scientific reports

OPEN

Quantum annealing applications, challenges and limitations for optimisation problems compared to classical solvers

Finley Alexander Quinton[1]✉, Per Arne Sevle Myhr[1,2], Mostafa Barani[1], Pedro Crespo del Granado[1]✉ & Hongyu Zhang[1]

Quantum computing is rapidly advancing, harnessing the power of qubits' superposition and entanglement for computational advantages over classical systems. However, scalability poses a primary challenge for these machines. By implementing a hybrid workflow between classical and quantum computing instances, D-Wave has succeeded in pushing this boundary to the realm of industrial use. Furthermore, they have recently opened up to mixed integer linear programming (MILP) problems, expanding their applicability to many relevant problems in the field of optimisation. However, the extent of their suitability for diverse problem categories and their computational advantages remains unclear. This study conducts a comprehensive examination by applying a selection of diverse case studies to benchmark the performance of D-Wave's hybrid solver against that of industry-leading solvers such as CPLEX, Gurobi, and IPOPT. The findings indicate that D-Wave's hybrid solver is currently most advantageous for integer quadratic objective functions and shows potential for quadratic constraints. To illustrate this, we applied it to a real-world energy problem, specifically the MILP unit commitment problem. While D-Wave can solve such problems, its performance has not yet matched that of its classical counterparts.

The field of Quantum Computing (QC) has seen great improvements over the last decade with companies like IBM and Google investing heavily in order to develop resources capable of solving challenging computational problems[1,2]. Recently, also Microsoft announced their new topological qubit design[3], showcasing a new path in obtaining more error-resistant scalable quantum computers. Although rapid progress coincides with modern times, the idea of using "simple" systems governed by Quantum Mechanics (QM) to perform computation was proposed already in the 1980s by both Richard Feynman[4] and Paul Benioff[5] independently. There are numerous ways to approach building a quantum computer, but the two main categories are gate-based and annealing-based QC. Just like bits in a classical computer, quantum computers use quantum bits, commonly referred to as qubits. Over the years, many different technologies have been suggested and tested as physical implementations of qubits. The first attempts in the 90s used nuclear magnetic resonance[6], later, a plethora of different ideas has shown promise such as cold atoms[7], trapped ions[8], photon-based[9] and others[10,11]. The current largest quantum annealers are pioneered by D-Wave[12] and consist of loops of superconducting material[13]. It has been shown that in specific cases, QC can be exponentially faster than Classical Computing (CC), such as for Shor's algorithm[14], or the quantum Fourier transform algorithm[15].

The fundamental idea behind QC is to use quantum phenomena such as superposition and entanglement to solve computationally difficult problems. In addition to solving computationally expensive problems, QC may be suited for solving problems that classical computers cannot, such as simulating nature[16,17], as these systems are governed by the same rules of QM. It has been shown that for several NP-hard problems, where classical approaches fail to efficiently find a solution, QC can be a promising way to solve them[18–21].

In gate-based QC, the classical, binary bits are replaced by controllable qubits capable of being in non-binary states. Adiabatic quantum annealers, however, function on a different principle. They prepare an ensemble of qubits in the ground state of some initial Hamiltonian, before allowing the system to evolve abiding by the adiabatic theorem[22] to some final state encoding the desired solution. This principle will be further explained in Section "Adiabatic theorem".

[1]Department of Industrial Economics and Technology Management, Norwegian University of Science and Technology, 7491 Trondheim, Norway. [2]Centre for Cosmology, Particle Physics and Phenomenology, Université Catholique de Louvain, 1348 Louvain-la-Neuve, Belgium. ✉email: quinton.f.alexander@ntnu.no; pedro@ntnu.no

Although quantum annealers are easier to scale towards more qubits, they are far more limited in their applications due to the lack of controllability of the individual qubits. While in gate-based quantum computers qubits can be controlled and manipulated with precise unitary operations, the annealing-based computers of D-Wave controls theirs with tunable biases and the coupling strengths between them. D-Wave's quantum annealers are not universal quantum computers, and therefore can not run arbitrary algorithms. They are, however, especially well suited for optimisation problems where scalability is one of the most determining factors in solving relevant, modern, real-world problems. For a more thorough overview of the synergies between operational research and quantum computing, see[23,24].

The development of quantum computers has finally reached a point where they are comparable to classical computers for an ever-increasing diverse pool of problems, some of which we wish to present here. Several formulations of vehicle routing problems were proposed for utilising quantum computers in[21]. Syrichas & Crispin[25] proposed a simplified but systematic approach to tune quantum annealing parameters for vehicle routing problems. Bernal et al.[26] presented the perspective of quantum computing for chemical engineering and provided examples in computational chemistry and molecular simulation where QC may become relevant. Rosenberg et al.[27] developed a metaheuristic solver to solve large quadratic unconstrainted binary optimisation problems, which utilise D-Wave's quantum annealers. On the other hand, Glover, Lewis, and Kocheberger[28] analysed the logical and inequality implications of reducing problem size to use QC to solve larger problems.

Quantum annealing may prove significantly beneficial for stochastic programming. In[29] an algorithm based on the concepts of quantum theory was proposed and was applied to a stochastic job shop scheduling problem. In addition to real-world applications, several studies also investigate utilising quantum computing in classical optimisation algorithms, such as[30,31].

Although D-Wave's quantum annealers have been applied to solve problems such as rescheduling of urban railroads[32], traffic flow optimisation[33], and dynamic portfolio optimisation[34], the performance or capability of quantum annealers in solving different classes of optimisation problems compared with industrial leading classical solvers is not clear[35]. As pointed out in[32], the current implementations of quantum annealing in optimisation are limited in size and not yet upscaled to real-world situations. Furthermore, the exploration of quantum annealing in scheduling problems is limited. Given the rapid evolution of QC, regular benchmarking becomes crucial to highlight emerging developments and showcase the current state of this highly promising field, as emphasised by[24]. Benchmarks like[36] have shown that D-Wave can compete with classical solvers like Tabu Search[37] and Simulated Annealing (SA)[38] for energy coalition formation and is ahead of gate-based applications like the Quantum Approximate Optimization Algorithm[39]. However, no benchmark of industry-leading solvers has been conducted that evaluates their performance against that of D-Wave's hybrid solvers for different optimisation problem categories.

This research aims to address this gap by utilizing the largest publicly available adiabatic quantum annealers, last updated in August 2023, to solve various optimisation problems, including integer/binary linear programming (BLP) with linear and quadratic constraints, integer/binary quadratic programming (BQP), and mixed-integer linear programming. To assess performance, we systematically compare D-Wave's Advantage quantum annealers, as described in[40], with the state-of-the-art classical solvers CPLEX[41], Gurobi[42], and IPOPT[43]. Recognizing the limitations of fully QC algorithms, we employ D-Wave's hybrid solvers[44]. This approach seamlessly integrates the strengths of both classical and quantum computation, as explained in more detail in Section "Hybrid computing". This study serves as a valuable resource for researchers across various fields dealing with optimisation problems, enabling them to assess the suitability of current D-Wave quantum annealers for their respective applications.

The subsequent sections of the paper are organized as follows: Section "Quantum annealing" provides a brief introduction to the physical and technical background of quantum annealing, encompassing the adiabatic theorem and D-Wave's architecture in their treatment of constraints and the hybrid computing approach employed. Section "Case study" presents various case studies for different classes of optimisation problems. Then, Section The unit commitment problem—an energy system application showcases the application of D-Wave's hybrid constrained quadratic model (CQM) solver[45] on an energy system problem formulated as Mixed-Integer Linear Programming (MILP).

Quantum annealing

Simulated Annealing (SA) is used for approximating the global optimal solution, often used when an approximate global solution is preferred over an exact local minimum[38]. The fundamental principle of SA involves a two-phase process: initially "heating" the system, followed by a gradual "cooling" that guides it into its globally lowest energy state. It is important to emphasize that this "heating" and "cooling" are metaphorical concepts, as no physical heating or cooling is applied; rather, the entire process is mathematically simulated. The inspiration for SA comes from the physical annealing process often used to change the characteristics of different materials[46]. D-Wave utilizes a quantum variation of this process, known as quantum annealing. In quantum annealing, the quantum mechanical phenomenon of tunnelling through a potential barrier, rather than the classical process of "climbing" over it, is utilized to escape local minima and potentially find a global one, as explained by Kadowaki and Nishimori (1998)[47]. This fundamental characteristic is pivotal in understanding the potential computational advantages of quantum annealers over their classical counterparts, particularly because the "hill climbing" process in SA can be time-consuming for a substantial subset of problems.

Quantum annealers exhibit narrower applicability compared to gate-based quantum computers and are, for example, unable to execute Shor's algorithm[14]. By design, they are made to handle optimisation problems and are especially suited for quadratic unconstrained binary optimisation (QUBO) problems of the form

$$\min_x \quad \left(\sum_i h_i x_i + \sum_{i<j} Q_{ij} x_i x_j \right), \tag{1}$$

where $x \in \{0,1\}$ and $Q_{ij}, h_i \in \mathbb{R}^n$.

