Hindawi Publishing Corporation
Modelling and Simulation in Engineering
Volume 2014, Article ID 571374, 8 pages
http://dx.doi.org/10.1155/2014/571374

Research Article

Optimal ILP-Based Approach for Gate Location Assignment and Scheduling in Quantum Circuits

Naser Mohammadzadeh,[1] Tayebeh Bahreini,[1] and Hossein Badri[2]

[1] *Computer Engineering Department, Shahed University, Tehran, Iran*
[2] *Department of Industrial Engineering, Amirkabir University of Technology, Tehran, Iran*

Correspondence should be addressed to Tayebeh Bahreini; t.bahreini@shahed.ac.ir

Received 7 July 2013; Revised 18 November 2013; Accepted 19 November 2013; Published 4 February 2014

Academic Editor: Azah Mohamed

Physical design and synthesis are two key processes of quantum circuit design methodology. The physical design process itself decomposes into scheduling, mapping, routing, and placement. In this paper, a mathematical model is proposed for mapping, routing, and scheduling in ion-trap technology in order to minimize latency of the circuit. The proposed model which is a mixed integer linear programming (MILP) model gives the optimal locations for gates and the best sequence of operations in terms of latency. Experimental results show that our scheme outperforms the other schemes for the attempted benchmarks.

1. Introduction

Quantum effects have been a major concern in classical computers in metal-oxide-semiconductor technology (CMOS) as the feature size shrinks into the 10 s of nanometers range [1]. Quantum effects such as entanglement and superposition are amplified in quantum computers. They operate on the entangled superposition states and this is where the power of quantum algorithms comes from.

A quantum circuit is a model for quantum computation. In a large picture, the quantum circuit design flow includes two main tasks: synthesis and physical design (Figure 1). The physical design consists of scheduling, mapping, and placement processes. In this paper, an ILP-based approach is proposed for scheduling, routing, and mapping processes. The proposed approach takes an initial netlist and a layout, and maps and schedules the gates on the layout.

A quantum circuit is defined as a sequence of quantum operations acting on one or multiple qubits. These operations could be categorized in two groups: (1) single or multiqubit logical gates and (2) single qubit measurement. In this paper, only one- and two-qubit operations are considered due to some practical limitations on many quantum circuit technologies [2].

Ion-trap technology is a quantum technology where every universal element for quantum computation has been realized with a clear scalable communication model [3, 4]. In this technology, an ion is the physical demonstration of a qubit and a gate location is a location wherein a gate is performed. Each ion could be trapped or physically moved between traps by applying pulse sequences to discrete electrodes (Figure 2). Each qubit is measured by stimulating the target ion with a different frequency laser pulse [5].

In this paper, library of macroblocks which is defined in [6] (Figure 3) is used due to two major advantages. The main reason is that by using this library, some low level details could be removed and it is not necessary to consider the variations in ion types, size of electrodes, and precise voltage levels needed for trapping and moving ions. All of these details are condensed within the macroblocks [7, 8].

In this library, a 3×3 structure of trap regions and electrodes forms each macroblock. Each structure has some ports to allow qubits to move between the macroblocks. Gate locations are indicated by black squares. Various orientations of each macroblock could be used in a layout.

The paper is continued as follows: an overview of the prior work is presented in Section 2, followed by the details of the proposed model in Section 3. An illustrative example

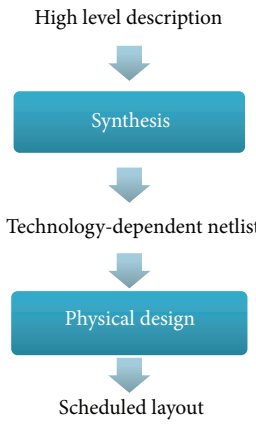

Figure 1: Quantum circuit design flow [9].

