

Article

Application of Quantum Computers and Their Unique Properties for Constrained Optimization in Engineering Problems: Welded Beam Design

Dawid Ewald 

Department of Intelligent Systems, Faculty of Telecommunications, Computer Science and Electrical Engineering, Bydgoszcz University of Science and Technology, 85-796 Bydgoszcz, Poland; dawid.ewald@pbs.edu.pl

Abstract: The welded beam design problem represents a real-world engineering challenge in structural optimization. The objective is to determine the optimal dimensions of a steel beam and weld length to minimize cost while satisfying constraints related to shear stress (τ), bending stress (σ), critical buckling load (P_c), end deflection (δ), and side constraints. The structural analysis of this problem involves the following four design variables: weld height (x_1), weld length (x_2), beam thickness (x_3), and beam width (x_4), which are commonly denoted in structural engineering as h, l, t, b respectively. The structural formulation of this problem leads to a nonlinear objective function, which is subject to five nonlinear and two linear inequality constraints. The optimal solution lies on the boundary of the feasible region, with a very small feasible-to-search-space ratio, making it a highly challenging problem for classical optimization algorithms. This paper explores the application of quantum computing to solve the welded beam optimization problem, utilizing the unique properties of quantum computers for constrained optimization in engineering problems. Specifically, we employ the D-Wave quantum computing system, which utilizes quantum annealing and is particularly well-suited for solving constrained optimization problems. The study presents a detailed formulation of the problem in a format compatible with the D-Wave system, ensuring the efficient encoding of constraints and objective functions. Furthermore, we analyze the performance of quantum computing in solving this problem and compare the obtained results with classical optimization methods. The effectiveness of quantum computing is evaluated in terms of computational efficiency, accuracy, and its ability to navigate complex, constrained search spaces. This research highlights the potential of quantum algorithms in tackling real-world engineering optimization problems and discusses the challenges and limitations of current quantum hardware in solving practical industrial application issues.



Academic Editor: Carlo Mastroianni

Received: 30 January 2025

Revised: 1 March 2025

Accepted: 4 March 2025

Published: 4 March 2025

Citation: Ewald, D. Application of Quantum Computers and Their Unique Properties for Constrained Optimization in Engineering Problems: Welded Beam Design. *Electronics* **2025**, *14*, 1027. <https://doi.org/10.3390/electronics14051027>

Copyright: © 2025 by the author. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: quantum computing in engineering optimization; d-wave quantum optimization; welded beam design with quantum computing

1. Introduction

Optimisation problems constitute a fundamental category of mathematical challenges aimed at identifying the best possible solution within defined constraints. In practical terms, this involves maximising or minimising an objective function, which represents efficiency, cost, or another property of the analysed system. The resolution of optimisation problems is particularly significant in the field of engineering, where the design and optimisation of structures, processes, and technological systems play a pivotal role [1,2].

In engineering, optimisation problems frequently encompass complex design decisions, such as minimising material costs, maximising structural strength, or improving system efficiency. These problems are inherently challenging due to numerous constraints, including stresses, loads, deformations, and geometric limitations. The optimisation of real-world engineering problems, such as the design of mechanical components or load-bearing structures, is critical for reducing production costs, enhancing safety and performance, and improving the environmental efficiency of engineering processes.

A well-documented example of an optimisation problem in engineering is the welded beam design problem. This task involves determining the optimal dimensions of structural components to minimise production costs while adhering to constraints related to stresses, buckling, and deflection. This problem serves as a prominent benchmark in the literature for evaluating the performance of advanced optimisation algorithms [3].

In recent years, quantum computers, such as systems developed by D-Wave, have emerged as a promising tool for tackling challenging optimisation problems. Quantum computers leverage phenomena of quantum mechanics, including superposition, entanglement, and quantum tunnelling, enabling the simultaneous exploration of numerous solutions in the search space. D-Wave's specific approach, known as quantum annealing, is particularly effective in solving combinatorial optimisation problems. This method exploits the natural tendency of physical systems to move towards a state of minimum energy, facilitating the identification of low-energy states that correspond to optimal or near-optimal solutions [4–6].

Thus, optimisation using advanced tools like D-Wave systems opens new horizons in addressing complex engineering challenges while improving the precision and efficiency of the design process [7,8].

This integration of cutting-edge computational methodologies into engineering optimisation marks a transformative step towards solving intricate problems with enhanced speed and accuracy.

2. The Welded Beam Problem as an Optimization Problem

The optimization of a welded beam design involves finding the optimal dimensions of structural elements to minimize production costs while simultaneously meeting the specified strength and geometric constraints, as illustrated in the diagram (Figure 1), where the key variable names are highlighted [9–11].

