# Quantum Architecture Search: A Survey

Darya Martyniuk
*Fraunhofer FOKUS*
Berlin, Germany
darya.martyniuk@fokus.fraunhofer.de

Johannes Jung
*Freie Universitat Berlin*
*Fraunhofer FOKUS*
Berlin, Germany
johannes.jung@fokus.fraunhofer.de

Adrian Paschke
*Freie Universitat Berlin*
*Fraunhofer FOKUS*
Berlin, Germany
adrian.paschke@fokus.fraunhofer.de

***Abstract*—Quantum computing has made significant progress in recent years, attracting immense interest not only in research laboratories but also in various industries. However, the application of quantum computing to solve real-world problems is still hampered by a number of challenges, including hardware limitations and a relatively under-explored landscape of quantum algorithms, especially when compared to the extensive development of classical computing. The design of quantum circuits, in particular parameterized quantum circuits (PQCs), which contain learnable parameters optimized by classical methods, is a non-trivial and time-consuming task requiring expert knowledge. As a result, research on the automated generation of PQCs, known as quantum architecture search (QAS), has gained considerable interest. QAS focuses on the use of machine learning and optimization-driven techniques to generate PQCs tailored to specific problems and characteristics of quantum hardware. In this paper, we provide an overview of QAS methods by examining relevant research studies in the field. We discuss main challenges in designing and performing an automated search for an optimal PQC, and survey ways to address them to ease future research.**

***Index Terms*—Quantum architecture search, Quantum neural architecture circuit search, Automatic circuit generation, Variable ansatz, Quantum circuit structure search, QAS, AutoML**

## I. Introduction

Advancing quantum computing involves developing and improving corresponding hardware and software systems. In recent years, research endeavors have led to rapid and impressive progress both in the physical realization of quantum computing concepts and in the development of software tools. However, there are still several serious challenges for the research community to overcome before quantum systems can be applied to real-world use cases. Machine learning (ML) and optimization algorithms can be used to open up the potential of hardware devices and expand the possibilities for programming these devices to effectively tackle complex problems. When addressing a particular task with quantum computing, the automation of algorithm design and its execution, including compilation and selection of a suitable device, are promising research directions. Quantum architecture search (QAS) [1] represents a variety of techniques tailored to automate the process of finding an optimal parametrized quantum circuit (PQC).

This work was partially funded by the BMWK project EniQmA (01MQ22007A) and and Einstein Research Unit on Quantum Digital Transformation (ERU-2020-607). *(Darya Martyniuk and Johannes Jung are co-first authors.)*

A PQC is a crucial component of variational quantum algorithms (VQAs), which have gained great interest in the community and have been successfully applied to various domains, such as chemical simulation [2], combinatorial optimization [3], and ML [4], [5]. It depends on learnable parameters $\vec{\theta}$, representing angles of quantum rotation gates. During the iterative execution of a VQA, a classical optimization algorithm adjusts $\vec{\theta}$ in order to minimize the underlying cost function [6]. In PQCs, a subroutine that consist of a sequential application of gates to specific qubits and thus dictates how quantum computation is performed, is called an *ansatz*. Despite the intuitive principle of PQCs, the manual design of a beneficial circuit ansatz is non-trivial, since the criteria for an optimal ansatz in a particular scenario is still an active research area. Moreover, PQCs may encounter trainability issues, such as barren plateaus [7]. To assists developers in creating powerful PQCs, several manually designed ansatz patterns have been presented in the literature, e.g., unitary coupled cluster (UCC) and hardware-efficient ansatz (HEA) [8]. Although the ansatz patterns are useful, they usually need to be tuned in terms of hyperparameters, which is a time-consuming process with no guarantee of success in fitting the particular scenario. Moreover, quantum noise and hardware constraints further impede the performance of ansatz patterns. To overcome these limitations, QAS methods aim to automatically find the optimal structure and the set of parameters for a PQC, tailored to both the underlying problem and the quantum hardware.

In the recent years, several promising techniques for the automated generation of PQCs have been presented. In this paper, we survey QAS methods and its open challenges.

