Advances in Reliability, Safety and Security, Part 3 – Kołowrocki, Dąbrowska (eds.)
 ISBN 978-83-68136-15-9 (printed), ISBN 978-83-68136-02-9 (electronic)

Advances in Reliability, Safety and Security

ESREL 2024
Monograph Book Series

Quantum Based Optimization For Critical Infrastructure Health Monitoring

Gabriel San Martín Silva, Enrique López Droguett

Civil and Environmental Engineering Department, University of California Los Angeles, Los Angeles, California, USA
The John B. Garrick Institue for the Risk Sciences, University of California Los Angeles, Los Angeles, California, USA

Abstract

Over the last few decades, the increased availability and affordability of remote sensing equipment have opened the field for the implementations of sophisticated approaches to perform data-driven health monitoring in a wide variety of engineering systems. An industry sector where these advances hold the potential to cause a large impact is Structural Health Monitoring, mostly due to the high capital costs involved in the design and construction of structural systems, and the human risk involved in manual inspections. However, the scale and complexity of modern structural systems make it impossible to install sensors in all relevant locations within a structure. This limitation forces practitioners to select an optimal sensor configuration under certain criteria for fitness, giving rise to the problem commonly known as the Sensor Placement Optimization problem. Due to its combinatorial nature, this problem is impossible to solve exactly in practical situations, requiring the use of heuristic and metaheuristic approaches. While genetic algorithms and simulated annealing are examples of successful techniques used to find near-optimal solutions, the increasing size and complexity of structural systems motivates the exploration of novel alternatives. Quantum computing is an exciting field that offers a series of interesting approaches to solve combinatorial problems. However, their use and capabilities within the Sensor Placement Optimization context are currently underexplored. To bridge this important gap, this paper presents a novel approach to solve Sensor Placement Optimization problems by transforming the Modal Strain Energy criteria into a form suitable for quantum algorithms. Additionally, we provide an exploration of different training techniques for the quantum technique, in the hope that their results may inspire and guide future research in the area. Our experiments and approach are validated using a numerical modal of a Warren Truss bridge.

Keywords: quantum computing, combinatorial optimiziation, Sensor Placement Optimization, QAOA

1. Introduction

Within the field of Structural and Infrastructure Health Monitoring, one of the key challenges is the optimal placement of sensing equipment, a problem commonly known as Sensor Placement Optimization (SPO) (Hassani and Dackermann, 2023). The SPO problem can be stated as finding the best k-out-of-N candidate locations to place sensors within a system, in accordance with some criteria for fitness. In structural engineering, the candidate locations usually coincide with the degrees of freedom of the structure. The SPO problem is a combinatorial optimization problem that scales exponentially with the size of the structural system. For that reason, it is considered intractable in practical situations, requiring practitioners to rely on heuristics or meta-heuristics approaches to obtain suitable solutions. Common approaches used in existing literature have included genetic and evolutionary algorithms as well as simulated annealing (Ostachowicz et al., 2019; Tan and Zhang, 2020). However, the increasing size and complexity of structural systems motivate the exploration of novel approaches that can find near-optimal solutions in scenarios where an exceedingly large solution space exists.

Over the last few years, quantum computing has attracted the interest of researchers and practitioners alike due to its potential to solve challenging tasks more efficiently than classical computers. In this regard, a promising application is quantum-based combinatorial optimization (Heng et al., 2022). While a wide variety of

approaches exists within quantum computation to solve optimization problems, the most promising one is the Quantum Approximate Optimization Algorithm (Farhi et al., 2014), or QAOA for short. The reasons for this are threefold. First, QAOA is meant to be executed on a gate-based quantum device, a universal architecture that is expected to be the predominant hardware implementation in the near future. Second, it has attracted considerable attention since its discovery in 2014; nowadays multiple variants and modifications of QAOA exist to solve a wide range of different optimization problems (see (Blekos et al., 2024) for a complete review). Finally, the QAOA can be categorized as a meta-heuristic technique and consequently can be used in any binary optimization problem with an objective function that accepts a quadratic formulation.

However, despite QAOA's potential, its application to the field of Structural Health Monitoring remains unexplored. There exists a large gap between the novel approaches heralded by quantum computing theory and their application to practical problems in Structural Health Monitoring (SHM). One of the few examples of such an application is given by Speziali et al. (2021), who applied quantum annealing to find a near-optimal sensor configuration in a water distribution system. In the author's opinion, several pieces are missing to bridge this relevant gap. First, the QAOA is designed to find optimal solutions for Quadratic Unconstrained Binary Optimization problems, and most, if not all objective functions used in structural SPO do not have a quadratic form. Second, there exists a lack of common ground between traditional quantum computing research and applied fields, such as structural health monitoring and reliability. Finally, the QAOA presents multiple design choices, and general guidelines for these types of decisions have not been studied within the context of structural health monitoring.