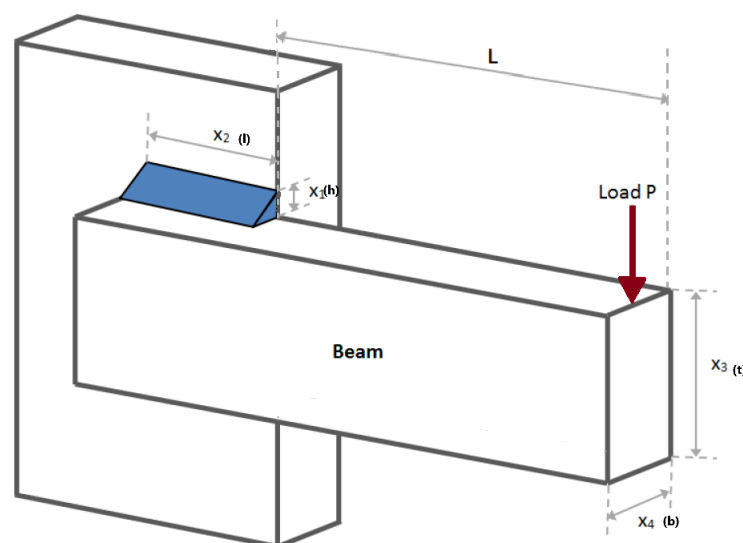


Figure 1. The welded beam problem formulated as an optimization problem.

2.1. Design Variables

1. $x_1 = h$: height of the weld.
2. $x_2 = l$: length of the weld.
3. $x_3 = t$: thickness of the beam.
4. $x_4 = b$: width of the beam.

2.2. Objective Function

The fabrication cost $f(X)$ to be minimized is as follows:

$$f(X) = 1.10471x_1^2 + 0.04811x_3x_4(L + x_2) \quad (1)$$

where $L = 14.0$ in.

Meaning: The objective function represents the manufacturing cost of the welded beam. Minimizing this function helps optimize material and production costs.

Explanation: It is crucial for the economic efficiency of the project, as optimization should lead to cost savings while meeting structural requirements.

2.3. Constraints

The design is subject to the following constraints:

1. Shear stress constraint

$$\tau(X) - \tau_{max} \leq 0 \quad (2)$$

Meaning: Ensures that shear stresses in the beam do not exceed the allowable limit τ_{max} , preventing structural failure.

Explanation: Exceeding this limit could lead to weld joint failure.

2. Bending stress constraint

$$\sigma(X) - \sigma_{max} \leq 0 \quad (3)$$

Meaning: Prevents bending stress from exceeding σ_{max} , ensuring beam stability.

Explanation: The structure must withstand loads without plastic deformation or fractures.

3. Deflection constraint

$$\delta(X) - \delta_{max} \leq 0 \quad (4)$$

Meaning: Controls the beam's deflection under load.

Explanation: Excessive deflection may lead to structural failure and usability issues.

4. Geometric constraint

$$x_1 - x_4 \leq 0 \quad (5)$$

Meaning: Ensures that the weld height x_1 does not exceed the beam width x_4 .

Explanation: Satisfying this condition ensures proper weld quality and structural integrity.

5. Buckling load constraint

$$P - P_c(X) \leq 0 \quad (6)$$

Meaning: Guarantees that the critical buckling load is higher than the applied load, preventing buckling failure.

Explanation: Exceeding the critical load may cause sudden instability and failure.

6. Minimum weld thickness constraint

$$0.125 - x_1 \leq 0 \quad (7)$$

Meaning: Ensures that the weld has a minimum acceptable height for adequate strength. Why is it important? Too small a weld may weaken the joint.

7. Cost constraint

$$f(X) - 5.0 \leq 0 \quad (8)$$

Meaning: Imposes a maximum allowable manufacturing cost.

Explanation: Ensures the economic feasibility of the design.

2.4. Design Variables' Ranges

$$0.1 \leq x_1 \leq 2, \quad 0.1 \leq x_2 \leq 10, \quad 0.1 \leq x_3 \leq 10, \quad 0.1 \leq x_4 \leq 2 \quad (9)$$

2.5. Material Properties and Parameters

- Applied force: $P = 600$ lb.
- Beam length: $L = 14$ in.
- Maximum deflection: $\delta_{\max} = 0.25$ in.
- Young's modulus: $E = 30 \times 10^6$ psi.
- Shear modulus: $G = 12 \times 10^6$ psi.
- Maximum shear stress: $\tau_{\max} = 13,600$ psi.
- Maximum bending stress: $\sigma_{\max} = 30,000$ psi.

2.6. Stress and Deflection Formulas

1. Shear stress (τ):

$$\tau = \sqrt{\tau'^2 + \frac{\tau' \tau''}{x_2^2 R} + \tau''^2} \quad (10)$$

where

$$\tau' = \frac{P}{\sqrt{2}x_1x_2}, \quad \tau'' = \frac{M}{R}, \quad M = P\left(L + \frac{x_2}{2}\right), \quad R = \sqrt{\frac{x_2^2}{4} + \left(\frac{x_1 + x_3}{2}\right)^2}. \quad (11)$$

Meaning: Defines shear stress in the structure.

Explanation: Helps prevent cracks and weld failures.

2. Bending stress (σ):

$$\sigma = \frac{6PL}{x_4x_3^2} \quad (12)$$

Meaning: Determines stress due to bending in the beam.

Explanation: Ensures the structure is resistant to fractures and deformations.

3. Deflection (δ):

$$\delta = \frac{6PL^3}{Ex_3^2x_4} \quad (13)$$

Meaning: Measures the extent of beam deflection under load.

