Quantum Information & Computation
ISSN 1533-7146 Volume 26 (2026) pp.154–179

Article

Quantum Circuit Optimization Techniques: A Literature Survey

Mariia Makarova [1,2,3], Davide Pastorello [3,4,5], Valter Cavecchia [6], Enrico Blanzieri [3,4,6], Kamil Khadiev [7,8] and Alexander Rumyantsev [1,2,*]

[1] Institute of Applied Mathematical Research, Karelian Research Centre of RAS, Pushkinskaya Str. 11, Petrozavodsk 185910, Russia
[2] Department of Applied Mathematics and Cybernetics, Petrozavodsk State University, Lenina Pr. 33, Petrozavodsk 185910, Russia
[3] Department of Information Engineering and Computer Science, University of Trento, Via Sommarive 9, Povo Trento 38123, Italy
[4] Trento Institute of Fundamental Physics and Applications, Via Sommarive 14, 38123 Povo Trento, Italy
[5] Department of Mathematics, Universit'a di Bologna, Piazza di Porta San Donato 5, 40126 Bologna, Italy
[6] Institute of Materials for Electronics and Magnetism (CNR), Via Alla Cascata 56/c, 38123 Povo Trento, Italy
[7] Institute of Computational Mathematics and Information Technologies, Kazan Federal University, Kremlyovskaya Str. 18, Kazan 420008, Russia
[8] Zavoisky Physical-Technical Institute, FRC Kazan Scientific Center of RAS, Sibirsky Tract, 10/7, Kazan 420029, Russia
[*] Corresponding author: ar0@krc.karelia.ru

Received date: 12 January 2026; Accepted date: 3 February 2026; Published online: 15 February 2026

Abstract: We present a literature survey of the mathematical aspects of optimization and optimal synthesis of quantum circuits. The optimization techniques we consider perform the synthesis of quantum circuit by minimizing a given cost function. A total of 179 papers were filtered by inclusion/exclusion criteria and reviewed according to representation models adopted, the techniques used or developed, the kind of benchmarks adopted and the comparison against competitors performed. The survey shows that a very diverse literature has been accumulating in the recent years and although it resists a common formalization, some trends can be individuated. In particular, there is a lack of shared benchmarks with some exceptions. Moreover, many approaches change the representation of the problem to a formulation other than the simple sequence of gates. Final conclusions are drawn and possible lines of future research are suggested.

Keywords: quantum computing; circuit synthesis; quantum circuit optimization

1. Introduction

Quantum technologies are currently under the spotlight of scientific and industrial communities for their disruptive potential in outperforming classical solutions of information processing. In particular, there is intense research activity towards scalable quantum computers with promising results. The technical challenge of providing large-scale, fault-tolerant, programmable quantum hardware must be accompanied by a relevant effort in developing suitable mathematical techniques aimed at making the implementation of quantum algorithms as efficient as possible.

Quantum circuits (QC) provide a universal model for quantum computation in terms of sequences of elementary quantum gates on qubit registers that are, mathematically, unitary operators on Hilbert spaces. Quantum algorithms are typically expressed in terms of QC assuming the availability of a set of elementary gates. The problem of *QC synthesis* is basically given by the quest of constructing a QC implementing a given algorithm. Moreover, in general, there are several ways in composing quantum gates for the implementation of the algorithm, then a remarkable task, say *QC optimization*, is finding the less expensive implementation in terms of available resources, according to some criteria.

DOI: 10.2478/qic-2026-0008

In this paper, we survey the literature related to QC optimization. Instead of formalizing the notions and giving some background on quantum computing (these tasks are done in the Appendices), in Section 2 we define the review question, inclusion/exclusion criteria, and search strategy of the papers to be reviewed. As a result of strategy implementation, a systematic analysis is provided over 179 papers. To perform the analysis, in Section 3 we describe the dimensions: optimization models, criteria, and methods, and support the taxonomy with illustrative examples. A concise stratification is done in a summary table, which is located on a shared cloud resource due to size constraints. The survey is intended to be a guide to the relevant literature in the research area. We finalize the paper with a conclusion.

2. Methodology

In this section, we summarize the key settings of this survey paper, characterize the literature body, and perform a comparison to relevant reviews available in the literature. The notions used hereafter are explained in the Appendices in more detail.

2.1. Focused Review Question

In this review, we focus on various *mathematical aspects of QC synthesis and optimization* techniques relevant to the Noisy Intermediate-Scale Quantum (NISQ) era devices. We avoid physical aspects of circuit implementation unless they are relevant to the corresponding mathematical problem stated in the paper. Both heuristic and rigorous techniques are studied.

2.2. Inclusion/Exclusion Criteria

We included journal articles and conference proceedings papers relevant to the review question, based on the content of the paper. The papers primarily dealing with physical aspects of the circuit synthesis/optimization were excluded unless the corresponding mathematical result relevant to the review question had been significantly elaborated therein. Similarly, relevant papers from closely related fields (such as *reversible circuit* synthesis) were included if the contents of the paper had significant focus on quantum aspects. Multi-valued quantum logic (qudits) was mostly excluded.

The papers related to optimization of generic (theoretic) QC or those running on quantum hardware were included unless the focus was on some hardware specificity (e.g., papers optimizing QC fidelity by scheduling optimization were excluded), or the optimization was related to a specific class/circuit. We excluded papers relevant to parametrized QC, especially those related to variational optimization approaches, e.g., variational quantum eigensolver (VQE) or quantum approximate optimization algorithm (QAOA), due to ambiguity in the naming (optimization here was the type of operation that was done by this particular class of QC in hybrid computing). Optimization for simulation, both quantum simulation and simulation of a QC on classical hardware (including e.g. optimization of tensor network structure) were excluded.

Papers related to optimization of a particular circuit were mostly excluded because of the lack of generality. For the same reason, we excluded most of the papers related to QC layout/mapping problems (most of them were related to specific hardware), unless a generic circuit or generic method was considered (e.g., mapping as a part of QC transpilation).

Papers dealing with quantum machine learning were excluded, and this field is an interesting but rather separate branch of research. In a similar manner, we excluded distributed quantum computing and QC approximation-related optimization (e.g., by the unitary matrix approximation or by Monte-Carlo techniques). The decision for the papers related to the QC compilation or QC synthesis was made individually, according to the content of the article.

2.3. Search Strategy

To gather the literature body, we used a hybrid search strategy that combines the following three steps.

Keywords search was performed by means of indexing systems: Web of Science, Scopus, Crossref and Google Scholar. The relevant keywords were "quantum circuit optimization" and "quantum circuit synthesis". Filtering was performed by manual exclusion based on the abstract and/or content of the paper.

Snowball search was used to extract relevant papers from the references of the corresponding literature body obtained at other stages.

Data clearance was done to avoid repetitions if a series of papers (say, a conference paper and an extended journal paper) was identified; the extended/more recent item was kept in the literature body. We deliberately decided to exclude PhD theses since those are to be based on the formally published results in the form of articles.

2.4. Literature Body Characteristics

The literature used in this review was published up to 2025. This review is based on 179 articles [1–179], which have been published in relevant journals, conference proceedings, and open sources such as ArXiV. In a few cases, when a conference (short) or unpublished (say, ArXiV) version of the article was available together with the journal (complete) or published version, we included the latter only. With similar motivation, we deliberately excluded theses, say [180–188], since those results are to be available in articles.

To make this summary complete, we mention a few interesting articles that contain various circuit optimization techniques but are out of scope due to the inclusion/exclusion criteria selected. These are: a fundamental paper by A. Yu. Kitaev on the factorization problem, where an extension of Shor's results was done [189]; C. Jones' method for optimizing the Toffoli gate [190]; optimal small circuits generation paper [191], to name a few.

Finally, we mention several fundamental books relevant to the subject [192–194].

2.5. Relevant Surveys

It is important to mention a review paper [195]. Here, the authors characterize existing methodologies and open problems of reversible circuit synthesis. They survey papers devoted to reversible logic, not only to QC. The background on reversible logic and QC is given, followed by a comparison of these types of circuits. Existing physical implementation approaches are also concerned.

The authors provide a detailed general description of representation models, optimization criteria, and optimization techniques. Compared to our review, they highlight less representation models and optimization criteria, but they pay attention to limitations of the approaches. This review is based on 15 papers. The authors explain the existing benchmark families, their functionality and purpose.

The list of open problems of state of the art of reversible circuits synthesis is also formulated; e.g., they note that technological mapping is proposed only for specific applications and the number of elementary gates could be estimated more precisely.

Another relevant survey is [196]. This paper is dedicated to comparative practical analysis of *evolutionary algorithms*, which are used for *reversible* circuit synthesis. In contrast to [196], we do not perform numerical experiments and study implementation aspects; however, we focus on mathematical aspects and study the problem in a broader sense.

While a revision of this article was prepared, a survey paper on the topic of QC optimization [197], which covers a wide range of hardware-related and hardware-agnostic techniques, was published. Compared to those, we are targeting the mathematical problems/techniques related to QC optimization rather than optimization approaches, and in the present article, the literature body is wider.

3. Literature Review

In this section, we structure the literature body on the basis of several important characteristics: optimization model used (Section 3.1), optimization criteria (Section 3.2), and optimization method (Section 3.3). Details on quality evaluation, if applicable, are given in Appendix 3. When introducing or discussing related entities, we cite the corresponding papers where such a model or criterion was used, to illustrate the taxonomy. Note that these citations are *not intended to be exhaustive* and serve as examples. A complete, ordered list of papers and their descriptions is given in the summary table. Due to the huge size of the list of articles and in an effort to provide additional filtering capabilities, we decided to present the summary table as a separate document, available at https://docs.google.com/spreadsheets/d/13ZrxgFcxkCm4MiuQmZWeGxl9m2K0AoW_SZaq9AFYnpU/edit?usp=sharing.

3.1. QC Optimization Models

In this section, we summarize the models that are behind the corresponding circuit synthesis and optimization techniques. We note that there is a significant heterogeneity in the descriptions below: the items in the list correspond both to the *mathematical models* that are used for optimization or synthesis purposes and to the models used for, say, representation of a circuit (e.g. graph models). These may coincide, i.e. the commonly used *gate sequence* representation of the circuit may be transformed into the *unitary matrix* useful for the optimization algorithm and vice versa. Thus, a single paper may appear in several items of the list if the corresponding models are utilized therein. Sometimes we also highlight the specific *mathematical problem* that is used in the optimization phase.

It should be noted that the optimization technique may also work with the original circuit representation, which is common for, say, template optimization techniques (the techniques and methods are discussed in Section 3.3). To facilitate understanding, we refine the *circuit type* (and hence, architecture and/or the elementary gate set) used if it is the key target of the paper.

In this work, we consider an n-qubits circuit. Researchers consider QC from different points of view and different restrictions. We collect these models in groups and present them in the following list.

Standard QC

– Circuit as a **Sequence of gates**. This is the most common representation allowing one to directly use the quantum program and apply optimization techniques such as pattern matching. Researchers also use the following approaches to improve the optimization results.

  * Parallel algorithms [109] to improve the runtime of the circuit synthesis.
  * Splitting the circuit. The researchers first split the circuit into big blocks and optimize them, then optimize each block independently [168].
  * The circuit can be presented as a ZX-diagram, and researchers use the tensor network-like structure to optimize it [93].
  * Isometry based representation [128].

  Furthermore, **graph models** are frequently used to state the corresponding mathematical problem. These can be

  * Based on the generic dependency graph [70,122] or the logical data precedence graph [78].
  * Based on qubit line adjacency graph [118].
  * Multi-commodity network flow problem in a staged digraph where nodes at each stage correspond to computational basis states, arcs between stages correspond to gates and costs are defined at gate level and commodities representing output rows in truth table [89].

  Some researchers [60,104,118,139,152,159,168,169] suggested heuristic procedures to minimize the number of gates for noisy models.
  Other researchers optimize circuits from the depth point of view. The authors of [12,30] used *meet-in-the-middle* technique and another expectational time techniques [74] to improve the search algorithm.
  It is known that increasing the ancilla qubits allows us to get better results. The discussion of a trade-off between the number of ancilla qubits and the number of gates can be found in [105].
  When we develop a QC, not only decomposition of an arbitrary unitary is important, but also the decomposition of the main and most expensive parts of the circuit. Optimization of the implementation of Multi-control Toffoli gates can be found in [15]. Optimization of the implementation of Multi-control rotation gate can be found in [198,199]. At the same time, the presented technique requires good precision of the rotation gates. The trade-off between precision and the number of CNOT-gates is presented in [200,201].
  Among the classes of QC considered, one should also mention the ones having probabilistic structure, such as the so-called *repeat-until-success* circuits [19,45].

– Naturally reversible, QC that provide boolean output for boolean input can be considered as **Reversible logic circuits** (aka reversible Boolean circuits/reversible circuits). Being a subset of QC that essentially performs the *permutation* of a boolean input, these allow one to use corresponding representations such as multiple control Toffoli [111,139] to optimize them, say, by the rewrite rules [105]. Another options would be to use some equivalent algebraic construction and corresponding apparatus, such as the so-called positive Davio lattice [100], binary decision diagram [81,161], exclusive sum of products [36,79], Kronecker functional lattice diagram [100], cycle-based permutation representation [7,178], graph based algorithms and data structures [125], Toffoli networks [106], to name a few.