We summarize **our contributions** as follows:

- We outline the relation of QAS to other fields.
- We provide a structured introduction in QAS methods.
- We discuss techniques to increase efficiency of QAS.
- We introduce ideas for further research direction.

This work includes six sections. Section II outlines a brief history of QAS and its connection to other fields. In Section III, components of QAS are introduced. Different search strategies are discussed in Section IV. For each strategy, a short overview is given, followed by summaries of selected studies. In Section V, we review techniques the improve the efficiency of QAS. Possible directions for future research are summarized in Section VI.

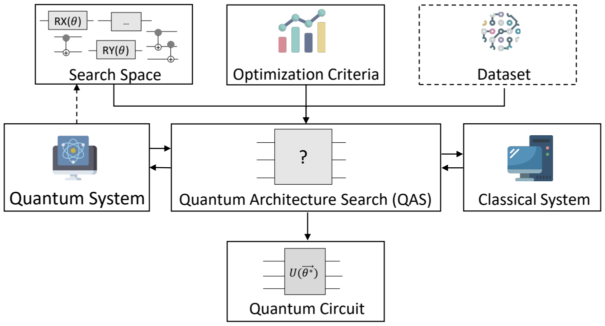


Fig. 1. General overview of QAS: Given a search space (which can be constrained by hardware characteristics), performance criteria, and, in some applications, a dataset, QAS aims to automatically find an optimal PQC with parameters $\vec{\theta}^*$ that maximizes the performance. Quantum and classical systems can be accessed to perform the evaluation of candidate circuits.

## II. QAS Research and Relations to Other Fields

Initial research into the automatic design of quantum circuits dates back to the late 1990s and early 2000s. In early work, researchers used evolutionary algorithms to discover alternative architectures for quantum circuits that are simpler than those constructed manually [9], [10], more efficient than classical [11], and designed without the need for in-depth knowledge of quantum physics [12]. With the growing accessibility of quantum hardware, simulation environments, and software tools, the interest on the automatic generation of quantum circuits for specific problems has been increased. Since then, a wide range of ML techniques such as deep reinforcement learning [13]–[16], differentiable algorithms [1], [17]–[20], and Bayesian optimization [21]–[23] have been used to generate or optimize quantum circuits in various application areas, e.g., chemistry [24]–[30], ML [18], [31], [32], and optimization [15], [33]. The motivation driving the research remains in encoding problems into quantum systems without requiring extensive expertise in quantum computing, designing efficient and noise-resilient circuits in light of limitations of noisy intermediate-scale quantum (NISQ) hardware, and overcoming algorithmic-specific issues, e.g., trainability issues. It should be noted that over the years, the automated design of quantum circuits has been termed in the scientific literature also as quantum circuit design search [9], [10], [22], quantum circuit search [34], [35], ansatz architecture search [36], adaptive variational quantum algorithms [37], quantum circuit learning [38], [39], quantum neural architecture search (QNAS) [21], and hybrid optimization [40]. We used these notations to identify relevant studies for this survey. However, some of the publications cover the generation of quantum circuits in general rather than PQCs. Therefore, although these studies have influenced existing QAS methods, they are outside the scope of this survey.

Many QAS techniques introduced in recent years have been strongly inspired by *neural architecture search (NAS)*, which is the process of automated engineering of neural network architectures for a given task. NAS methods have already been successfully applied to a variety of tasks, such as large-scale image classification [41], segmentation [42], and text classification [43]. NAS aligns closely with *hyperparameter optimization (HPO)*, which aims to automate the search for optimal hyperparameters of a ML model, i.e., parameters used for the setup of the model or the optimizer (e.g., learning rate, type of optimizer) [44]. NAS and HPO methods can be considered as subfields of *automated machine learning (AutoML)*. AutoML intends to automate the entire pipeline of a ML model including data preparation and processing, feature engineering, algorithm and architecture selection (e.g., with NAS), and HPO. In contrast to related classical fields, QAS extends the idea of architecture search beyond (Q)ML models because of the variety of VQA applications. Moreover, due to the current high impact of the concrete hardware on the success of a circuit execution, a lot of QAS studies take into account hardware characteristics developing hardware-aware QAS methods [8], [25], [33], [34]. The resilience of QAS solutions to noise is another crucial property in the current NISQ-era. It has been shown that the presence of noise causes the estimated values of the cost function for given parameters to differ from those in a noiseless environment [45]. Thus, a number of QAS approaches [34], [39], [45] define noise resilience as an important property of candidate circuits.