Following up on our previous work in the area (San Martín and López Droguett, 2024), this paper attempts to bridge the aforementioned gap by presenting three main contributions:

1. A novel formulation of the commonly used SPO criteria called "Modal Strain Energy" as a QUBO instance in order to solve it using the QAOA approach.
2. A math-based explanation of Quantum Computing theory, prioritizing a functional point of view centered on what a quantum computer does and how it can be used to perform practical engineering computation.
3. The exploration of different training approaches for the QAOA with the objective of guiding the community towards future developments.

This article is structured as follows: Section 2 provides a concise introduction to SPO theory within a structural context, with a particular focus on the Modal Strain Energy criteria and how it can be transformed into a QUBO instance. In Section 3, we offer a brief yet comprehensive introduction to quantum computing theory, emphasizing a mathematical explanation while avoiding quantum mechanics wherever possible. Section 4 presents the theory behind the QAOA approach, detailing its implementation in a quantum computer simulator and its application to solving the SPO problem. Section 5 validates the proposed approach using a case study involving a Warren Truss bridge numerical model. Finally, in Section 6, we provide concluding remarks with a focus on outlining research avenues that we believe hold promise for future exploration.

2. Sensor placement optimization in structural engineering

Most SHM strategies rely on the common hypothesis that changes in the dynamical properties within a structural system are highly correlated with the presence of degradation. As such, damage is usually detected by comparing the dynamical characterization of a given system against a healthy baseline constructed at the beginning of the structure's service life or by using a finely tuned numerical system.

The main dynamical properties of a structural system are summarized in its mass, damping, stiffness, and modal shape matrices, all of which can be easily obtained using finite element analysis, even for large-scale structures. Given the relevance of detecting changes in these dynamical properties for the characterization of the health state of a structural system, most SPO strategies focus on positioning sensors in locations where a large amount of dynamical information can be measured.

A commonly used criterion in SPO is the Modal Strain Energy (MSE) (Tan and Zhang, 2020) maximization, where the objective is to find a candidate sensor configuration S that maximizes the amount of elastic energy that can be sensed by the equipment installed in the structure. Empirically, it has been found that configurations that capture a large amount of elastic energy correlate with higher signal-to-noise ratios, favoring the system characterization and therefore the detection of existent damage.

Mathematically, the MSE captured by a given candidate sensor configuration S is given by (1), where N_{DOF} is the total number of degrees of freedom within the structure where sensors can be placed, ϕ_{ij} is the i-th component of the j-th modal shape vector and k_{ij} is the (i, j) component of the stiffness matrix K. Formally, the candidate sensor configuration S is a collection of indices representing the degrees of freedom where sensors are proposed to be installed.

$$MSE(S) = \sum_{i=1}^{N_{DOF}} \sum_{j=1}^{N_{DOF}} \sum_{p\in S} \sum_{q\in S} \left|\phi_{pi} k_{pq} \phi_{qj}\right| \tag{1}$$

For reasons that will become evident in Section 4, when we describe the QAOA approach, we need to transform the MSE formulation presented in (1) into a Quadratic Unconstrained Binary Optimization (QUBO) formulation. To this end, first, we encode the candidate sensor configuration S into a binary vector $\vec{x}$ using the formula presented in (2), where the index $i \in \{1, \dots, N_{DOF}\}$.

$$x_i = \begin{cases} 1, & DOF_i \in S \\ 0, & DOF_i \notin S. \end{cases} \tag{2}$$

Using this representation, we can rewrite (1) as a QUBO instance, presented in (3).

$$MSE(\vec{x}) = \sum_{i=1}^{N_{DOF}} \sum_{j=1}^{N_{DOF}} \sum_{p=1}^{N_{DOF}} \sum_{q=1}^{N_{DOF}} \left|\phi_{pi} k_{pq} \phi_{qj}\right| x_p x_q \,. \tag{3}$$

Equation (3) can be used directly as the objective function to find the optimal sensor configuration for a given structure. Formally, the problem we are trying to solve using quantum computing is described in (4), where we have included an equality constraint to indicate that we only wish to place n_s sensors in the structure, a common limitation in practice due to budgetary reasons.

$$\begin{aligned} &\max_{\vec{x}\in\{0,1\}} MSE(\vec{x}) \\ &s.t. \sum_{i=1}^{N_{DOF}} x_i = n_s \end{aligned} \tag{4}$$

3. Quantum computing background

A quantum computer is a physical machine capable of ***creating***, ***modifying***, and ***measuring*** a quantum state. A quantum state, in mathematical terms, is a complex vector $|\psi_1\rangle$ of arbitrary dimensionality, where the ket notation ($|\cdot\rangle$) is used for vector representation. The simplest possible quantum state corresponds to the case $N = 2$. In this case, the quantum state is commonly referred to as a quantum bit or ***qubit***. The most general example of a qubit is the vector $|\psi\rangle = [c_0 \;\; c_1]^T$, where c_0 and c_1 are complex numbers.