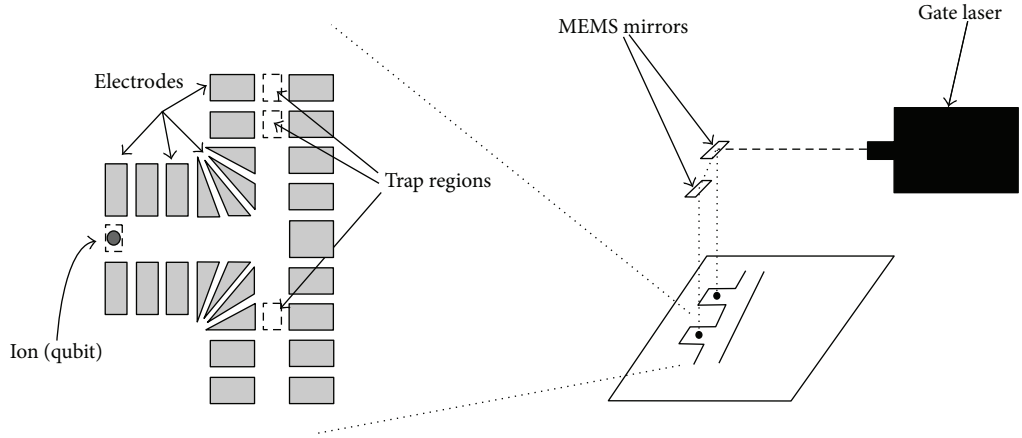

Figure 2: Simplified ion-trap technology view [5].

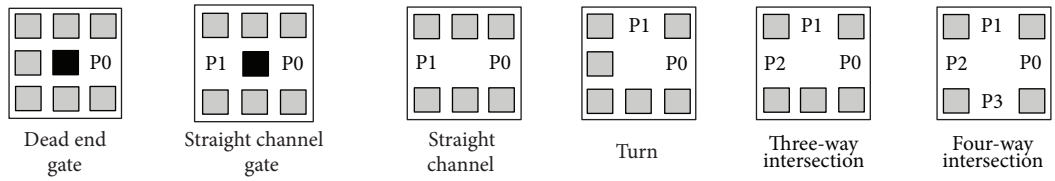

Figure 3: Library of basic macroblocks used in this paper [6].

is presented in Section 4. Section 5 shows the experimental results, and Section 6 concludes the paper.

2. Related Work

There are a lot of techniques proposed for optimization in synthesis process [6, 10–14], but a few studies have been done on optimization problems in the quantum physical stage.

Metodi et al. [15] presented a physical operations scheduler (QPOS) which takes a description of a circuit and a definite physical layout and produces a sequence of operations which indicates the required communication and possible parallelism in the circuit. Dousti and Pedram [16] presented an algorithm for scheduling, placement, and routing of a quantum circuit which decreases the circuit latency. They developed a heuristic for placement called MVFB which improved the center placement used in the previous quantum CAD tool [17, 18]. Balensiefer et al. [17, 18] proposed a design flow and compilation technique to consider fault-tolerance and developed some tools to appraise layouts. Whitney et al. [9] proposed a computer-aided design flow for the layout, scheduling, and control of ion-trap-based quantum circuits. They proposed two heuristics for layout design. The first heuristic is a greedy algorithm that is appropriate for small circuits. Dataflow-based algorithm was proposed for larger circuits aiming at placement and routing.

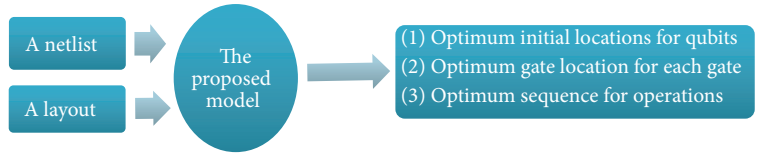

Figure 4: Inputs and outputs of the proposed model.

Maslov et al. [19] proposed heuristics for the mapping of quantum circuits onto molecules used in liquid state NMR quantum computing technology. Their algorithm starts with a molecule to be used for computation, modeled as a weighted graph with edges representing atomic couplings within the molecule. The dataflow graph of the circuit is mapped onto the molecule graph with an effort to minimize overall circuit latency.

Mohammadzadeh et al. [8] proposed an optimization technique applying gate location changing (GLC) to reduce the latency of quantum circuits. The proposed technique finds critical paths by the use of layout and scheduling information and reduces their latency by modifying locations of the gates on the critical path. These authors gave an introduction to the physical synthesis concept in quantum circuits [20–22] which includes techniques that change the netlist using layout information to improve latency and/or area as the main performance indicators of the circuits.

Yazdani et al. [23] presented a physical design flow for quantum circuits in ion-trap technology which consists of two parts. First, a scheduler takes a description of a circuit and finds the best order for the execution of its quantum gates using ILP. Then a layout generator receives the schedule produced by the scheduler and generates a layout for this circuit using a graph-drawing algorithm [24]. This work uses ILP only to schedule quantum circuits.