Several works survey techniques and breakthroughs in automated generation of quantum circuits. Reference [46] provide a survey on evolving quantum algorithms using genetic programming. However, as VQAs had not yet been introduced at the time the work was created, the automated generation of PQCs is not taken into account here. A brief overview of QAS techniques with emphasis on search strategies is provided in [47]. Some of QAS methodologies are outlined in [48] in the context of a broader review of ansatz designing techniques. However, given the active research effort on this topic and novel ideas offered in the recent years, there is a need of a comprehensive overview of QAS methods and challenges.

## III. Background

In this work, we define *QAS* as follows. Given a task to be solved, QAS aims to automatically design a PQC optimized against specified performance criteria. Besides to the *task* that specifies the search objective, the process of QAS can incorporate input *data*, e.g., in case of quantum ML applications, or additional *constraints*, e.g., on properties of quantum hardware such as type of supported gates, number of qubits, or qubit connectivity, as illustrated on Fig. 1. QAS methods can be delineated along four dimensions depicted on Fig. 2. Note that this categorization is heavily inspired by NAS-related surveys [49], [50].

***Search space*** specifies the set of potential architectures that can be discovered. Due to the large search space, it is typical for QAS studies to constrain it, e.g., by manually constructing the set of gates available for the circuit design. The constraints can be derived from the authors' experience or task's requirements. Alternatively, the constrained gate set may be derived from hardware properties, such as the native gate set supported by a particular quantum machine.

TABLE I
SEARCH STRATEGIES AND TASKS IN QAS STUDIES

| Strategy | Task | Reference |
|---|---|---|
| Reinforcement Learning | Combinatorial Optimization (QAOA) | [15] |
| | Ground State Approximation (VQE) | [61], CRLQAS [45] |
| | Diagonalization (VQSD) | RL-VQSD [62] |
| Evolutionary Algorithms | Ground State Approximation (VQE) | QuantumNAS [34], EVQE [33], MoG-VQE [63], QCEAT [64] |
| | Classification | QuantumNAS [34], MQNE [31], [32] |
| Generative Models | Ground State Approximation (VQE) | GQE [65] |
| Random Search (with One-Shot or Predictor Evaluation) | Classification | [52],Élivágar [35], [51], GSQAS [55], [66] |
| | Ground State Approximation (VQE) | [52], TF-QAS [67], [51],GSQAS [55], PQAS-AL [68], [69] |
| Differentiable QAS | Noise Reduction | DQAS [1], QuantumDARTS [18] |
| | Ground State Approximation (VQE) | QuantumDARTS [18] |
| | Classification | QuantumDARTS [18] |
| | Combinatorial Optimization (QAOA) | DQAS [1], QuantumDARTS [18] |
| | Quantum Reinforcement Learning | DQAS-RL [19] |
| | Compilation | MetaQAS [70], [20] |
| Bayesian Optimization | Quantum Program Synthesis | [21] |
| | Combinatorial Optimization (QAOA) | [21] |
| | Distribution Approximation | [21] |
| | Classification | [22], QES [23] |
| Adaptive Methods | Quantum Data Compression (Quantum Autoencoder) | VAns [27] |
| | Ground State Approximation (VQE) | ADAPT-VQE [24], VAns [27], Rotoselect [71], |
| | Compilation | VAns [27], QAQC [72], NACL [39] |
| | Combinatorial Optimization (QAOA) | AAS [36] |
| | Quantum Program Synthesis | [73] |
| | Observable Extraction | NACL [39] |
| | State Preparation | NACL [39] |
| Monte-Carlo Tree Search | Error Detection | [58] |
| | Linear Equations | [58] |
| | Ground State Approximation (VQE) | [28], [58] |
| | Combinatorial Optimization (QAOA) | [58], MCTS-QAOA [40] |
| | Quantum Program Synthesis | MCGS [57] |
| | Classification | MCGS [57] |
| | Cellular Automation | MCGS [57] |

and mutation to create new individuals. Evaluate the fitness of each individual. Select individuals via the selection strategy to form the next generation.