3.1. Forming quantum states

A quantum computer is generally composed of several independent qubits, which can be combined together to form quantum states of larger dimensionality. Mathematically, the operation used to join qubits together is the Kronecker product operation ($\otimes$). The result of applying this operation to a pair of qubits $|\psi_1\rangle = [c_{10}\; c_{11}]^T$ and $|\psi_2\rangle = [c_{20}\; c_{21}]^T$ is shown in (5).

$$|\psi_1\rangle \otimes |\psi_2\rangle = \begin{bmatrix} c_{10}c_{20} \\ c_{10}c_{21} \\ c_{11}c_{20} \\ c_{11}c_{21} \end{bmatrix} \tag{5}$$

Equation (5) shows that the result of applying the Kronecker operator to qubits $|\psi_1\rangle$ and $|\psi_2\rangle$ is a quantum state of dimensionality equal to four. By extension, the application of the Kronecker product to a set of k qubits results in a quantum state of dimensionality equal to 2^k. In this sense, the sizes of the quantum states that a quantum computer can hold are restricted to the powers of 2.

Quantum states, like any other vector, can also be described as a linear combination using a set of basis vectors. The most common set of basis vectors used in practice is the set of unitary orthogonal basis vectors given by $\{|e_i\rangle\}_{i=1}^{2^k}$, where $|e_i\rangle$ is a unitary vector with a 1 in the i-th position and 0 otherwise. Using this basis, a quantum state can be written as:

$$|\psi\rangle = \sum_{i=1}^{2^k} c_i |e_i\rangle, \quad c_i \in \mathbb{C} \tag{6}$$

This decomposition will become important later in this section when we describe how quantum computing is used to solve practical problems.

3.2. Modifying quantum states

A quantum state $|\psi_1\rangle \in \mathbb{C}^{2^k \times 1}$ can be transformed into a different quantum state $|\psi_2\rangle \in \mathbb{C}^{2^k \times 1}$ by the application of a unitary matrix $U \in \mathbb{C}^{2^k \times 2^k}$ using the traditional rules of matrix multiplication from linear algebra. A unitary matrix fulfills that its inverse is also its conjugate transpose, i.e., $UU^\dagger = U^\dagger U = I$. As a result of that, they are norm preserving, meaning that they preserve the norm of the vectors they operate on.

Unitary operations can also be applied successively to a quantum state in order to generate multiple transformation steps, as shown in (7).

$$|\psi_M\rangle = \left(\prod_{j=1}^{M} U_j \right) |\psi_1\rangle \tag{7}$$

Notably, these unitary operations can also depend on external, user-defined parameters. These types of matrices allow for the encoding of external data into the quantum computer, such as the coefficients controlling an objective function of an optimization problem.

3.3. Measurement of quantum states

In the previous section, we reviewed how a quantum computer forms and modifies quantum states. Now, we turn our attention to how a quantum computer is capable of reading information from the quantum state.

A fundamental rule of quantum mechanics states that the 2^k complex coefficients that compose a quantum state cannot be read directly. Instead, a quantum state can be "measured", which is an operation that allows the quantum computer to estimate the square norm of its complex coefficients. By performing a measurement, the quantum computer allows us to stochastically obtain one of the basis vectors $|e_i\rangle$ of the quantum state. The basis vector that is retrieved depends on the following probability distribution:

$$p(|e_i\rangle) = \|c_i\|^2 \tag{8}$$

In other words, the probability of obtaining the basis vector $|e_i\rangle$ when performing a measurement operation over the quantum state is the squared norm of the corresponding complex coefficient. Equation (8) allows us to derive three important conclusions regarding the measurement of quantum states. First, the quantum state can be interpreted as a mathematical object encoding a categorical probability distribution over 2^k elements, where k is the number of qubits used to form it. Second, while the measurement process facilitates the estimation of the squared norm of the complex coefficients of a quantum state (i.e., the probability of obtaining each basis vector), all the information contained in the real and imaginary parts is lost. Third, (8) provides a justification for the requirement of matrices in quantum computing to be unitary: these matrices preserve the norm of vectors upon which they are applied and, therefore, allow for the underlying probability distribution to remain valid.