To the best of our knowledge, there is no ILP model in the literature in which mapping, scheduling, and routing are considered simultaneously. Focusing on this issue, in this paper, a model is proposed for them in the physical design stage. The proposed model gives the optimal mapping (i.e., locations of the gates) as well as the optimal routing and scheduling in terms of latency.

3. The Proposed Approach

In this section, a mixed integer linear programming model is presented for scheduling, routing, and mapping of quantum circuits. This model is proposed to minimize latency of a quantum circuit. It takes an initial netlist and a layout as inputs and finds the optimum initial locations for qubits, optimum gate location for each gate, and the optimum sequence of operations in such way that the latency of the circuit is minimized (Figure 4).

Suppose that each gate has a processing time ($\mathbf{p}_i$), where $i \in \{1, 2, \ldots, n\}$. For one-qubit gates, the processing time is 1 μs and for two-qubit gate is 10 μs as mentioned in Table 1. Each gate has a starting time ($\mathbf{S}_i$) that is a variable. Due to qubit congestion and gate dependencies, all gates cannot execute concurrently. Obviously, the finishing time for each gate would be $\mathbf{S}_i + \mathbf{p}_i$.

Table 1: Latency values for various physical operations in ion-trap technology [25].

Physical operation	Latency (μs)
One-qubit gate	1
Two-qubit gate	10
Straight movement	1
Turn	10

Let $j \in \{1, 2, \ldots, n\}$ be a set of locations and each gate is located at one place. Parameter $d_{jj'}$ indicates the distance between location j and location j' and is calculated using values mentioned in Table 1. Strict Manhattan distance is not an accurate measure to calculate distance between locations, because preliminary studies in ion-trap technology have shown that turning corners and traversing intersections will be more time consuming than moving straight through a one-way channel [26]. Therefore, in this paper, a modified version of Manhattan distance is used that considers the latencies of both straight movement and turn in, calculating the distance between gate locations. The value of variable x_{ij} denotes location of each gate. Variable x_{ij} is 1 if gate i is located at location j.

Each gate may have two kinds of dependencies to other gates. If gate i must be executed before gate i' and it changes the value of qubit q and gate i' needs this qubit, therefore, gate i' has to wait until execution of gate i is finished. In the proposed model, set $y_{ii'}$ represents this type of dependency. The other kind of dependency which is represented by set $g_{ii'}$ occurs when both gate i and gate i'need qubit q in this case data congestion would occur. The binary variable $f_{ii'}$ is used to model the second type of dependency, if $f_{ii'} = 1$ it means that gate i is executed at first and if $f_{ii'} = 0$ it means that gate i' is executed at first. In the following formulation, M indicates a very big number. The proposed model is summarized as follows.

MILP Model:

$$\text{Minimize } t \tag{1}$$

subject to

$$S_i + P_i \le t, \quad \forall i \in \{1, \ldots, n\}, \tag{2}$$

$$S_i + P_i + d_{jj'} + \left(x_{ij} + x_{i'j'} - 2\right) \cdot M \le S_{i'}, \quad \forall \left(i, i'\right) \in y, \quad \forall j, j' \in \{1, \ldots, n\}, \tag{3}$$