In [63], a multi-objective genetic VQE is build, called MoG-VQE, which aims to optimize the ansatz structure while minimizing the two-qubit gate count. Genes are characterized as blocks composed of one or several CNOT gates and several rotation operators. Mutation operations add or delete those blocks to an individual at a random position. To reduce the time of the parameter optimization, the authors used an EA-based optimization scheme called the Covariance-Matrix Adaptation Evolutionary Strategy [78]. In [32], a multi-objective GA is implemented to search for resource-efficient feature maps for quantum kernels.

Reference [33] introduces the Evolutionary Quantum Eigensolver (EVQE). It implements a weight-sharing strategy (see Section V-A) and initializes the parameters of a rotation gates such that the gates perform the identity transformation, yielding to a much faster convergence than random initialization. Genes are characterized as layers, which only include universal rotations, controlled universal rotations and identity gates. The algorithm explicitly forbids crossover operations, arguing that combining two circuits does not guarantee that any of the parent properties will be inherited in terms of cost evaluation. The asexual reproduction strategy provides a similarity metric, defined by the number of mutations to a common ancestor of two individuals. This similarity metric then can be used to encourage diversity within a population by penalizing individuals that are too similar to each other, such that the risk of falling into local optima reduces.

In [31], an algorithm termed Markovian quantum neuroevolution (MQNE) is introduced. MQNE encodes one-layered circuits into nodes of a directed graph and identifies quantum circuits with paths in that graph. For the circuit generation, only adjacent controlled $R_x$ gates and universal single-qubit rotation gates are allowed. A directed edge between two nodes is added if two one-layer circuits do not produce ambiguities when executed successively. Then, an evolutionary search is introduced to find a good path in the resulting graph.

A robust and resource-efficient multi-species EA, Quantum Circuit Evolution of Augmenting Topologies (QCEAT), is proposed in [64]. QCEAT maintains multiple subpopulations that initially evolve independently but eventually migrate into each other. This approach allows greater diversity and exploration. To improve the robustness of the individuals, the authors enhance the fitness function with robustness measures.

### *C. Differentiable QAS*

Differentiable QAS methods relax the discrete search space of quantum architectures into a continuous and differentiable domain. For the search space $\mathcal{S} = \{s_1, s_2, ..., s_n\}$ of quantum architectures, every candidate solution $s \in \mathcal{S}$ is assigned a probability $p_s(\alpha_s)$, depending on a variable $\alpha_s$. A loss function $L(\alpha)$ has to be defined based on the probability distribution $p(\alpha) = (p_1(\alpha_1), ..., p_n(\alpha_n))$ such that it is differentiable in $\alpha$ in some sense. Thus, the minimum $\alpha^* = \arg\min_\alpha L(\alpha)$ can be attained through a gradient descend method. In the end, Deep RL, allow multiple agents to learn independently and in parallel in different copies of the environment, e.g., as realized in [108], the development of QAS methods specifically designed for parallel processing needs to be investigated in future research.

A closely related approach involves distributing the search across multiple devices that can compute different parts of the task simultaneously. A distributed search across multiple small-scale QPUs interconnected by quantum links is introduced in [69]. Distributed quantum computing (DQC) involves, in addition to local gates operating on the qubits of a single QPU (data qubits), also non-local gates affecting pairs of qubits on different devices (communication qubits). Thus, distributed QAS need to incorporate the graph representation of the distributed quantum system as an input and take into account local and non-local types of gates as well as the optimization of implementation methods for non-local gates. Although the distributed QAS may enable generation of circuits for large-scale problems, the search complexity is greater than in a single QPU scenario. This need to be further investigated, combined with open issues in the DQC research.

