Genetic Algorithms for Quantum Circuit Design
–Evolving a Simpler Teleportation Circuit–

Taro Yabuki
yabuki@miv.t.u-tokyo.ac.jp
Department of Frontier Informatics,
Graduate School of Frontier Sciences,
The University of Tokyo, Japan

Hitoshi Iba
iba@miv.t.u-tokyo.ac.jp
Department of Frontier Informatics,
Graduate School of Frontier Sciences,
The University of Tokyo, Japan

Abstract

We propose a method to apply genetic algorithms to the quantum circuit design. We show by experiments that without any deep knowledge of the problem it is possible to evolve a circuit for the quantum teleportation simpler than ever known.
keyword: genetic algorithms, quantum teleportation, quantum computer, quantum computing, quantum circuit.

1 Introduction

1.1 The history of quantum computation

The notion of a quantum computer was devised by Benioff in 1980. It was based upon the research trend toward a high-density chip and the research of thermodynamics of computation. Feynman thought that the quantum computer is suitable for the simulation of quantum mechanical phenomena. In 1985, Deutsch formularized a quantum Turing machine (Deutsch, 1985). However, there was no practical task for which the quantum computer surpassed the classical one. As a result, the research was piled up.

Since Shor discovered how to factorize a large integer on the quantum computer in polynomial time in 1994 (Shor, 1994), this field has attracted much attention from researchers. Because the reliability of the public key cryptography used currently is grounded on the difficulty to factorize a large integer.

Various interdisciplinary researches between physics and information science have been made since then[A].

[A] For beginners of this field, refer to an introductory review (Steane, 1998). You can get many papers from http://xxx.lanl.gov/archive/quant-ph, and information from http://www.qubit.org/

1.2 Why genetic algorithms?

Only a few quantum algorithms that probably surpass the classical one have been discovered since Shor. This is partly because we are not used to the quantum computation.

A quantum circuit consists of a set of unitary transformations and measurements that execute a quantum algorithms. There are many difficulties in designing a quantum circuit as follows:

- We do not know how to design a quantum circuit to solve a given problem.
- Even though we know the target unitary transformation, it is not necessarily known how to compose it of primitive unitary transformations.
- Even if we can compose a quantum circuit, there is no yardstick for judging its efficiency.
- Because there is little knowledge about a search space, we cannot predict the effect of local change on the circuit property.

To avoid these difficulties, genetic algorithms (GAs) are suitable to the quantum circuit design. In using genetic algorithms, we can start with random candidates, and we need only an evaluation of the entire circuit. Because it is difficult to keep a superposition of plural states long, a quantum bit (qubit) cannot pass through many gates. Therefore, a simpler circuit is desirable. We can also make a circuit simpler and easy to implement. For instance, Williams and Gray made a circuit simpler than ever known by using GA (Williams, 1999), and Spector et al. used GP to design a circuit more efficient than the classical computer currently used (Spector, 1999).

1.3 Goal of this paper

Motivated by the above-mentioned previous studies, we apply genetic algorithms to designing a quantum teleportation circuit. Quantum teleportation is a typical example of quantum computing. In the past study, a circuit was evolved when a correct man-made specification was provided. We empirically show that only essential conditions and desirable outputs of a circuit are enough to evolve a correct and simpler circuit.

2 Quantum teleportation

Quantum teleportation is a scheme by which the state of a qubit can be transported from one point to another by communicating just two classical bits.

Suppose that Alice wants to transmit the information as to the 1-qubit state, i.e., $|f\rangle = a|0\rangle + b|1\rangle$, to Bob in a distant place. If she knows 'a' and 'b', she can transmit them in a classical way. At first sight, if she does not know 'a' and 'b', she can send the message by no means except when she carries a qubit directly. Because if she measures the qubit $|f\rangle$, the state changes to either a state $|0\rangle$ or a state $|1\rangle$, and because she cannot copy the state [B]. However, the quantum teleportation by Bennett et al. enables that (Bennett, 1993).

2.1 The idea of quantum teleportation

The idea of quantum teleportation is described in fig.1. The procedure is given below:

1. Make an EPR-pair. Send one of them to Alice and the other to Bob.
2. Alice entangles the received qubit and her own qubit. Thus all qubits are entangled.
3. Alice measures two qubits that belong to her. This measurement also has an effect on the qubit that belongs to Bob.
4. When Alice informs Bob of the measurement result, he can restore the information by a proper operation.