QUBO problems are very closely related to the Ising model (Eq. 2) and can be easily transformed into this form by applying an affine transformation $x \mapsto 2x - 1$, effectively mapping $x \in \{0,1\}$ to $\{-1,1\}$. Under this transformation, the variables x_i, x_j correspond to classical spin variables $s_i = \pm 1$ leading to the classical Ising spin system:

$$E_{ising}(s) = \sum_i h_i s_i + \sum_{i>j} J_{i,j} s_i s_j, \tag{2}$$

where $J_{i,j}$ represents the spin interaction strength and h_i external field. Traditionally, the Ising model is used in the field of statistical mechanics where the variables represent the binary states of a particle that can either be "spin down" or "spin up".

In quantum mechanics, these variables are replaced by Pauli operators $\sigma_{x,y,z}$. Specifically, spin states are often expressed by terms of the Pauli σ_z matrix which correspond to spin projections along the z-axis. The Hamiltonian describing such a quantum system takes the form:

$$H(\sigma) = \frac{B(s)}{2} \left(\sum_i h_i \sigma_z^{(i)} + \sum_{i>j} J_{i,j} \sigma_z^{(i)} \sigma_z^{(j)} \right), \tag{3}$$

where $B(s)$ is a prefactor with dimensions of energy and $J_{i,j}$ determines the interaction strength between neighboring spins. Since all the terms in this Hamiltonian involve σ_z operators, they mutually commute. However, quantum annealers such as D-Wave introduce and additional transverse field involving σ_x terms, which do not necessarily commute with σ_z. This non-commutativity is crucial for quantum annealing to leverage the adiabatic theorem[22], requiring an initial Hamiltonian with a dominant transverse field component as a starting point before gradually transitioning into the problem Hamiltonian.

Adiabatic theorem

In quantum mechanics, operators are used to extract information about physical systems. The most relevant of these operators in our case is the Hamilton operator H, usually just referred to as the Hamiltonian. The Hamilton corresponds to the total energy of the quantum mechanical system. By assuming that the time-dependent part varies sufficiently slow and that there is a big enough gap between the eigenstates of H, the solutions to the time-dependent Schödinger equation[48] are obtained in terms of the eigenfunctions of the instantaneous Hamiltonian,

$$H(t)\psi_a(t) = E_a(t)\psi_a(t), \tag{4}$$

where E_a is the energy of eigenstate ψ_a. The scale of "sufficiently slow" depends on the specific energy gap between the quantum states of the system and can range from less than a microsecond to a few seconds. If $H(t)$ varies slowly in time, a system initially in a non-degenerate state $\psi(t = t_0)$ with energy $E_a(t = t_0)$ will evolve into the corresponding state $\psi_a(t)$ with energy $E_a(t)$ at a later time t without making any transitions between energy levels. This is known as the adiabatic theorem and is the fundamental principle utilized in quantum annealing. For a more detailed description of the adiabatic theorem and a complete proof thereof, see[22].

To perform calculations using a quantum annealer, the physical collection of qubits, hereafter referred to as the system, is prepared in an easy-to-solve initial Hamiltonian H_I. After the system has been prepared in H_I , external magnetic fields are applied to change the initial H_I into a complex Hamiltonian H_F. Suppose this process follows the adiabatic theorem described above. In that case, the system will remain in its ground state for the duration of the anneal and the final state will be the solution of the complex Hamiltonian. *As* $[\sigma_x, \sigma_z] \neq 0$, H_I and H_F do not share a common set of eigenstates. As a result, the system must undergo quantum transitions to evolve from one eigenstate to another. If the two Hamiltonians did commute, the eigenstates would remain unchanged throughout the evolution, preventing any quantum dynamics from occurring. The procedure can slowly anneal the system from the quantum Hamiltonian (Eq. (5)) to the solution of the classical Ising spin system shown in Eq. (2). After the annealing process, the collapsed spin states will present a low-energy solution. The annealing quantum Ising Hamiltonian is as follows:

$$H_{\text{Ising}} = \underbrace{\frac{-A(s)}{2} \left(\sum_i \sigma_x^{(i)} \right)}_{\text{Initial state Hamiltonian}} + \underbrace{\frac{B(s)}{2} \left(\sum_i h_i \sigma_z^{(i)} + \sum_{i>j} J_{i,j} \sigma_z^{(i)} \sigma_z^{(j)} \right)}_{\text{Final state Hamiltonian}}. \tag{5}$$

The annealing process begins at $t = 0$ and ends at time $t = t_f$, where $A(0) \gg B(0)$ and $A(t_f) \ll B(t_f)$. At time $t = t_f$, the qubits have been de-phased to a classical system and the Pauli matrix can be replaced by the classical $s_i = \pm 1$, resulting in Eq. (2).

D-Wave architecture

The quantum annealers developed by D-Wave are currently the most promising systems capable of utilizing quantum properties for computation. Since 2011, D-Wave has steadily improved the capabilities of their quantum processing units (QPU) by upgrading both the hardware through multiple generations of system topologies and their own developed software *Ocean*[49], thereby allowing for remote problem solving on their quantum annealers. D-Wave quantum annealers use loops of superconducting material as qubits to build their QPUs. This approach fixes the physical layout, also known as the (hardware graph), of the QPU.

The most powerful QPU available is the *Pegasus* topology[50]. In the Pegasus topology, each qubit is coupled to 15 other qubits, which is a great improvement from the 6 couplings of the previous generation. It consists of a square lattice of unit cells which themselves contain 24 qubits. The Advantage QPU, used for the computations performed in this work, is a 16×16 lattice of unit cells, totalling $24 \times 16 \times (16 - 1) = 5760$ qubits. Due to the densely packed spatial arrangement of qubits and the physical manufacturing process, it is expected that some qubits may exhibit suboptimal performance. D-Wave estimates that approximately 5% of the qubits in the Pegasus topology may have imperfections. This dense graph structure of the Pegasus systems has allowed for a significant increase in the number of useful qubits, making the Advantage QPU the most promising system to solve the optimisation problems presented in Sections "Integer/binary linear programming (BLP)", "Integer/Binary Linear Programming with quadratic constraints" and "Integer/binary quadratic programming (BQP)".

Quantum devices like those from D-Wave are subject to several causes of errors, including thermal noise, control errors, flux bias drifts and decoherence, which can lead to wrong solutions. To mitigate some of these, D-Wave has implemented Drift Correction techniques[51] to compensate for slow fluctuations in qubit parameters by periodically measuring and adjusting flux biases to maintain computational accuracy. Additionally, D-Wave has demonstrated on the new Advantage-2 prototype Zero-Noise Extrapolation (ZNE)[52] using controlled noise amplification and extrapolation to estimate error-free results. However, ZNE has not yet been implemented on commercially available systems. Generally, D-Wave uses statistical sampling by running multiple annealing cycles to generate a distribution of solutions, mitigating the noise introduced by these faulty qubits.

Due to their physical implementation, current quantum annealers can only encode BLP and BQP problems, so terms and constraints have to be transformed into this form[35]. For detailed information on Pegasus and other D-Wave topologies refer to[53]. Additionally, the next generation of topology, called Zephyr[54], has already been announced with even higher connectivity. When working with any of these topologies, the embedding of the problem onto the physical hardware is of high importance. While D-Wave does provide a large amount of functionalities to help in accomplishing this process, this is no easy task and requires significant knowledge of the architecture and the problem at hand.

Constraints

Once the Hamiltonian is set, the annealing process will result in a minimal energy configuration of the system. No restrictions regarding conditions or constraints on the binary variables can be enforced directly into this procedure. Constraints can, however, be accounted for by adding quadratic penalty terms directly to the Ising Hamiltonian, also known as Lagrangian relaxation[55]. By penalizing the unwanted solutions with large enough penalty factors, a cost is added to make some of the solutions unfavourable.

A linear equality constraint $P(\mathbf{x}) = \left(\sum_i^N a_i x_i - b\right)$ with binary variables $x_i = \{x_i : \forall i \in N\}$ can be written as

$$\lambda \left(\sum_i^N a_i x_i - b \right)^2 \equiv \lambda P(\mathbf{x})^2, \tag{6}$$

where b is some constant that has to be met. However, when confronted with a linear inequality constraint of the form $P(\mathbf{x}) = \left(\sum_i^N a_i x_i - b\right) \leq 0$, auxiliary slack variables, $\mathbf{y} = \{y_j : \forall j \in J\}$ and their corresponding weights w_j are used to transform inequality constraints into equality constraints[56] so that Eq. (6) can be written as

$$\lambda \left(\sum_i^N a_i x_i + \sum_j^W w_j y_j - b \right)^2 \equiv \lambda P(\mathbf{x})^2. \tag{7}$$

The number of slack variables W needed can be determined based on the coefficients a_i and b_i, with an upper limit of $W = b$.

The entire new QUBO, after adding all constraints, can then be written as

$$F(\mathbf{x}) = \mathrm{Obj}(\mathbf{x}, Q) + \sum_k \lambda_k P_k(\mathbf{x})^2, \tag{8}$$

with coupling values Q and summing over all constraints k. With this approach, constrained optimisation problems can be reformulated into a solvable QUBO. We note that other approaches than Lagrangian relaxation have been considered and are actively researched to account for the constraints[57]. Furthermore, the penalty strength λ in Eq. (8) has to be chosen with care. Contrary to classical computing, there is a far more limited dynamic range of possible pre-factors and coupling strengths that can be applied. Too large values cause a

tremendous amount of distortion of the distribution in the ground state. At the same time, too small values will not ensure the satisfaction of the constraint as breaking it could then still result in optimal, unfeasible solutions. The best penalty strength is thereby the lowest possible value that still satisfies the constraints. There is no consensus on how to consistently guarantee this behaviour, and only some guidelines can be found either empirically or algebraically[33,58].

