The Reversible Network Cascade Based on Reversible Logic Gate Coding Method

Hui Li, Zhijin Guan, Shanli Chen, Yuxin Chen
Nantong University, Nantong, 226019, China
e-mail: guan_zj@nuaa.edu.cn

Abstract—Reversible computing is an emerging area of research and reversible logic synthesis is its important aspect. This paper provides reversible network cascade method based on reversible logic gate coding method. According to the number of vertical lines and control bits, which from small to large, the method constructs automatically the different output vectors corresponding network meanwhile the algorithm can generate all the reversible network. Compared with the three variables reversible function test in Benchmark, its control gates amounts are much less and the cost is much lower.

Keywords— reversible logic gate; coding; reversible gate cascade

I. INTRODUCTION

Reversible logic synthesis is the important aspects of the reversible computing research[1][2][3][4][5][6] meanwhile reversible logic synthesis is the basis of quantum information technology. According to the reversible network’s no fan-out and no feedback constraint condition and limitation, reversible logic synthesis is to use reversible logic gate given to implement the relevant reversible logic network meanwhile make the cost as low as possible. Reversible logic gate cascade is one of the key issues of the reversible logic synthesis. Reversible logic gate network is the number of the input and output are equal and input vectors and output vectors are one to one mapping reversible logic gate collection. Therefore, the input vectors state can only be reconstructed by the output vectors, which are described by the way of the function as following: the function is reversible, if every input vectors of function can only be mapped by only one output vector. An $n$ variables reversible function can also be defined as integer set $\{0,1,\ldots,2^n-1\}$ self-mapping.

It has been proposed a variety of reversible logic gate, such as Feynman's controlled-not gate, Toffoli gate and Fredkin gate[7][8][9] and so on. How to use specified much lower cost reversible logic gate network has become a hot-spot area of reversible logic synthesis computing in recent years.

This paper mainly focuses on how to realize the $n$ input and $n$ output Toffoli gate algorithm, which main purpose is to find how many kinds of corresponding results according to the $n$ input vectors (according to the formula we know the result is $2^n$) and output corresponding circuit combination figures.

II. REVERSIBLE NETWORK CONSTRUCTING

A. Reversible logic gate

If a logic gate is reversible, then the number of its input / output bits must be equal. An $n$ input $n$ output binary logic gate is referred to as $n$-bit reversible logic gate. This paper mainly describes the Toffoli reversible logic gate series as following:

A single-input and single-output is referred to as the NOT gate in traditional logic gate, its role is to convert 0 into 1, 1 into 0. With regard to reversible logic gate, NOT gate is also applicable. The reversible logic network is generally indicated in Fig. 1. NOT gate is uncontrolled bit Toffoli gate.

The two basic input in $\{0,1\}$ are corresponding vector $\mathbf{C}_0, \mathbf{C}_1 = [0,1]$.

![Figure 1. NOT gate](image1)

A control bit Toffoli gate is single control reversible logic gate, also known as the control- not gate and be called for short as CNOT gate. The reversible logic gate simultaneously act on the two bits, one of bits used as a control bit, the other as a target bit; when the control-bit display 1, it will make the target bit’s value convert contrary state, when the control-bit display 0, the target value unchanged. In other words, to use the control-bit state value to determine the target bit value whether converts contrary state or not. The so-called CNOT gate input is referred to as the two-bit original state, the output is the two-bit state after being operated by CNOT gate, as shown in Fig. 2.

![Figure 2. CNOT gate](image2)

Dual-control gate also known as Control- Control–not-gate, which was proposed by Toffoli[8] who comes from Massachusetts Institute of Technology, the United States, another name for it is Toffoli gate, as shown in Fig. 3.

![Figure 3. CCNOT gate](image3)
Corresponding to CNOT gate, CCNOT gate has two control bits and a target bit. When the two control bit both are "1", the target bit will make a "NOT" operation.

B. Reversible logic gate cascade

Using the reversible logic gate to construct reversible network is one of the key issues of the reversible logic synthesis. The same input / output results can use the different reversible gate cascade. The first chapter describes the basic principles of using the reversible logic gate to construct reversible network, that is to say reversible network requires the same number of the input and output. In accordance with the reversible network structure, whatever kind of reversible logic gate are used to construct reversible networks, no fan-out and no feedback are natural constraint condition while constructing a reversible network.

This paper mainly takes Toffoli gate series (NOT gate, CNOT gate is referred to as a special form of Toffoli gate, which has no control bit and one control bit) about reversible logic gate cascade.

C. Toffoli gate network model

For a confirmed input / output bit Toffoli network, the basic elements of constructing the network are Toffoli gate, including the Toffoli gate input bits are less than or equal to network input bits.

N input / output network model can be described as:
1. The network has n horizontal lines.
2. The network has k (k >= 0) vertical lines.
3. The number of input bits for Toffoli gates (which are referred to the sum of the number of control bits and target bits) from the every vertical lines can be the same or different.
4. The network input vectors which from (0,0,0, ..., 0) to (1,1,1, ..., 1).

D. Encoding the network input /output bit and vertical lines

In order to facilitate the description and construction of the Network, we are to encode the network input / output bits and the vertical lines, the rules of encoding as following:

For n input / output network, encoding the horizontal lines from top to bottom separately as 0,1,2 ,..., n-1; its input / output ports code are the same. Encoding the vertical line from left to right separately as 1 ~ k. Shown as Fig. 4.

According to the network input and output bits coding, you can distinguish the different types of Toffoli gate or homo-type Toffoli gate which from the same vertical line while matching a different horizontal line has the different meanings. For example, Fig. 5 indicates the distinguishing and coding of the three input / output reversible logic gate. A, B are not homo-type, so they represent the different meanings in the network, while C, D are the homo-type, but their coding are different, so they represent the different meanings. E, F are the homo-type and coding are the same, so they have the same meanings.

