 优化电路的算法 SimplyRule2。

输入：存放量子电路的数组 Gates。

输出：存放量子电路的数组 Gates，其中的门有所删简。

```
1. struct NumberPosition
2. { int count;
3.   int position;  };
4. NumberPosition DifNumPos={0,0};
5. char * p1=Gates, * p2=p1+4;
6. for (i=0;i<=3;i++)
7.   {if (p1[i])!=p2[i])
8.       {DifNumPos. count+=1;
9.         if (DifNumPos. count>1)
10.            {reset DifNumPos;break;}
11. DifNumPos. position=i;}}
12. if (DifNumPos. position!  =0)
13.   {p1[DifNumPos. position]='*';
14.    for (k=0;k<=3;k++) * p2++='*';
15. GatesNumber--;reset DifNumPos;}
```

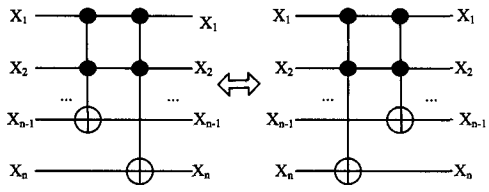
第 6—11 行，检查相邻的门是否只有一个控制端不同。

第 12—14 行，保留一个门并删除其中不同的控制端。

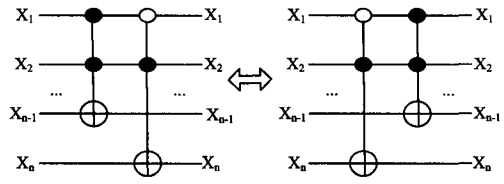
3. 若 EGT 门 $A=EGT(x_1,x_2,\cdots,x_{k-1};x_k)$ 与 $B=EGT(y_1,y_2,\cdots,y_{l-1};y_l)$ 满足：

条件 1：$x_k\notin\{y_1,y_2,\cdots,y_{l-1}\}$ and $y_l\notin\{x_1,x_2,\cdots,x_{k-1}\}$，或

条件 2：$\exists x_i\in\{x_1,x_2,\cdots,x_{k-1}\}$ and $y_i\in\{y_1,y_2,\cdots,y_{l-1}\}$，满足 $x_i=\overline{y_i}$，则 A 与 B 互不影响，可以交换 A 与 B 的位置，如图 7 所示。



(a)每一个EGT门的受控端都不在另一个门的控制线上时两门可交换



(b)同一条控制线上两个EGT门的控制条件不同时两门可交换

图 7　优化规则 3

证明：首先证明 A 和 B 两个 EGT 门在条件 1 下可交换。不失一般性，假设 A 和 B 控制端均为 $(x_1,x_2,\cdots,x_{n-2})$，A 受控端为 x_{n-1}，B 受控端为 x_n，依据 EGT 门的定义：先经过 A 门输出为 $(x_1,x_2,\cdots,x_{n-2},x_{n-1}\oplus\prod_{k=1}^{n-2}x_k,x_n)$，再经过 B 门输出为 $(x_1,x_2,\cdots,x_{n-2},x_{n-1}\oplus\prod_{k=1}^{n-2}x_k,x_n\oplus\prod_{k=1}^{n-2}x_k)$。A 与 B 交换后，先经过 B 门输出为 $(x_1,x_2,\cdots,x_{n-2},x_{n-1},x_n\oplus\prod_{k=1}^{n-2}x_k)$，再经过 A 门输出为 $(x_1,x_2,\cdots,x_{n-2},x_{n-1}\oplus\prod_{k=1}^{n-2}x_k,x_n\oplus\prod_{k=1}^{n-2}x_k)$。由此可知，在此条件下，A 和 B 交换前后，经过这两个 EGT 门的输出结果相同。所以在条件 1 成立下，A 和 B 可交换。