State of the art of quantum annealing

The most capable quantum annealers as of today are owned by D-Wave. Therefore, in this paper, we focus on analysing and comparing the performance of their most advanced system against that of classical computers. While D-Wave promotes the capabilities of their machines in solving optimisation problems with 1 million variables and 100,000 constraints[40], it is essential to note that these claims are strongly problem-dependent and not currently realizable simultaneously. The limit of 1 million variables is only valid for unconstrained problems with the possibility of adding constraints via penalties manually. For constrained models, the Advantage system can handle 500,000 variables and 100,000 constraints. These numbers, however, are only the case for D-Wave's hybrid solvers, which split the problem into different partitions, where some are solved classically and others on the QPU depending on how suited these sub-problems are for either of the systems. As an example, continuous variables are usually run on classical computers, as here quantum annealers are not suspected to provide any computational advantage[59]. For pure quantum computation, without any partitioning, the capabilities of the Advantage system are significantly reduced and dependent on the exact embedding of the problem onto the QPU topology. Additionally, the larger the problem size, the "smaller" the logical qubits get, resulting in reduced connectivity and poorer solution quality. The maximum number of variables for a fully connected QUBO on the Advantage hardware is currently at 180, meaning the architecture can only support up to this amount (not considering any faulty qubits) of fully interacting logical variables[60]. For larger fully connected QUBOs, D-Wave uses hybrid workflows and partitioning of the problem into several sub-problems.

Hybrid computing

One of the main issues with modern quantum computing and quantum annealers is their scalability. For most industrial problems, the current number of qubits is not sufficient to run purely via quantum methods. However, hybrid workflows pave the way to tackle this issue by identifying and decomposing the problem into several sub-problems. Hereby, the strengths of both classical computers and quantum annealers for suited tasks can be utilized similarly to GPUs used for training machine learning models. D-Wave's *Leap Hybrid Solvers*[44] are cloud-based solvers and are the currently leading hybrid quantum annealing algorithm. Due to its proprietary nature, it acts like a black box, however, some generally known features are summarized in the following. When identifying and decomposing the problem into several sub-problems, the *LeapHybridCQMSolver*, designed for constrained quadratic models, automatically implements an appropriate penalty term for the constraint with suitable Lagrangian multipliers. D-Wave does not guarantee that this setting is optimal as these might differ for certain problems.

After the preprocessing, a hybrid workflow[44] is implemented utilizing classical heuristics like simulated annealing and Tabu search[37]. Simultaneously, the quantum hardware is used to run smaller binary quadratic unconstrained optimisation problems with the potential to boost the classical counterparts, noting that linear binary problems can easily be transformed into QUBOs. Furthermore, the hybrid solvers also take over the task of embedding the problem onto the quantum hardware and run the QUBO partitions several times, creating a sampleset of solutions that then can be incorporated into the hybrid workflow again. As both the quantum annealing and the integrated classical methods are heuristic, the entire output is also heuristic. Thereby the solver does not guarantee the optimal solutions but returns a sampleset of several results. The entire hybrid approach is probabilistic. Finally, the exact nature of these solvers is not known, however, D-Wave enables the creation of custom hybrid workflows which, if properly implemented, might outperform the *LeapHybridSolvers*.

Case study

This section presents different case studies and their results. The exploration covers various classes of optimisation problems: Integer/Binary Linear Programming (BLP), while also considering the influence of additional constraints of both linear and quadratic nature in Section "Integer/Binary Linear Programming (BLP)", and Integer/Binary Quadratic Programming (BQP) in Section "Integer/binary quadratic programming (BQP)". To assess the performance of D-Wave's Advantage system for industrial use, we benchmark the solution quality and computational time of D-Wave's *LeapHybridCQMSolver*[45] against results produced by the widely adopted classical solvers CPLEX[41], Gurobi[42], and IPOPT[43] (version 3.14.12). The classical solvers are executed on a dedicated computer cluster equipped with dual 3.5 GHz Intel Xeon Gold 5115 CPUs, each featuring 10 cores and 96 GB of RAM. Noting here that for the time comparison, we measured the total run time as outputted by the cloud service, which includes the time it takes to return the full sample-set of on average 100 sample solutions. From these solutions, we first filter out the feasible set before selecting the solution with the lowest energy value. The time required for these two steps was not included in the computational time measurement for obtaining sufficient solutions. While these steps typically take less than 1s on average, their impact may only be considerable for very short total run times.

Integer/binary linear programming (BLP)

Initially, our focus is on exploring optimisation problems characterized by variables that are exclusively either integer or binary values, with linear objective functions and constraints. We consider how solution quality and execution times vary with the number of assigned variables for the hybrid CQM solver and compare its performance with the introduced classical ones. The problem is formulated as:

$$\min \quad \sum_{i=1}^{N} \mu_i x_i \tag{9}$$

$$\text{s.t.} \quad \sum_{i=1}^{N} x_i = C, \tag{10}$$

where μ_i are random constants between (0,1) and $x_i \in \{0, 1\}$ are binary variables. This restricts the sum over all binary variables to equal a fixed constant C. We first run the solvers with 10,000 to 100,000 binary variables. A comparison of the solutions and the computational time for this range of binary variables is shown in Fig. 1a. Here, we average over 5 runs, each providing a sampleset of approximately 100 solutions from which the best feasible one is selected. As we experienced little deviation in the results we deemed higher statistics not to be necessary. Additionally, as access to quantum computers is still very limited, large statistics are not realisable. IPOPT is designed to handle large-scale nonlinear problems[43] and is not suited for integer variables. As IPOPT approximates the binary variables with continuous ones with very small bounds around zero and one, the derived objective function value with these numbers can score a lower value than what is actually true. We excluded IPOPT's objective function values in Fig. 1b, however, we kept this solver throughout this work as it shows important results in Section "Integer/binary quadratic programming (BQP)". Looking at Fig. 1a, D-Wave stays relatively constant in computation time of about 6s up to 38,000 variables, from whereon it increases more drastically when the number of variables is increased further. Viewing the objective function value, we see that IPOPT behaves similarly to the other solvers for a low number of variables, but starts diverging from the rest as the number of variables increases. It can also be noted that the computational time of IPOPT increases rapidly compared to the other classical solvers. CPLEX and Gurobi start below 1s and gradually increase with an increasing number of variables. D-Wave finds the same optimal solution as Gurobi and CPLEX, resulting in three overlapping lines. The observed sawtooth pattern of the solutions is due to the randomness of the μ –values as they are newly drawn for each run with a specific seed. When comparing computational time in Fig. 1b, the hybrid solver requires significantly more time as it approaches its scalability limit. This behaviour could be explained by the need for an extremely high amount of partitioning into subproblems as the number of variables increases[35]. Although the optimal solution has been found, D-Wave's hybrid solvers show no computational advantage.

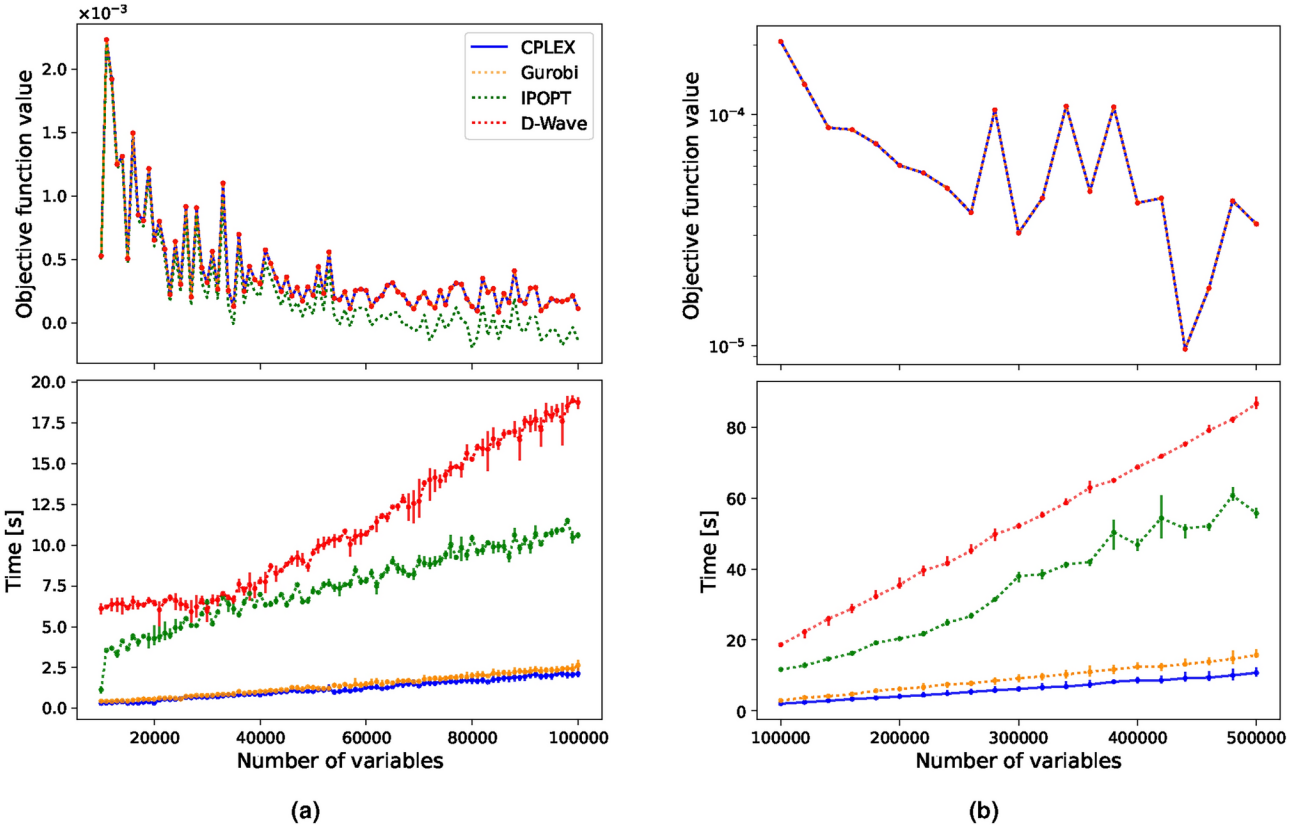

Fig. 1. Mean objective function value and computational time comparison between CPLEX, Gurobi, IPOPT and D-Wave's hybrid solver for the **BLP** with a single linear constraint. The results are averaged over 5 runs and the minimum and maximum deviation from the mean value is shown by the vertical lines.