Explanation: Ensures compliance with displacement limits in construction standards.

4. Buckling load (P_c):

$$P_c = \frac{4.013E\sqrt{\frac{x_3^2x_4^6}{36L^2}}\left(1 - \frac{x_3}{2L}\sqrt{\frac{E}{4G}}\right)}{L^2} \quad (14)$$

Meaning: Defines the critical load at which the beam may buckle.

Explanation: Ensures structural stability under real-world conditions.

3. Strength Constraints

The Strength Constraints Section contains key inequalities that define the physical and mechanical limitations of the welded beam. These constraints are essential in ensuring

the structure meets the safety and strength requirements while minimizing production costs. In the context of quantum optimization, this problem is formulated as a constrained optimization task, where these conditions must be considered when transforming the problem into a model suitable for quantum annealing.

In quantum computing systems such as D-Wave, these constraints are typically implemented as penalty functions in the Quadratic Unconstrained Binary Optimization (QUBO) model. This means that each constraint is incorporated into the objective function as an additional term, where violations increase the cost function. As a result, the quantum algorithm not only minimizes the production cost but also ensures compliance with engineering constraints.

1. Shear stress constraint g_1

$$g_1(\vec{X}) = \tau(\vec{X}) - \tau_{\max} \leq 0 \quad (15)$$

where

$$\tau(\vec{X}) = \sqrt{\tau_1^2 + 2\tau_1\tau_2\frac{x_2}{2R} + \tau_2^2} \quad (16)$$

$$\tau_1 = \frac{P}{\sqrt{2}x_1x_2}, \quad \tau_2 = \frac{M}{J}, \quad M = P\left(L + \frac{x_2}{2}\right) \quad (17)$$

$$R = \sqrt{\frac{x_2^2}{4} + \left(\frac{x_1 + x_3}{2}\right)^2}, \quad J = 2\left(\sqrt{2}x_1x_2\left(\frac{x_3^2}{4} + \left(\frac{x_4 + x_2}{2}\right)^2\right)\right) \quad (18)$$

Meaning: Ensures that shear stresses in the beam do not exceed the allowable value. Shear stress is one of the primary failure mechanisms in structures, making its control essential.

Explanation: In quantum optimization, this constraint is formulated as a penalty added to the objective function, increasing when stress exceeds the critical value. The D-Wave system searches for solutions where shear stress remains within acceptable limits.

2. Normal stress constraint g_2

$$g_2(\vec{X}) = \sigma(\vec{X}) - \sigma_{\max} \leq 0 \quad (19)$$

where

$$\sigma(\vec{X}) = \frac{6PL}{x_4x_3^2} \quad (20)$$

Meaning: Controls the stresses resulting from bending moments. Excessive stresses can lead to plastic deformation or structural failure.

Explanation: Quantum optimization aims to find values for design parameters (beam thickness, length, and height) that provide bending resistance while minimizing material cost. The penalty for exceeding $\sigma_{\max}$ in QUBO ensures that the system avoids unsafe configurations.

3. Deflection constraint g_3

$$g_3(\vec{X}) = \delta(\vec{X}) - \delta_{\max} \leq 0 \quad (21)$$

where

$$\delta(\vec{X}) = \frac{6PL^3}{Ex_4x_3^3} \quad (22)$$

Meaning: Limits the maximum allowable deflection of the beam. Excessive deflection can lead to functional issues and structural instability.

Explanation: Quantum annealing enables the optimization of geometric parameters to ensure minimum deflection without excessive material usage. Including this constraint in the QUBO model forces the solution to satisfy the rigidity requirements.

4. Weld geometry constraint g_4

$$g_4(\vec{X}) = x_1 - x_4 \leq 0 \quad (23)$$

Meaning: Ensures that the weld height does not exceed the beam width, which is crucial for stability and weld quality.

Explanation: In quantum annealing, this condition influences the proper shaping of the solution so that the obtained values of design variables are realistic. Including this inequality in QUBO restricts the solution space to physically valid combinations.

5. Buckling load constraint g_5

$$g_5(\vec{X}) = P - P_c(\vec{X}) \leq 0 \quad (24)$$

where

$$P_c = \frac{4.013E \sqrt{\frac{x_3^2 x_4^6}{36L^2}} \left(1 - \frac{x_3}{2L} \sqrt{\frac{E}{4G}}\right)}{L^2} \quad (25)$$

Meaning: Prevents exceeding the critical load that leads to buckling failure. This is one of the key constraints in structural design.

Explanation: Quantum annealing enables the evaluation of various beam dimension configurations to find the most efficient geometry that meets the buckling resistance requirements. Adding this constraint to the QUBO model eliminates solutions that lead to structural instability.

6. Minimum weld height constraint g_6

$$g_6(\vec{X}) = 0.125 - x_1 \leq 0 \quad (26)$$

Meaning: Ensures that the weld has a minimum acceptable height to provide sufficient strength. A weld that is too small may lead to weak joints and failure under stress.

Explanation: In quantum optimization, this constraint is crucial for ensuring structural integrity while optimizing material usage. In the QUBO model, a penalty is applied if x_1 (weld height) is below the minimum requirement, preventing physically unrealistic solutions.