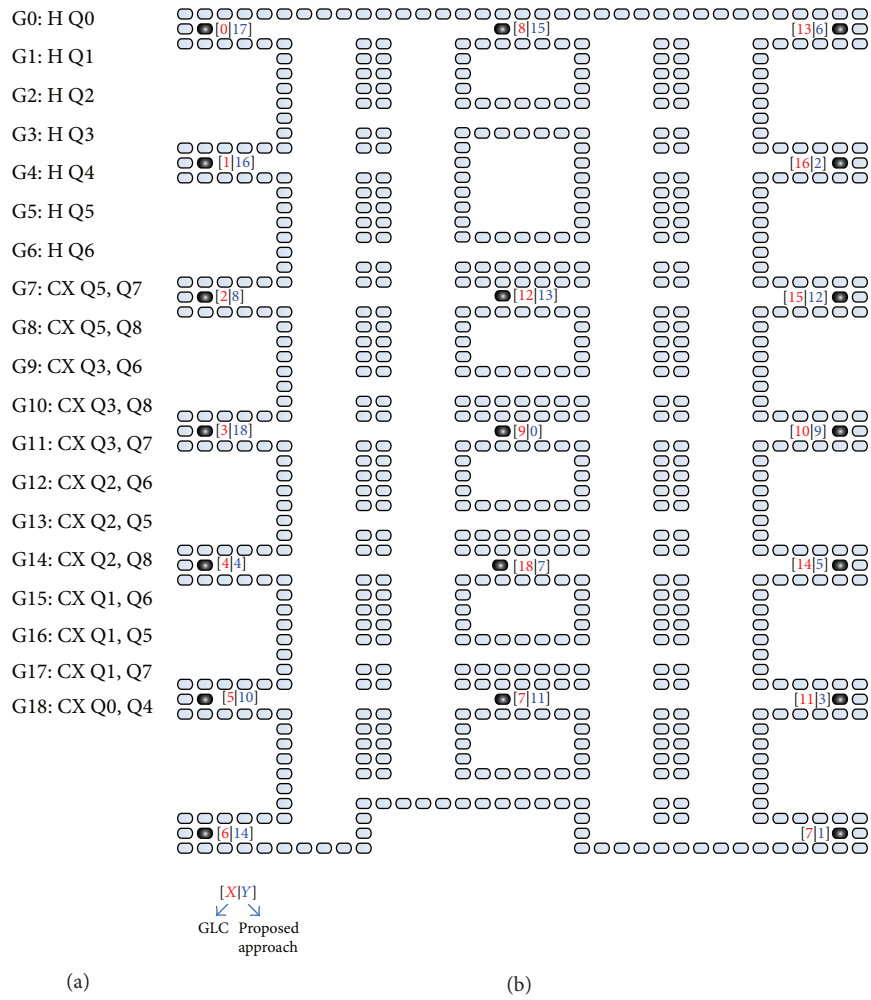

Figure 5: (a) A circuit netlist. (b) The gate locations assigned by GLC technique and the proposed approach.

$$S_i + P_i + d_{jj'} + \left(x_{ij} + x_{i'j'} - 2\right) \cdot M \leq S_{i'} + M \cdot f_{ii'}, \quad \forall \left(i, i'\right) \in g, \quad \forall j, j' \in \{1, \ldots, n\}, \tag{4}$$

$$S_{i'} + P_{i'} + d_{jj'} + \left(x_{ij} + x_{i'j'} - 2\right) \cdot M \leq S_i + M \cdot \left(1 - f_{ii'}\right), \quad \forall \left(i, i'\right) \in g, \quad \forall j, j' \in \{1, \ldots, n\}, \tag{5}$$

$$\sum_{j=1}^{n} x_{ij} = 1 \quad \forall i \in \{1, \ldots, n\}, \tag{6}$$

$$\sum_{i=1}^{n} x_{ij} = 1 \quad \forall j \in \{1, \ldots, n\}, \tag{7}$$

$$x_{ij} \in \{0, 1\}, \quad \forall i, j \in \{1, \ldots, n\}, \tag{8}$$

$$f_{ii'} \in \{0, 1\}, \quad \forall i, i' \in \{1, \ldots, n\}, \tag{9}$$

$$S_i \geq 0, \quad \forall i \in \{1, \ldots, n\}, \tag{10}$$

$$t \geq 0. \tag{11}$$

Formula (1) illustrates the objective—the maximum completion time (critical path)—to be minimized. Formula (2) defines the critical path as the maximum completion time of gates. Formula (3) illustrates priority dependencies. Formulas (4) and (5) are used to prevent congestions. Based on these two constraints, either gate i would be started first and gate i' starts after finishing time of gate i plus the transferring time of the common qubit or gate i' starts first and gate i starts after finishing time of gate i' plus the transferring time of the common qubit. Formula (6) ensures that each gate location is only assigned to one gate and (7) ensures that each gate is located in only one location. Constraints (9)-(10) are related to definition of variables.

4. An Example

To illustrate the idea of the proposed approach, an example is presented in this section. Figure 5(a) shows QASM [27] instruction sequence of circuit [9-1-3] [28]. The example has been solved by GLC technique [8] and the proposed approach. Physical latencies shown in Table 1 have been

Table 2: Benchmarks used for evaluation of the proposed model.