– **Nearest neighbor** (NN) circuits. Many types of quantum computers (for example, quantum devices based on superconductors) do not allow us to apply two-qubit gates to an arbitrary pair of qubits but have a graph that represents such a restriction. It is called a qubit connectivity graph.

Vertices of the graph correspond to qubits, and two-qubit gates can be applied only to qubits corresponding to vertices connected by an edge. We say that such a graph represents the qubit topology for a device. The most simple connectivity graph is a chain, where all qubits are presented as a line, and each qubit can interact only with the next and the previous qubits. This architecture is called Linear Nearest-neighbor (LNN) or one-dimensional Nearest neighbor (1D NN) architecture. Mottonen et. al. [198] suggest a method to represent any unary transformation using only 1.5 times more gates, comparing to the circuit without restrictions on graph connectivity. The result for an arbitrary qubit connectivity graph is based on similar results for the uniformly controlled rotation gate [202,203].

Different heuristics approaches for mapping a circuit to the LNN architecture can be found in [28,41,95,166,178] and using specific strategies [3,6,9,10,18,50,87,138]. One of the algorithms that are used for optimizing a circuit for the LNN architecture is A*. The authors of [22,138] used it to find the best permutation of the qubit order that allows us to minimize the number of gates in the case of the LNN architecture. Other researchers use the Minimum linear arrangement problem as a target problem for optimizing a circuit [1]. Interesting approaches include ones based on the Boolean satisfiability problem [138], pseudo-Boolean optimization [13], and integer programming [42].

Other types of NN architecture were considered. In the case of a 2D grid, heuristic solutions and specific strategies were suggested [4,20,28,41].

In the case of a 3D grid, heuristic solutions were suggested [11,13,41].

Circuit with Multi-qubit gates. Some types of quantum computers allow one to use Multi-qubit gates, that is why researchers optimize such circuits as well [14].

One-way quantum computation is a model of universal quantum computations in which a specific highly entangled state called a cluster state allows one to perform quantum computation by single-qubit measurements [8].

It can be seen that the circuit as a sequence of gates is the most widely used representation (e.g. [168]). This, however, restricts the possible methods, and the researchers imply other representations of the objects of interest to solve the optimization problems, such as graph-based [118], or matrix-based [119]. Furthermore, a few exotic examples include meta-optimization of the hyperparameters of the synthesis algorithm [169].

3.2. Optimization Criteria

There are many simple and complex criteria used for QC synthesis and optimization. These usually are to be minimized (smaller-better) and most of them are motivated by some specific hardware requirements (e.g. the number of gates the machine can handle) or physical requirements (e.g. decoherence time of a qubit). Hereafter we summarize these, based on the literature considered, and discuss in brief their meaning accordingly.

Below we describe the simple (one-parametric) optimization criteria and give the references where corresponding criteria are used, including the cases when a specific criterion is used as part of a more complex one.

Depth of the circuit [12,30,31,57,65,68] that indicates the longest path from the input to the output of the circuit, which is inversely proportional to the coherence time (larger - better), including

- **T-depth** being the largest number of sequential T-gates in the circuit that cannot be conducted in parallel [2,17,119];
- **Depth of measurement pattern** being the maximum number of levels of operations for the execution of the pattern due to the dependencies of measurement and correction commands [8];

Gate count, i.e., the number of gates in the circuit [7,25,30,36,50,59,63,65,68,76,79,104,106,111,118,139,159,171,178], including

- **CNOT count**, i.e., the number of CNOT gates in the circuit [16,60,87,128,136,140,142,168,204] motivated by the fact that such gates are error-prone and this criterion is proportional to depth;
- **SWAP count**, i.e., the number of SWAP gates [1,3,4,6,9–11,13,18,20,22,28,41,50,95,166] widely used in architecture-specific (e.g. NN) circuit optimization;

- **T-count** i.e. the total number of T gates or Hermitian transposes of the T gate in a QC [2,5,19,45,56,74,77,93,94,109,160] since such gates have significant resource cost;
- **MS-count**, i.e., the number of entangling Mølmer–Sørensen gates [14,130] which is important for the gate set used in the so-called trapped-ion technology;
- **Hadamard count** [94];
- **Primitive quantum one-qubit and two-qubit gate count** [12];
- **Elementary one-qubit gate count** [115];

Cost, including

- **Two-qubit cost** i.e. the number of two-qubit gates [105] and named as *the number of entanglements* in a pattern required to be applied at the beginning to produce the cluster (highly entangled) state in [8];
- **Quantum cost** being the number of basic quantum gates required to implement the given function [18,25,31,36,89,100,125,137,138,152,161,165];
- **Interaction cost** [31] and, specific to the NN architecture, the *NN cost* [3,138,167], being the sum of distances between gate qubits of any two-qubit gates;
- **Clifford+T gate cost** [17];

Number of lines in the circuit (where ancillae lines can be used) [15,150];

Number of ancillae lines [113];

Number of levels in the circuit meaning the number of sub-sequences of commuting gates that can be applied in parallel [104];

Size of pattern, being the number of qubits in a measurement pattern in one-way quantum computation [8];

In a few works, complex criteria were used such as:

- weighted additive inverse of errors and costs [99]:

$$\alpha\left(1-\frac{Error}{MaxError}\right)+\frac{\beta}{Cost},$$

Finally, a few works were focused on practical aspects of the implementation of the algorithms, such as:

- testcase-based empirical error [169];
- runtime [117], including synthesis time [65,81];
- memory consumption [117];
- system size, or method scalability [122].

For various technologies, there are several bounds known on the optimization criteria for a generic circuit. We summarize a few sources on these in Appendix 3.

It is also interesting to note that due to diverse perspective architectures studied in the literature, specific criteria used are sometimes more common for specific architectures, e.g. SWAP count is common for the NN architecture, whereas MS-count is typical for trapped-ion technology.

3.3. QC Optimization Methods

In this section, we summarize the optimization methods used in the reviewed papers. In general, there are two basic options to synthesize optimized circuits: either by using the exact method (which guaranties to find a global minimum) or by using some (meta) heuristic approach to obtain nearly optimal solution. However, even the exact optimization method, while delivering the global minimum for the given optimization criterion, may synthesize the circuit that differs from the original (unitary operator) with a given tolerance (we elaborate more on this *quality evaluation* issue in Appendix C). As such, both synthesis options are equally valuable from the point of view of solution quality.

In the list below we structure the papers according to the more general procedure names, and give more details in the summary table.

- Rewrite rules/templates based simplification (usually performed in a greedy way) [36,50,63,93,104–106,111,118,125,138,169], in particular,
  - enhanced with cost metric evaluation [3,9];
  - randomized [169];
  - based on specific properties of generalized Toffoli gates [140] or Toffoli gates with multiple target lines [165];
  - using negative and positive control lines in CNOT [152];
  - enhanced with reinforcement learning [68];
  - used in exhaustive search to construct a database of circuits [19];
  - including the lines reordering [11,138,167] and ancillae adding [105];
  - reduction rules (simplification, interchanging, and commutation) [12] including relative phase gate substitutions [56];
  - peephole optimization [48] including relaxed peephole [98];
- Heuristics [65,68,87,118,119]
  - using properties of the so-called ϵ-nets in Solovay–Kitaev algorithm implementation [12,117];
  - based on functional diagram dependency matrices (local and global) [150];
  - based on A^* search [7,22,59,60,168,204];
  - based on subtree-to-gate mapping lookup table [81];
  - based on decomposition and symmetry rules [100];
  - based on operating with total Hamming distance [171];
  - lookahead-based [1,10,95];
  - graph partitioning [6];
  - based on qubit count [20];
  - based on preference index of qubits [4];
  - Skipping Table Algorithm [12] that skips redundant sequences to speed up the exhaustive search;
- Metaheuristics based:
  - genetic algorithms for circuit synthesis [11,25,99,139,159];
  - Harmony search with local optimization of SWAP gate insertion [28];
  - graph traversal by ant colony [41];
- Decomposition based schemes [15,94,128,137], in particular, based on the cosine-sine decomposition [8,136,142], quantum Shannon decomposition [8,136], multiobjective QC decomposition [8];
- Numerical optimization [65], including gradient-based optimization [130] and Quasi-Newton method of Broyden, Fletcher, Goldfarb, and Shanno (BFGS) [14];
- Sequential generation (with error correction) of sub-circuits returning the desired result with high probability in exponential [19] or polynomial [5] time, or with a fixed (given) number of steps guaranteed by 'fallback' circuit [45], and using such a method as a framework for nonlinear arithmetics [160];
- Cycle-based permutation representation: optimized using total Hamming distance [178], bin packing problem for cycle distribution among the available registers [31], using A^* search [7];
- Boolean satisfiability based [76,77,167], in particular, Pseudo-Boolean optimization (PBO) [13,166];
- Large-scale approaches, including hierarchical synthesis of QC [122] and circuit partitioning [65,168];
- Meet-in-the-middle [30,74,109];
- Multi-commodity Network flow combined with arc-subset selection problem [89];
- Variational hybrid quantum-classical approach with quantum cost evaluation [57];
- Exhaustive search with speedup using heuristic search tree pruning [79], rules-based substitutions [161];
- Positive Davio Lattice Diagram construction [18];
- SWAP insertion methods [20];
- Global consideration of transformation matrix [17];
- Matroid partitioning algorithm [2];
- Steiner tree problem [16];

It is important to mention that in some papers the optimality of some circuit may be given *by design* of the corresponding synthesis method, e.g.

- in [76] the corresponding SAT problem gives a solution (i.e. the function is SAT), if there exists a circuit of a given (fixed) depth d, and thus optimality is obtained by solving a number of problems with increasing d;
- in [45] the so-called probabilistic QC with fallback consists of a finite (fixed, given) number of sub-circuits followed by a fallback circuit (the last step is offered with small probability) with a given precision and low (average) cost;
- in [14], the solution is obtained in an iterative way by increasing the target (number of entangling gates) sequentially, and thus obtaining the minimum.

3.4. Software and Benchmarks

We note that among the benchmarks used, a few are relatively popular between researchers, these include

RevLib, the benchmark for reversible functions, including the QC [205], http://revlib.org/;

Maslov, Reversible logic synthesis benchmark page by D. Maslov, available at http://www.cs.uvic.ca/dmaslov/ since 2002, and the new version available at https://reversiblebenchmarks.github.io/.

LGSynth, one of the benchmarks available at https://ddd.fit.cvut.cz/www/prj/Benchmarks/.

optimizer, Benchmark QC before and after optimization available at https://github.com/njross/optimizer, using techniques from [118],

RevKit, a toolkit for reversible circuit design, available at http://www.revkit.org,

Cirq, available at https://github.com/quantumlib/Cirq.

We also mention a few synthesis/optimization tools that were introduced or used in the corresponding papers.

BayeSyn synthesizer [169] - QC generation by high-level languages (C or C++) framework implementation via stochastic synthesis;

QFAST package for Python [65];

pQCS written in C++11 [109];

RMRLS written in C [79];

QULASYN [18];

Revkit https://github.com/msoeken/revkit used in the paper [161].

4. Discussion

A large number of reviewed papers deals with direct manipulation on the gates in circuit representation. In this case, the search space of the optimization coincides with the configuration space of the circuit itself. Not surprisingly, in this case the cost measures are mainly statistics computed on the circuit characteristics like depths, gate (or specific gate) counts and quantum cost. Exceptions use testbase empirical error [169].

Some of the papers are targeting architecture-specific NN circuits that are relevant to the commonly used NN architecture. Note also that while the majority of the papers are optimizing circuits of finite length, a few circuits are by design non-deterministic and possibly have infinite length, such as the RUS circuits [19]. This opens a way to studying the *convergence* of circuit optimization algorithms.

The most original and promising papers, however, are the ones in which a different representation is chosen in order to perform the optimization. In this case, we appreciate from our analysis of the literature how diverse and wide the choice of representations can be. In this category of papers, the representation is usually chosen in a way to exploit an optimizer already developed for such a representation. Among the most popular representations of such a type are various graph models that allow one to use (classical) graph traversal/search algorithms and various (meta)heuristics.

Let us consider the tendencies that emerge over time. It is possible to see how papers that actually change the representation in order to perform the optimization appear in more recent years. Moreover, in recent years the comparison against an existing compiler has been more and more common. Surprisingly, even recent papers sometimes lack comparison with competitors and present just validation with simple circuits or against baselines.

It is also important to mention the relationship between optimization and compilation. It is reasonable that some sort of optimization can in principle be performed by the compiler. However, it is important to distinguish between the optimization process that we are considering and the compiler that have to take into account the actual physical architecture of the target machine. A successful proposal for the optimization could be included in a compiler but we argue, given that the QC research area is still relatively recent, research should be pursued separately.

It is worth to mention that no attempt has been reviewed to target QC for Quantum Machine Learning. In our opinion, an actual optimization of the circuit that takes the data into account could be an interesting prospective to take. In this regard, also the training of quantum neural networks represented in terms of parametric circuits can be meant as a circuit synthesis where a classical backpropagation can be used but also the parameter shift rule for differentiating the function implemented by the circuit and performing gradient descent can be implemented. Moreover it is worth to cite a couple of recent papers that go in the directions of integrating machine learning [68,169].