7. Cost constraint g_7

$$g_7(\vec{X}) = 1.10471x_1^2 + 0.04811x_3x_4(14.0 + x_2) - 5.0 \leq 0 \quad (27)$$

Meaning: Imposes a maximum allowable manufacturing cost, ensuring that the optimization does not lead to excessive expenses.

Explanation: In quantum optimization, cost minimization is the primary goal, but it must be balanced with structural constraints. This condition is incorporated into the QUBO model as an additional penalty, ensuring that the computed solution remains within economically viable limits.

The experiment was replicated based on the works of [12,13], and all numerical constants are the same as those in the cited studies. In the subsequent sections of the article, I refer to the results of the optimization using classical methods, ensuring that the adopted numerical constants and constraints are identical to reliably compare the effectiveness of the methods.

4. Quadratic Unconstrained Binary Optimization (QUBO)

Quadratic Unconstrained Binary Optimization (QUBO) is an optimization problem in which we seek a vector of binary variables $\mathbf{x} \in \{0, 1\}^n$ that minimizes (or maximizes) a quadratic objective function as follows [14–16]:

$$f(\mathbf{x}) = \mathbf{x}^T \mathbf{Q} \mathbf{x}, \quad (28)$$

where $\mathbf{Q}$ is an $n \times n$ square matrix (often symmetric). Written in detail, this is as follows:

$$f(\mathbf{x}) = \sum_{i=1}^n \sum_{j=1}^n Q[i, j] x_i x_j. \quad (29)$$

The term Unconstrained indicates that, in its baseline form, QUBO does not include additional explicit constraints other than the binary nature of the variables. In practical scenarios that require specific constraints (e.g., limiting the sums of certain variables), these are introduced as penalty terms in the objective, preserving the quadratic form.

4.1. Basic Elements of QUBO

1. Binary variables

Each component x_i of the vector $\mathbf{x}$ can only be 0 or 1.

2. The matrix $\mathbf{Q}$

- Diagonal elements $\mathbf{Q}[i, i]$ capture the contribution of each variable x_i (the “linear” part in the binary sense, since $x_i^2 = x_i$ for $x_i \in \{0, 1\}$).
- Off-diagonal elements $\mathbf{Q}[i, j]$ (for $i \neq j$) describe the interaction between variables x_i and x_j .

3. Objective function

It is given by $\mathbf{x}^T \mathbf{Q} \mathbf{x}$. Typically, one aims to minimize this function, though maximizing is also possible by inverting the sign of the relevant terms in $\mathbf{Q}$.

4. Constraint penalties

In real-world applications, constraints (e.g., $\sum_i x_i \leq k$) are introduced by adding penalty terms to the objective; the following represents an example:

$$\lambda \left(\sum_{i=1}^n x_i - k \right)^2, \quad (30)$$

where λ is sufficiently large so that violating the constraint becomes too costly.

4.2. Applications of QUBO

- **Combinatorial optimization problems.** The following represents an example [17]:
 - Max-Cut: Partitioning a graph’s vertices into two sets to maximize the sum of edges “cut” by that partition.
 - SAT (Satisfiability): Logical formulas can be mapped to QUBO by introducing penalties for unsatisfied clauses.
 - Traveling Salesman Problem (TSP): Minimizing the total route distance while visiting each city exactly once.
- **Business optimization** [18]
 - Knapsack problem: Choosing a subset of items under capacity constraints.
 - Production planning and scheduling: Allocating tasks to resources.
 - Resource allocation: Selecting investment projects with limited budgets.
- **Machine learning** [19]

- Clustering: Minimizing a chosen error function to group data points.
- Feature selection: Choosing an optimal subset of features for classification tasks.

4.3. Solving QUBO

Classical methods

- Simulated annealing, genetic algorithms, tabu search: Metaheuristics that search the space of solutions [20].
- Hybrid methods Combining exact search (e.g., integer programming) with heuristic approaches [21].

Quantum methods

- Quantum annealing: Realized on machines like D-Wave, where QUBO is expressed as an equivalent Ising problem [22].
- Gate-based algorithms: For example, QAOA (Quantum Approximate Optimization Algorithm), requiring additional implementation steps and an active area of research [23].

QUBO is a universal model for binary optimization, allowing the representation of numerous complex real-world and theoretical problems. It has a single unified form $\mathbf{x}^T \mathbf{Q} \mathbf{x}$ that is naturally compatible with both classical heuristic approaches and quantum annealers. Its broad range of applications spans from purely theoretical problems (such as SAT or Max-Cut) to practical tasks in business (scheduling and resource allocation) and machine learning (feature selection and clustering). By adding penalty terms, it is straightforward to incorporate various constraints while preserving the core quadratic structure.

5. Formulating Welded Beam Design as QUBO

5.1. Objective Function

The cost function to be minimized is given by the following:

$$f(X) = 1.10471x_1^2 + 0.04811x_3x_4(14.0 + x_2). \quad (31)$$

5.2. Meaning of the Formula

The objective function minimizes the total cost of fabricating the beam, considering the following:

- The **welding cost**, which depends on the weld height x_1 .
- The **material cost**, which depends on the beam thickness x_3 , beam width x_4 , and weld length x_2 .