Benchmark	Number of gate	Number of qubit	Number of extended variables	Number of extended constraints
[6-0-2]	6	6	5	199
[4-2-2]	7	4	62	806
[5-2-2]	8	5	78	1049
[6-2-2]	10	6	116	1831
[5-0-3]	11	5	145	4148
[5-1-3]	12	5	156	5653
[ham3-D1]	12	3	189	11413
[ham3-D2]	13	3	203	16771
[mod5-D4]	14	5	239	13959
[1bitAdder-rd32]	16	4	274	35377
[7-1-3]	18	7	351	25651
[10-0-2]	18	10	342	17875
[9-1-3]	19	9	381	22440
[9-3-2]	20	9	408	28871
[7-0-3]	20	7	472	50461
[13-1-3]	39	13	1605	334738
[11-1-5]	47	11	4479	880661

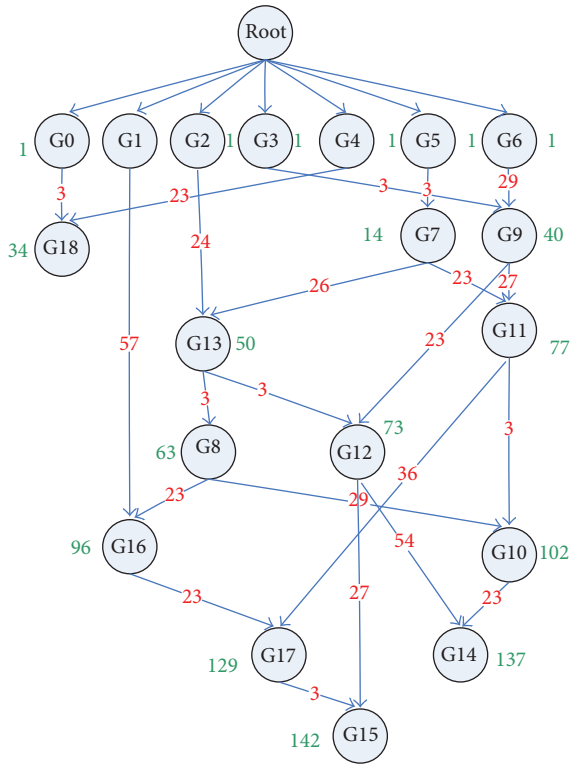

Figure 6: Dataflow graph of the example based on the proposed model.

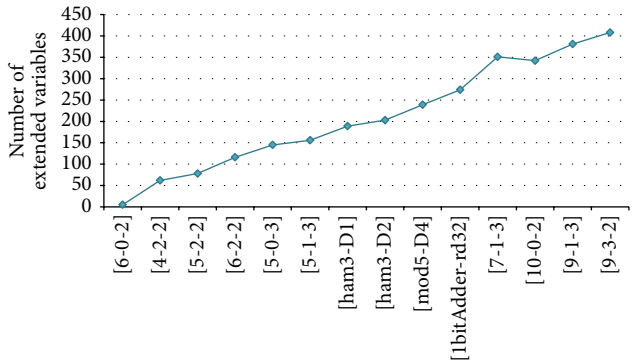

Figure 7: The number of extended variables in each class of benchmarks.

used to calculate operation latency. In this example, latency obtained by the GLC technique is 189 μs, while for the proposed model it is 142 μs. Figure 5(b) depicts the locations assigned to each gate by the GLC technique and the proposed model. The dataflow graph of the circuit which is attained by the proposed approach is shown in Figure 6.

5. Experimental Results

The computational analysis presented in this section is to evaluate the proposed model. Different benchmarks were selected from [28, 29]. The proposed model was fed with the data of these benchmarks and solved using the CPLEX MIP solver in GAMS tool [30]. The software was run on a Dual core 2.26 GHz processor with 2 GB of RAM. After solving the benchmarks, numbers of extended variables, number of extended constraints, and the latency were calculated. Table 2 shows structure of the benchmarks with the numbers of extended variables and number of extended constraints.

In MILP models, size of the model is a function of total number of extended variables and constraints. By increasing

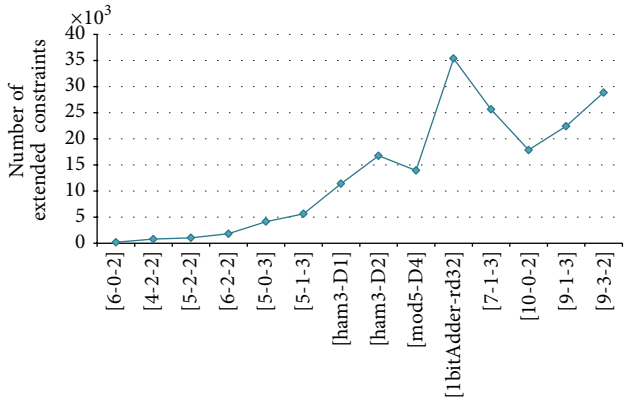

Figure 8: The number of extended constraints in each class of benchmarks.