3.4. Quantum computing framework

So far, we have reviewed the three main operations that a quantum computer can perform. However, one question remains: how can the quantum computer utilize these operations to solve a combinatorial optimization problem?

The framework used by quantum computing to solve combinatorial optimization tasks is summarized in Figure 1.

First, an initial quantum state $|\psi_1\rangle \in \mathbb{C}^{2^k \times 1}$ is formed using the Kronecker operation over k qubits. Then, each one of the candidate solutions of the original optimization problem is identified with one of the basis vectors of the quantum state. In general, quantum computers can easily generate an initial quantum state encoding a uniform probability distribution over all its basis vectors. Consequently, at the beginning of the computations, all solutions are equally suitable. In the following stage, the initial quantum state is modified by applying a series of user-defined unitary operations. This process effectively transforms the probability distribution encoded into the initial quantum state and often receives the name of *quantum circuit design*. In this stage, the objective function of the optimization problem is encoded into the quantum computer, using an approach called QAOA, which is described in Section 4. Finally, the quantum state, $|\psi_M\rangle \in \mathbb{C}^{2^k \times 1}$ that results from this transformation is measured, obtaining one of the basis vectors.

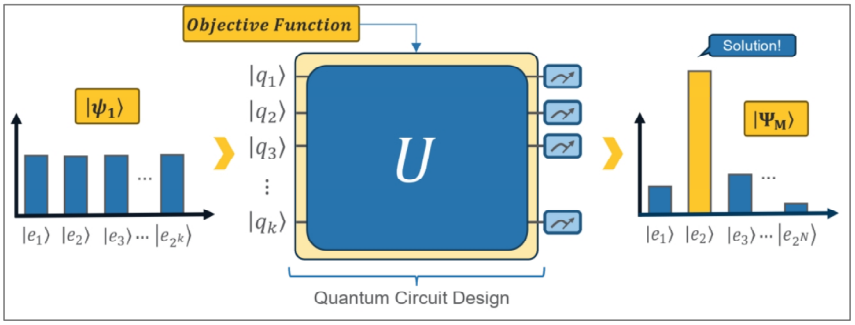


Fig. 1. Framework used by quantum computing to solve combinatorial optimization tasks.

These steps are repeated multiple times, allowing the estimation of the probability distribution encoded in $|\psi_M\rangle$. As such, a quantum computer can be summarized as a machine capable of altering categorical probability distributions over exponentially larger event sets. For the case of combinatorial optimization problems, this is very useful as the discrete set of feasible solutions can be directly identified with each one of the basis vectors. The goal is that during the quantum circuit design stage, the unitary operation applied to $|\psi_1\rangle$ generates a final probability distribution that assigns a high likelihood to solutions that minimize (or maximize) the problem's objective function.

In the following section, we will review how this unitary operation is formed using the Quantum Approximate Optimization Algorithm (QAOA) (Farhi et al., 2014).

4. Quantum combinatorial optimization: the Quantum Approximate Optimization Algorithm (QAOA)

We begin this section by providing a brief physical motivation for quantum-based combinatorial optimization. As described in Section 3, a quantum computer is a physical machine capable of holding and modifying a quantum state. This quantum state is not a virtual object, but the actual physical state of the quantum computer itself. As such, it has all the properties that one would expect from a quantum system. In particular, it has an energy configuration that will tend to a local minimum if the system is left to evolve naturally. The general idea behind quantum-based optimization is to interpret the problem's objective function and the set of candidate solutions as the system's energy and the system's states, respectively. By doing this, solving the original optimization problem can be thought of as "evolving" the quantum computer's state to find its minimum (or maximum) energy state. In practice, we would expect the final measurement process to assign a higher likelihood to those solutions that are optimal or near optimal.

The process of "evolution" of a quantum state is based on the Quantum Adiabatic Theorem (Crosson et al., 2014), which roughly states that if a system is initially prepared in a minimum energy state for configuration A, and then is slowly transitioned towards configuration B, then the final state will correspond to a minimum energy state of configuration B. The main algorithm to implement this adiabatic evolution process in quantum computers is the Quantum Approximate Optimization Algorithm (QAOA) (Farhi et al., 2014). A main shortcoming of the QAOA is that it is only designed to find near-optimal solutions for Quadratic Unconstrained Binary Optimization (QUBO) problems (Glover et al., 2019). However, as explained later in this section, we can slightly modify some components of the algorithm to allow the incorporation of constraints over the solution vectors.