There seem to be no shared and commonly-adopted benchmarks, in fact is hard to rank the methods in a sensible way, because there are no shared state-of-the-art to compare against. Some exceptions, however, are the RevLib and Maslov benchmark sources. Still we see that it is becoming common to compare against compilers.

5. Conclusion

We have reviewed the papers that deal with mathematical aspects of quantum gates circuit optimization and synthesis. The area appears to be wide and diverse, with no shared benchmarks, representations, and an easy-to-identify stare-of-the-art approach. That makes the area interesting for research and possibly contributions. Moreover, we argue that it is still possible to work on optimization separately from the actual practical implementation of a quantum compiler.

We expect that the research area of QC optimization will eventually meet the trend of application of machine learning techniques as deep learning to the optimization process itself. Moreover, we also expect the development of optimization specific techniques devoted to QML algorithms. As argued in this review, optimization techniques are central in QC synthesis processes. In a natural way, the seek of an optimized QC implementing a given computation can be cast into the training of a machine learning model. In this sense, classical algorithms of machine learning can be applied to learn a synthesis process, for example adapting QC belonging to a given collection. An instance-based learning approach can be adopted where the synthesis is carried on from a training set of QC which have already been synthesized. An example of this approach is given in [206], where a hybrid quantum classical algorithm is proposed, in which the runs of an adiabatic quantum computer are iterated in order to find a representation of a given optimization problem in a Hamiltonian operator.

Another possibility that we foresee is the actual application of specific-purpose machines like quantum annealers to the actual task of QC optimization. If the QC synthesis can be formulated in terms of a combinatorial optimization problem, one can suppose to exploit quantum resources to accomplish the task. A few research papers are targeting this direction, e.g., [207]. Being formulated in QUBO form, the problem may be solved by, the quantum approximate optimization algorithm (QAOA) on a gate-based machine, but also runs on a quantum annealer can be considered for an efficient QC optimization.

The limits of the current quantum machines are such that an optimization step will be probably proved to be necessary in order to bridge the gap between theoretical quantum algorithms and practical implementations. To this end the characteristics of the hardware will dictate which will be the cost functions to be optimized, and optimization will be included eventually in any architecture-specific compiler. The development of standard and shared benchmarks could accelerate the development of techniques and the emergence of a recognizable state of the art. Moreover, multi-objective optimization has not been fully addressed yet in this realm. These circumstances make the optimization of QC an interesting and important field of research in the near future.

Author Contributions
Conceptualization, M.M., E.B. and A.R.; M.M. and A.R. collected the metadata; M.M., E.B., A.R. and K.K. analyzed and interpreted the results found in the literature; M.M., E.B., A.R., K.K., D.P. and V.C. wrote the paper; edited the paper and reviewed the drafts. All authors have read and agreed to the published version of the manuscript.

Funding
This research received no external funding.

Conflicts of Interest
The authors declare no conflicts of interest.

Data Availability Statement
No new data were created during this study.

Acknowledgments
The work has been supported by the Q@TN consortium. We thank the referees for their constructive criticism that helped us complete this work.

Appendix A. Quantum Computing Basics

Within this section, well-known material on quantum computing is briefly explained, for an explicit source see e.g., [193]. In applications, often Dirac formalism is adopted, and the so-called "bra-ket" notation used. This means that $|\psi\rangle$ is the column vector and $\langle\psi|$ is the conjugate row vector in a (finite dimensional) Hilbert space $\mathcal{H}$ that is a complex vector space equipped with an inner product $\langle \ | \ \rangle$. The physical states of a quantum system are represented by the positive linear operators on $\mathcal{H}$ with unit trace called *density matrices*. The set of density matrices:

$$\mathcal{S}(\mathcal{H}) = \{\rho \in \mathcal{L}(\mathcal{H}) : \rho \geq 0, \mathrm{tr}\rho = 1\}, \tag{A1}$$

where $\mathcal{L}(\mathcal{H})$ is the space of linear operators on $\mathcal{H}$, is convex and its extreme elements are the 1-dimensional orthogonal projectors $\rho = |\psi\rangle\langle\psi|$ with $\| \ \psi \ \| = 1$, called *pure states*. Therefore, the pure states are in one-to-one correspondence to the projective rays of $\mathcal{H}$ then a pure state can be completely represented by a normalized vector $|\psi\rangle \in \mathcal{H}$ up to a multiplicative phase factor. The *mixed states* are density matrices given by convex combination of pure states, i.e. elements in $\mathcal{S}(\mathcal{H})$ that are not pure. Without explicit specifications, we deal with pure states.

A qubit is any quantum system described in a 2-dimensional Hilbert space, it turns out to be a quantum analog of a bit in binary logic. The special symbols $|0\rangle$ and $|1\rangle$ correspond to the *computational basis states* of a two-state qubit system and are given as

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \quad |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}.$$

Any qubit state $|\psi\rangle$ is a *superposition* of the states $|0\rangle$ and $|1\rangle$, meaning that

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle = \begin{bmatrix} \alpha \\ \beta \end{bmatrix}, \quad \alpha, \beta \in \mathbb{C}, |\alpha|^2 + |\beta|^2 = 1.$$

The components, called *amplitudes*, of a qubit have a specific physical meaning, since the *measurement* of a qubit would produce the value $|0\rangle$ with probability $|\alpha|^2$ and $|1\rangle$ with probability $|\beta|^2$, which resembles a Bernoulli random variable. In particular, the unbiased coin is represented by the state $|+\rangle$ such that $\alpha = \beta = 1/\sqrt{2}$. However, it is important to note that measurement of a single qubit can be made only once, since the measurement procedure essentially *collapses* the qubit to one of the binary states reproducing the same state under subsequent measurements.

As a general property of quantum systems, composition of multiple qubits as well as multi-qubit circuits are performed via the Kronecker product of matrices:

$$\boldsymbol{A} \otimes \boldsymbol{B} = \begin{pmatrix} a_{11}\boldsymbol{B} & \dots & a_{1n}\boldsymbol{B} \\ \vdots & \ddots & \vdots \\ a_{n1}\boldsymbol{B} & \dots & a_{nn}\boldsymbol{B} \end{pmatrix}$$

In particular, considering the product $|\psi\rangle \otimes |\varphi\rangle$ of two 1-qubit states, we obtain the state of a qubit pair described by a vector in $\mathbb{C}^4$. To shorten the notation, very common abbreviations are used such as:

$$|\psi\rangle \otimes |\varphi\rangle \equiv |\psi\rangle|\varphi\rangle \equiv |\psi\varphi\rangle.$$

The compact notation often applies to the vectors of the computational basis, e.g. $|00\rangle = (1,0,0,0)^T$ (where a row vector is transposed). In general, since the state of a n-qubit system is a vector in $\mathbb{C}^{2^n}$, then in quantum computing one can represent data in a space whose dimension scales exponentially in the number of qubits. A system of multiple qubits may not always be decomposed into a product of individual qubits due to the so-called *entanglement* demonstrated e.g. by the following *Bell state*

$$|\Psi\rangle = \frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|11\rangle, \tag{A2}$$

such a state is not decomposable to a Kronecker product of two 1-qubit states. Entangled states, like the Bell state (A2), encode quantum correlations that cannot be reproduced classically as proved by the experimental violation of the Bell inequalities. Due to its dramatic deviation from classical phenomena, entanglement is one of the main resources exploited in quantum computing.

Similarly to a single qubit, an n-qubit system, also called *n-qubit register*, can be in a quantum superposition of the all possible 2^n binary states numbered lexicographically. As such, the state of a n-qubit system is represented by a vector $|\psi\rangle$ of length n as

$$|\psi\rangle = \sum_{i\in\{0,1\}^n} \alpha_i |i\rangle, \qquad \sum_{i\in\{0,1\}^n} |\alpha_i|^2 = 1,$$

where the components α_i have similar meaning, and i is the n-digit binary string corresponding to the pure n-qubit state. It is worth noting that a measurement is possible to be taken over a part of the system of n qubits which will subsequently reduce the number of components of a vector by fixing them, causing re-normalization in the coefficients.

Constructing a QC is performed using several basic elements known as quantum gates and acting as operators on the corresponding vectors. Algebraically these correspond to matrices that define operators on the state space of the model, where the only restriction for the matrix $\boldsymbol{U}$ used to perform a quantum operation is that $\boldsymbol{U}$ should be *unitary*, that is, $\boldsymbol{U}^\dagger \boldsymbol{U} = \boldsymbol{I}$, where $\boldsymbol{U}^\dagger = (\boldsymbol{U}^T)^*$ is the transposed and complex conjugate matrix of $\boldsymbol{U}$ and $\boldsymbol{I}$ is the identity matrix of corresponding dimension. In particular, the following so-called Pauli matrices $\sigma_x, \sigma_y, \sigma_z$ define the single-qubit gates:

$$\sigma_x = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \quad \sigma_y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}, \quad \sigma_z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}, \tag{A3}$$

where σ_x is acting as a quantum NOT circuit which swaps the components of a vector $|\psi\rangle$ (a.k.a. *bit flip*), σ_y flips both the components and interacts with their real and imaginary parts, and σ_z flips the sign of the second component of the vector $|\psi\rangle$ (sometimes called *phase flip*). The eigenvectors of σ_z form the computational basis.

A remarkable 1-qubit gate is the *Hadamard gate* defined as follows:

$$H := \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}. \tag{A4}$$

There is a well-known graphical representation of quantum gates that resembles the notation of logical gates in digital computing. The graphical representation of the Hadamard gate is:

$$-\!\!\!-\!\!\!-\boxed{H}-\!\!\!-\!\!\!- \qquad .$$

The Hadamard gate realizes a change of basis $\{|0\rangle, |1\rangle\} \mapsto \{|+\rangle, |-\rangle\}$ of a 1-qubit Hilbert space where $|\pm\rangle = (|0\rangle \pm |1\rangle)/\sqrt{2}$.

The 1-qubit gate appending a relative phase in the input state, it is defined by:

$$P_\phi := \begin{pmatrix} 1 & 0 \\ 0 & e^{i\phi} \end{pmatrix}, \tag{A5}$$

where $\phi \in \mathbb{R}$, and hence $P_\pi = \sigma_z$. The corresponding graphical representation is simply:

$$-\!\!\!-\!\!\!-\boxed{P_\phi}-\!\!\!-\!\!\!- \qquad .$$

Two more specific 1-qubit gates $S := P_{\pi/2}$ and $T := P_{\pi/4}$ are important in defining a universal set of quantum gates.

Multiple qubit gates can also be defined, a remarkable example of a 2-qubit gate is the *controlled NOT* ($CNOT$) gate given by

$$\boldsymbol{U}_{CN}|\psi\rangle = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \end{bmatrix} = \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_4 \\ \alpha_3 \end{bmatrix}.$$

It may be noted that the second component is flipped only if the first component is in state 1, that is, $|10\rangle$ is changed to $|11\rangle$, whereas $|11\rangle$ becomes $|10\rangle$. The graphical representation of the $CNOT$ is

In order to construct QC from single gates in algebraic formalism, Kronecker product is used to indicate applying parallel gates on independent qubits, while matrix multiplication corresponds to sequential application of gates. This, however, has the price of exponential dependence of the matrix size on the number of qubits, as a consequence a general quantum computation cannot be efficiently simulated by classical computing.

As such, a QC for n qubits corresponds to a unitary matrix $\boldsymbol{U}$ of size 2^n that performs the necessary computation in terms of a product $\boldsymbol{U}|\psi\rangle$, where $|\psi\rangle$ is the initial state of the system. As in classical computing, where a small set of logical gates such as $\{AND, OR, NOT\}$, can be used to realize any computation, in quantum computing there is a similar notion of universality. A set of quantum gates is said to be *universal* for quantum computation if, for all $n \in \mathbb{N}$, any QC for n qubits can be approximated to arbitrary accuracy by a composition of only those gates. One can prove that the set $\{H, S, T, CNOT\}$ is universal for quantum computation [193].

Appendix B. Synthesis, Optimization, Adaptation and Decomposition of QC

QC are a universal representation of quantum computations. However, procedure of quantum computation can be described in alternative ways, for example *adiabatic quantum computing* (AQC) is a universal model of quantum computing as well. Moreover, to some extend, quantum computations can be also described in natural language, possibly introducing some ambiguity.

By synthesis of QC we mean a process of obtaining a (more precisely at least one) QC from a different representation of a quantum algorithm. The foundation of this topic was laid in [208] generalizing approach for constructing Toffoli gate from 5 $2-$bit gates proposed in [209] and thus building up n-bit operations with quantum $1-$ and $2-$qubit gates. Further, the QC synthesis problem is described briefly in [186], where the term synthesis is defined as a compilation of the desired operator into a QC understandable for a quantum computer. In general, the process of synthesis depends on the kind of the initial representation and on possible constraints and additional requirements on the target QC. Among the synthesis processes of QC we point out the following relevant procedures:

- **Optimization**: a synthesis process in which the target QC is required to minimize one or more cost criteria.
- **Adaptation**: a synthesis process which provides the target circuit modifying one or more intermediate QC that do not implement the initial representation of the considered algorithm.
- **Composition**: a synthesis process from given elementary gates and/or circuits until the corresponding unitary matrix becomes equal (or approximately equal) to the specified matrix [99].
- **Decomposition**: a synthesis process whose initial representation is a given unitary matrix [210] or already a QC and the target circuit is constrained to be a composition of elementary gates from a given set.