Table 3: The latency of the benchmark circuits achieved by the proposed approach compared with [8].

Benchmark	Latency (μs)		Improvement (%)
	GLC [8]	The proposed model	
[4-2-2]	108	108*	0
[5-2-2]	96	76*	20.83
[5-0-3]	138	132*	4.34
[ham3-D1]	286	185	35.31
[ham3-D2]	323	230	28.79
[mod5-D4]	228	162	28.94
[1bitAdder-rd32]	310	277	10.64
[10-0-2]	283	175	38.16
[7-0-3]	288	252	12.50
[13-1-3]	417	314	24.70

*Optimal solution.

the number of gates as well as complexity of data flow graph, the number of extended variables (Figure 7) and constraints (Figure 8) are increased. The number of extended variables varies from 5 to 4479, while total number of extended constraints is between 199 and 880661.

To evaluate the proposed model, the latency of each benchmark is compared to results of the GLC technique [8] that is the best in the literature. Table 3 shows the computational results.

Experimental results show that the proposed model could find the minimum latency for small circuits and improve the latency of benchmarks up to 38%. For larger circuits, because of the exponential increase in the size of the problem, MILP solver cannot find the optimum solution, but it still improves the latency. Figure 9 depicts a comparable view on the results of the GLC algorithm and those of the proposed model for each benchmark.

As mentioned in Section 2, Yazdani et al. [23] proposed an ILP-based approach that uses ILP to schedule a circuit. To evaluate the effectiveness of our approach over theirs, a comparison has been done between the results obtained from

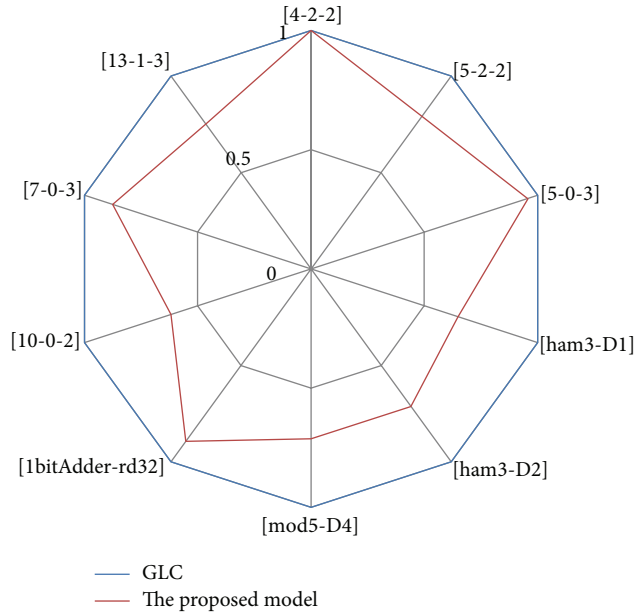

Figure 9: Improvement in the latency (μs) of the benchmarks achieved by the proposed model.

the proposed model, the ILP-based approach [23], and GLC method [8]. The computational results in Table 4 show the approach proposed in this paper that improves the average latency by about 25.5% and 36.1% compared with GLC and previous ILP-based approach, respectively. Figure 10 presents a comparable view on the results of the proposed model, GLC algorithm [8, 23].

In sum, the proposed model adopts an integrated approach in which gate location problem, routing, and scheduling are optimized simultaneously. The GLC technique [8] and the ILP-based model proposed in [23] are dominated by the proposed model in all attempted benchmarks. The GLC technique follows a hierarchal approach for gate location, routing, and scheduling of the circuits, while in [23] an ILP-based model is used to schedule a circuit and then the placement procedure is done on the scheduled circuit.

6. Conclusion

In this paper, a mathematical model for mapping, routing, and scheduling in ion-trap technology was proposed for minimizing the latency of the quantum circuit. The proposed model follows an integrated approach in which gate location assigning, routing, and scheduling are optimized simultaneously. An example was analyzed to illustrate the application of the proposed model. Moreover, some analysis was done on different size of benchmarks to evaluate the efficiency of the proposed model in comparison to the best work reported in the literature. The proposed model could offer a considerable amount of improvement in all attempted circuits. The results showed that it is possible to obtain optimal or near optimal solutions within a reasonable amount of time for small sizes instances. However, when the size of

Table 4: The latency of the benchmark circuits achieved by the proposed approach compared with [8, 23].