Optimizing this function helps identify the most cost-effective structural configuration while still satisfying all mechanical and strength constraints.

5.3. Meaning of Individual Components in the Formula

1. Constant numbers (coefficients)

- **1.10471**—a constant related to the welding cost, defining how the weld height x_1 impacts the total cost.
- **0.04811**—a constant determining the material cost of the beam based on its width x_4 and thickness x_3 .
- **14.0**—a fixed value added to the weld length x_2 , accounting for the basic structural parameters and additional costs.

2. Structural variables

- **x_1 (height of weld)**—the weld height, which affects the amount of material used for welding and the cost of the welding process.

- **x_2 (length of weld)**—the weld length, where a longer weld increases material consumption and production cost.
- **x_3 (thickness of beam)**—the beam thickness, which influences the weight and strength of the structure.
- **x_4 (width of beam)**—the beam width, where a wider beam results in higher material consumption.

To express this in QUBO, the following are required:

- Each design variable (x_1, x_2, x_3, x_4) is discretized into a finite range of values, represented as binary vectors.
- Each variable x_i is described as follows:

$$x_i = \sum_{j=1}^n 2^{j-1} q_{ij}, \quad (32)$$

where $q_{ij} \in \{0, 1\}$ are binary variables, and n is the number of bits.

The objective function in binary form is as follows:

$$f(Q) = 1.10471 \left(\sum_{j=1}^n 2^{j-1} q_{1j} \right)^2 + 0.04811 \left(\sum_{j=1}^n 2^{j-1} q_{3j} \right) \left(\sum_{j=1}^n 2^{j-1} q_{4j} \right) (14.0 + \sum_{j=1}^n 2^{j-1} q_{2j}) \quad (33)$$

5.4. Constraints

Each constraint is transformed into a quadratic expression with a penalty C_g for violations. The penalty terms are added to the objective function as follows:

$$F(Q) = f(Q) + \lambda \sum_g C_g, \quad (34)$$

where λ is the weighting coefficient.

5.5. Constraints

1. Shear stress (g_1):

$$\tau(Q) - \tau_{\max} \leq 0, \quad (35)$$

where $\tau(Q)$ is computed in binary form as described above.

Penalty:

$$C_1 = \max(0, \tau(Q) - \tau_{\max})^2. \quad (36)$$

2. Bending stress (g_2):

$$\sigma(Q) - \sigma_{\max} \leq 0. \quad (37)$$

Penalty:

$$C_2 = \max(0, \sigma(Q) - \sigma_{\max})^2. \quad (38)$$

3. Deflection (g_3):

$$\delta(Q) - \delta_{\max} \leq 0. \quad (39)$$

Penalty:

$$C_3 = \max(0, \delta(Q) - \delta_{\max})^2. \quad (40)$$

4. Geometry (g_4):

$$x_1 - x_4 \leq 0. \quad (41)$$

Penalty:

$$C_4 = \max(0, x_1 - x_4)^2. \quad (42)$$

5. Buckling load (g_5):

$$P - P_c(Q) \leq 0. \quad (43)$$

Penalty:

$$C_5 = \max(0, P - P_c(Q))^2. \quad (44)$$

6. Minimum thickness (g_6):

$$0.125 - x_1 \leq 0. \quad (45)$$

Penalty:

$$C_6 = \max(0, 0.125 - x_1)^2. \quad (46)$$

7. Cost limit (g_7):

$$f(Q) - 5.0 \leq 0. \quad (47)$$

Penalty:

$$C_7 = \max(0, f(Q) - 5.0)^2. \quad (48)$$

5.6. QUBO Matrix

The objective function and penalties can be written as a quadratic binary form, as follows:

$$F(Q) = \sum_i a_i q_i + \sum_{i,j} b_{ij} q_i q_j. \quad (49)$$

where

- a_i represents the linear weights for the binary variables,
- b_{ij} represents the quadratic interaction coefficients between the variables.

For each constraint C_g , the appropriate a_i and b_{ij} elements in the QUBO matrix Q are calculated and added to the objective function.

6. Pseudocode

For the research, the dwave-ocean-sdk library provided by D-Wave in Python 3.12.7 was used. This allows us to formulate a problem and run it on their quantum computer. The company imposes a time limit on users.

Below is the pseudocode Algorithm 1 for formulating and solving the welded beam design problem using a quantum annealer.