其次证明 A 和 B 两 EGT 门在条件 2 下可交换。不失一般性，假设 x_1 为 A 与 B 两门的控制端之一，A 在 x_1 上的控制条件为 1，B 在 x_1 上的控制条件为 0。$x_1=1$ 时，无论 A 和 B 如何交换，只有 A 门可能起作用，B 门绝对不会起作用。$x_1=0$ 时，无论 A 与 B 如何交换，只有 B 门可能起作用，A 门绝对不会起作用。所以在条件 2 成立下，在某一时刻，A 和 B 两门中只可能有一个门起作用，所以 A 与 B 可交换。

算法 4　按照优化规则 3 优化电路的算法 SimplyRule3。

输入：存放量子电路的数组 Gates。

输出：存放量子电路的数组 Gates，其中的门有所移动。

```
1. char * p1=Gates, * p2=p1+4;int k1=0,k2=0;
2. while(p1[k1]!='3')  k1++;
3. while(p2[k2]!='3')  k2++;
4. {for (i=0;i<=3;i++)
5. {if (p1[k2]=='*'&&p2[k1]=='*' || p1[i]=='1' &&p2[i]
   =='0' || p1[i]=='0'&&p2[i]=='1')
6.   {Swap(gate1,gate2);break;}
```

交换相邻的两个门，将受控端在同一条线上的量子门尽量排列在一起，为合并与移除优化操作做准备。

4.2　算法举例

考虑例 2，生成可逆电路的步骤如下：

①将例 2 中的置换表示成表 1 所示的矩阵，并将其存入 *Matrix* 数组，*Matrix*={0,3,2,1,4,5,6,7,8,10,12,14,15,13,11,9}，数组下标代表矩阵的行号，数组元素代表本行中 1 所在列号。②扫描 *Matrix* 数组，寻找 *Matrix*[*i*]!=*i* 的元素，发现 *Matrix*[1]=3，则在 *Matrix* 数组中查找 1，*Matrix*[3]=1，此时无论是调整矩阵两行元素还是调整矩阵行号均使用(3,1)建立邻接矩阵(0011,0001)，在此情况下，我们选择调整矩阵元素的方法生成电路，即从后向前生成电路。依据定义 4 产生图 8 右数第一个门，交换元素(3,1)，此时 *Matrix*={0,1,2,3,4,5,6,7,8,10,12,14,15,13,11,9}。继续扫描发现 *Matrix*[9]=10，*Matrix*[15]=9，因为矩阵这两行中 1 的位置的汉明距离为 *Hd*(10,9)=2，这两行行号的汉明距离为 *Hd*(9,15)=2，*Hd*(9,10)≤*Hd*(9,15)，所以采用将矩阵第 9 行中 1 的列号从 10 调整到 9 的方法建立电路，以数字对(10,9)建立邻接矩阵(1010,1011,1001)，依据定义 4 产生图 8 右数第二个门，交换数组元素(10,11)，产生图 8 右数第三个门，交换数组元素(11,9)，此时 *Matrix*={0,1,2,3,4,5,6,7,8,9,12,14,15,13,10,11}。继续扫描发现 *Matrix*[10]=12，*Matrix*[14]=10，因为 *Hd*(12,10)=2，*Hd*(14,10)=1，*Hd*(12,10)>*Hd*(14,10)，所以采用调整行号的方法建立电路，以 (14,10)建立邻接矩阵(1110,1010)，产生图 8 左数第一个门并交换行号(14,10)。依此继续，直到 *Matrix*={0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15}时，算法终止，产生的量子可逆逻辑电路如图 8 所示。③利用优化规则对电路实施优化，如图 9 所示。至此，电路综合过程结束。