Benchmark	Latency (μs)			Improvement (%)	
	The proposed model	GLC [8]	Yazdani et al. [23]	Over GLC [8]	Over Yazdani et al. [23]
[6-0-2]	14	38	23	63.15	39.13
[6-2-2]	76	121	121	37.19	37.19
[5-1-3]	152	191	239	20.41	36.40
[7-1-3]	155	205	263	24.39	41.06
[9-1-3]	142	170	383	16.47	62.92
[9-3-2]	192	202	259	4.95	25.86
[11-1-5]	595	677	659	12.11	9.71
Average				25.5	36.1

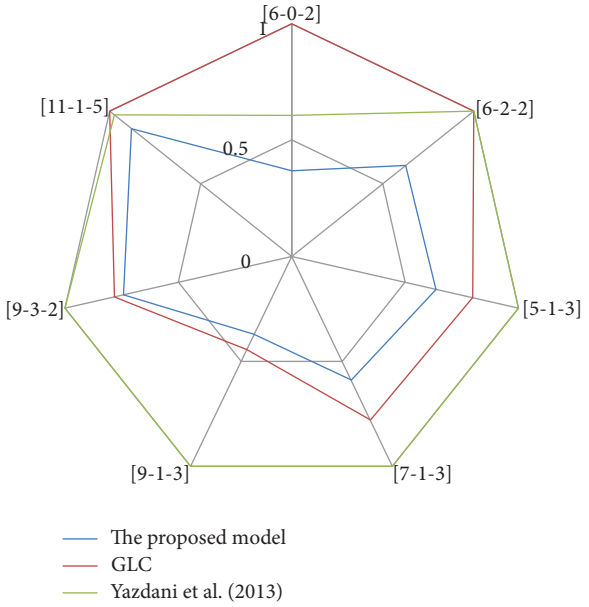

Figure 10: Improvement in the latency (μs) of the benchmarks achieved by the proposed model.

benchmarks was larger, more time and space are required to achieve the optimal solution. The future work is to develop some heuristic solution methods for optimization of large benchmarks.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

[1] R. P. Feynman, "Quantum mechanical computers," *Foundations of Physics*, vol. 16, no. 6, pp. 507–531, 1986.

[2] A. Barenco, C. H. Bennett, R. Cleve et al., "Elementary gates for quantum computation," *Physical Review A*, vol. 52, no. 5, pp. 3457–3467, 1995.

[3] H. Häffner, C. F. Roos, and R. Blatt, "Quantum computing with trapped ions," *Physics Reports*, vol. 469, no. 4, pp. 155–203, 2008.

[4] D. Kielpinski, C. Monroe, and D. J. Wineland, "Architecture for a large-scale ion-trap quantum computer," *Nature*, vol. 417, no. 6890, pp. 709–711, 2002.

[5] T. S. Metodi and F. T. Chong, *Quantum Computing for Computer Architects*, vol. 1 of *Synthesis Lectures in Computer Architecture*, 2006.

[6] M. Whitney, *Practical fault tolerance for quantum circuits [Ph.D. thesis]*, University of California, Berkeley, Calif, USA, 2009.

[7] S. B. Bravyi and A. Y. Kitaev, "Quantum codes on a lattice with boundary," *Quantum Computers and Computing*, vol. 2, pp. 43–48, 2001.

[8] N. Mohammadzadeh, M. S. Zamani, and M. Sedighi, "Gate location changing: an optimization technique for quantum circuits," *World Scientific International Journal of Quantum Information*, vol. 10, no. 3, Article ID 1250037, 20 pages, 2012.

[9] M. Whitney, N. Isailovic, Y. Patel, and J. Kubiatowicz, "Automated generation of layout and control for quantum circuits," in *Proceedings of the 4th Conference on Computing Frontiers*, pp. 83–94, May 2007.

[10] D. Maslov, C. Young, D. M. Miller, and G. W. Dueck, "Quantum circuit simplification using templates," in *Proceedings of the Design, Automation and Test in Europe (DATE '05)*, pp. 1208–1213, March 2005.