Figure 2 shows the QAOA quantum circuit diagram corresponding to the transition from configuration A to configuration B. First, an initial state is prepared by operation U_0. This initial quantum state encodes a uniform probability distribution, assigning equal weight to all candidate solutions. Subsequently, a set of unitary operations referred to as the cost (U_C) and mixing (U_M) unitaries, are applied a total of p times to the initial quantum state. The cost operation U_C is the one that encodes the objective function of the original optimization problem and therefore represents the configuration B. The mixing operation U_M is a very simple matrix that represents configuration A. These unitary operations are parameterized by the set of tunable parameters $\{\gamma_j, \beta_j\}_{j=1}^{p}$, which control how fast should the transition from configuration A to configuration B occur. According to quantum computing theory, a high number of circuit applications p tends to correlate with finding better solutions for the original optimization problem.

The performance of a particular configuration of the QAOA is not trivial to compute. The reason for this is that the output of QAOA is not a single solution, but a probability distribution over all the elements that constitute the feasible solution space. To this end, we first proposed the following performance metric for a single solution, denominated normalized performance, $r(\vec{x})$:

$$r(\vec{x}) = \begin{cases} \dfrac{MSE(\vec{x}) - MSE_{min}}{MSE_{max} - MSE_{min}}, & if \sum_{i=1}^{N_{DOF}} x_i = n\_s \\ 0, & otherwise \end{cases} \tag{9}$$

where MSE_{min} and MSE_{max} are the MSE obtained by the worst and best feasible configurations, respectively. The normalized performance metric assigns a value of 1 for the best configuration and a value of 0 for the worst and unfeasible configurations. Using this formulation, we can compute the performance of QAOA by computing the expected normalized performance under the probability distribution generated by the quantum circuit, as described in (10).

$$r_{QAOA}\left(\{\gamma_j, \beta_j\}_{j=1}^{P}\right) = \sum_{i=1}^{2^k} p_{QAOA}(\vec{x}_i) \cdot r(\vec{x}_i) \tag{10}$$

where $p_{QAOA}(\vec{x}_i)$ is the probability assigned to the basis vector $|e_i\rangle$ that represents the solution vector $\vec{x}_i$.

In a practical implementation, the QAOA uses a quantum computer in tandem with a classical computer. The quantum computer is tasked with estimating the probability distribution generated by a particular selection of the parameter set $\{\gamma_j, \beta_j\}_{j=1}^{P}$, while the classical computer is used to compute the normalized performance and propose new parameter sets that hopefully march toward an optimum. Consequently, the QAOA can be interpreted as an algorithm that facilitates the transformation of an optimization problem over binary variables into one that depends now on a continuous set of tunable parameters. As such, we can employ a much wider set of techniques to find near-optimal solutions, including powerful gradient-based techniques.

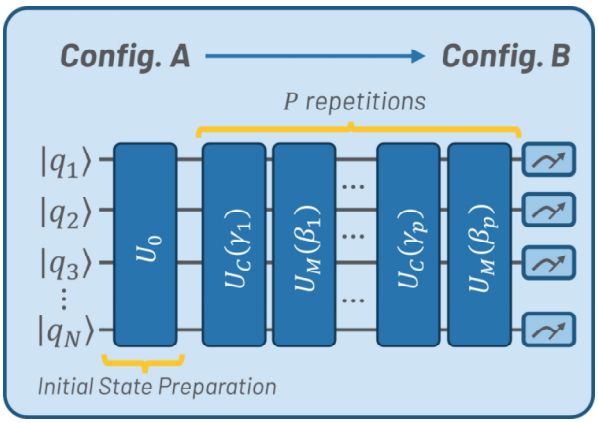


Fig. 2. Quantum circuit design of the Quantum Approximate Optimization Algorithm (QAOA).

Nevertheless, the SPO problem that we wish to solve in this paper involves an equality constraint; see (4)). In practical terms, this constraint implies that the problem is no longer a QUBO. However, equality constraints can easily be incorporated into QAOA by modifying the initial state operator and mixing operation, U_0 and U_M, to alternative operators denoted in the literature as U_0^{XY} and U_M^{XY} (Brandhofer et al., 2022). These operators modify the behavior of the original QAOA algorithm, restricting the space of solutions that it can evaluate to only those that fulfill the equality constraint. For a more detailed explanation of QAOA, including how the unitary operations U_0, U_0^{XY}, U_C, U_M and U_M^{XY} are constructed from a series of basic unitary matrices, the reader is referred to our long-format paper on the topic (San Martin and Lopez Droguett, 2023).