[B] Assume that there exists a copy gate C which works as $C|a\rangle|0\rangle = |a\rangle|a\rangle$ and $C|b\rangle|0\rangle = |b\rangle|b\rangle$. However, since

$$\begin{aligned} C(\alpha|a\rangle| + \beta|b\rangle)0\rangle &= \alpha|a\rangle|a\rangle + \beta|b\rangle|b\rangle \\ &\neq (\alpha|a\rangle| + \beta|b\rangle)(\alpha|a\rangle| + \beta|b\rangle), \end{aligned}$$

it is inconsistent with the first assumption. Thus, there can be no copy gate.

In the first step, an EPR-pair $(|00\rangle + |11\rangle)$ is made[C]. Then, the zeroth (right) qubit is sent to Bob, and the first (left) qubit to Alice. We use the second qubit to represent the qubit that Alice originally had. At this point, the state is described as

$$\begin{aligned} &(a|0\rangle + b|1\rangle) \otimes (|00\rangle + |11\rangle) \\ &\qquad = a|000\rangle + b|100\rangle + a|011\rangle + b|111\rangle. \end{aligned}$$

After Alice operates CNOT_{21}, and H_1 [D], and measures the first and the second qubits, the state becomes one of the four states, i.e., $|00\rangle(a|0\rangle + b|1\rangle)$, $|01\rangle(a|1\rangle + b|0\rangle)$, $|10\rangle(a|0\rangle - b|1\rangle)$, or $|11\rangle(a|1\rangle - b|0\rangle)$. For example, if she measures 11 and informs Bob of the result, he knows that his own qubit is $(a|1\rangle - b|0\rangle)$. Then, he can restore the state $|f\rangle = a|0\rangle + b|1\rangle$, i.e., Alice's original state, by applying the operation $\begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$. When she gets a different result, the same type of operation can be used to restore the original information. Note that Alice's original state is destroyed and that Bob gets the same state. Thus, this is regarded as a teleportation. Alice informs Bob of her measurement result in a classical way (i.e., the speed is less than that of light). Therefore, the speed of this teleportation is slower than the light speed.

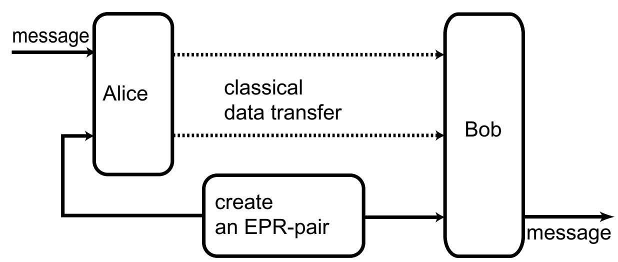

Figure 1: Idea of quantum teleportation.

2.2 Circuit details

Brassard proposed a concrete circuit (fig.2) to realize the idea described above (Brassard, 1996). In fig.2, L, R, S, T, and M mean $\frac{1}{\sqrt{2}}\begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix}$, $\frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ -1 & 1 \end{pmatrix}$, $\begin{pmatrix} i & 0 \\ 0 & 1 \end{pmatrix}$, $\begin{pmatrix} -1 & 0 \\ 0 & -i \end{pmatrix}$, and the measurement, respectively. Note that there are 11 gates in this circuit. As

[C] In this paper, we omit a normalization factor.

[D] CNOT_{21} is a controlled-NOT gate which has the second qubit as its target and the first qubit as its control. H_1 is a gate which operates $\mathrm{H} = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$ on the first qubit.

we shall see later in section 4, the circuit for the above teleportation can be realized with only eight gates.

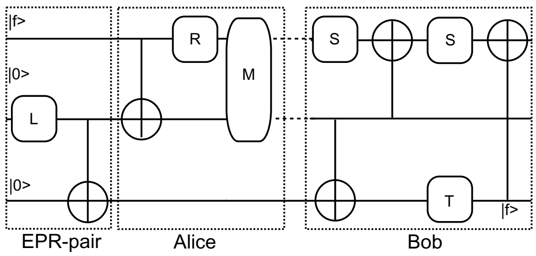

Figure 2: Brassard's circuit (Brassard, 1996).