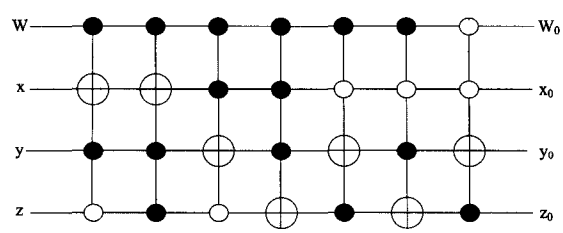


图 8　实现置换 f 的电路

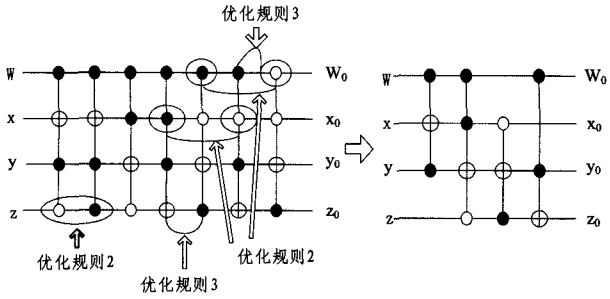


图 9　对 f 电路优化图解

5　算法性能分析

考虑到置换中前 2^{n-1} 个数中的每一个输出与输入之间最多有 n 位不同，依据定义 4 要将输出调整到与输入相同，即最多需要 n 个 EGT 门。如果前 2^{n-1} 个数中的每一个输出与输入都相同，则剩下的 2^{n-1} 个数中的每一个输出与输入之间最多有 $n-1$ 位不同，则要将后 2^{n-1} 个数中的前 2^{n-2} 个数的输出调整到与输入相同，即最多需要 $n-1$ 个 EGT 门，而如果前 $2^{n-1}+2^{n-2}$ 个数中的每一个输出与输入都相同，则剩下的 2^{n-2} 个数中的每一个输出与输入之间最多有 $n-2$ 位不同，则要将后 2^{n-2} 个数中的前 2^{n-3} 个数中的输出调整到与输入相同，即最多需要 $n-2$ 个 EGT 门，以此类推，直到剩下最后两个门时仅需一个 EGT 门。所以 BMET 算法实现 n 量子可逆逻辑电路所需 EGT 门最多为：

$$L(n)=n\cdot 2^{n-1}+(n-1)\cdot 2^{n-2}+(n-2)\cdot 2^{n-3}+\cdots+2\cdot 2^{1}+1=(n-1)\cdot 2^{n}+1$$

目前较为成熟的非穷尽搜索的量子可逆逻辑电路综合算法为构造性算法[15]，其使用的 EGT 门最多为 $2n\cdot 2^{n}$ 个，在最坏情况下比 BMET 算法多 $(n+1)\cdot 2^{n}-1$ 个。BMET 算法还使用了双向综合技术和优化规则来优化电路，优化后门的数量和复杂度都将进一步降低。

BMET 算法可直接构建给定置换的量子可逆逻辑电路，其中仅用一个 $\Theta(2^n)$ 规模的线性数组来存储量子可逆逻辑电路的酉矩阵，所以空间要求低。而文献[12-14]采用基于广度优先搜索的综合算法，该算法在最坏情况下需要记住所有 $(2^n)!/2$ 个电路才能综合某一特定置换的电路，其空间复杂度超过 $(2^n)!/2$。BMET 算法的时间复杂度为构建与优化量子可逆逻辑电路的算法的时间复杂度之和，在最坏情况下不超过 2^{n+3}。而基于广度优先搜索的综合算法与其空间复杂度分析一样，时间复杂度大于 $(2^n)!/2$。所以和文献[12-14]相比，BMET 算法的空间复杂度和时间复杂度都呈指数级降低。

由于当 $n\geqslant 4$ 时，文献[12-14]需要记住 $(2^n!)/2$ 个电路，已远超过经典计算机的内存空间，又由于它们均采用三量子 CNP 门库，因此它们只能综合出部分 n 量子偶置换电路。BMET 算法合理采用 n 量子 EGT 门，可以综合任一给定奇或偶置换的可逆逻辑电路。