5. Case study: truss structural system

Figure 3 shows the case study used to validate the proposed approach presented in Section 4, consisting of a Warren truss bridge numerical model. The 2D structural model consists of 18 identical elements of length $L = 2[m]$, cross-sectional area $A = 1.42 \times 10^{-2}\ [m^2]$, density $\rho = 7850\ \left[\frac{kg}{m^3}\right]$, and Young's modulus $E = 200\ [GPa]$. This configuration results in a structure with 22 degrees of freedom. Of these, only 18 are non-supported and therefore represent valid candidate locations for sensors. The left and right supports restrict the movement in both its vertical and horizontal degrees of freedom. For all the experiments shown in this paper, we assume that $n_s = 4$ sensors will be installed in the structure. Thus, the total number of feasible sensor configurations is $\binom{18}{4} = 3060$.

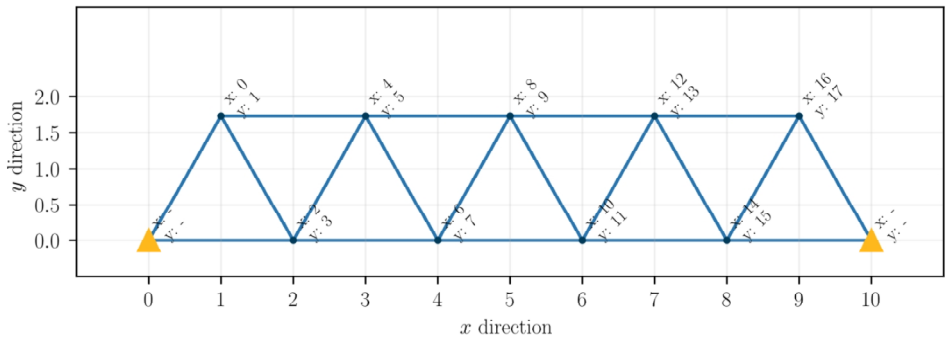


Fig. 3. Numerical model of the Warren truss bridge used as a validation example in this paper.
Note that the number of candidate degrees of freedom is 18.
Consequently, if $n_s = 4$ sensors are to be installed in the structure,
and the total number of candidate configurations is $\binom{18}{4} = 3060$.

The mass, stiffness, and modal matrices are obtained by applying the Finite Element (FE) methodology to this 2D structure, using truss elements for each structural member (Ferreira and Fantuzzi, 2020). To validate the proposed approach, the case study is designed such that the SPO problem is solvable using an exhaustive search method. This enables the comparison of the solutions obtained with the QAOA against a true baseline via the normalization performance metric; see (10)). Table 1 shows the top 3 configurations in accordance with the MSE criteria.

Table 1. Top-3 sensor configurations in accordance with the Modal Strain Energy criteria. Obtained using an exhaustive search approach.

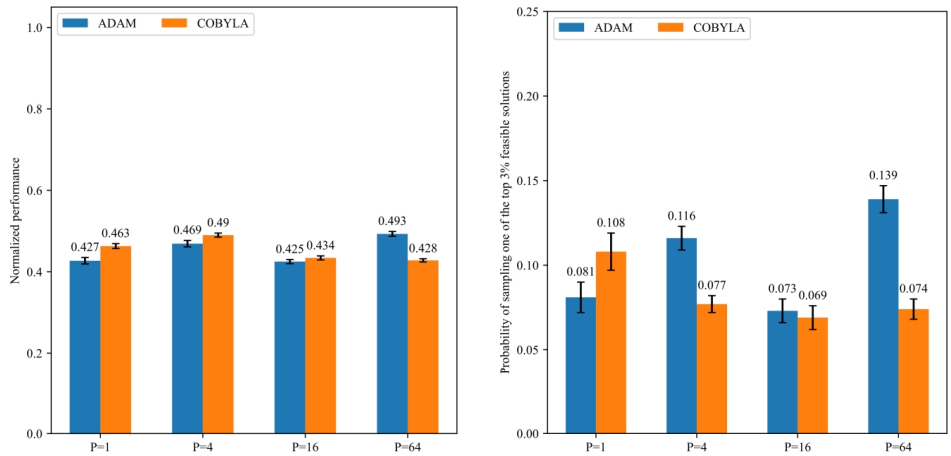
Solution Ranking	Normalized Performance, $r(\vec{x})$	Sensor Locations [DOFs]
1st	1.0	[4, 6, 14, 16]
2nd	0.968	[10, 12, 14, 16]
3rd	0.968	[4, 6, 8, 10]