Algorithm 1 Welded beam design QUBO implementation

```

1: Input: Problem parameters:  $\tau_{\max}, \sigma_{\max}, \delta_{\max}, E, G, P, L$ 
   Variable bounds for  $x_1, x_2, x_3, x_4$ ; Number of bits  $n = 4$ 
2: Output: Best solution and corresponding energy from the quantum annealer
3: // Define physical and design parameters
4: Set  $\tau_{\max} = 13,600, \sigma_{\max} = 30,000, \delta_{\max} = 0.25$ 
5: Set  $E = 30 \times 10^6, G = 12 \times 10^6, P = 600, L = 14$ 
6: // Define variable bounds
7: Define bounds:
    $x_1 \in [0.1, 2], x_2 \in [0.1, 10], x_3 \in [0.1, 10], x_4 \in [0.1, 2]$ 
8: // Discretize continuous variables into binary representation
9: Set num_bits = 4
10: for each variable  $x$  in  $\{x_1, x_2, x_3, x_4\}$  do
11:   Compute:
       
$$\text{scale\_factor} = \frac{\text{upper bound} - \text{lower bound}}{2^n - 1}$$

12: end for

```

Algorithm 1 *Cont.*

```

13: // Define binary-to-continuous conversion function
14: function BINARY_TO_CONTINUOUS(binary_vector, variable)
15:     scale ← corresponding scale_factor for the variable
16:     Compute:

           continuous_value =  $\sum_{i=0}^{n-1} \text{binary\_vector}[i] \times 2^i \times \text{scale} + \text{lower bound}$ 

17:     return continuous_value
18: end function
19: // Create the QUBO model
20: Initialize empty QUBO model: QUBO_model
21: // Add cost function to the QUBO model
22: for each variable  $x$  in  $\{x_1, x_2, x_3, x_4\}$  do
23:     for  $i = 0$  to num_bits−1 do
24:         Add linear term with name “x_i” and coefficient:

            $1.10471 \times (\text{scale\_factor})^2$ 

25:     end for
26: end for
27: // Add constraints as penalty terms to QUBO
28: function ADD_CONSTRAINTS(QUBO_model)
29:     // Shear stress constraint ( $\tau$ )
30:     Compute shear stress and add penalty if  $\tau > \tau_{\max}$ 
31:     // Bending stress constraint ( $\sigma$ )
32:     Compute bending stress and add penalty if  $\sigma > \sigma_{\max}$ 
33:     // Deflection constraint ( $\delta$ )
34:     Compute deflection and add penalty if  $\delta > \delta_{\max}$ 
35:     return updated QUBO_model
36: end function
37: Call add_constraints(QUBO_model)
38: // Return constructed QUBO model
39: return QUBO_model
40: // Solve QUBO using a quantum annealer
41: function SOLVE_QUBO(QUBO_model)
42:     Initialize sampler using DWaveSampler and EmbeddingComposite
43:     Run sampler on QUBO_model with num_reads = 100
44:     return best solution and its energy
45: end function
46: // Main program flow
47: QUBO_model ← result of welded_beam_qubo() function
48: response ← solve_qubo(QUBO_model)
49: Print best solution and corresponding energy from response

```

Explanation of Components

1. **Problem parameters:** The physical and design parameters (e.g., maximum shear stress $\tau_{\max}$, bending stress $\sigma_{\max}$, deflection $\delta_{\max}$, Young’s modulus E , shear modulus G , applied load P , and beam length L) set the limits for the design.
2. **Variable bounds:** Bounds for the design variables (x_1, x_2, x_3, x_4) define the range in which each variable may vary.
3. **Discretization:** Continuous variables are discretized into binary form using a fixed number of bits (num_bits = 4). A scale factor is computed for each variable to map the binary representation to its continuous value. This is described by Equation (32)

4. **Binary-to-continuous conversion:** The `binary_to_continuous` function converts a binary vector into a continuous variable value using the computed scale factor and the variable's lower bound.
5. **QUBO model creation:** An empty QUBO (Quadratic Unconstrained Binary Optimization) model is initialized. This model will later incorporate both the cost function and constraint penalties.
6. **Cost function:** The cost function is added as linear terms to the QUBO model for each binary variable. The coefficients are determined by the scale factors. The cost function is described by Equation (31). In the QUBO version, the equation is described as (33).
7. **Constraints as penalties:** Physical constraints (shear stress, bending stress, and deflection) are added as penalty terms. If any constraint is violated, a penalty increases the overall cost. The detailed implementation is left as a placeholder.
8. **Solving the QUBO:** The QUBO is solved using a D-Wave quantum annealer. The `solve_qubo` function sets up the sampler, runs the annealing process (with `num_reads = 100`), and returns the best solution.
9. **Main flow:** The pseudocode concludes by constructing the QUBO model, solving it, and printing the best solution along with its energy.

7. Results

Below are the results of our research on structural optimization using the quantum properties of the D-Wave system. The results of other algorithms, such as the Whale Optimization Algorithm (WOA), Gravitational Search Algorithm (GSA), Goose Algorithm (GOOSE), and Particle Swarm Optimization (PSO), are sourced from the literature. The names in Table 1, Quantum Start 1 and Quantum Start 2, represent the results of optimization using the D-Wave system.

Table 1. Comparison of algorithm performance.

Algorithm Name	Best Result	Result Source
WOA	1.732	(Mirjalili & Lewis, 2016) [24]
PSO	1.742	(Mirjalili & Lewis, 2016) [24]
GOOSE	3.188	arxiv 2024 [24]
GSA	3.576	(Mirjalili & Lewis, 2016) [24]
Quantum Start 1	1.234	–
Quantum Start 2	1.016	–

Meanwhile, Tables 2 and 3 present the parameters for the lowest cost values obtained for the solutions Quantum Start 1 and Quantum Start 2.

Table 2. Parameters for Quantum Start 1.