Increasing the complexity by adding more constraints

Next, we focus on determining whether D-Wave could continue to find the optimal solution at its scalability limit when increasing the complexity of the BLP (Eq. (9)) by adding additional constraints. As the constraints are added as penalty terms, we want to see if the solution quality can be maintained even though the computational time is significantly higher. We consider some arbitrary although feasible constraints listed below.

$$\begin{aligned}
&\sum_{i}^{N} x_i \geq C, \\
&\sum_{i}^{N} x_i = \sum_{j}^{N} x_j, \quad i \in 2\mathbb{N}, j \in 2\mathbb{N}+1, \\
&\sum_{i}^{N/2} x_i \leq \sum_{i=N/2}^{N} x_i, \\
&\sum_{i}^{N} x_i = 0, \quad i/5 \in \mathbb{N}, \\
&\sum_{i}^{N} \mu_{i+1} x_i \leq \sum_{i}^{N} \mu_{i-1} x_i.
\end{aligned} \tag{11}$$

The objective function value and computational time averaged over 5 runs together with the minimum and maximum deviation as vertical lines are shown in Fig. 2. We observe that the solution quality stagnates with more constraints and in particular for the 5th added constraint a significant variability in the result with an approximate increase up to a factor of five. D-Wave's computational time is significantly higher reaching above 600s on the fully constrained problem. Furthermore, we ran the same problem with just 10,000 variables where D-Wave was able to find the optimal value with all 5 constraints added in every one of the 5 runs. This is a clear indication that D-Wave's CQM hybrid solver can not ensure optimal solutions at its variable limit. Although no thorough study on the best setting for the Lagrange multipliers was conducted, minor adjustments to their values showed no improvements, supporting our conclusion that D-Wave fails to provide optimal solutions at its variable limit with an increasing number of constraints.

Integer/binary linear programming with quadratic constraints

In the previous section, all added constraints were linear. The examination, in this section, involves the consideration of a quadratic constraint while keeping the objective function linear. To this end, we added another factor to the constraint from Eq. (10) so that it now reads

$$\sum_{i,j=1}^{N} x_i x_j \geq C, \tag{12}$$

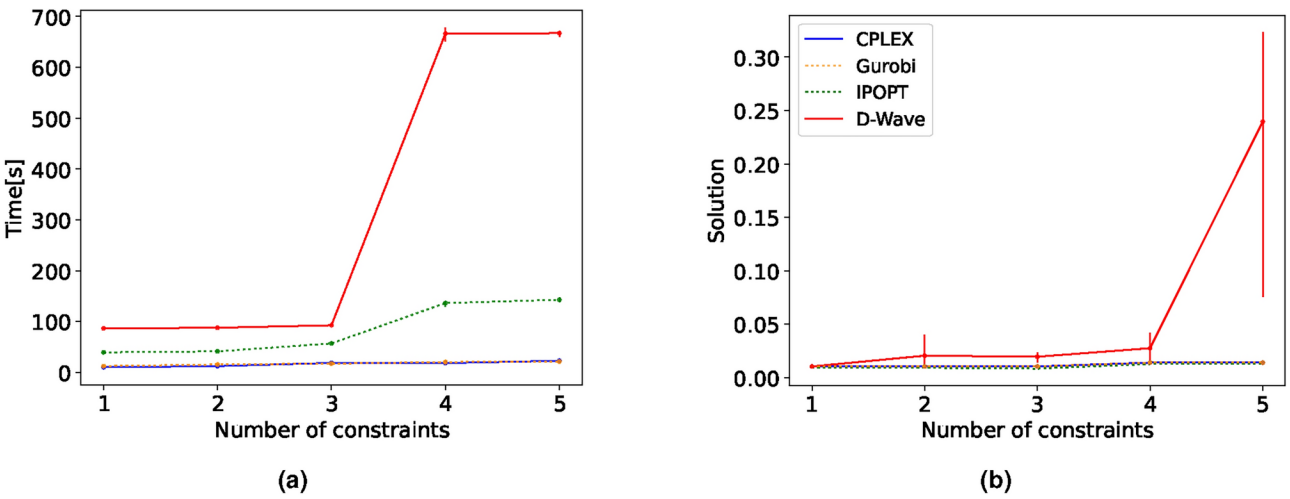

Fig. 2. Mean objective function value (**a**) and computational time (**b**) comparison between CPLEX, Gurobi, IPOPT and D-Wave's hybrid solver for the **BLP** with increasingly more constraints. The results are averaged over 5 runs and the minimum and maximum deviation from the mean value is shown by the vertical lines.

where $x_{i,j} \in \{0, 1\}$, resulting in N^2 terms. At first, we viewed the solution quality and computational time for an increasing number of variables, shown in Fig. 3a, followed by a stepwise increase in the complexity of the constraint by increasing the value of C in Fig. 3b.

In the first scenario, we observe that D-Wave finds worse solutions than both CPLEX and IPOPT whereas IPOPT's value ranks lower due to the approximate representation of the binary variables as already mentioned in the previous section.

The solution obtained by CPLEX corresponds to the actual optimal solution from which D-Wave is continuously approximately a factor of two off. Notably, although D-Wave does not find the optimal solution, the best solution found in every run is the same, resulting in no observable error bars. This indicates that D-Wave gets stuck in a local minima. Nonetheless, D-Wave solves the problem faster than IPOPT from 500 variables and CPLEX from 600 variables onwards. Although D-Wave did not find the optimal solution a single time, the computational time is unaffected by the increase of variables.

We experience similar behaviour when stepwise increasing the value of the constraint. D-Wave is again slightly off the optimal solution, however, it stays relatively constant in computational time and solves the problem almost consistently faster than IPOPT. Although the solution is suboptimal, it can be seen in Fig. 3b that also for quadratic constraints with a linear objective function, the hybrid solver is not able to find the optimal solution in any run. We note that we did not adjust the Lagrange multipliers and kept the ones chosen by default by the hybrid solver. We do not rule out that customised settings for the parameters could be beneficial to the quality of the solution.

Integer/binary quadratic programming (BQP)

Binary quadratic problems are NP-hard[61] and thus challenging to solve classically, even for a relatively small number of variables and constraints. As the structure of BQP resembles the Ising Hamiltonian, Eq. (5), these problems are especially suited for quantum annealers[62]. By multiplying with and summing over an additional set of binary variables x_j in Eq. (9), we acquire a BQP of the form