3 Proposed method

3.1 Assumptions

We compose a circuit under the following conditions:

- Circuit Features
  1. 3-qubit system.
  2. In EPR-pair generation, only the zeroth and the first qubits can be operated.
  3. Alice can operate only the first and the second qubits.
- Search heuristics
  1. The measurement is allowed only once. Alice measures the first and the second qubits.
  2. A gate set of $\{\mathrm{CNOT}, \mathrm{L}, \mathrm{R}\}$ is chosen[E].

3.2 Gene representation

We use a one-dimensional gene which is a fixed length arrangement of letters. The letter set is {0,1,2,3}. An example of gene is decribed below:

$$\overbrace{112|231|001|\underline{\mathbf{3}}31|}^{\text{EPR-pair}}\overbrace{132|012|221|\underline{\mathbf{3}}02|}^{\text{Alice's part}}\overbrace{001|100|002|201}^{\text{Bob's part}}$$

Each gene is interpreted with a codon, i.e., a three-letter unit. The first letter indicates a kind of gate, whereas the second and the third letters indicate qubits that will be operated. The first codon whose first letter is 3 is interpreted as the partition between EPR-pair generation and Alice's part. The second codon whose first letter is 3 is used for the partition between Alice's and Bob's part. In other words, this corresponds to Alice's measurement. The relationship between a codon and a gate depends on which region the codon is in.

[E] This gate set is not for the universal quantum Turing machine. It followed Williams and Gray's experimental setting described later.

	0	1	2	3	
0	CNOT_{01}	CNOT_{10}			0
	CNOT_{01}	CNOT_{10}			1
	CNOT_{01}	CNOT_{10}			2
					3
1	L_0	L_1			0
	L_0	L_1			1
	L_0	L_1			2
					3
2	R_0	R_1			0
	R_0	R_1			1
	R_0	R_1			2
					3
3	separator				*

Table 1: Genetic code for the EPR-pair generation.

We explain how each gene is interpreted by using the above example. The forth codon is the partition between EPR-pair generation and Alice's part. The next codon that starts with 3 is the eighth one, i.e., 302. It is interpreted as Alice's measurement. The codons after the measurement are interpreted as Bob's part.

	0	1	2	3	
0		CNOT_{12}	CNOT_{21}		0
		CNOT_{12}	CNOT_{21}		1
		CNOT_{12}	CNOT_{21}		2
					3
1		L_1	L_2		0
		L_1	L_2		1
		L_1	L_2		2
					3
2		R_1	R_2		0
		R_1	R_2		1
		R_1	R_2		2
					3
3	measurement				*

Table 2: Genetic code for Alice's part.

In the EPR-pair generation, codons are interpreted in accordance with table 1. For example, 112 is decoded as L_1, 231 as nothing, and 001 as CNOT_{01}. In Alice's part, table 2 is used, whereas Bob uses the rule described in table 3. For example, in Alice's part, 132 is decoded as nothing, 012 as CNOT_{12}, 221 as R_2, and 302 as measurement. As a result of this interpretation, the above gene gives the circuit shown in fig. 7.

	0	1	2	3	
0		CNOT$_{10}$	CNOT$_{20}$		0
	CNOT$_{01}$		CNOT$_{21}$		1
	CNOT$_{02}$	CNOT$_{12}$			2
					3
1	L$_0$	L$_1$	L$_2$		0
	L$_0$	L$_1$	L$_2$		1
	L$_0$	L$_1$	L$_2$		2
					3
2	R$_0$	R$_1$	R$_2$		0
	R$_0$	R$_1$	R$_2$		1
	R$_0$	R$_1$	R$_2$		2
					3
3					*

Table 3: Genetic code for Bob's part.

3.3 Fitness definition

Each individual is evaluated in the following ways:

1. Make three random numbers, i.e., $\alpha, \beta, \gamma \in [0, 2\pi]$.
2. Prepare three initial states (p, q), i.e., $(e^{i\beta}\cos\alpha, e^{i\gamma}\sin\alpha)$, $(e^{i\gamma}\cos\beta, e^{i\alpha}\sin\beta)$, and $(e^{i\alpha}\cos\gamma, e^{i\beta}\sin\gamma)$.[F]
3. Derive a circuit and its unitary transformation by using the tables described in section 3.2.
4. Operate the unitary transformation or the measurement on the states $(p|0\rangle + q|1\rangle) \otimes |0\rangle \otimes |0\rangle = (p, 0, 0, 0, q, 0, 0, 0)$ in order.
5. If a circuit outputs a final state similar to the desirable one, its fitness is enlarged.
6. Change random numbers every 50 generations.