Note that the four processes described above do not exhaust the whole possible synthesis processes and a synthesis process can include more than one. Some authors adopt slightly different versions of these definition, for example reducing synthesis to decomposition.

Approximation of any unitary operator acting on n qubits with arbitrary precision ε by the so-called universal set of gates is guaranteed by Solovay–Kitaev theorem [193] using $O(n^2 4^n \log^c(n^2 4^n/\varepsilon))$ gates. However, this is quite a large number of gates and thus optimization is required. The general idea of QC optimization is to obtain a QC that satisfies some requirements, e.g. hardware requirements of a real machine, the set of quantum gates used for circuit synthesis and topology [60], specific connection architecture used [28,41], depth [31] etc. As such, various optimization schemes are used, either based on the so-called search algorithms (including Monte-Carlo), or using some information about the specific system, such as undertaken by Bayesian optimization as in [169].

It should be noted that a few related problems to QC synthesis and optimization are:

- the so-called QC *compilation* [211] or transformation [212], which is, adaptation of the general QC into a specific hardware (as well as some specific related techniques such as instantiation [60]) and turns out to be NP-hard [213,214],
- the quantum *tomography* which attempts to reconstruct or approximate the QC on the basis of multiple output measurements for given input [215],
- optimization related to the data loading into the quantum program, i.e. quantum state preparation [216, 217],
- last but not least, quantum computing *simulation* [218] is a necessary attribute in the development of the new circuits with the help of conventional computers and supercomputing hardware.

Authors use different technique for synthesis of quantum circuits. Let us list some of them. Classical (reversible) logic based: binary decision diagram (BDD) [81,161], exclusive-sum-of-products (ESOP) [36,79,165], Toffoli network [77,106,167], node dependency matrix [150], Kronecker functional lattice diagram (KFLD) [100], heuristic and A* algorithms [25,59,99,119,171], representation as a series of cycles [7,7,31,137,171,178], Bayesian optimization [169], SAT problem on a Boolean function encoding the QC of a given depth (satisfiable if such a circuit exists) [76], synthesis with respect to LNN architecture [16,138].

The universal circuit optimization results for arbitrary unitary can be significantly improved if we know the structure of the unitary matrix. So, researchers consider improvement of quantum circuits for specific important problems. Researchers [60,104,118,139,152,159,168,169] suggested a heuristic procedure to minimize the number of gates for a noisy model in the case of QFT and QAOA and other algorithms. Circuits for specific problems with respect to NN architecture are also can be interesting: QFT [219–226], quantum hashing [200,220,221], Shor's algorithm [223] and others.

These problems are beyond the scope of this review unless a significant part of the corresponding papers are related to circuit optimization and/or synthesis.

Appendix C. Bounds and Quality Evaluation

The circuit size of an arbitrary n-qubit operation is bounded, and some of these bounds are constructive. Those bounds may be given for an arbitrary size n, or for some specific size. Many of the constructive results are now embedded as optimization techniques in the software packages for quantum computing.

Any unitary that represents an algorithm on n qubits can be represented using 4^n gates [198,199]. Later, this result was improved. The approach that uses $\frac{23}{48}4^n$ CNOT gates (and about the same one qubit gates) was shown [128,227]. These results are close to the lower bound [228,229] for an arbitrary unitary transformation that is $\frac{1}{4}4^n$.

In particular, for a trapped ion technology, the number of so-called MS-gates is lower bounded by [130]

$$\frac{4^n - 3n - 1}{2n + 1}.$$

The CNOT count of a universal n-qubit circuit is lower bounded by [230]

$$\frac{4^n - 3n - 1}{4}.$$

Another optimization is [136]. The CNOT cost of n-qubit Toffoli gate is at least $2n$, even if ancillae are permitted [231].

The depth of an arbitrary n-qubit Clifford transformation is upper bounded by $7n - 4$ over LNN architecture, and $1.5n + O(\log^2(n))$ in all-to-all connected architecture [232].

For the arbitrary 2-qubit gate, the lower bound on CNOT count is 3 [233,234], and it is constructive (i.e., a procedure exists for constructing such a chain), used together with at most 15 rotation gates.

The practical limitations of the circuit parameters that provide quantum supremacy are summarized in [235]. A few synthesis-related metrics are summarized in [236], whereas upper bounds for various synthesis technologies are given in [237].

Mapping the specific unitary operator $\boldsymbol{U}$ into a QC and considering all the constraints will give a new operator $\boldsymbol{U}_G$. However, the latter may be not identical to the former, and there is some tolerance for acceptable deviation. Otherwise, the deviation of $\boldsymbol{U}_G$ from $\boldsymbol{U}$ is used to construct a goal function for the optimization algorithm. Among the norms, in the literature the following are considered:

- The L^p norm, i.e. the norm in an L^p space, defined in general case as

$$||\boldsymbol{U} - \boldsymbol{U}_G||_{L^p} = \left[\sum_{i=1}^{n}\sum_{j=1}^{m}|u_{ij} - u_{Gij}|^p\right]^{1/p}$$

including

  – L^1 norm, known as Manhattan distance (in case of vectors), [99]

$$||\boldsymbol{U} - \boldsymbol{U}_G||_{L^1} = \sum_{i=1}^{n}\sum_{j=1}^{m}|u_{ij} - u_{Gij}|$$

  – Euclidean norm [130]

$$||\boldsymbol{U} - \boldsymbol{U}_G||_E = \sqrt{\sum_{i=1}^{n}\sum_{j=1}^{m}(u_{ij} - u_{Gij})^2}$$

  – Maximum norm

$$||\boldsymbol{U} - \boldsymbol{U}_G||_{L^\infty} = \max_{i,j}|u_{ij} - u_{Gij}|$$

- Trace norm of matrix $\boldsymbol{U}$ is the sum of singular values of this matrix. The singular values are the square roots of the eigenvalues of $\boldsymbol{U}\boldsymbol{U}^*$.

$$||\boldsymbol{U}||_1 = Tr(\sqrt{\boldsymbol{U}\boldsymbol{U}^*}).$$

- Diamond norm is the trace norm of the output of a trivial extension of a linear map, maximized over all possible inputs with trace norm at most one.
  Let $\Phi : M_n(\mathbb{C}) \to M_m(\mathbb{C})$ be a linear transformation, where $M_n(\mathbb{C})$ denotes $n \times n$ complex matrices, let $1_n : M_n(\mathbb{C}) \to M_n(\mathbb{C})$ be the identity map on $n \times n$, and $X \in M_{n^2}(\mathbb{C})$. Then the diamond norm of Φ is given by

$$||\Phi||_\diamond = \max_{X;||X||_1 \leq 1} ||(\Phi \otimes 1_n)X||_1,$$

  where $||\cdot||_1$ denotes the trace norm.
  The diamond norm induces the diamond distance, which in the particular case of completely positive, trace non-increasing maps E, F is given by

$$d_\diamond(E, F) = ||E - F||_\diamond = \max_{\rho} ||(E \otimes 1_n)\rho - (F \otimes 1_n)\rho||_1,$$

  where the maximization is done over all density matrices ρ of dimension n^2.
- Hilbert–Schmidt norm defined by Hilbert–Schmidt inner product [60]

$$\langle \boldsymbol{U}, \boldsymbol{U}_G \rangle_{HS} = Tr(\boldsymbol{U}^* \boldsymbol{U}_G).$$

  Based on this representation, the following distance function is suggested in [60]

$$D(\boldsymbol{U}, \boldsymbol{U}_G) = 1 - \langle \boldsymbol{U}, \boldsymbol{U}_G \rangle / 2^n.$$

- Frobenius norm [65]

$$||\boldsymbol{U}_G^* \boldsymbol{U} - \boldsymbol{I}|| = 2 - 2Re(Tr(\boldsymbol{U}_G^* \boldsymbol{U})),$$

  which induces a distance function [65,204]

$$\Delta_F(\boldsymbol{U}_G, \boldsymbol{U}) = 1 - \frac{|\langle \boldsymbol{U}, \boldsymbol{U}_G \rangle_{HS}|}{d},$$

where d is the state space dimension. (A similar version of this function, $\sqrt{\Delta_F}$, was used in [5]).

It can be mentioned that sometimes instead of the measurements on the (original and approximating) unitary matrices, the output of two circuits is compared in terms of the probability distribution of the results it generates. In such a case, the comparison is done by some other norms, such as the total variation norm [168] (defined as the half of the L_1 norm applied to the corresponding probability vectors).

To discuss the relation between the quality evaluation measures presented in this section and optimization criteria from Section 3.2, we note that usually the problem of circuit optimization is considered as (un)constrained optimization problem, where the criterion is one (or several) of the criteria defined in Section 3.2, and the constraint (if any) is given using one of the aforementioned norms. However, a dual problem can also be studied, where the distance is to be optimized, and the constraint is defined using one of the metrics from Section 3.2. The rather diverse problem statements in the literature body of this study prevent us from giving a concise mathematical description of the generic circuit optimization problem.

References

1. A. Shafaei, M. Saeedi and M. Pedram (2013). Optimization of quantum circuits for interaction distance in linear nearest neighbor architectures, in *2013 50th ACM/EDAC/IEEE Design Automation Conference (DAC)*, pp. 1–6.
2. M. Amy, D. Maslov and M. Mosca (2014). Polynomial-time t-depth optimization of clifford+t circuits via matroid partitioning, *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 33(10), pp. 1476–1489.
3. A. Bhattacharjee, C. Bandyopadhyay, R. Wille, R. Drechsler and H. Rahaman (2019). Improved look-ahead approaches for nearest neighbor synthesis of 1d quantum circuits, in *2019 32nd International Conference on VLSI Design and 2019 18th International Conference on Embedded Systems (VLSID)*, pp. 203–208.
4. A. Bhattacharjee, C. Bandyopadhyay, R. Wille, R. Drechsler and H. Rahaman (2018). A novel approach for nearest neighbor realization of 2d quantum circuits, in *2018 IEEE Computer Society Annual Symposium on VLSI (ISVLSI)*, pp. 305–310.
5. A. Bocharov, M. Roetteler and K. M. Svore (2015). E?icient synthesis of universal repeat-until-success quantum circuits, *Physical Review Letters*, 114(8).
6. A. Chakrabarti, S. Sur-Kolay and A. Chaudhury (2011). Linear nearest neighbor synthesis of reversible circuits by graph partitioning.
7. K. Datta, B. Ghuku, D. Sandeep, I. Sengupta and H. Rahaman (2013). A cycle based reversible logic synthesis approach, in *2013 Third International Conference on Advances in Computing and Communications*, pp. 316–319.
8. M. Houshmand, M. Sedighi, M. S. Zamani and K. Marjoei (2017). Quantum circuit synthesis targeting to improve one-way quantum computation pattern cost metrics, *ACM Journal on Emerging Technologies in Computing Systems (JETC)*, 13, pp. 1 – 27.
9. A. Kole, K. Datta, R. Wille and I. Sengupta (2017). A nearest neighbor quantum cost metric for the reversible circuit level, in *TENCON 2017 - 2017 IEEE Region 10 Conference*, pp. 2943–2948.
10. R. Wille, O. Keszocze, M. Walter, P. Rohrs, A. Chattopadhyay and R. Drechsler (2016). Look-ahead schemes for nearest neighbor optimization of 1D and 2D quantum circuits, in *2016 21st Asia and South Pacific Design Automation Conference (ASP-DAC)*, pp. 292–297, IEEE, Macao, Macao.
11. A. Kole, K. Datta and I. Sengupta (2018). A new heuristic for \$n\$ -dimensional nearest neighbor realization of a quantum circuit, *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 37, pp. 182–192.
12. C. Lin, A. Chakrabarti and N. Jha (2014). Ftqls: Fault-tolerant quantum logic synthesis, Very Large Scale Integration (VLSI) Systems, *IEEE Transactions* on, 22, pp. 1350–1363.
13. A. Lye, R. Wille and R. Drechsler (2015). Determining the minimal number of swap gates for multi-dimensional nearest neighbor quantum circuits, in *The 20th Asia and South Pacific Design Automation Conference*, pp. 178–183.
14. E. A. Martinez, T. Monz, D. Nigg, P. Schindler and R. Blatt (2016). Compiling quantum algorithms for architectures with multi-qubit gates, *New Journal of Physics*, 18(6), p. 063029.
15. D. Miller, R. Wille and Z. Sasanian (2011). Elementary quantum gate realizations for multiple-control toffoli gates, Proceedings of *The International Symposium on Multiple-Valued Logic*.