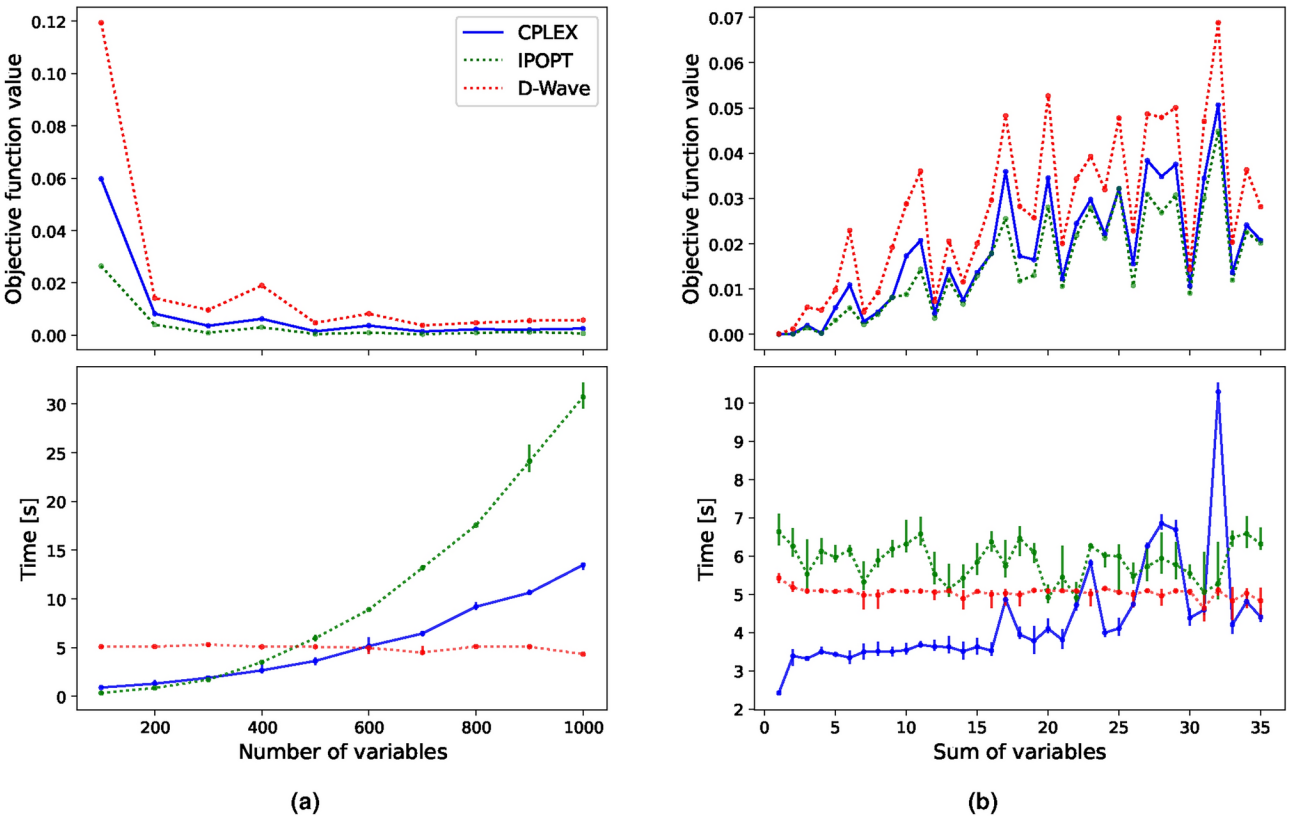

Fig. 3. Mean objective function value (top) and computational time (bottom) comparison between two classical solvers and D-Wave's hybrid solver for the **BLP** with a single quadratic constraint. In (**a**) the complexity scales with the number of variables. In (**b**), the variables are fixed to 500 whereas the complexity scales with increasing C as stated in Eq. (10). All results are averaged over 5 runs, and the vertical lines show the minimum and maximum deviation from the mean value. Gurobi's solve time is significantly larger than all other solvers and is thereby excluded here for visibility.

$$\min \quad \sum_{i,j=1}^{N} \mu_{ij} x_i x_j \tag{13}$$

$$\text{s.t.} \quad \sum_{i=1}^{N} x_i = C, \quad x_i, x_j \in \{0, 1\}. \tag{14}$$

To benchmark the different solvers for this BQP, we increase the complexity by incrementally increasing the value of the constant C. Thereby, the complexity of the task is increased as the feasible solution space increases. This is preferred over a further increase of N to avoid an unnecessarily large memory usage as the number of terms in the objective function increases with N^2.For the comparisons in this section, we choose a sufficiently large problem with $N = 500$, although different sizes were tested, showing the same tendencies. The results are shown in Fig. 4.

Considering both solution and run time, we observe a clear computational advantage of the hybrid solver to CPLEX. We cut CPLEX off after 1000s of runtime, resulting in sub-optimal solutions when the solution space increases. Gurobi performed even worse than CPLEX for this problem and was excluded in the figure for visibility reasons. While IPOPT shows comparable solutions and run time to D-Wave, the solutions are still not optimal. D-Wave finds consistently optimal solutions in less computational time, which again results in no error bars. To more clearly visualise this, we show the relative difference in solution quality between D-Wave's hybrid solver and the objective function value of both IPOPT and CPLEX in Fig. 5.

We see from Fig. 5 that CPLEX performs well in the lower complexity region and that D-Wave does not find exclusively better solutions than CPLEX here, however, due to the capped computational time, D-Wave outperforms consistently for higher values of C. While IPOPT's relative difference becomes increasingly small, CPLEX, with a capped solve-time of 1000s, seems to settle at a deviation of $\sim 25\%$ from the optimal solution. This shows that D-Wave's hybrid solvers are a suitable alternative for BQPs, even showing a computational advantage for this specific case. However, in the field of optimisation, there is a limited amount of problems that correspond to these pure BQPs.

The unit commitment problem—an energy system application

With the release of the LeapHybridCQM solver[45], D-Wave opened itself up to a wider range of problems, such as Mixed Integer Linear Problems (MILP), with a wide spread of applications in various fields. To evaluate its performance, we benchmark D-Wave's current performance on a real-world Unit Commitment problem, a complex optimisation challenge in the power sector that plays a crucial role in power system operation and planning. The Unit Commitment Problem involves determining the optimal scheduling of generating units—deciding when each unit should be turned on or off—over a given time horizon. The goal is to meet the forecasted load at minimum total production cost while satisfying a range of constraints, including power balance, reserve requirements, transmission limits, and individual unit constraints such as generation limits, minimum up/down times, and ramp rates[63]. For this purpose, a well-established MILP formulation of the Unit Commitment problem is chosen[64], which consists of both continuous and binary variables while maintaining a fully linear structure in its objective function and constraints. The problem is tested at three different scales as defined in Table 1. The objective function of a unit commitment model is as follows:

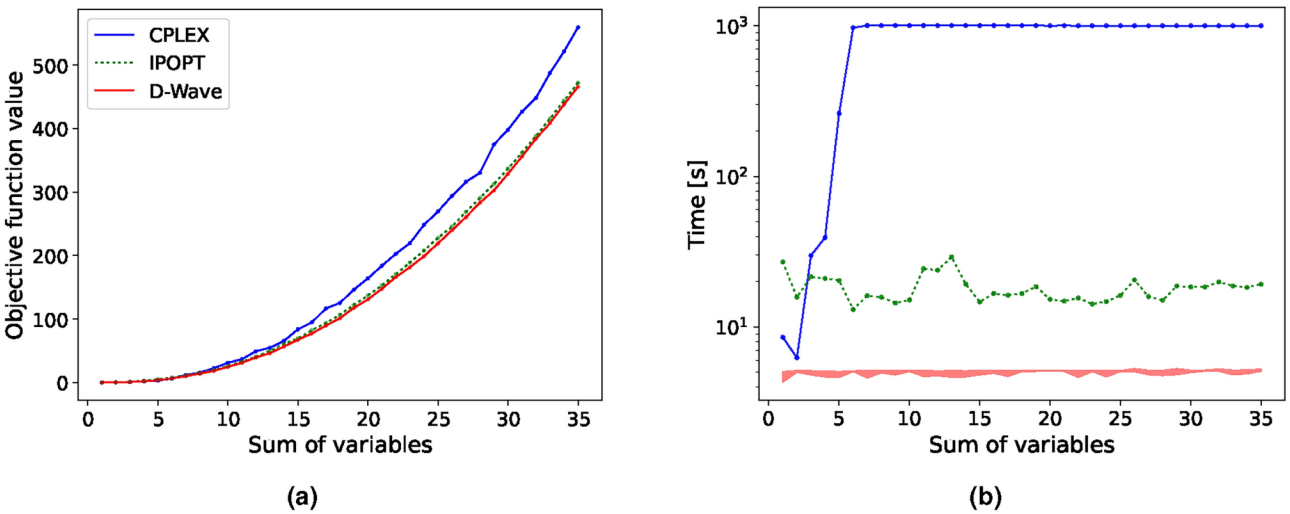

Fig. 4. Comparison of the objective function value (**a**) and computational time (**b**) between D-Wave and the previously introduced classical solvers for **BQP** with 500 binary variables. The results are averaged over 5 runs. The run time is limited to 1000s, resulting in CPLEX not finding the optimal solution for increasing C.

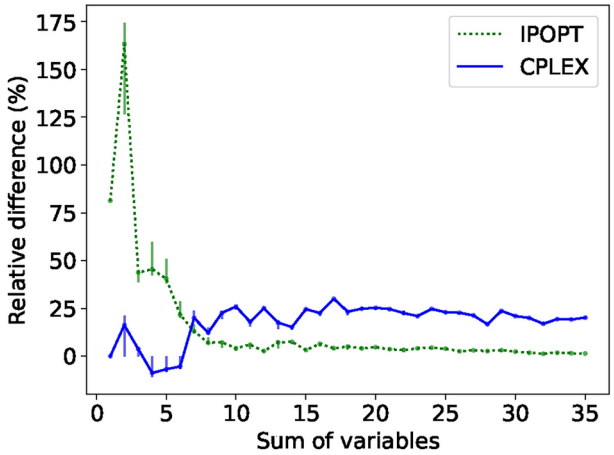

Fig. 5. Mean relative difference of D-Wave's hybrid solution and those supplied by CPLEX and IPOPT for the **BQP**. The vertical lines indicate the difference between CPLEX's and IPOPT's mean solution to the maximum and minimum solution provided by D-Wave within all 5 runs.