If we measure the first and the second qubits of the state (a_i), the state will become one of $(a_0, a_1, 0, 0, 0, 0, 0, 0)$, $(0, 0, a_2, a_3, 0, 0, 0, 0)$, $(0, 0, 0, 0, a_4, a_5, 0, 0)$, and $(0, 0, 0, 0, 0, 0, a_6, a_7)$. We trace all branches.

The desirable final state is $(a|00\rangle + b|01\rangle + c|10\rangle + d|11\rangle) \otimes (p|0\rangle + q|1\rangle) = (ap, aq, bp, bq, cp, cq, dp, dp)$. Note that the ratio of two components in order is $p : q$. Each individual makes 12 final states, i.e., there are three initial states, and each one diverges to four states by the measurement. We use a suffix j to indicate each final state, namely, we write the final state $\boldsymbol{a}_j = (a_{j,i})$, where, $j = 1, \cdots, 12$ and $i = 0, \cdots, 7$. The gap of the final state and the desirable one is expressed as

$$error_j \equiv \frac{1}{n} \sum_{i=0,2,4,6} \left| \frac{a_{j,i}}{a_{j,i+1}} - \frac{p}{q} \right|.$$

Where n is the number of a pair $(a_{j,i}, a_{j,i+1})$ $(i = 0, 2, 4, 6)$ that is not $(0, 0)$ and the summation is taken over such pairs. If the final state is $\mathbf{0}$, then $error_j = 100$. If the final state is equal to the desirable one, then $error_j = 0$.

The fitness f of an individual is defined as $f = 1/(1 + 10 \sum error_j)$. However, if f is 1, i.e., if the circuit is correct, the bonus of (1/(gate number)) is added to f, so as to apply a selection pressure based upon the circuit size.

3.4 GA parameters

After initializing all individuals randomly, we use the sigma scaling, i.e., $f' = f - (\bar{f} - 2\sigma)$. We use the roulette-wheel selection. Two-point crossover is adopted with probability 0.7. The mutation probability is (1/(gene length)). All 5,000 individuals are replaced every generation up to 1,000 generations. We conduct experiments ten times for various gene lengths with different random seeds.

4 Results

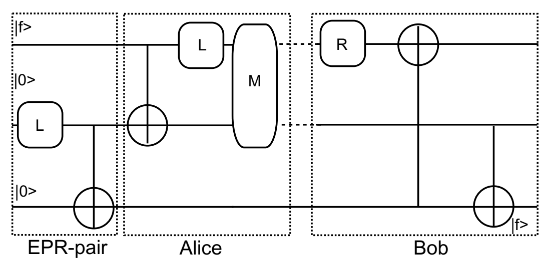

Figure 3: Evolved circuit for teleportation.

The simplest circuit evolved is shown in fig. 3. This circuit consists of eight gates, while Brassard's circuit has eleven gates (see fig. 2). Actually, it is easily verified that if Alice measures $|00\rangle$, $|01\rangle$, $|10\rangle$, and $|11\rangle$, the final state is $(|00\rangle - |10\rangle)(p|0\rangle + q|1\rangle)$, $(|11\rangle - |01\rangle)(p|0\rangle + q|1\rangle)$, $(|00\rangle + |10\rangle)(p|0\rangle + q|1\rangle)$, and $(|01\rangle + |11\rangle)(p|0\rangle + q|1\rangle)$, respectively. Thus, we can confirm that the state of the second qubit is teleported to the zeroth qubit.

Fig. 4 plots the relation between the gene length and the number of gates in the evolved circuit[G]. The

[F] We use three initial states to make $error_j$ independent from random numbers.

[G] Error bars show the standard deviation.

shorter the gene length becomes, the simpler the evolved circuit is. However, fig. 5 indicates that if the gene length becomes short, the success probability is also reduced.