16. B. Nash, V. Gheorghiu and M. Mosca (2020). Quantum circuit optimizations for nisq architectures, *Quantum Science and Technology*, 5(2), p. 025010.
17. P. Niemann, R. Wille and R. Drechsler (2018). Improved synthesis of Clifford+T quantum functionality, in *2018 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, pp. 597–600, IEEE, Dresden, Germany.
18. M. Perkowski, D. Shah and M. Kameyama (2011). Synthesis of quantum circuits in linear nearest neighbor model using positive davio lattices, *Facta universitatis - series: Electronics and Energetics*, 24.
19. A. Paetznick and K. M. Svore (2014). Repeat-until-success: non-deterministic decomposition of single-qubit unitaries, *Quantum Inf. Comput.,* 14(15-16), pp. 1277–1301.
20. R. R. Shrivastwa, K. Datta and I. Sengupta (2015). Fast qubit placement in 2d architecture using nearest neighbor realization, in *2015 IEEE International Symposium on Nanoelectronic and Information Systems*, pp. 95–100.
21. Y. Zhang, Z. Guan, L. Ji, Q. F. Luan and Y. Wang (2020). A method of mapping and nearest neighbor optimization for 2-d quantum circuits, *Quantum Information and Computation*, 20(3 & 4), pp. 194–212.
22. A. Zulehner, S. Gasser and R. Wille (2017). Exact Global Reordering for Nearest Neighbor Quantum Circuits Using A∗, in I. Phillips and H. Rahaman (editors), *Reversible Computation*, vol. 10301, pp. 185–201, Springer International Publishing, Cham. Series Title: Lecture Notes in Computer Science.
23. N. Abdessaied, M. Amy, M. Soeken and R. Drechsler (2016). Technology Mapping of Reversible Circuits to Clifford+T Quantum Circuits, in *2016 IEEE 46th International Symposium on Multiple-Valued Logic (ISMVL)*, pp. 150–155.
24. A. Abdollahi and M. Pedram (2006). Analysis and Synthesis of Quantum Circuits by Using Quantum Decision Diagrams, in *Proceedings of the Design Automation & Test in Europe Conference*, vol. 1, pp. 1–6.
25. M. Y. Abubakar, L. T. Jung, N. Zakaria, A. Younes and A.-H. Abdel-Aty (2017). Reversible circuit synthesis by genetic programming using dynamic gate libraries, *Quantum Information Processing*, 16(6), p. 160.
26. A. V. Aho and K. M. Svore (2003). *Compiling Quantum Circuits using the Palindrome Transform*. Version Number: 1.
27. N. Ain Binti Adnan, K. Kushida and S. Yamashita (2016). A pre-optimization technique to generate initial reversible circuits with low quantum cost, in *2016 IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 2298–2301, IEEE, Montreal, QC.
28. M. G. Alfailakawi, I. Ahmad and S. Hamdan (2016). Harmony-search algorithm for 2D nearest neighbor quantum circuits realization, *Expert Systems with Applications*, 61, pp. 16–27.
29. M. B. Ali, T. Hirayama, K. Yamanaka and Y. Nishitani (2015). Quantum Cost Reduction of Reversible Circuits Using New Toffoli Decomposition Techniques, in *2015 International Conference on Computational Science and Computational Intelligence (CSCI)*, pp. 59–64.
30. M. Amy, D. Maslov, M. Mosca and M. Roetteler (2013). A Meet-in-the-Middle Algorithm for Fast Synthesis of Depth-Optimal Quantum Circuits, *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 32(6), pp. 818–830. Publisher: Institute of Electrical and Electronics Engineers (IEEE).
31. M. Arabzadeh, M. Saheb Zamani, M. Sedighi and M. Saeedi (2013). Depth-optimized reversible circuit synthesis, *Quantum Information Processing*, 12(4), pp. 1677–1699.
32. P. M. Arpita, K. Datta, R. Vemula and I. Sengupta (2015). Optimization of reversible circuits using triple-gate templates at quantum gate level, in *2015 International Conference on Electronic Design, Computer Networks & Automated Verification (EDCAV)*, pp. 120–124, IEEE, Shillong, India.
33. S. Ashhab, N. Yamamoto, F. Yoshihara and K. Semba (2022). Numerical analysis of quantum circuits for state preparation and unitary operator synthesis, *Physical Review A*, 106(2), p. 022426.
34. J.-H. Bae, P. M. Alsing, D. Ahn and W. A. Miller (2020). Quantum circuit optimization using quantum Karnaugh map, *Scientific Reports*, 10(1), p. 15651.
35. T. Bahreini and N. Mohammadzadeh (2015). An MINLP Model for Scheduling and Placement of Quantum Circuits with a Heuristic Solution Approach, *ACM Journal on Emerging Technologies in Computing Systems*, 12(3), pp. 1–20.
36. C. Bandyopadhyay, S. Parekh and H. Rahaman (2018). Improved circuit synthesis approach for exclusive-sum-of-product-based reversible circuits, *IET Computers & Digital Techniques*, 12(4), pp. 167–175.
37. A. Banerjee and A. Pathak (2009), *An algorithm for minimization of quantum cost*. Version Number: 2.
38. B. Barán, A. Carballude and M. Villagra (2021). A Multiobjective Approach for Nearest Neighbor Optimization of N-Dimensional Quantum Circuits, *SN Computer Science*, 2(1), p. 19.

39. A. Basak, A. Sadhu, K. Das and K. K. Sharma (2019). Cost Optimization Technique for Quantum Circuits, *International Journal of Theoretical Physics*, 58(9), pp. 3158–3179.
40. C. Beaudoin, K. Phalak and S. Ghosh (2024). AltGraph: Redesigning Quantum Circuits Using Generative Graph Models for E?icient Optimization, in *Proceedings of the Great Lakes Symposium on VLSI 2024*, pp. 44–49, ACM, Clearwater FL USA.
41. A. Bhattacharjee, C. Bandyopadhyay, A. Mukherjee, R. Wille, R. Drechsler and H. Rahaman (2021). An ant colony based mapping of quantum circuits to nearest neighbor architectures, *Integration*, 78, pp. 11–24.
42. D. Bhattacharjee and A. Chattopadhyay (2017). *Depth-Optimal Quantum Circuit Placement for Arbitrary Topologies*. _eprint: 1703.08540.
43. L. Biswal, C. Bandyopadhyay, A. Chattopadhyay, R. Wille, R. Drechsler and H. Rahaman (2016). Nearest-Neighbor and Fault-Tolerant Quantum Circuit Implementation, in *2016 IEEE 46th International Symposium on Multiple-Valued Logic (ISMVL)*, pp. 156–161.
44. L. Biswal, R. Das, C. Bandyopadhyay, A. Chattopadhyay and H. Rahaman (2018). A template-based technique for e?icient Clifford+T-based quantum circuit implementation, *Microelectronics Journal*, 81, pp. 58–68.
45. A. Bocharov, M. Roetteler and K. M. Svore (2015). E?icient synthesis of probabilistic quantum circuits with fallback, *Physical Review A*, 91(5), p. 052317.
46. A. Bocharov and K. M. Svore (2012). Resource-Optimal Single-Qubit Quantum Circuits, *Physical Review Letters*, 109(19), p. 190501.
47. K. Booth, M. Do, J. Beck, E. Rieffel, D. Venturelli and J. Frank (2018). Comparing and Integrating Constraint Programming and Temporal Planning for Quantum Circuit Compilation, *Proceedings of the International Conference on Automated Planning and Scheduling*, 28, pp. 366–374.
48. S. Bravyi, R. Shaydulin, S. Hu and D. Maslov (2021). Clifford Circuit Optimization with Templates and Symbolic Pauli Gates, *Quantum*, 5, p. 580. ArXiv:2105.02291 [quant-ph].
49. M. Ghosh, N. Dey, D. Mitra and A. Chakrabarti (2020). 2D Qubit Placement of Quantum Circuits Using LONGPATH, in R. Chaki, A. Cortesi, K. Saeed, and N. Chaki (editors), *Advanced Computing and Systems for Security*, vol. 996, pp. 127–141, Springer Singapore, Singapore. Series Title: Advances in Intelligent Systems and Computing.
50. A. Chakrabarti and S. Sur-Kolay (2007). Rules for Synthesizing Quantum Boolean Circuits Using Minimized Nearest-Neighbour Templates, in *15th International Conference on Advanced Computing and Communications (ADCOM 2007)*, pp. 183–189, IEEE, Guwahati, Assam, India.
51. M. Chen, Y. Zhang and Y. Li (2021). *A Quantum Circuit Optimization Framework Based on Pattern Matching*, SPIN, 11(03), p. 2140008.
52. X. Cheng, Z. Guan and W. Ding (2018). Mapping from multiple-control Toffoli circuits to linear nearest neighbor quantum circuits, *Quantum Information Processing*, 17(7), p. 169.
53. X. Cheng, Z. Guan and P. Zhu (2020). Nearest Neighbor Transformation of Quantum Circuits in 2D Architecture, *IEEE Access*, 8, pp. 222466–222475.
54. A. M. Childs, E. Schoute and C. M. Unsal (2019). Circuit Transformations for Quantum Architectures, *LIPIcs*, Volume 135, TQC 2019, 135, pp. 3:1–3:24.
55. Y.-H. Chou, C.-Y. Hua, H.-P. Chen, E.-T. Hsu, Y.-C. Jiang and S.-Y. Kuo (2024). Adapting Developing Quantum Circuit Synthesis with a Multi -objective Quantum- inspired Optimization, in *2024 IEEE International Conference on Quantum Computing and Engineering (QCE)*, pp. 559–560, IEEE, Montreal, QC, Canada.
56. D. Clarino, S. Kuroda and S. Yamashita (2025). Reducing T-Count in Quantum Circuits Using Alternate Forms of the Relative Phase Toffoli Gate, *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, E108.A(1), pp. 1–10.
57. P. J. Coles, S. Khatri, R. LaRose, A. T. Sornborger, L. Cincio and A. Poremba (2019). Quantum-assisted quantum compiling, *arXiv*:1807.00800 [quant-ph].
58. A. J. Da Silva and D. K. Park (2022). Linear-depth quantum circuits for multiqubit controlled gates, *Physical Review A*, 106(4), p. 042602.
59. K. Datta, G. Rathi, I. Sengupta and H. Rahaman (2012). Synthesis of Reversible Circuits Using Heuristic Search Method, in *2012 25th International Conference on VLSI Design*, pp. 328–333, IEEE, Hyderabad, India.
60. M. G. Davis, E. Smith, A. Tudor, K. Sen, I. Siddiqi and C. Iancu (2020). Towards Optimal Topology Aware Quantum Circuit Synthesis, in *2020 IEEE International Conference on Quantum Computing and Engineering (QCE)*, pp. 223–234, IEEE, Denver, CO, USA.

61. R. Wille, N. Quetschlich, Y. Inoue, N. Yasuda and S.-i. Minato (2016). Using $$\pi $$ DDs for Nearest Neighbor Optimization of Quantum Circuits, in S. Devitt and I. Lanese (editors), *Reversible Computation*, vol. 9720, pp. 181–196, Springer International Publishing, Cham. Series Title: Lecture Notes in Computer Science.
62. J. Ding and S. Yamashita (2020). Exact Synthesis of Nearest Neighbor Compliant Quantum Circuits in 2-D Architecture and Its Application to Large-Scale Circuits, *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 39(5), pp. 1045–1058.
63. G. W. Dueck, A. Pathak, M. M. Rahman, A. Shukla and A. Banerjee (2018). Optimization of Circuits for IBM's Five-Qubit Quantum Computers, in *2018 21st Euromicro Conference on Digital System Design (DSD)*, pp. 680–684, IEEE, Prague.
64. R. Duncan, A. Kissinger, S. Perdrix and J. Van De Wetering (2020). Graph-theoretic Simplification of Quantum Circuits with the ZX-calculus, *Quantum*, 4, p. 279.
65. E. Younis, K. Sen, K. Yelick and C. Iancu (2021). Qfast: Conflating search and numerical optimization for scalable quantum circuit synthesis, in *2021 IEEE International Conference on Quantum Computing and Engineering (QCE)*, p. 232–243, IEEE.
66. H. Fan, V. Roychowdhury and T. Szkopek (2005). Optimal two-qubit quantum circuits using exchange interactions, *Physical Review A*, 72(5), p. 052323.
67. H. Fan, C. Guo and W. Luk (2022). Optimizing quantum circuit placement via machine learning, in *Proceedings of the 59th ACM/IEEE Design Automation Conference*, pp. 19–24, ACM, San Francisco California.
68. T. Fösel, M. Y. Niu, F. Marquardt and L. Li (2021). Quantum circuit optimization with deep reinforcement learning. *ArXiv*:2103.07585 [quant-ph].
69. X. Zhang, H. Xiang, and T. Xiang (2019). An E?icient Quantum Circuits Optimizing Scheme Compared with QISKit (Short Paper), in H. Gao, X. Wang, Y. Yin, and M. Iqbal (editors), *Collaborative Computing: Networking, Applications and Worksharing*, vol. 268, pp. 467–476, Springer International Publishing, Cham. Series Title: Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering.
70. X. Gao, Z. Guan, S. Feng and Y. Jiang (2023). Quantum Circuit Template Matching Optimization Method for Constrained Connectivity, *Axioms*, 12(7), p. 687.
71. P. Gokhale, S. Koretsky, S. Huang, S. Majumder, A. Drucker, K. R. Brown and F. T. Chong (2021). Quantum Fan-out: Circuit Optimizations and Technology Modeling, in *2021 IEEE International Conference on Quantum Computing and Engineering (QCE)*, pp. 276–290, IEEE, Broomfield, CO, USA.
72. C. P. Williams and A. G. Gray (1999). Automated Design of Quantum Circuits, in G. Goos, J. Hartmanis, J. Van Leeuwen, and C. P. Williams (editors), *Quantum Computing and Quantum Communications*, vol. 1509, pp. 113–125, Springer Berlin Heidelberg, Berlin, Heidelberg. Series Title: Lecture Notes in Computer Science.
73. J. Gorard, M. Namuduri and X. D. Arsiwalla (2021). ZX-Calculus and Extended Wolfram Model Systems II: Fast Diagrammatic Reasoning with an Application to Quantum Circuit Simplification. _eprint: 2103.15820.
74. D. Gosset, V. Kliuchnikov, M. Mosca and V. Russo (2014). An Algorithm for the T-Count, *Quantum Info. Comput.,* 14(15–16), pp. 1261–1276. Place: Paramus, NJ Publisher: Rinton Press, Incorporated.
75. T. Goubault De Brugière, M. Baboulin, B. Valiron and C. Allouche (2020). Quantum circuits synthesis using Householder transformations, *Computer Physics Communications*, 248, p. 107001.
76. D. Grosse, R. Wille, G. W. Dueck and R. Drechsler (2008). Exact Synthesis of Elementary Quantum Gate Circuits for Reversible Functions with Don't Cares, in *38th International Symposium on Multiple Valued Logic (ismvl 2008)*, pp. 214–219, IEEE, Dallas, TX.
77. D. Grosse, R. Wille, G. Dueck and R. Drechsler (2009). Exact Multiple-Control Toffoli Network Synthesis With SAT Techniques, *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 28(5), pp. 703–715.
78. G. G. Guerreschi and J. Park (2018). Two-step approach to scheduling quantum circuits, *Quantum Science and Technology*, 3(4), p. 045003.
79. P. Gupta, A. Agrawal and N. Jha (2006). An Algorithm for Synthesis of Reversible Logic Circuits, *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 25(11), pp. 2317–2330.
80. L. Gyongyosi and S. Imre (2020). Circuit Depth Reduction for Gate-Model Quantum Computers, *Scientific Reports*, 10(1), p. 11229.
81. A. Hawash, A. Awad and B. Abdalhaq (2020). Reversible Circuit Synthesis Time Reduction Based on Subtree-Circuit Mapping, *Applied Sciences*, 10(12), p. 4147.