	Number of variables	Number of constraints	Time periods [h]	Segments
Full scale	44,544	42,899	48	4
Reduced-1	11,136	10,169	12	4
Reduced-2	1418	1526	2	1

Table 1. Parameters of the full scale and the two reduced versions of the Unit Commitment problem adjusted from[64] and using the rts_gmlc data set therein.

$$\min \sum_{g\in G}\sum_{t\in T}\left(\underbrace{c_g(t)+CP_g^1\,u_g(t)}_{i}+\underbrace{\sum_{s=1}^{S_g}\left(CS_g^s\delta_g^s(t)\right)}_{ii}\right), \tag{15}$$

where the objective function consists of two components: (i) the production cost and (ii) the startup cost. $c_g(t)$ represents the cost of the power produced by thermal generator *g* above its minimum output at time *t*, CP_g^1 represents the cost of operating at the minimum power output for generator *g*, while $u_g(t)$ is a binary variable that indicates the commitment status of generator *g* at time *t*. In the final term, CS_g^s represents the startup cost in category *s* for generator *g*, where $s \in S_g$ represents the startup categories for thermal generator *g*, with $s = 1$ corresponding to the hottest state (shortest downtime) and $s = S_g$ to the coldest state (longest downtime). The variable $\delta_g^s(t)$ indicates the startup in category *s* for thermal generator *g* at time *t*. These three terms are summed over all periods in time $t \in T$ and all generators $g \in G$. For a detailed explanation of this formulation, see[64,65].

For the full-scale problem, the default pre-set minimum run time is set by D-Wave at 20s. In this limit, the hybrid solver was not able to find a single feasible solution, shown in Fig. 6a. We therefore solved the problem 6 times with an increased minimum solve time of 150s and 2 times of 400s, shown in Fig. 6b. For comparison, the near-optimal solution with a gap of 0.05% was found by Gurobi in 180s with a value of 1.23×10^6. We can observe a clear increase in solution quality by the hybrid solver as in both cases feasible solutions were found. The best feasible solution for the 150s case yielded within 6 runs an objective function value of 5.9×10^6, while two runs with 400s run time yielded a value of 5.7×10^6. This is a clear indication that the minimum run time chosen by the solver itself is not optimal and has to be adjusted specifically for the problem at hand as a balance between computational time and solution quality. Next, an investigation was conducted to observe how D-Wave would perform if the problem was downsized, as stated in Table 1. First, the Reduced-1 problem with 12 time periods and 4 segments was examined, making it approximately one-fourth of the size of the initial problem. We kept the default minimum run-time setting of 10s, as this is already significantly longer than Gurobi's solve time of 1.34s for the optimal solution without a gap. The solution distribution of 70 runs is shown in Fig. 7a, while the deviation of each run to the classical optimal solution is presented in Fig. 7b.

Although D-Wave found at least one feasible solution in every run except for one, its best solution differs by approximately a factor of 10 from the optimal. The Reduced-2 problem is reduced even further to just 2 time periods and a single segment of linearisation. The solution distribution over 70 runs is shown in Fig. 8a and the deviation from the optimal classical solution in Fig. 8b. The hybrid solver finds a feasible solution in every run with the best run being $\sim 2.4\%$ off from the optimal one. When comparing solve time, D-Wave takes on average 5s while Gurobi finds the best solution in less than 1s thereby showing no quantum computing advantage. We

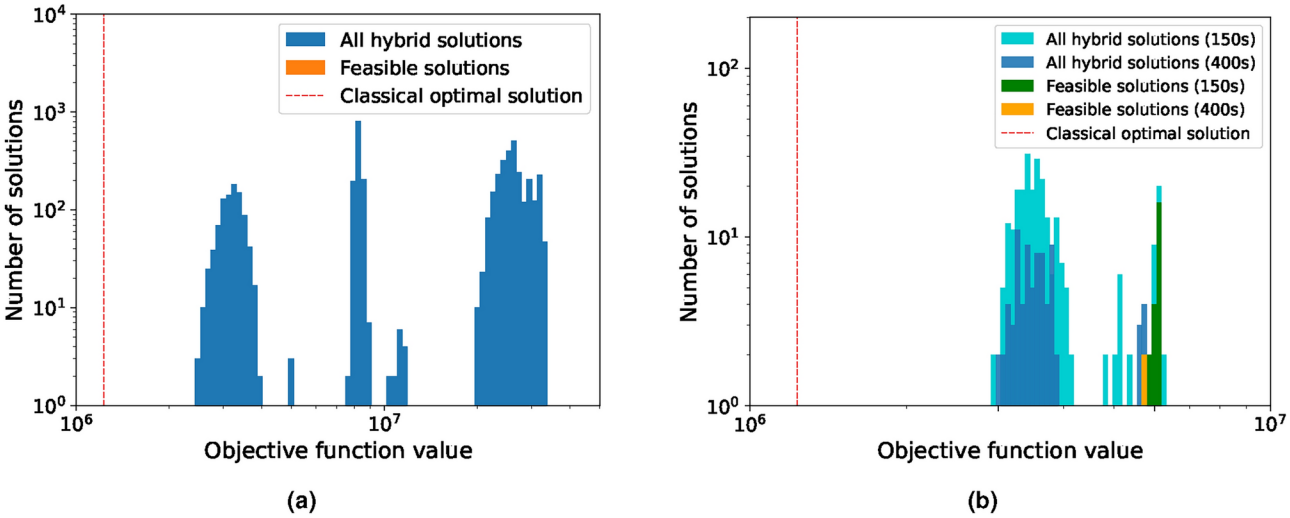

Fig. 6. Solution distribution from D-Wave's LeapHybridCQM solver for the full-scale unit commitment problem. (**a**) shows the results of 50 runs with the solver's default chosen run time of 20s. (**b**) shows the solution for the same problem with 6 runs for an increased run time of 150s and 2 runs for one of 400s. Each run supplies a sampleset with on average 100 solutions each.

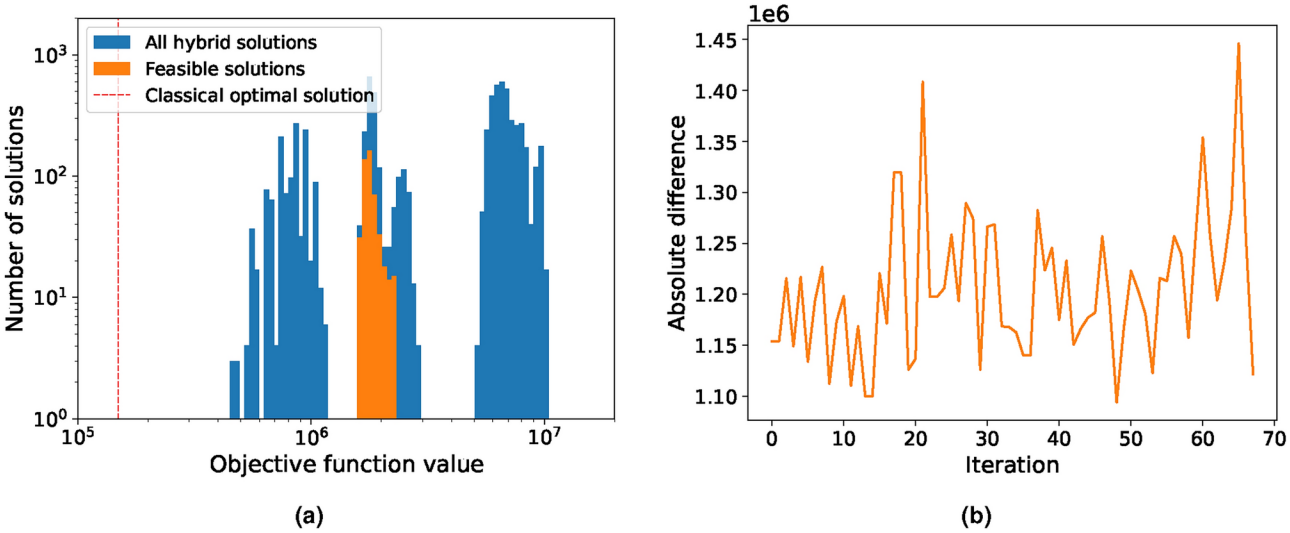

Fig. 7. Solution distribution of the Reduced-1 unit commitment problem. (**a**) shows the objective function values of 70 runs (blue) given by D-Wave's LeapHybridCQM solver with an average of 100 solutions each. The feasible solutions are highlighted in orange. (**b**) shows the absolute difference between the optimal classical and the best feasible hybrid solution for each run for the Reduced-1 problem. Only one of the 70 runs did not obtain a feasible solution.

note that the problem in this reduced state is no longer representable of a real-world problem, as a 2h time frame is too short for a unit commitment problem. We conclude that D-Wave's hybrid solver is currently not in a state to effectively solve large-scale MILP unit commitment problems of this type, considering both computational time and solution quality. It is also worth mentioning that minor adjustments to some of the Lagrange multipliers for the constraints did not show any improvements.

Conclusions

We provided an introduction to quantum annealing and conducted a benchmark of D-Wave's current generation of quantum-annealing-based hybrid solvers against state-of-the-art classical algorithms. Through a comparison of computational times and the corresponding best solutions across several case studies spanning a wide range of optimisation problems, we conclude that a state-of-the-art hybrid solver running on the most advanced quantum annealer has reached a point where it is competitive with the best classical approaches for a limited set of real-world problems.