结束语　酉矩阵是量子电路的数学模型，能够形式化地描述量子系统的演化和量子计算的数学属性。本文首先给出量子可逆逻辑电路的酉矩阵表示方法，在此基础上，又提出基于矩阵初等变换的量子可逆逻辑电路双向综合算法。与构造性算法相比，电路中门的数量随着量子比特的增加而大幅减少。与基于广度优先搜索的算法相比，该算法可综合任一置换(奇或偶置换)的量子可逆逻辑电路，并且算法的时间复杂度和空间复杂度都呈指数级降低。

进一步的工作为：寻找更加有效的优化规则和优化策略来对生成的量子可逆逻辑电路进行优化。

参 考 文 献

[1] Nielsen M A, Chuang I L. Quantum Computation and Quantum Information [M]. Cambridge: Cambridge University Press, 2000

[2] Bennett C. Logical reversibility of computation [J]. IBM Journal of Research and development, 1973, 17(6): 525-532

[3] Barcnco A, Bennett C, et al. Elementary gates for quantum computation [J]. Physical Review A, 1995, 52(5): 3457-3467

[4] Yang Liu, Long Gui-lu, Yang Sun. Analytic one-bit and CNOT Gate Constructions of General n-Qubit Controlled Gates [J]. International Journal of Quantum Information, 2008, 6(3): 447-462

[5] Song Xiao-yu, Yang Guo-wu, Perkowski M, et al. Algebraic characteristics of reversible gates [M]. Theory of Computing System, 2004: 311-391

[6] Iwama K, Kambayashi Y, Yamashita S. Transformation rules for designing CNOT-based quantum circuits [C] // Proceedings of Design Automation Conference, 2002. New Orleans, 2002: 419-424

[7] Miller D M, Maslov D, Gueck G W. Spectral and two-place decomposition techniques in reversible logic [C] // Proceedings of the 45th IEEE International Midwest Symposium on Circuits and Systems, 2002. Tulsa, 2002: 493-496

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

[9] Miller D M, Maslov D, Dueck G W. A transformation based algorithm for reversible logic synthesis [C] // Proceedings of the International Conference on Computer-Aided Design, 2003. 2003: 318-323

[10] Shende V V, Prasad A K, Markov I L, et al. Synthesis of reversible logic circuits [J]. IEEE Transactions on Circuits and System-I, 2003, 22(6): 710-722

[11] Yang Guo-wu, Song Xiao-yu, et al. Fast synthesis of exact Minimal reversible circuits using group theory [C] // Proceedings of the 2005 conference on Asia South Pacific design automation, 2005. Shanghai, 2005: 1002-1005

[12] Yang Guo-wu, Song Xiao-yu, Hung W N, et al. Bi-direction Synthesis for Reversible Circuits [C] // Proceedings of the IEEE Computer Society Annual Symposium on VLSI, 2005. Tampa: IEEE, 2005

[13] Yang Guo-wu, Song Xiao-yu, Hung W N, et al. Bi-directional Synthesis of 4-bit Reversible Circuits [J]. The Computer Journal, 2008, 51(2): 207-215

[14] Li Zhi-qiang, Chen Han-wu, Xu Bao-wen. Fast algorithms for 4-qubit reversible logic circuits synthesis [J]. Acta Electronic Sinica, 2008, 36(11): 2081-2089

[15] Yang Guo-wu, Xie Fei, Song Xiao-yu, et al. A constructive algorithm for reversible logic synthesis [C] // Proceedings of 2006 IEEE Congress on Evolutionary Computation, 2006 . Vancouver BC: IEEE, 2006

[16] Dixon J D, Mortimer B. Permutation Group [M]. New York: Springer, 1996

[17] Toffoli T. Reversible computing [J]. Computer Science, 1980, 85: 632-644

[18] Weiss M A. Data Structures and Algorithm Analysis in C (Second Edition) [M]. Beijing: POSTS & TELECOM PRESS, 2005