All experiments are performed using a quantum computing simulation environment executed on a traditional computer using PennyLane 0.33.1 (Bergholm et al., 2020) and Python 3.10.8. The workstation is equipped with an AMD Ryzen Threadripper PRO 3955WX 16-Core processor and 128 GB of RAM. The use of a simulation environment enables the execution of quantum computer algorithms without requiring access to quantum hardware. The focus is placed on examining the impact of two design factors: the number of circuit repetitions and the selection of classical optimization techniques on the outcomes generated by the (QAOA). To this end, we investigate values for P within the set $\{1, 4, 16, 64\}$. Furthermore, we optimize the parameter set $\{\gamma_j, \beta_j\}_{j=1}^{p}$ from a starting point identified through the computation of a preliminary exploration of the optimization landscape (San Martín and López Droguett, 2024). The optimization is carried using two distinct algorithms: COBYLA (Larson et al., 2019), a gradient-free method, and ADAM (Kingma and Ba, 2014), which requires the computation of gradients. The code to obtain the results shown in this paper is available at (Gabriel San Martin Silva, 2024).

Figure 4(a) shows the normalized performance achieved by QAOA under different values of P and different optimization techniques. The error bar indicates the standard deviation achieved after 5 optimization runs. From the figure, it is clear that COBYLA demonstrates marginally superior performance when a lower number of circuit repetitions is employed, while at higher values of p, the situation is reversed. This phenomenon can be attributed to ADAM's proficiency in optimizing functions over a larger number of parameters (for example, in Deep Learning contexts). Despite these variations, the normalized performance does not exhibit a substantial increase with higher values of circuit repetitions, as was expected. This suggests a potential influence of the relatively modest size of the problem. The normalized performances generally fluctuate between 0.4 and 0.5, with the most favorable outcome observed when employing the ADAM optimizer with $p = 64$.

While the results portrayed in Figure 4(a) may not immediately appear remarkable, it is important to remember that the QAOA does not return a single candidate solution, but rather a distribution over all feasible solutions. In practical settings, the QAOA circuit would be measured multiple times, proposing the best solution found as the near-optimal configuration. Consequently, an alternative metric to assess the quality of the solutions found by the QAOA is to compute the probability of sampling one of the top-performing solutions. As long as this probability is not negligible, executing the optimized QAOA circuit a reasonable number of times and preserving the best solution ensures that near-optimal solutions will be found. Figure 4(b) shows the probability of sampling one of the top 3% performing sensor configuration from the QAOA distribution under varying circuit depth levels and both optimization techniques tested.

From the figure, COBYLA only significantly surpasses the performance obtained with ADAM for the smallest circuit tested, corresponding to $p = 1$. For all other cases, ADAM is able to achieve higher probabilities of sampling a near-optimal solution. The greater performance is obtained for the case $p = 64$, where ADAM achieves a 13.9% probability of sampling one of the top 3% sensor configurations. These results indicate the importance of assessing the QAOA performance under different lenses, as the normalized performance may not provide a fully comprehensive picture of its capabilities.



(a) Normalized Performance metric for varying number of circuit repetitions.

(b) Probability of obtaining one of the top 3% feasible solutions for varying number of circuit repetitions.

Fig. 4. (a) Normalized performance obtained by the QAOA algorithm under different number of circuit repetitions using the COBYLA and ADAM optimizers. (b) Probability of sampling one of the 3% feasible solutions under different numbers of circuit repetitions using the COBYLA and ADAM optimizers.

6. Concluding remarks

This paper has proposed a novel approach to perform Sensor Placement Optimization in structural systems by leveraging the capabilities of Quantum Computing techniques. In particular, it applied the Quantum Approximate Optimization Algorithm (QAOA) to find near-optimal solutions using as an objective function a modified version of the Modal Strain Energy criteria. The proposed methodology's efficacy was validated using an 18-element Warren truss bridge numerical model. In particular, the number of circuit repetitions within the

QAOA and the classical optimizer used to tune the parameter set $\{\gamma_j, \beta_j\}_{j=1}^{p}$ were varied to derive practical design guidelines.

The results clearly showed that for a high number of circuit repetitions, the optimizer ADAM seems to be a better option to find near-optimal sensor configurations. This result may be attributed to the fact that the ADAM optimizer was designed for training deep learning models consisting of thousands (or even millions) or optimizable parameters, and as such it seems like a preferable option for QAOA circuits where the parameter set is large. Additionally, the results outline the importance of testing the performance of QAOA using different metrics. For example, even when the normalized performance metric did not show remarkable results for the QAOA approach, a closer look at the probability of sampling near-optimal solutions reveals that the proposed methodology is capable of producing high-quality solutions in non-trivial instances.