82. M. M. Rahman and G. W. Dueck (2013), Properties of Quantum Templates, in D. Hutchison, T. Kanade, J. Kittler, J. M. Kleinberg, F. Mattern, J. C. Mitchell, M. Naor, O. Nierstrasz, C. Pandu Rangan, B. Steffen, M. Sudan, D. Terzopoulos, D. Tygar, M. Y. Vardi, G. Weikum, R. Glück, and T. Yokoyama (editors), *Reversible Computation*, vol. 7581, pp. 125–137, Springer Berlin Heidelberg, Berlin, Heidelberg. Series Title: Lecture Notes in Computer Science.
83. N. Abdessaied, M. Soeken and R. Drechsler (2014). Quantum Circuit Optimization by Hadamard Gate Reduction, in D. Hutchison, T. Kanade, J. Kittler, J. M. Kleinberg, A. Kobsa, F. Mattern, J. C. Mitchell, M. Naor, O. Nierstrasz, C. Pandu Rangan, B. Steffen, D. Terzopoulos, D. Tygar, G. Weikum, S. Yamashita, and S.-i. Minato (editors), *Reversible Computation*, vol. 8507, pp. 149–162, Springer International Publishing, Cham. Series Title: Lecture Notes in Computer Science.
84. N. Abdessaied, R. Wille, M. Soeken and R. Drechsler (2013). Reducing the Depth of Quantum Circuits Using Additional Circuit Lines, in D. Hutchison, T. Kanade, J. Kittler, J. M. Kleinberg, F. Mattern, J. C. Mitchell, M. Naor, O. Nierstrasz, C. Pandu Rangan, B. Steffen, M. Sudan, D. Terzopoulos, D. Tygar, M. Y. Vardi, G. Weikum, G. W. Dueck, and D. M. Miller (editors), *Reversible Computation*, vol. 7948, pp. 221–233, Springer Berlin Heidelberg, Berlin, Heidelberg. Series Title: Lecture Notes in Computer Science.
85. Z. Sasanian and D. M. Miller (2013). Reversible and Quantum Circuit Optimization: A Functional Approach, in D. Hutchison, T. Kanade, J. Kittler, J. M. Kleinberg, F. Mattern, J. C. Mitchell, M. Naor, O. Nierstrasz, C. Pandu Rangan, B. Steffen, M. Sudan, D. Terzopoulos, D. Tygar, M. Y. Vardi, G. Weikum, R. Glück, and T. Yokoyama (editors), *Reversible Computation*, vol. 7581, pp. 112–124, Springer Berlin Heidelberg, Berlin, Heidelberg. Series Title: Lecture Notes in Computer Science.
86. R. Iten, R. Moyard, T. Metger, D. Sutter and S. Woerner (2022). Exact and Practical Pattern Matching for Quantum Circuit Optimization, *ACM Transactions on Quantum Computing*, 3(1), pp. 1–41.
87. T. Itoko, R. Raymond, T. Imamichi and A. Matsuo (2020). Optimization of quantum circuit mapping using gate transformation and commutation, *Integration*, 70, pp. 43–50.
88. T. Jones and S. C. Benjamin (2022), Robust quantum compilation and circuit optimisation via energy minimisation, *Quantum*, 6, p. 628.
89. J. Jung and I.-C. Choi (2021). A multi-commodity network model for optimal quantum reversible circuit synthesis, *PLOS ONE*, 16(6), p. e0253140.
90. J. Jung, K. Dalmeijer and P. Van Hentenryck (2024). A New Optimization Model for Multiple-Control Toffoli Quantum Circuit Design, *LIPIcs*, Volume 307, CP 2024, 307, pp. 16:1–16:20.
91. M. Sarvaghad-Moghaddam, P. Niemann and R. Drechsler (2018). Multi-objective Synthesis of Quantum Circuits Using Genetic Programming, in J. Kari and I. Ulidowski (editors), *Reversible Computation*, vol. 11106, pp. 220–227, Springer International Publishing, Cham. Series Title: Lecture Notes in Computer Science.
92. W. Hattori and S. Yamashita (2018). Quantum Circuit Optimization by Changing the Gate Order for 2D Nearest Neighbor Architectures, in J. Kari and I. Ulidowski (editors), *Reversible Computation*, vol. 11106, pp. 228–243, Springer International Publishing, Cham. Series Title: Lecture Notes in Computer Science.
93. A. Kissinger and J. van de Wetering (2020). Reducing the number of non-clifford gates in quantum circuits, *Physical Review A*, 102(2).
94. V. Kliuchnikov, D. Maslov and M. Mosca (2013). Fast and e?icient exact synthesis of single-qubit unitaries generated by Clifford and T gates, *Quantum Information and Computation*, 13(7&8), pp. 607–630.
95. A. Kole, K. Datta and I. Sengupta (2016). A Heuristic for Linear Nearest Neighbor Realization of Quantum Circuits by SWAP Gate Insertion Using n-Gate Lookahead, *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, 6(1), pp. 62–72.
96. T. G. De Brugière, M. Baboulin, B. Valiron, S. Martiel and C. Allouche (2020). Quantum CNOT Circuits Synthesis for NISQ Architectures Using the Syndrome Decoding Problem, in I. Lanese and M. Rawski (editors), *Reversible Computation*, vol. 12227, pp. 189–205, Springer International Publishing, Cham. Series Title: Lecture Notes in Computer Science.
97. Y. Li, W. Liu, M. Li and Y. Li (2023). Quantum circuit compilation for nearest-neighbor architecture based on reinforcement learning, *Quantum Information Processing*, 22(8), p. 295.
98. J. Liu, L. Bello and H. Zhou (2021). Relaxed Peephole Optimization: A Novel Compiler Optimization for Quantum Circuits, in *2021 IEEE/ACM International Symposium on Code Generation and Optimization (CGO)*, pp. 301–314, IEEE, Seoul, Korea (South).

99. M. Lukac, M. Perkowski, H. Goi, M. Pivtoraiko, C. H. Yu, K. Chung, H. Jeech, B.-G. Kim and Y.-D. Kim (2003). Evolutionary Approach to Quantum and Reversible Circuits Synthesis, *Artificial Intelligence Review*, 20(3/4), pp. 361–417.
100. M. Lukac, D. Shah, M. Perkowski and M. Kameyama (2014). Synthesis of Quantum Arrays from Kronecker Functional Lattice Diagrams, *IEICE Transactions on Information and Systems*, E97.D(9), pp. 2262–2269.
101. D. Maslov and B. Zindorf (2022). Depth Optimization of CZ, CNOT, and Clifford Circuits, *IEEE Transactions on Quantum Engineering*, 3, pp. 1–8.
102. D. Maslov and G. Dueck (2006). Level Compaction in Quantum Circuits, in *2006 IEEE International Conference on Evolutionary Computation*, pp. 2405–2409.
103. D. Maslov, C. Young, D. Miller and G. Dueck (2005). Quantum Circuit Simplification Using Templates, in *Design, Automation and Test in Europe*, pp. 1208–1213, IEEE, Munich, Germany.
104. D. Maslov, G. Dueck, D. Miller and C. Negrevergne (2008). Quantum Circuit Simplification and Level Compaction, *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 27(3), pp. 436–444.
105. D. Maslov and M. Saeedi (2011). Reversible Circuit Optimization via Leaving the Boolean Domain, *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 30(6), pp. 806–816. ArXiv: 1103.0215.
106. D. Maslov, G. W. Dueck and D. M. Miller (2007). Techniques for the synthesis of reversible Toffoli networks, *ACM Transactions on Design Automation of Electronic Systems*, 12(4), p. 42.
107. D. Maslov and Y. Nam (2018). Use of global interactions in e?icient quantum circuit constructions, *New Journal of Physics*, 20(3), p. 033018.
108. A. Matsuo, W. Hattori and S. Yamashita (2019). Reducing the Overhead of Mapping Quantum Circuits to IBM Q System, in *2019 IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 1–5.
109. O. D. Matteo and M. Mosca (2016). Parallelizing quantum circuit synthesis, *Quantum Science and Technology*, 1(1), p. 015003.
110. G. Meuli, M. Soeken and G. De Micheli (2018). SAT-based $\{$CNOT, T$\}$ Quantum Circuit Synthesis, in J. Kari and I. Ulidowski (editors), *Reversible Computation*, Lecture Notes in Computer Science, pp. 175–188, Springer International Publishing, Cham.
111. D. M. Miller, D. Maslov and G. W. Dueck (2003). A transformation based algorithm for reversible logic synthesis, in *Proceedings of the 40th annual Design Automation Conference*, pp. 318–323, ACM, Anaheim CA USA.
112. I. Moflic and A. Paler (2023). Cost Explosion for E?icient Reinforcement Learning Optimisation of Quantum Circuits, in *2023 IEEE International Conference on Rebooting Computing (ICRC)*, pp. 1–5, IEEE, San Diego, CA, USA.
113. M. Mohammadi, A. Niknafs, M. Eshghi and G. W. Dueck (2012). DESIGN AND OPTIMIZATION OF SINGLE AND MULTIPLE-LOOP REVERSIBLE AND QUANTUM FEEDBACK CIRCUITS, *Journal of Circuits, Systems and Computers*, 21(03), p. 1250018.
114. N. Mohammadzadeh, M. S. Zamani and M. Sedighi (2009). Improving Latency of Quantum Circuits by Gate Exchanging, in *2009 12th Euromicro Conference on Digital System Design, Architectures, Methods and Tools*, pp. 67–73.
115. M. Möttönen, J. J. Vartiainen, V. Bergholm and M. M. Salomaa (2004). Quantum Circuits for General Multiqubit Gates, *Physical Review Letters*, 93(13), p. 130502.
116. H. Nagarajan, O. Lockwood and C. Coffrin (2021). QuantumCircuitOpt: An Open-source Framework for Provably Optimal Quantum Circuit Design, in *2021 IEEE/ACM Second International Workshop on Quantum Computing Software (QCS)*, pp. 55–63.
117. A. B. Nagy (2021). On an implementation of the Solovay-Kitaev algorithm, *arXiv*:quant-ph/0606077.
118. Y. Nam, N. J. Ross, Y. Su, A. M. Childs and D. Maslov (2018). Automated optimization of large quantum circuits with continuous parameters, *npj Quantum Information*, 4(1), p. 23.
119. P. Niemann, R. Wille and R. Drechsler (2020). Advanced exact synthesis of Clifford+T circuits, *Quantum Information Processing*, 19(9), p. 317.
120. P. Niemann, A. Gupta and R. Drechsler (2019). T-depth Optimization for Fault-Tolerant Quantum Circuits, in *2019 IEEE 49th International Symposium on Multiple-Valued Logic (ISMVL)*, pp. 108–113.
121. S. Niu, A. Hashim, C. Iancu, W. A. De Jong and E. Younis (2024). Effective Quantum Resource Optimization via Circuit Resizing in BQSKit, in *Proceedings of the 61st ACM/IEEE Design Automation Conference*, pp. 1–6, ACM, San Francisco CA USA.