Parameter	Value
x1 (Height of weld)	0.3533
x2 (Length of weld)	6.0400
x3 (Height of beam)	2.0800
x4 (Width of beam)	2.0000

Table 3. Parameters for Quantum Start 2.

Parameter	Value
x1 (Height of weld)	0.3533
x2 (Length of weld)	4.7200
x3 (Height of beam)	4.7200
x4 (Width of beam)	0.8600

In both cases of quantum optimization, the energy obtained was 5,954,303.3448 and 2,452,399.0500, respectively. In one case, the energy was 1640.2799 for the best solution of 0.5646, which is an exception and should be subjected to detailed investigation. As can be seen, optimization using D-Wave yields good results. Moreover, the properties of this system suggest that the optimization of highly complex structures can be performed very quickly and with excellent results.

8. Discussion

The presented findings of this study indicate the significant impact of optimization on solving engineering problems, particularly in structural design. This type of optimization plays a crucial role in cost reduction, which leads to decreased financial and environmental burdens. Minimizing material consumption and designing more efficient structures can contribute to the sustainable development of structural engineering and other technical fields.

The obtained results were compared with the literature data, where classical heuristic optimization methods such as the Whale Optimization Algorithm (WOA), Gravitational Search Algorithm (GSA), Goose Algorithm (GOOSE), and Particle Swarm Optimization (PSO) were applied. It has been demonstrated that leveraging the properties of quantum computers in the structural optimization process can yield results that are at least comparable, and in some cases superior, to traditional methods. This is a significant discovery suggesting the potential application of quantum algorithms across a broad range of engineering problems.

However, one of the key challenges associated with using quantum computers, such as D-Wave, is the scalability of solutions. Current quantum systems have a limited number of qubits and a specific connection topology, making it difficult to directly map complex problems. To overcome these limitations, the following strategies are applied:

Quantum-classical hybrid methods—a combination of classical and quantum methods, where computationally intensive tasks are delegated to classical computers, while key optimization components are executed quantumly. Hierarchical problem decomposition—large problems are divided into smaller subproblems, which are optimized separately, and their results are then combined. Approximation and model reduction—the elimination of less significant variables and the use of simplified mathematical representations, reducing hardware requirements. These approaches enable the expansion of quantum methods' applications despite the current technological constraints.

In this study, the available computation time on the D-Wave computer was utilized to test the optimization of the welded beam problem. However, limited access to computational resources poses a barrier to further research. Obtaining additional funding for extending the experiments would allow for a more thorough analysis of the effectiveness of the applied methods.

Based on the current capabilities of the D-Wave Advantage system, it can be estimated that, following the QUBO approach, approximately 1666 decision variables can be optimized, assuming an average use of three qubits per variable. This represents a significant step towards the practical utilization of quantum computers in engineering. However, this number depends on the complexity of the problem, the topology of connections between qubits, and the embedding strategy used in implementation.

In conclusion, quantum optimization can serve as a viable alternative to classical heuristic methods. The presented findings indicate that it can deliver solutions of comparable or higher quality, making it a promising research direction for the optimization of engineering problems. Future work should focus on expanding the analyzed cases and assessing the scalability and effectiveness of the method in more complex problems.

9. Conclusions

The conducted research has demonstrated that quantum optimization can serve as an effective alternative to classical heuristic methods in solving engineering problems, particularly in structural design. The obtained results suggest that utilizing quantum computers enables the achievement of comparable or superior solutions while also offering the possibility to significantly accelerate the computational processes.

Despite the limited access to advanced quantum systems, the results of the conducted experiments provide a solid foundation for further research.

In conclusion, quantum optimization exhibits substantial potential in engineering, and its continued development may lead to significant cost savings and environmental benefits.

Funding: This research received no external funding

Informed Consent Statement: Not applicable.