Fig. 6 shows the relation between the gene length and the generations when the correct circuits are evolved. When the gene is 120-letter length, i.e., it can encode up to 40 gates, the correct circuits are evolvable in the earlier generations. Even if we consider the computational burden to evaluate a longer gene, using a 120-letter gene seems to be the most effective. However fig. 4 shows that if we use a 120-letter gene, we cannot necessarily evolve a simple circuit.

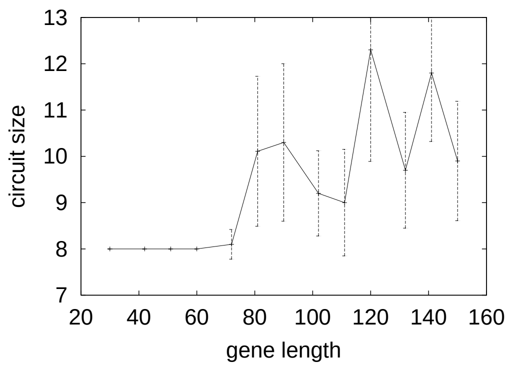

Figure 4: Number of gates in evolved circuit.

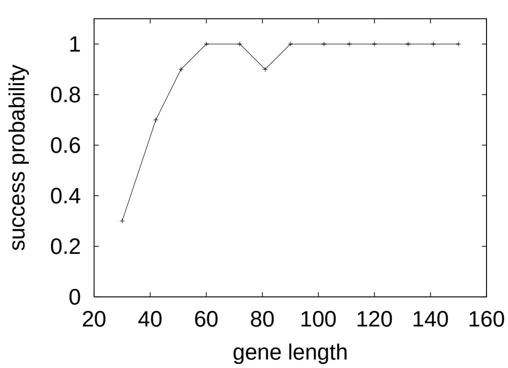

Figure 5: Probability of success.

5 Discussion

5.1 Search efficiency

When we use a 60-letter gene, i.e., the circuit size is restricted up to 20 gates, we always have found the simplest (i.e., 8-gate) circuit. This circuit was found around 350 generations on the average (the standard deviation is 280). Before the circuit was evolved,

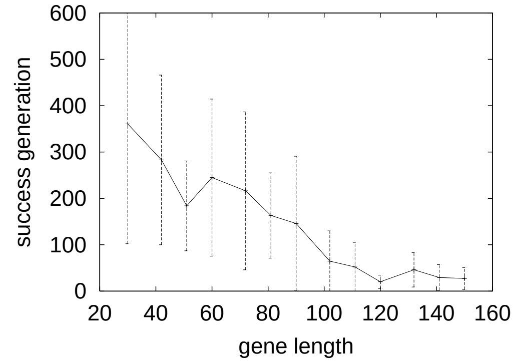

Figure 6: The generation of success.

$5,000$ individuals $\times$ 350 generations $\sim 1.8 \times 10^6$ candidates had been generated. On the other hand, there are 14 kinds of different gates[H] and the number of possible circuits is $20^{14} \sim 10^{18}$. Since the density of solutions in the search space is not necessarily known, we should not compare these numbers directly. But the above search result looks promising for evolving a quantum circuit.

5.2 Previous works

Williams and Gray also conducted an experiment in evolving a teleportation circuit by using GA (Williams, 1999). Note that the circuit shown in fig. 2 consists of the following three parts (the first and the third parts represent unitary transformations):

1. $\mathrm{L}_1, \mathrm{CNOT}_{01}, \mathrm{CNOT}_{21}$, and R_2
2. measurement
3. $\mathrm{S}_2, \mathrm{CNOT}_{01}, \mathrm{CNOT}_{12}, \mathrm{S}_2, \mathrm{T}_0$, and CNOT_{20}

For example, the matrix of the first unitary transformation is,

$$U = \frac{1}{2}\begin{pmatrix} 1 & 0 & -1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & -1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 & -1 & 0 \\ 1 & 0 & -1 & 0 & 0 & 1 & 0 & 1 \\ -1 & 0 & -1 & 0 & 0 & 1 & 0 & 1 \\ 0 & -1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & -1 & 0 & -1 & 1 & 0 & -1 & 0 \\ -1 & 0 & -1 & 0 & 0 & 1 & 0 & -1 \end{pmatrix}.$$

This is easily verified by composing unitary transformations correspondent to the gates shown in fig. 2. They used a gene to represent a row of primitive unitary transformations, i.e., $\{\mathrm{CNOT}, \mathrm{L}, \mathrm{R}\}$. Let S be the

[H] There are six kinds of CNOT gates, three kinds of L and R gates, the measurement and the separator.

matrix expressed by the gene. They defined a fitness function f as follows:

$$f(S,U) \equiv \sum_{i=1}^{8}\sum_{j=1}^{8} |U_{ij} - S_{ij}|.$$

Note that small f means a large probability to survive.