122. A. Pavlidis and D. Gizopoulos (2016). Hierarchical Synthesis of Quantum and Reversible Architectures, *International Journal of Parallel Programming*, 44(5), pp. 1028–1053.
123. J. Paykin, A. T. Schmitz, M. Ibrahim, X.-C. Wu and A. Y. Matsuura (2023). PCOAST: A Pauli-Based Quantum Circuit Optimization Framework, in *2023 IEEE International Conference on Quantum Computing and Engineering (QCE)*, pp. 715–726, IEEE, Bellevue, WA, USA.
124. M. Pedram and A. Shafaei (2016). Layout Optimization for Quantum Circuits with Linear Nearest Neighbor Architectures, *IEEE Circuits and Systems Magazine*, 16(2), pp. 62–74.
125. A. K. Prasad, V. V. Shende, I. L. Markov, J. P. Hayes and K. N. Patel (2006). Data structures and algorithms for simplifying reversible circuits, *ACM Journal on Emerging Technologies in Computing Systems*, 2(4), pp. 277–293.
126. Z. Qi, A. Matsuo and S. Yamashita (2024). Optimization for gaussian elimination-based nna-compliant circuit synthesis method by inserting cnot gates, in *2024 IEEE International Conference on Quantum Computing and Engineering (QCE)*, p. 971–978, IEEE.
127. Y. Qian, Z. Guan, S. Zheng and S. Feng (2023). A Method Based on Timing Weight Priority and Distance Optimization for Quantum Circuit Transformation, *Entropy*, 25(3), p. 465.
128. Raban Iten, Roger Colbeck, Ivan Kukuljan, Jonathan Home and Matthias Christandl (2016). Quantum circuits for isometries, *PHYSICAL REVIEW A*, 93(3), p. 032318.
129. M. M. Rahman, G. W. Dueck and J. D. Horton (2014). An Algorithm for Quantum Template Matching, *ACM Journal on Emerging Technologies in Computing Systems*, 11(3), pp. 1–20.
130. T. Goubault de Brugière, M. Baboulin, B. Valiron and C. Allouche (2019). Synthesizing Quantum Circuits via Numerical Optimization, in J. M. F. Rodrigues, P. J. S. Cardoso, J. Monteiro, R. Lam, V. V. Krzhizhanovskaya, M. H. Lees, J. J. Dongarra, and P. M. Sloot (editors), *Computational Science – ICCS 2019*, vol. 11537, pp. 3–16, Springer International Publishing, Cham. Series Title: Lecture Notes in Computer Science.
131. B. Rosenhahn and T. J. Osborne (2023). Monte Carlo graph search for quantum circuit optimization, *Physical Review A*, 108(6), p. 062615.
132. D. Ruffinelli and B. Barán (2017). Linear nearest neighbor optimization in quantum circuits: a multiobjective perspective, *Quantum Information Processing*, 16(9), p. 220.
133. D. Ruffinelli and B. Baran (2016). A multiobjective approach to linear nearest neighbor optimization for 2D quantum circuits, in *2016 XLII Latin American Computing Conference (CLEI)*, pp. 1–8.
134. C. Ruican, M. Udrescu, L. Prodan and M. Vladutiu (2009). Genetic algorithm based quantum circuit synthesis with adaptive parameters control, in *2009 IEEE Congress on Evolutionary Computation*, pp. 896–903.
135. M. Saeedi, M. S. Zamani and M. Sedighi (2007). Algebraic Characterization of CNOT-Based Quantum Circuits with its Applications on Logic Synthesis, in *10th Euromicro Conference on Digital System Design Architectures, Methods and Tools (DSD 2007)*, pp. 339–346.
136. M. Saeedi, M. Arabzadeh, M. S. Zamani and M. Sedighi (2011). Block-Based Quantum-Logic Synthesis, *Quantum Info. Comput.,* 11(3), pp. 262–277. Place: Paramus, NJ Publisher: Rinton Press, Incorporated.
137. M. Saeedi, M. S. Zamani, M. Sedighi and Z. Sasanian (2010). Reversible circuit synthesis using a cycle-based approach, *ACM Journal on Emerging Technologies in Computing Systems*, 6(4), pp. 1–26.
138. M. Saeedi, R. Wille and R. Drechsler (2011). Synthesis of quantum circuits for linear nearest neighbor architectures, *Quantum Information Processing*, 10(3), pp. 355–377.
139. T. N. Sasamal, A. K. Singh and A. Mohan (2015). Reversible Logic Circuit Synthesis and Optimization Using Adaptive Genetic Algorithm, *Procedia Computer Science*, 70, pp. 407–413.
140. M. Sedlák and M. Plesch (2008). Towards optimization of quantum circuits, *Open Physics*, 6(1).
141. G. Sergioli (2017). *Quantum circuit optimization for unitary operators over non-adjacent qudits*. Version Number: 1.
142. V. Shende, S. Bullock and I. Markov (2006). Synthesis of quantum-logic circuits, *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 25(6), pp. 1000–1010.
143. R. D. d. Silva, E. Pius and E. Kashefi (2013). Global Quantum Circuit Optimization. _eprint: 1301.0351.
144. M. Soeken, G. W. Dueck, M. M. Rahman and D. M. Miller (2016). An extension of transformation-based reversible and quantum circuit synthesis, in *2016 IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 2290–2293.
145. M. Soeken, D. M. Miller and R. Drechsler (2013). Quantum circuits employing roots of the Pauli matrices, *Physical Review A*, 88(4), p. 042322.

146. M. Soeken, R. Wille, G. W. Dueck and R. Drechsler (2010). Window optimization of reversible and quantum circuits, in *13th IEEE Symposium on Design and Diagnostics of Electronic Circuits and Systems*, pp. 341–345.
147. D. Srinivasan, K. Chakrabarti, N. Chopra and A. Dutt (2023). Quantum Circuit Optimization through Iteratively Pre-Conditioned Gradient Descent, in *2023 IEEE International Conference on Quantum Computing and Engineering (QCE)*, pp. 443–449, IEEE, Bellevue, WA, USA.
148. K. Staudacher, T. Guggemos, S. Grundner-Culemann and W. Gehrke (2023). Reducing 2-QuBit Gate Count for ZX-Calculus based Quantum Circuit Optimization, *Electronic Proceedings in Theoretical Computer Science*, 394, pp. 29–45.
149. A. Younes (2018). Using Reed-Muller Expansions in the Synthesis and Optimization of Boolean Quantum Circuits, in S. Stepney and A. Adamatzky (editors), *Inspired by Nature*, vol. 28, pp. 113–141, Springer International Publishing, Cham. Series Title: Emergence, Complexity and Computation.
150. S. Stojković, R. Stanković, C. Moraga and M. Stanković (2020). Reversible Circuits Synthesis from Functional Decision Diagrams by using Node Dependency Matrices, *Journal of Circuits, Systems and Computers*, 29(05), p. 2050079.
151. L. Sünkel, D. Martyniuk, D. Mattern, J. Jung and A. Paschke (2023). GA4QCO: Genetic Algorithm for Quantum Circuit Optimization. _eprint: 2302.01303.
152. O. Susam and M. Altun (2014). An e?icient algorithm to synthesize quantum circuits and optimization, in *2014 21st IEEE International Conference on Electronics, Circuits and Systems (ICECS)*, pp. 570–573, IEEE, Marseille, France.
153. M. Szyprowski and P. Kerntopf (2011). An approach to quantum cost optimization in reversible circuits, in *2011 11th IEEE International Conference on Nanotechnology*, pp. 1521–1526, IEEE, Portland, OR, USA.
154. Y.-y. Tan, X.-y. Cheng, Z.-j. Guan, Y. Liu and H. Ma (2018). Multi-strategy based quantum cost reduction of linear nearest-neighbor quantum circuit, *Quantum Information Processing*, 17(3), p. 61.
155. A. Matsuo and S. Yamashita (2019). An E?icient Method for Quantum Circuit Placement Problem on a 2-D Grid, in M. K. Thomsen and M. Soeken (editors), *Reversible Computation*, vol. 11497, pp. 162–168, Springer International Publishing, Cham. Series Title: Lecture Notes in Computer Science.
156. G. Meuli, B. Schmitt, R. Ehlers, H. Riener and G. De Micheli (2019). Evaluating ESOP Optimization Methods in Quantum Compilation Flows, in M. K. Thomsen and M. Soeken (editors), *Reversible Computation*, vol. 11497, pp. 191–206, Springer International Publishing, Cham. Series Title: Lecture Notes in Computer Science.
157. W. Van Dam, G. M. D'Ariano, A. Ekert, C. Macchiavello and M. Mosca (2007). Optimal Quantum Circuits for General Phase Estimation, *Physical Review Letters*, 98(9), p. 090501.
158. A. Oddi and R. Rasconi (2018). Greedy Randomized Search for Scalable Compilation of Quantum Circuits, in W.-J. Van Hoeve (editor), *Integration of Constraint Programming, Artificial Intelligence, and Operations Research*, vol. 10848, pp. 446–461, Springer International Publishing, Cham. Series Title: Lecture Notes in Computer Science.
159. C. Ruican, M. Udrescu, L. Prodan and M. Vladutiu (2009). Quantum Circuit Synthesis with Adaptive Parameters Control, in L. Vanneschi, S. Gustafson, A. Moraglio, I. De Falco, and M. Ebner (editors), *Genetic Programming*, vol. 5481, pp. 339–350, Springer Berlin Heidelberg, Berlin, Heidelberg. Series Title: Lecture Notes in Computer Science.
160. N. Wiebe and M. Roetteler (2014). Quantum arithmetic and numerical analysis using Repeat-Until-Success circuits, *arXiv*:1406.2040 [quant-ph]. ArXiv: 1406.2040.
161. R. Wille and R. Drechsler (2009). BDD-based synthesis of reversible logic for large functions, in *Proceedings of the 46th Annual Design Automation Conference*, pp. 270–275, ACM, San Francisco California.
162. R. Wille, A. Lye and R. Drechsler (2014). Considering nearest neighbor constraints of quantum circuits at the reversible circuit level, *Quantum Information Processing*, 13(2), pp. 185–199.
163. R. Wille and R. Drechsler (2010). Effect of BDD Optimization on Synthesis of Reversible and Quantum Logic, *Electronic Notes in Theoretical Computer Science*, 253(6), pp. 57–70.
164. R. Wille, A. Lye and R. Drechsler (2014). Exact Reordering of Circuit Lines for Nearest Neighbor Quantum Architectures, *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 33(12), pp. 1818–1831.
165. R. Wille, M. Soeken, C. Otterstedt and R. Drechsler (2013). Improving the mapping of reversible circuits to quantum circuits using multiple target lines, in *2013 18th Asia and South Pacific Design Automation Conference (ASP-DAC)*, pp. 145–150, IEEE, Yokohama.