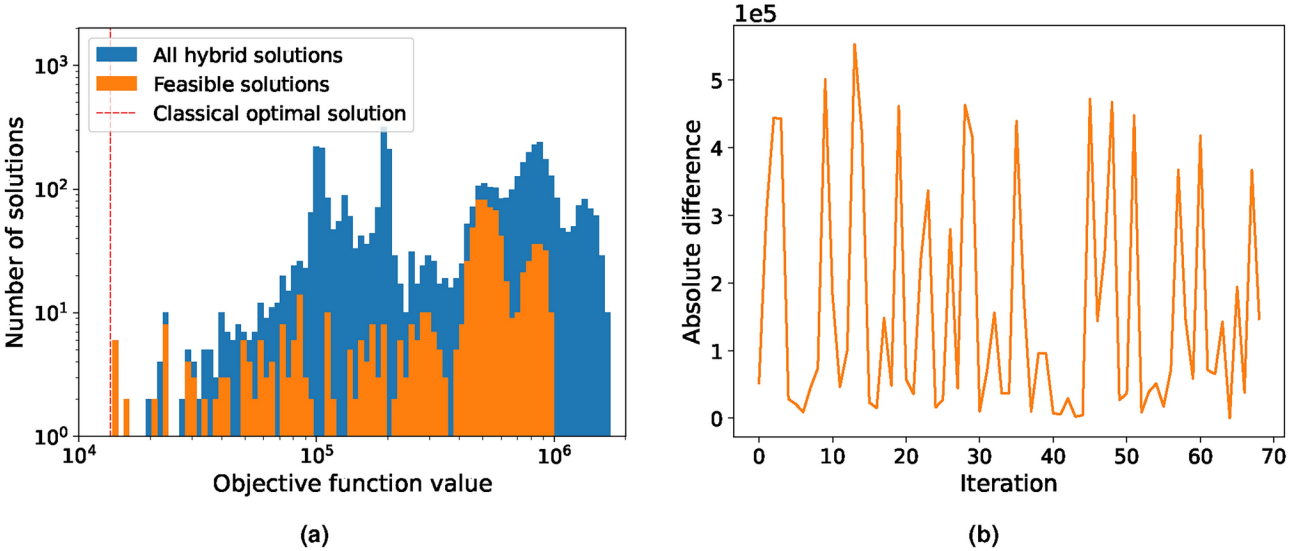

Fig. 8. Solution distribution of the Reduced-2 unit commitment problem. (**a**) shows the objective function values of 70 runs (blue) given by D-Wave's LeapHybridCQM solver with an average of 100 solutions each. The feasible solutions are highlighted in orange. (**b**) shows the absolute difference between the optimal classical and the best feasible hybrid solution for each run for the Reduced-2 problem. A feasible solution was found in every run.

For BLP, D-Wave's hybrid solver stays consistent with both CPLEX and Gurobi and finds near-optimal solutions for all our test cases. However, the computational time grows significantly when the number of variables increases.

Although the hybrid-solver approach enables larger problem sizes than what is currently possible for full quantum approaches, such problems still have to be heavily partitioned into many subproblems and stuck back together, leading to increasing computational time[35].

By increasing the complexity of the problem through systematically adding additional constraints, we conclude that the solution quality diverges more and more from the classical solvers for an increasing number of constraints at D-Wave's scalability limit. Additionally, the increase in run time is so drastic that the hybrid solver shows no benefit at this scale. Although we did see some competitive computational times by D-Wave when handling quadratic constraints in Section "Integer/Binary Linear Programming with quadratic constraints", the hybrid solver was not able to find the optimal solution with a factor of approximately two off in most runs. It is for the BQP that D-Wave's hybrid solver seems the most promising. Due to the mathematical structure of BQPs being so closely related to the Ising Hamiltonian of Eq. (5), D-Wave's hybrid solver performs better than all three classical solvers. The most competitive classical solver is IPOPT, but D-Wave does find slightly better solutions in a shorter amount of time. Compared with CPLEX, which due to its deterministic nature will find the optimal solution, D-Wave is superior in regards to computational time. Regarding the MILP unit commitment problem, the benchmark underscored the importance of fine-tuning the minimum run time to enhance the solution quality. Despite various adjustments and reduced problem cases, D-Wave's *LeapHybridCQMSolver* was unable to surpass Gurobi in terms of computational time or solution quality. As this specific problem has no quadratic binary instance in either the objective function or in any of the constraints, the case study indicates that these instances are currently necessary to potentially show a computational benefit. While it is promising that D-Wave can directly solve full MILP problems using their hybrid solvers, the computational advantage appears to be limited to BQP problems.

The findings presented in this paper indicate that D-Wave's hybrid solvers and Advantage quantum computers currently exhibit competitiveness with classical optimisation algorithms, but this holds true only for a limited range of problems. While D-Wave's hybrid solver performs well with binary quadratic problems, its competitiveness diminishes when dealing with non-binary, non-quadratic problems, or combinations thereof. Consequently, quantum annealers with broader capabilities have strong prospects for the future. The field of quantum annealing and quantum computing, in general, is undergoing rapid and promising advancements. Between 2020 to 2021 alone, D-Wave enhanced the performance of the Advantage machine, achieving an approximately eightfold increase in its ability to find solutions for Satisfiability problems and obtaining solutions for 3D lattice problems twice as fast[66]. They have already announced Zephyr for 2024[54]. As classical solvers are experiencing regular performance updates, benchmarking research, as demonstrated in this paper, becomes imperative.

Data availability

The rts_gmlc dataset used for the unit commitment problem can be found in[64]. All other datasets used and/or analyzed during the current study are available from the corresponding author upon reasonable request.

Received: 23 January 2024; Accepted: 26 March 2025
Published online: 13 April 2025