Williams and Gray reported that they rediscovered the same circuit for the first part, and discovered a smaller circuit with only four gates for the third part. Fig. 7 shows the circuit evolved by their method. It contains nine gates and is simpler than the one by Brassard.

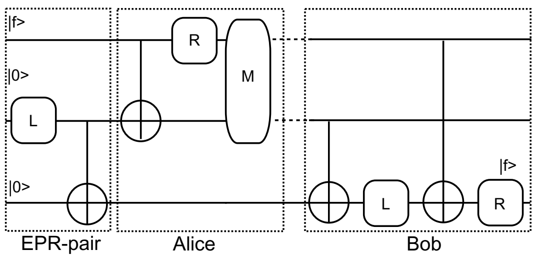

Figure 7: Williams's circuit (Williams, 1999).

However, this result is not satisfactory for the following reasons. First, they gave the target unitary transformation for the training. It was assumed that for the input $(p|0\rangle + q|1\rangle) \otimes |0\rangle \otimes |0\rangle$, the circuit should output $(a|00\rangle + b|01\rangle + c|10\rangle + d|11\rangle) \otimes (p|0\rangle + q|1\rangle)$. But we cannot compose the unitary transformation only by using this knowledge. Thus we need heuristics to obtain a whole unitary transformation. Besides, it was impossible to discover a much simpler circuit because they gave the target unitary transformation. The difference between our result and theirs is that the first and the second qubits in output are entangled.

Second, this method could compose a circuit shown in fig. 8 for the first part. But this circuit violates the second and the third conditions described in section 3.1. After the second qubit is operated, i.e., after Alice's operation begins, it is not allowed to operate the zeroth qubit. This is because the zeroth qubit belongs to Bob at that time [I] .

6 Conclusion

We devised an evolutionary method to design a quantum circuit and applied it to a quantum teleportation circuit design. We used only essential restrictions and assumptions for this evolution. As a result of experiments, we have confirmed that a correct circuit is evolvable which is simpler than ever known.

[I] In fact, it is only Brassard's circuit that composes the unitary transformation with less than four gates for the first part. Thus, it was quite natural for them to discover the correct circuit when they limited the circuit size.

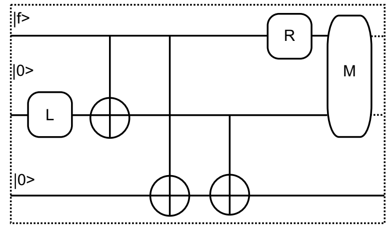

Figure 8: Inappropriate circuit.

References

[1] C. H. Bennett et al. Teleporting an unknown quantum state via dual classical and einstein-podolsky-rosen channels. *Phys. Rev. Lett.*, 70:1895, 1993.

[2] G. Brassard. Teleportation as a quantum computation. In *Proceedings of the Fourth Workshop on Physics and Computation*, page 44. New England Complex Systems Institute, 1996. quant-ph/9605035.

[3] D. Deutsch. Quantum theory, the church-turing principle and the univarsal quantum computer. In *Proceeding Royal Society London*, volume A400, page 97, 1985.

[4] P. Shor. Algorithms for quantum computation: Discrete logarithms and factoring. In *Proceedings of the 35th IEEE Symposium on Fundation of Computer Science*, page 124, 1994.

[5] L. Spector et al. Finding a better-than-classical quantum and/or algorithm using genetic programming. In *Proceedings of the 1999 Congress on Evolutionary Computation*. IEEE Press, 1999.

[6] A. Steane. Quantum computing. *Reports on Progress in Physics*, 61:117, 1998. quant-ph/9708022.

[7] C. P. Williams and A. G. Gray. Automated design of quantum circuits. In *QCQC'98 LNCS 1509*, page 113. Springer-Verlag, 1999.