[11] D. Maslov, G. W. Dueck, and D. M. Miller, "Simplification of toffoli networks via templates," in *Proceedings of the 16th Symposium on Integrated Circuits and System Design*, pp. 53–58, 2003.

[12] V. V. Shende, A. K. Prasad, K. N. Pate, I. L. Markov, and J. P. Hayes, "Scalable simplification of reversible circuits," in *Proceedings of the 12th International Workshop on Logic and Synthesis (IWLS '03)*, 2003.

[13] M. Sedlák and M. Plesch, "Towards optimization of quantum circuits," *Central European Journal of Physics*, vol. 6, no. 1, pp. 128–134, 2008.

[14] A. Aho and K. Svore, "The design and optimization of quantum circuits using the palindrome transform," in *Proceedings of the ERATO Conference on Quantum Information Sciences (EQIS '03)*, pp. 4–6, 2003.

[15] T. S. Metodi, D. S. Thaker, A. Cross, and F. T. Chong, "Scheduling physical operations in a quantum information processor," in *Defense and Security Symposium*, vol. 6244 of *Proceedings of SPIE*, International Society for Optics and Photonics, 2006, 62440T.

[16] M. J. Dousti and M. Pedram, "Minimizing the latency of quantum circuits during mapping to the ion trap circuit fabric," in *Proceedings of the Design, Automation & Test in Europe Conference & Exhibition (DATE '12)*, pp. 840–843, 2012.

[17] S. Balensiefer, L. Kreger-Stickles, and M. Oskin, "QUALE: quantum architecture layout evaluator," in *Quantum Information and Computation III*, vol. 5815 of *Proceedings of SPIE*, pp. 103–114, March 2005.

[18] S. Balensiefer, L. Kregor-Stickles, and M. Oskin, "An evaluation framework and instruction set architecture for ion-trap based quantum micro-architectures," in *Proceedings of the IEEE 32nd Interntional Symposium on Computer Architecture (ISCA '05)*, pp. 186–196, June 2005.

[19] D. Maslov, S. M. Falconer, and M. Mosca, "Quantum circuit placement: optimizing qubit-to-qubit interactions through mapping quantum circuits into a physical experiment," in *Proceedings of the 44th ACM/IEEE Design Automation Conference (DAC '07)*, pp. 962–965, San Diego, Calif, USA, June 2007.

[20] N. Mohammadzadeh, M. Sedighi, and M. S. Zamani, "Quantum physical synthesis: improving physical design by netlist modifications," *Microelectronics Journal*, vol. 41, no. 4, pp. 219–230, 2010.

[21] N. Mohammadzadeh, M. S. Zamani, and M. Sedighi, "Auxiliary qubit selection: a physical synthesis technique for quantum circuits," *Quantum Information Processing*, vol. 10, no. 2, pp. 139–154, 2011.

[22] N. Mohammadzadeh, M. S. Zamani, and M. Sedighi, "Improving latency of quantum circuits by gate exchanging," in *Proceedings of the IEEE 12th Euromicro Conference on Digital System Design (DSD '09)*, pp. 67–73, Patras, Greece, August 2009.

[23] M. Yazdani, M. S. Zamani, and M. Sedighi, "A quantum physical design flow using ilp and graph drawing," *Quantum Information Processing*, vol. 12, no. 10, pp. 3239–3264, 2013.

[24] "GDToolkit," 2013, http://www.dia.uniroma3.it/~gdt/gdt4/index.php.

[25] N. Isailovic, *An investigation into the realities of a quantum datapath [Ph.D. thesis]*, University of California, 2010.

[26] C. E. Pearson, D. R. Leibrandt, W. S. Bakr, W. J. Mallard, K. R. Brown, and I. L. Chuang, "Experimental investigation of planar ion traps," *Physical Review A*, vol. 73, no. 3, Article ID 032307, 12 pages, 2006.

[27] A. Cross, *Synthesis and evaluation of fault-tolerant quantum computer architectures [Ph.D. thesis]*, Massachusetts Institute of Technology, 2005.

[28] M. Grassl, "Circuits for quantum error-correcting codes," 2013, http://iaks-www.ira.uka.de/home/grassl/QECC/index.html.

[29] D. Maslov, G. Dueck, and N. Scott, "Reversible logic synthesis benchmarks page," 2013, http://www.cs.uvic.ca/~dmaslov/.

[30] "GAMS," 2013, http://www.gams.com/.