Regarding future research paths in the area, it is the authors' opinion that there is a need for developing new Quadratic Unconstrained Binary Optimization (QUBO) formulations for existing problems in infrastructure and structural health monitoring to allow their exploration using quantum-based optimization techniques.

References

Bergholm, V., Izaac, J., Schuld, M., Gogolin, C., Alam, M. S., Ahmed, S., Arrazola, J. M., Blank, C., Delgado, A., Jahangiri, S., McKiernan, K., Meyer, J. J., Niu, Z., Száva, A., Killoran, N. 2020. PennyLane: Automatic differentiation of hybrid quantum-classical computations. arXiv:1811.04968 [Physics, Physics:Quant-Ph]. http://arxiv.org/abs/1811.04968

Blekos, K., Brand, D., Ceschini, A., Chou, C.-H., Li, R.-H., Pandya, K., Summer, A. 2024. A review on Quantum Approximate Optimization Algorithm and its variants. Physics Reports, 1068, 1-66. https://doi.org/10.1016/j.physrep.2024.03.002

Brandhofer, S., Braun, D., Dehn, V., Hellstern, G., Hüls, M., Ji, Y., Polian, I., Bhatia, A. S., Wellens, T. 2022. Benchmarking the performance of portfolio optimization with QAOA. https://doi.org/10.48550/arXiv.2207.10555

Crosson, E., Farhi, E., Lin, C. Y.-Y., Lin, H.-H., Shor, P. 2014. Different Strategies for Optimization Using the Quantum Adiabatic Algorithm (arXiv:1401.7320). arXiv. https://doi.org/10.48550/arXiv.1401.7320

Farhi, E., Goldstone, J., Gutmann, S. 2014. A Quantum Approximate Optimization Algorithm. arXiv:1411.4028 [Quant-Ph]. http://arxiv.org/abs/1411.4028

Ferreira, A. J. M., Fantuzzi, N. 2020. Trusses in 2D Space. In A. J. M. Ferreira N. Fantuzzi (Eds.), MATLAB Codes for Finite Element Analysis: Solids and Structures, 57–75. Springer International Publishing. https://doi.org/10.1007/978-3-030-47952-7_4

Gabriel San Martin Silva. 2024. gsanmartinsilva/esrel2024_repo: Esrel2024_paper (v1.0.0) [Computer software]. [object Object]. https://doi.org/10.5281/ZENODO.11041141

Glover, F., Kochenberger, G., Du, Y. 2019. A Tutorial on Formulating and Using QUBO Models (arXiv:1811.11538). arXiv. https://doi.org/10.48550/arXiv.1811.11538

Hassani, S., Dackermann, U. 2023. A Systematic Review of Optimization Algorithms for Structural Health Monitoring and Optimal Sensor Placement. Sensors 23(6), Article 6. https://doi.org/10.3390/s23063293

Heng, S., Kim, D., Kim, T., Han, Y. 2022. How to Solve Combinatorial Optimization Problems Using Real Quantum Machines: A Recent Survey. IEEE Access, 10, 120106–120121. https://doi.org/10.1109/ACCESS.2022.3218908

Ostachowicz, W., Soman, R., Malinowski, P. 2019. Optimization of sensor placement for structural health monitoring: A review. Structural Health Monitoring 18(3), 963–988. https://doi.org/10.1177/1475921719825601

San Martin, G., Lopez Droguett, E. 2023. Quantum-Based Combinatorial Optimization for Structural Health Monitoring Applications (arXiv:2305.08738). arXiv. https://doi.org/10.48550/arXiv.2305.08738

San Martín, G., López Droguett, E. 2024. Quantum-Based Combinatorial Optimization for Optimal Sensor Placement in Civil Structures. Structural Control and Health Monitoring, 2024, e6681342. https://doi.org/10.1155/2024/6681342

Speziali, S., Bianchi, F., Marini, A., Menculini, L., Proietti, M., Termite, L. F., Garinei, A., Marconi, M., Delogu, A. 2021. Solving Sensor Placement Problems In Real Water Distribution Networks Using Adiabatic Quantum Computation. 2021 IEEE International Conference on Quantum Computing and Engineering (QCE), 463-464. https://doi.org/10.1109/QCE52317.2021.00079

Tan, Y., Zhang, L. 2020. Computational methodologies for optimal sensor placement in structural health monitoring: A review. Structural Health Monitoring 19(4), 1287-1308. https://doi.org/10.1177/1475921719877579