Data Availability Statement: Research institutions that wish to receive the data are requested to contact the author.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Ewald, D.; Czerniak, J.M.; Paprzycki, M. A New OFNBee Method as an Example of Fuzzy Observance Applied for ABC Optimization. In *Theory and Applications of Ordered Fuzzy Numbers*; Prokopowicz, P., Czerniak, J., Mikołajewski, D., Apiecione, Ł., Ślęzak, D., Eds.; Studies in Fuzziness and Soft Computing; Springer: Cham, Switzerland, 2017; Volume 356.
2. Czerniak, J.M.; Zarzycki, H.; Ewald, D. AAO as a new strategy in modeling and simulation of constructional problems optimization. *Simul. Model. Pract. Theory* **2017**, *76*, 22–33. [CrossRef]
3. Ewald, D.; Czerniak, J.M.; Zarzycki, H. Approach to Solve a Criteria Problem of the ABC Algorithm Used to the WBDP Multicriteria Optimization. In *Intelligent Systems'2014*; Angelov, P., Atanassov, K.T., Doukowska, L., Hadjiski, M., Jotsov, V., Kacprzyk, J., Kasabov, N., Sotirov, S., Szmidt, E., Zadrozny, S., Eds.; Advances in Intelligent Systems and Computing; Springer: Cham, Switzerland, 2015; Volume 322. [CrossRef]
4. D-Wave Documentation. Available online: https://docs.dwavesys.com/docs/latest/c_gs_2.html (accessed on 26 January 2025).
5. Glover, F.; Kochenberger, G.A.; Du, Y. *Quantum-Inspired Evolutionary Algorithms*; Studies in Computational Intelligence; Springer: Cham, Switzerland, 2019.
6. What Is the Hamiltonian? Available online: <https://support.dwavesys.com/hc/en-us/articles/360003684614-What-Is-the-Hamiltonian> (accessed on 26 January 2025).
7. D-Wave Hybrid Solver Documentation. Available online: https://docs.dwavesys.com/docs/latest/handbook_hybrid.html (accessed on 26 January 2025).
8. Introduction to D-Wave Systems. Available online: https://docs.dwavesys.com/docs/latest/handbook_intro.html (accessed on 26 January 2025).
9. Mirjalili, S.; Mirjalili, S.M.; Lewis, A. Grey Wolf Optimizer. *Adv. Eng. Softw.* **2014**, *69*, 46–61. [CrossRef]
10. Deb, K.; Sundar, J.; Rao, N., U.B.; Chaudhuri, S. Reference Point Based Multi-Objective Optimization Using Evolutionary Algorithms. *Int. J. Comput. Intell. Res.* **2006**, *2*, 273–286. Available online: <https://www.softcomputing.net/ijcir/vol2-issu3-paper4.pdf> (accessed on 26 January 2025).
11. Ray, T.; Liew, K.M. A Swarm Metaphor for Multiobjective Design Optimization. *Eng. Optim.* **2002**, *34*, 141–153. [CrossRef]
12. Deb, K. Optimal design of a welded beam via genetic algorithms. *AIAA J.* **1991**, *29*, 2013–2015. Available online: <https://api.semanticscholar.org/CorpusID:122733806> (accessed on 28 February 2025). [CrossRef]
13. Sarkar, M.; Roy, T.K. Multi-Objective Welded Beam Design Optimization Using T-Norm and T-Co-Norm Based Intuitionistic Fuzzy Optimization Technique. Proceedings of an Unspecified Conference. 2017. Available online: <https://api.semanticscholar.org/CorpusID:228104228> (accessed on 28 February 2025).
14. Kochenberger, G.; Hao, J.-K.; Glover, F.; Lewis, M.; Lü, Z.; Wang, H.; Wang, Y. The unconstrained binary quadratic programming problem: A survey. *J. Comb. Optim.* **2014**, *28*, 58–81. [CrossRef]
15. Glover, F.; Kochenberger, G.; Du, Y. A Tutorial on Formulating and Using QUBO Models. *arXiv* **2019**, arXiv:1811.11538. [CrossRef]

16. Chen, M.; Chun, J.; Xiang, S.; Wei, L.; Du, Y.; Wan, Q.; Chen, Y.; Chen, Y. Learning to Solve Quadratic Unconstrained Binary Optimization in a Classification Way. In Proceedings of the Thirty-Eighth Annual Conference on Neural Information Processing Systems (NeurIPS 2024), Vancouver, BC, Canada, 9–15 December 2024. Available online: <https://openreview.net/forum?id=p43ObIwJFW> (accessed on 27 February 2025).
17. Więckowski, A. Symulowane Wyzarzanie Problemu Kolorowania Grafu. 2019. Available online: https://www.researchgate.net/publication/335777671_Symulowane_wyzarzanie_problemu_kolorowania_grafu (accessed on 27 February 2025).
18. Herman, D.; Googin, C.; Liu, X.; Sun, Y.; Galda, A.; Safro, I.; Pistoia, M.; Alexeev, Y. Quantum computing for finance. *Nat. Rev. Phys.* **2023**, *5*, 450–465. [[CrossRef](#)]
19. Veszeli, M.T.; Vattay, G. Mean Field Approximation for solving QUBO problems. *arXiv* **2021**, arXiv:2106.03238. . [[CrossRef](#)] [[PubMed](#)]
20. Thamilselvan, R.; Balasubramanie, P. Integrating Genetic Algorithm, Tabu Search and Simulated Annealing For Job Shop Scheduling Problem. *Int. J. Comput. Appl.* **2012**, *48*, 42–47. [[CrossRef](#)]
21. Wang, Y.; Wang, S.; Wang, K. Hybrid Genetic Algorithm and Tabu Search for Solving Preventive Maintenance Scheduling Problems. *Mathematics* **2023**, *12*, 1881. [[CrossRef](#)]
22. Pelofske, E.; Bärtschi, A.; Eidenbenz, S. Quantum Annealing vs. QAOA: 127 Qubit Higher-Order Ising Problems on NISQ Computers. *arXiv* **2023**, arXiv:2301.00520. [[CrossRef](#)]
23. Boothby, K.; King, A.D.; Raymond, J. Next-Generation Topology of D-Wave Quantum Processors. *arXiv* **2020**, arXiv:2003.00133. [[CrossRef](#)]
24. Hamad, R.K.; Rashid, T.A. GOOSE Algorithm: A Powerful Optimization Tool for Real-World Engineering Challenges and Beyond. *arXiv* **2024**, arXiv:2307.10420. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.