## VII. Conclusion

The landscape of quantum computing is rapidly evolving. While the potential of quantum systems for solving complex problems is beyond doubt, the designing of quantum algorithms in the current NISQ-era remains challenging. Collaboration across research areas can accelerate the design of quantum methods and lead to a better understanding of the quantum computing paradigm. In this survey paper, we outlined techniques and research directions for the automated generation of PQCs using ML and optimization methods. Due to the inherent complexity of PQCs and the limitations of NISQ devices, QAS research faces significant time and resource challenges. We summarized established techniques to address the computational overhead and concluded the survey with discussing possible future research directions.

## References


[1] S.-X. Zhang, C.-Y. Hsieh, S. Zhang, and H. Yao, "Differentiable quantum architecture search," *Quantum Science and Technology*, vol. 7, no. 4, p. 045023, 2022.

[2] A. Peruzzo, J. McClean, P. Shadbolt, M.-H. Yung, X.-Q. Zhou, P. J. Love, A. Aspuru-Guzik, and J. L. O'brien, "A variational eigenvalue solver on a photonic quantum processor," *Nature communications*, vol. 5, no. 1, p. 4213, 2014.

[3] E. Farhi, J. Goldstone, and S. Gutmann, "A quantum approximate optimization algorithm," *arXiv preprint arXiv:1411.4028*, 2014.

[4] E. Farhi and H. Neven, "Classification with quantum neural networks on near term processors," *arXiv preprint arXiv:1802.06002*, 2018.

[5] M. Schuld, A. Bocharov, K. M. Svore, and N. Wiebe, "Circuit-centric quantum classifiers," *Physical Review A*, vol. 101, no. 3, p. 032308, 2020.

[6] M. Cerezo, A. Arrasmith, R. Babbush, S. C. Benjamin, S. Endo, K. Fujii, J. R. McClean, K. Mitarai, X. Yuan, L. Cincio *et al.*, "Variational quantum algorithms," *Nature Reviews Physics*, vol. 3, no. 9, pp. 625–644, 2021.

[7] M. Cerezo, A. Sone, T. Volkoff, L. Cincio, and P. J. Coles, "Cost function dependent barren plateaus in shallow parametrized quantum circuits," *Nature communications*, vol. 12, no. 1, p. 1791, 2021.

[8] A. Kandala, A. Mezzacapo, K. Temme, M. Takita, M. Brink, J. M. Chow, and J. M. Gambetta, "Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets," *nature*, vol. 549, no. 7671, pp. 242–246, 2017.

[9] C. P. Williams and A. G. Gray, "Automated design of quantum circuits," in *Quantum Computing and Quantum Communications*, C. P. Williams, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 1999, pp. 113–125.

[10] T. Yabuki and H. Iba, "Genetic algorithms for quantum circuit design-evolving a simpler teleportation circuit," in *Late Breaking Papers at the 2000 Genetic and Evolutionary Computation Conference*, 2000, pp. 421–425.

[11] L. Spector, H. Barnum, H. Bernstein, and N. Swamy, "Finding a better-than-classical quantum and/or algorithm using genetic programming," in *Proceedings of the 1999 Congress on Evolutionary Computation-CEC99 (Cat. No. 99TH8406)*, vol. 3, 1999, pp. 2239–2246 Vol. 3.

[12] B. I. Rubinstein, "Evolving quantum circuits using genetic programming," in *Proceedings of the 2001 congress on evolutionary computation (IEEE Cat. No. 01TH8546)*, vol. 1. IEEE, 2001, pp. 144–151.

[13] E.-J. Kuo, Y.-L. L. Fang, and S. Y.-C. Chen, "Quantum architecture search via deep reinforcement learning," *arXiv preprint arXiv:2104.07715*, 2021.