E. Reversible network constructing

Let network input vectors which from (0,0,0,...,0) to (1,1,1,...,1) and make the different meanings Toffoli gate gained by the way of constructing one vertical line and n horizontal lines coding as the basic elements.

Take three input / output network as an example as shown in Fig. 7, vertical line A has 0 as its target bit, 1 and 2 as its control bits, while vertical line B has 2 as its target bit, 0 and 1 as its control bits.
For \( n \) input / output network, the different output vectors are corresponding to the different network which has \( k \) \((k\geq0)\) vertical lines meanwhile every vertical line consists of the Toffoli gate. Therefore, as long as these Toffoli gates ordered in fixed sorting order, you can realize all \( n \) output vectors corresponding network.

Take three input / output as an example, according to the coding rules (vertical lines coding is not considered now): if a vertical line has 0 as its target bit, then the different meaning Toffoli gate have four cases: 0, 0 1, 0 2, 0 12; if a vertical line has 1 as its target bit, then the different meaning Toffoli gate have four cases: 1, 1 0, 1 2, 1 02; if a vertical line has 2 as its target bit, then the different meaning Toffoli gate have four cases: 2, 2 0, 2 1, 2 01. All of these 12 cases mentioned above become the basic elements of the three input / output network. As shown in Fig. 8.

Different output vectors network is the combination of these basic elements. The combinations of the basic elements from different vertical lines form the corresponding network basic states. The corresponding network may be constructed via all the network basic states combination which from the different vertical lines.

Take three input / output network as an example, as shown in Fig. 9, if 12 kinds of basic elements (also known as the basic states of network with only one vertical line) which from a vertical line are performed completely, obtaining a group of different output vectors. However, it is obvious that these vectors are not all the three input / output network output vectors. Another vertical line is added into network, then the two vertical lines can construct \( 12 \times 12 \) kinds different combinations (known as 144 different network basic states). If you have not yet obtained all of the output vectors, then continue to add the vertical lines into network and generate corresponding network basic states until obtain all the output vectors.

Different output vector reversible network constructing algorithm:

1. Network initialization (constructing \( n \) parallel lines without the Toffoli gate).
2. To add a vertical line into the network.
3. Constructing the network basic states collection
4. To initialize vertical lines (construct vertical blank lines).
5. Call a network basic state from the collection of network basic states and add the Toffoli gates represented by this basic state into vertical lines.
6. Determine whether the network generated exist or not? If not exist, then save the relevant network results; otherwise discard the relevant network results.
7. Determine whether all of the output vectors are obtained, if it is, then go to (9). Otherwise go to (4).
8. Determine whether the network basic states of the collection be invoked completely, if it is, then go to (2). Else go to (4).
9. End.

F. The indication of output vector corresponding to network

Because all the output vectors are known, in order to indicate the output vectors corresponding network, this paper uses the sequencing output Toffoli gate coding which from every vertical line to indicate the network consists of Toffoli gate under any a output vectors. The following Fig. 10 may be expressed as ((0 1), (1 2), (2 01)).

The algorithm is in accordance with vertical lines increased one by one and the basic network elements which from the every vertical line is added into the network sequentially. Therefore if there is a new network output vector and the output vector has no matching corresponding output vector of the output vector collection, then output its corresponding network. If obtain the matching output vector later in the output collection, then it’s vertical lines number must much more than or be equal to the output vector corresponding number of vertical lines. The combination method has realized the smallest degree of reduction numbers of the network Toffoli gates.

III. EXPERIMENT AND RESULT ANALYSIS

For the effectiveness of testing of the synthetic approach provided by this paper, we use C++ programming language to implement the algorithm described in this paper. Experiments can be performed under the condition: Pentium4 3.0G PC (512M RAM), Windows XP environment.
Adopt three variables reversible function testing standards which is recognized by the international craft brother to do the experiment, the purpose of the experiment is to find a different output vector corresponding $2^3 = 8! = 40320$ reversible logic circuits which from completing set. Fig. 11 selects 30 comprehensive results which from previous, middle and after in output vectors collection.

While constructing network, every Toffoli gate control bits are ascending orderly from small to large, so it is easy to realize the circuit and costs are much lower, among the three comparable input / output Benchmark examples, our control bits number are less than or equal to the corresponding standards. And Ham3 use the exchanging gate, its structure is much more complicated than our results. As shown in Table 1.

<table>
<thead>
<tr>
<th>Name</th>
<th>Benchmark Gates</th>
<th>CBits</th>
<th>We Gates</th>
<th>CBits</th>
</tr>
</thead>
<tbody>
<tr>
<td>3_17</td>
<td>6</td>
<td>7</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>Ham3</td>
<td>4</td>
<td>5</td>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>Ham3_2</td>
<td>5</td>
<td>6</td>
<td>5</td>
<td>6</td>
</tr>
</tbody>
</table>

IV. CONCLUSION

This paper proposed the reversible network cascade method based on reversible logic gate encoding number. The method is referred to the number of vertical lines from small to large, control bits from small to large to construct automatically the different output vectors corresponding the network algorithm, which can generate all the reversible network. Compared with three variable reversible function testing standards recognized by international craft brother, the control gates are much less, the cost are much lower. How to integrate a large-scale reversible network and further improve the efficiency of cascade is our next step research important topic.

ACKNOWLEDGMENT

This work is supported by the National Natural Science Foundation of China under grant No. 60873069, and Technology Research and Development Program of Nantong University under grant No.06Z053.

REFERENCES