166. R. Wille, A. Lye and R. Drechsler (2014). Optimal SWAP gate insertion for nearest neighbor quantum circuits, in *2014 19th Asia and South Pacific Design Automation Conference (ASP-DAC)*, pp. 489–494, IEEE, Singapore.
167. R. Wille, M. Saeedi and R. Drechsler (2010). Synthesis of Reversible Functions Beyond Gate Count and Quantum Cost. *ArXiv*:1004.4609 [quant-ph].
168. X.-C. Wu, M. G. Davis, F. T. Chong and C. Iancu (2021). Reoptimization of Quantum Circuits via Hierarchical Synthesis, in *2021 International Conference on Rebooting Computing (ICRC)*, pp. 35–46.
169. Y. Xiao, S. Nazarian and P. Bogdan (2021). A stochastic quantum program synthesis framework based on Bayesian optimization, *Scientific Reports*, 11(1), p. 13138.
170. G. Yang, W. Hung, X. Song and M. Perkowski (2005). Exact synthesis of 3-qubit quantum circuits from non-binary quantum gates using multiple-valued logic and group theory, in *Design, Automation and Test in Europe*, pp. 434–435 Vol. 1.
171. Y.-S. Yang, H.-K. Chen, S.-Y. Kuo, G.-J. Zeng and Y.-H. Chou (2015). A Novel E?icient Optimal Reversible Circuit Synthesis Algorithm, in *2015 IEEE International Conference on Systems, Man, and Cybernetics*, pp. 68–73, IEEE, Kowloon Tong, Hong Kong.
172. A. Younes * and J. F. Miller (2004). Representation of Boolean quantum circuits as reed–Muller expansions, *International Journal of Electronics*, 91(7), pp. 431–444.
173. M. Baioletti, R. Rasconi and A. Oddi (2021). A Novel Ant Colony Optimization Strategy for the Quantum Circuit Compilation Problem, in C. Zarges and S. Verel (editors), *Evolutionary Computation in Combinatorial Optimization*, vol. 12692, pp. 1–16, Springer International Publishing, Cham. Series Title: Lecture Notes in Computer Science.
174. S. Zhang, J. Wu and L. Li (2023). Characterization, synthesis, and optimization of quantum circuits over multiple-control Z -rotation gates: A systematic study, *Physical Review A*, 108(2), p. 022603.
175. S. Zhang, K. Huang and L. Li (2024). Depth-optimized quantum circuit synthesis for diagonal unitary operators with asymptotically optimal gate count, *Physical Review A*, 109(4), p. 042601.
176. Z. Zhang, Z. Guan, H. Zhang, H. Ma and W. Ding (2018). A method for synthesis and optimization for linear nearest neighbor quantum circuits by parallel processing, *Quantum Information and Computation*, 18(13&14), pp. 1095–1114.
177. X. Zhou, Y. Feng and S. Li (2020). A monte carlo tree search framework for quantum circuit transformation, in *Proceedings of the 39th International Conference on Computer-Aided Design*, pp. 1–7, ACM, Virtual Event USA.
178. W. Zhu, Z. Li, G. Zhang, S. Pan and W. Zhang (2018). A Reversible Logical Circuit Synthesis Algorithm Based on Decomposition of Cycle Representations of Permutations, *International Journal of Theoretical Physics*, 57(8), pp. 2466–2474.
179. M. Zomorodi-Moghadam and K. Navi (2016). Rotation-Based Design and Synthesis of Quantum Circuits, *Journal of Circuits, Systems and Computers*, 25(12), p. 1650152.
180. X. Wang (2024). *Quantum speedup, circuit decoupling, and stochastic modelling: on how quantum theory improves machine-learning, and how machine-learning helps to process quantum information*, Ph.D. thesis, Nanyang Technological University.
181. S. Hossain (2009), *Classical Search and Quantum Search Algorithms for Synthesis of Quantum Circuits and Optimization of Quantum Oracles*, Doctor of Philosophy in Electrical and Computer Engineering, Portland State University.
182. M. Amy (2013), *Algorithms for the Optimization of Quantum Circuits*, Master's thesis, University of Waterloo.
183. M. Amy (2019), *Formal Methods in Quantum Circuit Design*, PhD Thesis.
184. Z. Sasanian (2012), *Technology Mapping and Optimization for Reversible and Quantum Circuits*, PhD Thesis, University of Victoria.
185. M. Avitabile (2021), *Proposal for a multi-technology, template-based quantum circuits compilation toolchain*, PhD Thesis, Politecnico di Torino.
186. T. G. de Brugière (2020). *Methods for optimizing the synthesis of quantum circuits. (Méthodes pour l'optimisation de la synthèse de circuits quantiques)*, Ph.D. thesis, University of Paris-Saclay, France.
187. M. Saraivanov (2013). *Quantum Circuit Synthesis using Group Decomposition and Hilbert Spaces*, Master's thesis, Portland State University.
188. K. Staudacher (2021). *Optimization Approaches for Quantum Circuits using ZX-calculus*, Master's thesis, Universitaet Muenchen.

189. A. Y. Kitaev (1995). *Quantum measurements and the abelian stabilizer problem.*
190. C. Jones (2013). Low-overhead constructions for the fault-tolerant Toffoli gate, *Physical Review A*, 87(2), p. 022328.
191. O. Golubitsky and D. Maslov (2012). A Study of Optimal 4-Bit Reversible Toffoli Circuits and Their Synthesis, *IEEE Transactions on Computers*, 61(9), pp. 1341–1353.
192. A. J. Kitaev, A. C. Šen and M. N. Vjalyj (2002). Classical and quantum computation, no. volume 47 in Graduate studies in mathematics, American Mathematical Society, Providence, Rhode Island.
193. M. A. Nielsen and I. L. Chuang (2010). Quantum computation and quantum information, Cambridge University Press, Cambridge; New York, 10th anniversary ed edn.
194. N. Abdessaied and R. Drechsler (2016). Reversible and Quantum Circuits, Springer International Publishing.
195. M. Saeedi and I. L. Markov (2013). Synthesis and optimization of reversible circuits—a survey, *ACM Computing Surveys*, 45(2), pp. 1–34.
196. T. N. Sasamal, H. M. Gaur, A. K. Singh and A. Mohan (2020). Reversible Circuit Synthesis Using Evolutionary Algorithms, in A. K. Singh, M. Fujita, and A. Mohan (editors), *Design and Testing of Reversible Logic*, vol. 577, pp. 115–128, Springer Singapore, Singapore. Series Title: Lecture Notes in Electrical Engineering.
197. K. Karuppasamy, V. Puram, S. Johnson and J. P. Thomas (2025). A comprehensive review of quantum circuit optimization: Current trends and future directions, *Quantum Reports*, 7(1), p. 2.
198. M. Möttönen and J. J. Vartiainen (2006). Decompositions of general quantum gates, *Trends in quantum computing research*, p. 149. ArXiv preprint quant-ph/0504100.
199. V. Bergholm, J. J. Vartiainen, M. Möttönen and M. M. Salomaa (2005). Quantum circuits with uniformly controlled one-qubit gates, *Physical Review A*—Atomic, Molecular, and Optical Physics, 71(5), p. 052330.
200. A. Khadieva, O. Salehi and A. Yakaryı lmaz (2024). A representative framework for implementing quantum finite automata on real devices, in *Proceedings of UCNC 2024*, vol. 14776 of *LNCS*, pp. 163–177.
201. I. Zinnatullin and K. Khadiev (2026). E?icient algorithms for quantum hashing, in *Proceedings of UCNC 2025*, vol. 16364 of *LNCS*. ArXiv preprint arXiv:2507.07002, https://doi.org/10.1007/978-3-032-15641-9_21.
202. I. Zinnatullin, K. Khadiev and A. Khadieva (2023). E?icient implementation of amplitude form of quantum hashing using state-of-the-art quantum processors, *Russian Microelectronics*, 52(Suppl 1), pp. S390–S394.
203. I. Zinnatullin, K. Khadiev and I. Nikitin (2025). Connectivity-aware qubit mapping and cnot cost minimization for quantum hashing circuits, *Russian Microelectronics*.
204. M. G. Davis, E. Smith, A. Tudor, K. Sen, I. Siddiqi and C. Iancu (2019). Heuristics for Quantum Compiling with a Continuous Gate Set. *ArXiv*:1912.02727 [quant-ph].
205. R. Wille, D. Große, L. Teuber, G. W. Dueck and R. Drechsler (2008). Revlib: An online resource for reversible functions and reversible circuits, in *38th International Symposium on Multiple Valued Logic (ismvl 2008)*, pp. 220–225.
206. D. Pastorello, E. Blanzieri and V. Cavecchia (2021). Learning adiabatic quantum algorithms over optimization problems, *Quantum Machine Intelligence*, 3(1), p. 2.
207. M. A. Maltseva, E. Blanzieri and A. S. Rumyantsev (2024). Quantum Circuit Optimization Via Graph Partitioning by Quantum Annealing, *Lobachevskii Journal of Mathematics*, 45(10), pp. 5126–5140.
208. A. Barenco, C. H. Bennett, R. Cleve, D. P. DiVincenzo, N. Margolus, P. Shor, T. Sleator, J. A. Smolin and H. Weinfurter (1995). Elementary gates for quantum computation, *Physical Review A*, 52(5), p. 3457–3467.
209. T. Sleator and H. Weinfurter (1995). Realizable universal quantum logic gates, *Phys. Rev. Lett.,* 74, pp. 4087–4090.
210. Alexis De Vos and Stijn De baerdemacker (2016). Block-ZXZ synthesis of an arbitrary quantum circuit, *Phys. Rev. A*, 94(5), p. 052317.
211. J. Kusyk, S. M. Saeed and M. U. Uyar (2021). Survey on quantum circuit compilation for noisy intermediate-scale quantum computers: Artificial intelligence to heuristics, *IEEE Transactions on Quantum Engineering*, 2, pp. 1–16.
212. X. Zhou, S. Li and Y. Feng (2020). Quantum Circuit Transformation Based on Simulated Annealing and Heuristic Search, *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 39(12), pp. 4683–4694. ArXiv:1908.08853 [quant-ph].
213. A. Botea, A. Kishimoto and R. Marinescu (2021). On the Complexity of Quantum Circuit Compilation, *Proceedings of the International Symposium on Combinatorial Search*, 9(1), pp. 138–142.

214. P. Zhu, S. Zheng, L. Wei, X. Cheng, Z. Guan and S. Feng (2022). The complexity of quantum circuit mapping with fixed parameters, *Quantum Information Processing*, 21(10), p. 361.
215. C.-Y. Lai and H.-C. Cheng (2022). Learning Quantum Circuits of Some T Gates, *IEEE Transactions on Information Theory*, 68(6), pp. 3951–3964.
216. I. F. Araujo, D. K. Park, F. Petruccione and A. J. Da Silva (2021). A divide-and-conquer algorithm for quantum state preparation, *Scientific Reports*, 11(1), p. 6329.
217. M. Kölle, T. Schubert, P. Altmann, M. Zorn, J. Stein and C. Linnhoff-Popien (2024). A Reinforcement Learning Environment for Directed Quantum Circuit Synthesis, in *Proceedings of the 16th International Conference on Agents and Artificial Intelligence*, pp. 83–94, SCITEPRESS - Science and Technology Publications, Rome, Italy.
218. Z. YU, Y. Li, L. Liu and S. Feng (2022). Survey of Quantum Computing Simulation and Optimization Methods, *Computer Engineering*, 48(1), pp. 1–11. (In Chinese).
219. B. Park and D. Ahn (2023). Reducing cnot count in quantum fourier transform for the linear nearest-neighbor architecture, *Scientific Reports*, 13(1), p. 8638.
220. A. Khadieva (2026). Quantum hashing algorithm implementation, *Lobachevskii Journal of Mathematics*. ArXiv:quant-ph/2024.
221. K. Khadiev, A. Khadieva, Z. Chen and J. Wu (2025). Implementation of quantum fourier transform and quantum hashing for a quantum device with arbitrary qubits connection graphs, *Quantum Inforation and computation*, 25.
222. K. Khadiev, A. Khadieva, V. Sagitov and K. Khasanov (2025). Quantum circuit for quantum fourier transform for arbitrary qubit connectivity graphs, in *Proceedings of LATA 2026*, LNCS. ArXiv preprint arXiv:2510.09824.
223. A. Fowler, S. Devitt and L. Hollenberg (2004). Implementation of shor's algorithm on a linear nearest neighbour qubit array, *Quantum Information & Computation*, 4(4), pp. 237–251.
224. A. Barenco, A. Ekert, K.-A. Suominen and P. Törmä (1996). Approximate quantum fourier transform and decoherence, *Physical Review A*, 54(1), p. 139.
225. Y. Takahashi, N. Kunihiro and K. Ohta (2007). The quantum fourier transform on a linear nearest neighbor architecture, *Quantum Information & Computation*, 7(4), pp. 383–391.
226. B. Park and D. Ahn (2022). T-count optimization of approximate quantum fourier transform, *arXiv preprint* arXiv:2203.07739.
227. V. V. Shende, S. S. Bullock and I. L. Markov (2005). Synthesis of quantum logic circuits, in *Proceedings of the 2005 Asia and South Pacific Design Automation Conference*, ASP-DAC '05, p. 272–275, Association for Computing Machinery, New York, NY, USA.
228. V. V. Shende, I. L. Markov and S. S. Bullock (2004). Minimal universal two-qubit controlled-not-based circuits, *Phys. Rev. A*, 69, p. 062321.
229. V. Shende, I. Markov and S. Bullock (2004). Smaller two-qubit circuits for quantum communication and computation, in *Proceedings Design, Automation and Test in Europe Conference and Exhibition*, vol. 2, pp. 980–985 Vol.2.
230. V. V. Shende, I. L. Markov and S. S. Bullock (2004). Minimal universal two-qubit controlled-NOT-based circuits, *Physical Review A*, 69(6), p. 062321.
231. V. V. Shende and I. L. Markov (2009). On the CNOT-Cost of TOFFOLI Gates, *Quantum Info. Comput.*, 9(5), pp. 461–486. Place: Paramus, NJ Publisher: Rinton Press, Incorporated.
232. D. Maslov and W. Yang (2023). CNOT circuits need little help to implement arbitrary Hadamard-free Clifford transformations they generate, *npj Quantum Information*, 9(1), p. 96.
233. F. Vatan and C. Williams (2004). Optimal quantum circuits for general two-qubit gates, *Physical Review A*, 69(3), p. 032315.
234. G. Vidal and C. M. Dawson (2004). Universal quantum circuit for two-qubit transformations with three controlled-NOT gates, *Physical Review A*, 69(1), p. 010301.
235. A. Zlokapa, B. Villalonga, S. Boixo and D. A. Lidar (2023). Boundaries of quantum supremacy via random circuit sampling, *npj Quantum Information*, 9(1).
236. M. Saeedi, N. MohammadZadeh, M. Sedighi and M. S. Zamani (2008). Evaluation and Improvement of Quantum Synthesis Algorithms based on a Thorough Set of Metrics, in *2008 11th EUROMICRO Conference on Digital System Design Architectures, Methods and Tools*, pp. 490–493, IEEE, Parma, Italy.

237. N. Abdessaied, M. Amy, R. Drechsler and M. Soeken (2016). Complexity of reversible circuits and their quantum implementations, *Theoretical Computer Science*, 618, pp. 85–106.