[14] X. Zhu and X. Hou, "Quantum architecture search via truly proximal policy optimization," *Scientific Reports*, vol. 13, no. 1, p. 5157, 2023.

[15] K. A. McKiernan, E. Davis, M. S. Alam, and C. Rigetti, "Automated quantum programming via reinforcement learning for combinatorial optimization," *arXiv preprint arXiv:1908.08054*, 2019.

[16] S. Y.-C. Chen, "Asynchronous training of quantum reinforcement learning," *Procedia Computer Science*, vol. 222, pp. 321–330, 2023.

[17] Z. He, C. Chen, L. Li, S. Zheng, and H. Situ, "Quantum architecture search with meta-learning," *Advanced Quantum Technologies*, vol. 5, no. 8, p. 2100134, 2022.

[18] W. Wu, G. Yan, X. Lu, K. Pan, and J. Yan, "Quantumdarts: differentiable quantum architecture search for variational quantum algorithms," in *International Conference on Machine Learning*. PMLR, 2023, pp. 37 745–37 764.

[19] Y. Sun, Y. Ma, and V. Tresp, "Differentiable quantum architecture search for quantum reinforcement learning," in *2023 IEEE International Conference on Quantum Computing and Engineering (QCE)*, vol. 2. IEEE, 2023, pp. 15–19.

[20] Z. He, C. Chen, H. Situ, F. Zhang, S. Zheng, and L. Li, "A meta-trained generator for quantum architecture search," Jan. 2024, unpublished.

[21] T. Duong, S. T. Truong, M. Pham, B. Bach, and J.-K. Rhee, "Quantum neural architecture search with quantum circuits metric and bayesian optimization," in *ICML 2022 2nd AI for Science Workshop*, 2022.

[22] M. Pirhooshyaran and T. Terlaky, "Quantum circuit design search," *Quantum Machine Intelligence*, vol. 3, pp. 1–14, 2021.

[23] N. Nguyen and K.-C. Chen, "Quantum embedding search for quantum machine learning," *IEEE Access*, vol. 10, pp. 41 444–41 456, 2022.

[24] H. R. Grimsley, S. E. Economou, E. Barnes, and N. J. Mayhall, "An adaptive variational algorithm for exact molecular simulations on a quantum computer," *Nature communications*, vol. 10, no. 1, p. 3007, 2019.

[25] H. L. Tang, V. Shkolnikov, G. S. Barron, H. R. Grimsley, N. J. Mayhall, E. Barnes, and S. E. Economou, "qubit-adapt-vqe: An adaptive algorithm for constructing hardware-efficient ansätze on a quantum processor," *PRX Quantum*, vol. 2, no. 2, p. 020310, 2021.

[26] I. G. Ryabinkin, R. A. Lang, S. N. Genin, and A. F. Izmaylov, "Iterative qubit coupled cluster approach with efficient screening of generators," *Journal of chemical theory and computation*, vol. 16, no. 2, pp. 1055–1063, 2020.

[27] M. Bilkis, M. Cerezo, G. Verdon, P. J. Coles, and L. Cincio, "A semi-agnostic ansatz with variable structure for variational quantum algorithms," *Quantum Machine Intelligence*, vol. 5, no. 2, p. 43, 2023.

[28] F.-X. Meng, Z.-T. Li, X.-T. Yu, and Z.-C. Zhang, "Quantum circuit architecture optimization for variational quantum eigensolver via monto carlo tree search," *IEEE Transactions on Quantum Engineering*, vol. 2, pp. 1–10, 2021.

[29] D. A. Fedorov, Y. Alexeev, S. K. Gray, and M. Otten, "Unitary selective coupled-cluster method," *Quantum*, vol. 6, p. 703, 2022.

[30] R. A. Lang, I. G. Ryabinkin, and A. F. Izmaylov, "Unitary transformation of the electronic hamiltonian with an exact quadratic truncation of the baker-campbell-hausdorff expansion," *Journal of Chemical Theory and Computation*, vol. 17, no. 1, pp. 66–78, 2020.