References

1. Mooney, G. J., White, G. A., Hill, C. D. & Hollenberg, L. C. Whole-device entanglement in a 65-qubit superconducting quantum computer. *Adv. Quantum Technol.* **4**, 2100061. https://doi.org/10.1002/qute.202100061 (2021).
2. Arute, F. et al. Quantum supremacy using a programmable superconducting processor. *Nature* **574**, 505–510. https://doi.org/10.1038/s41586-019-1666-5 (2019).
3. Quantum, M. A. et al. Interferometric single-shot parity measurement in inas-al hybrid devices. *Nature* **638**, 651–655 (2025).
4. Feynman, R. P. Simulating physics with computers. *Int. J. Theor. Phys.* **21**, 467–488. https://doi.org/10.1007/BF02650179 (1982).
5. Benioff, P. The computer as a physical system: A microscopic quantum mechanical Hamiltonian model of computers as represented by turing machines. *J. Stat. Phys.* **22**, 563–591. https://doi.org/10.1007/BF01011339 (1980).
6. Chuang, I., Gershenfeld, N. & Kubinec, M. Experimental implementation of fast quantum searching. *Phys. Rev. Lett.* **80**, 3408–3411. https://doi.org/10.1103/PhysRevLett.80.3408 (1998).
7. Banchi, L., Bayat, A., Verrucchi, P. & Bose, S. Nonperturbative entangling gates between distant qubits using uniform cold atom chains. *Phys. Rev. Lett.* **106**, 140501. https://doi.org/10.1103/PhysRevLett.106.140501 (2011).
8. Bruzewicz, C. D., Chiaverini, J., McConnell, R. & Sage, J. M. Trapped-ion quantum computing: Progress and challenges. *Appl. Phys. Rev.* https://doi.org/10.1063/1.5088164 (2019).
9. Kok, P. et al. Linear optical quantum computing with photonic qubits. *Rev. Modern Phys.* **79**, 135. https://doi.org/10.1103/RevModPhys.79.135 (2007).
10. Kloeffel, C. & Loss, D. Prospects for spin-based quantum computing in quantum dots. *Annu. Rev. Condens. Matter. Phys.* **4**, 51–81. https://doi.org/10.1146/annurev-conmatphys-030212-184248 (2013).
11. Ladd, T. D. et al. Quantum computers. *Nature* **464**, 45–53. https://doi.org/10.1038/nature08812 (2010).
12. McGeoch, C. C. *Adiabatic Quantum Computation and Quantum Annealing: Theory and Practice* (Springer, 2022).
13. Devoret, M. H., Wallraff, A. & Martinis, J. M. *Superconducting Qubits: A Short Review*. https://doi.org/10.48550/arXiv.cond-mat/0411174 (2004).
14. Shor, P. W. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM J. Comput.* **26**, 1484–1509. https://doi.org/10.1137/s0097539795293172 (1997).
15. Cleve, R. & Watrous, J. Fast parallel circuits for the quantum Fourier transform. In *Proceedings 41st Annual Symposium on Foundations of Computer Science*, 526–536. https://doi.org/10.1109/SFCS.2000.892140 (2000).
16. Daley, A. J. et al. Practical quantum advantage in quantum simulation. *Nature* **607**, 667–676. https://doi.org/10.1038/s41586-022-04940-6 (2022).
17. Grimsley, H. R., Economou, S. E., Barnes, E. & Mayhall, N. J. An adaptive variational algorithm for exact molecular simulations on a quantum computer. *Nat. Commun.* **10**, 3007. https://doi.org/10.1038/s41467-019-10988-2 (2019).
18. Chatterjee, Y., Bourreau, E. & Rančić, M. J. Solving various np-hard problems using exponentially fewer qubits on a quantum computer. https://doi.org/10.48550/arXiv.2301.06978 (2023).
19. Au-Yeung, R., Chancellor, N. & Halffmann, P. Np-hard but no longer hard to solve? Using quantum computing to tackle optimization problems. *Front. Quantum Sci. Technol.* https://doi.org/10.3389/frqst.2023.1128576 (2023).
20. Van Dam, W., Mosca, M. & Vazirani, U. How powerful is adiabatic quantum computation? In *Proceedings 42nd IEEE Symposium on Foundations of Computer Science*, 279–287. https://doi.org/10.1109/SFCS.2001.959902(2001).
21. Harwood, S. et al. Formulating and solving routing problems on quantum computers. *IEEE Trans. Quantum Eng.* **2**, 1–17. https://doi.org/10.1109/TQE.2021.3049230 (2021).
22. Simon, B. Holonomy, the quantum adiabatic theorem, and berry's phase. *Phys. Rev. Lett.* **51**, 2167. https://doi.org/10.1103/PhysRevLett.51.2167 (1983).
23. Parekh, O. Synergies between operations research and quantum information science. *INFORMS J. Comput.* **35**, 266–273. https://doi.org/10.1287/ijoc.2023.1268 (2023).
24. Abbas, A. *et al.* Quantum optimization: Potential, challenges, and the path forward. https://doi.org/10.48550/arXiv.2312.02279 (2023).
25. Syrichas, A. & Crispin, A. Large-scale vehicle routing problems: Quantum annealing, tunings and results. *Comput. Oper. Res.* **87**, 52–62. https://doi.org/10.1016/j.cor.2017.05.014 (2017).
26. Bernal, D. E. et al. Perspectives of quantum computing for chemical engineering. *AIChE J.* **68**, e17651. https://doi.org/10.1002/aic.17651 (2022).
27. Rosenberg, G., Vazifeh, M., Woods, B. & Haber, E. Building an iterative heuristic solver for a quantum annealer. *Comput. Optim. Appl.* **65**, 845–869. https://doi.org/10.1007/s10589-016-9844-y (2016).
28. Glover, F., Lewis, M. & Kochenberger, G. Logical and inequality implications for reducing the size and difficulty of quadratic unconstrained binary optimization problems. *Eur. Oper. Res.* **265**, 829–842. https://doi.org/10.1016/j.ejor.2017.08.025 (2018).
29. Gu, J., Gu, M., Cao, C. & Gu, X. A novel competitive co-evolutionary quantum genetic algorithm for stochastic job shop scheduling problem. *Comput. Oper. Res.* **37**, 927–937. https://doi.org/10.1016/j.cor.2009.07.002 (2010).
30. Nannicini, G. Fast quantum subroutines for the simplex method. *Oper. Res.* https://doi.org/10.1287/opre.2022.2341 (2022).
31. Liu, Y. & Koehler, G. J. Using modifications to Grover's search algorithm for quantum global optimization. *Eur. J. Oper. Res.* **207**, 620–632. https://doi.org/10.1016/j.ejor.2010.05.039 (2010).
32. Koniorczyk, M., Krawiec, K., Botelho, L., Bešinović, N. & Domino, K. Solving rescheduling problems in heterogeneous urban railway networks using hybrid quantum-classical approach. https://doi.org/10.48550/arXiv.2309.0676 (2023).
33. Neukart, F. et al. Traffic flow optimization using a quantum annealer. *Front. ICT* **4**, 29. https://doi.org/10.3389/fict.2017.00029 (2017).
34. Mugel, S. et al. Dynamic portfolio optimization with real datasets using quantum processors and quantum-inspired tensor networks. *Phys. Rev. Res.* https://doi.org/10.1103/physrevresearch.4.013006 (2022).
35. Yarkoni, S., Raponi, E., Bäck, T. & Schmitt, S. Quantum annealing for industry applications: Introduction and review. *Rep. Progress Phys.* https://doi.org/10.1088/1361-6633/ac8c54 (2022).
36. Mohseni, N. *et al.* A competitive showcase of quantum versus classical algorithms in energy coalition formation. arXiv:2405.11917 (2024).
37. Glover, F. Tabu search: A tutorial. *Interfaces* **20**, 74–94. https://doi.org/10.1287/inte.20.4.74 (1990).
38. Kirkpatrick, S., Gelatt, C. D. Jr. & Vecchi, M. P. Optimization by simulated annealing. *Science* **220**, 671–680. https://doi.org/10.1126/science.220.4598.671 (1983).
39. Farhi, E., Goldstone, J. & Gutmann, S. A quantum approximate optimization algorithm. arXiv:1411.4028 (2014).
40. McGeoch, C. & Farré, P. *An overview* (D-Wave Systems Inc., Burnaby, BC, Canada, Tech. Rep, The d-wave advantage system, 2020).
41. Cplex, IBM ILOG. IBM ILOG CPLEX Optimization Studio 22.1.0. https://www.ibm.com/docs/en/icos/22.1.0?topic=documentation-introducing-ilog-cplex-optimization-studio-2210 (2022).
42. Gurobi Optimization, LLC. Gurobi Optimizer 10.0 documentation. https://www.gurobi.com/documentation/10.0/ (2022).

43. Wächter, A. & Biegler, L. T. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Math. Program.* **106**, 25–57. https://doi.org/10.1007/s10107-004-0559-y (2006).
44. D-Wave Systems Inc. D-Wave Hybrid Solver Service: An Overview. https://www.dwavesys.com/resources/white-paper/d-wave-hybrid-solver-service-an-overview/ (2020).
45. D-Wave Systems Inc. Hybrid Solvers for Quadratic Optimization. https://www.dwavesys.com/media/soxph512/hybrid-solvers-for-quadratic-optimization.pdf (2022).
46. Sheng, Z.-H. et al. Catalyst-free synthesis of nitrogen-doped graphene via thermal annealing graphite oxide with melamine and its excellent electrocatalysis. *ACS Nano* **5**, 4350–4358. https://doi.org/10.1021/nn103584t (2011).
47. Kadowaki, T. & Nishimori, H. Quantum annealing in the transverse Ising model. *Phys. Rev. E* **58**, 5355. https://doi.org/10.1103/PhysRevE.58.5355 (1998).
48. Schrödinger, E. An undulatory theory of the mechanics of atoms and molecules. *Phys. Rev.* **28**, 1049–1070. https://doi.org/10.1103/PhysRev.28.1049 (1926).
49. D-Wave Systems Inc. Ocean Software Suite. https://docs.ocean.dwavesys.com/en/stable/ (2023).
50. Boothby, K., Bunyk, P., Raymond, J. & Roy, A. Next-generation topology of d-wave quantum processors. https://doi.org/10.48550/arXiv.2003.00133 (2020).
51. Perdomo-Ortiz, A., O'Gorman, B., Fluegemann, J., Biswas, R. & Smelyanskiy, V. N. Determination and correction of persistent biases in quantum annealers. *Sci. Rep.* **6**, 18628 (2016).
52. Amin, M. H. *et al.* Quantum error mitigation in quantum annealing. arXiv:2311.01306 (2023).
53. Katzgraber, H. G., Hamze, F. & Andrist, R. S. Glassy chimeras could be blind to quantum speedup: Designing better benchmarks for quantum annealing machines. *Phys. Rev. X* **4**, 021008. https://doi.org/10.1103/PhysRevX.4.021008 (2014).
54. Boothby, K., King, A. D. & Raymond, J. Zephyr topology of d-wave quantum processors. https://www.dwavesys.com/media/2uznec4s/14-1056a-a_zephyr_topology_of_d-wave_quantum_processors.pdf (2021).
55. Fisher, M. L. The Lagrangian relaxation method for solving integer programming problems. *Manag. Sci* **27**, 1–18. https://doi.org/10.1287/mnsc.27.1.1 (1981).
56. Hillier, F. S. *Introduction to Operations Research* (McGraw Hill, 2001).
57. Yu, S. & Nabil, T. Applying the Hubbard–Stratonovich transformation to solve scheduling problems under inequality constraints with quantum annealing. *Front. Phys.* **9**, 730685. https://doi.org/10.3389/fphy.2021.730685 (2021).
58. Yarkoni, S. *et al.* Solving the shipment rerouting problem with quantum optimization techniques. In Mes, M., Lalla-Ruiz, E. & Voß, S. (eds.) *Computational Logistics*, 502–517. https://doi.org/10.1007/978-3-030-87672-2_33 (2021).
59. Ottaviani, D. & Amendola, A. Low rank non-negative matrix factorization with d-wave 2000q. https://doi.org/10.48550/arXiv.1808.08721 (2018).
60. Kuramata, M., Katsuki, R. & Nakata, K. Larger sparse quadratic assignment problem optimization using quantum annealing and a bit-flip heuristic algorithm. In *2021 IEEE 8th International Conference on Industrial Engineering and Applications (ICIEA)*, 556–565. https://doi.org/10.1109/ICIEA52957.2021.9436749 (2021).
61. Hu, H. & Sotirov, R. The linearization problem of a binary quadratic problem and its applications. *Ann. Oper. Res.* **307**, 229–249. https://doi.org/10.1007/s10479-021-04310-x (2021).
62. Farhi, E. et al. A quantum adiabatic evolution algorithm applied to random instances of an np-complete problem. *Science* **292**, 472–475. https://doi.org/10.1126/science.1057726 (2001).
63. Knueven, B., Ostrowski, J. & Watson, J.-P. On mixed-integer programming formulations for the unit commitment problem. *INFORMS J. Comput.* **32**, 857–876 (2020).
64. Coffrin, C. Power grid lib - unit commitment. https://github.com/power-grid-lib/pglib-uc/tree/master (2019). GitHub repository.
65. Morales-España, G., Latorre, J. M. & Ramos, A. Tight and compact MILP formulation for the thermal unit commitment problem. *IEEE Trans. Power Syst.* **28**, 4897–4908. https://doi.org/10.1109/TPWRS.2013.2251373 (2013).
66. D-Wave. The Most Connected and Powerful Quantum Computer Built for Business. https://www.dwavesys.com/solutions-and-products/systems/ (2022).

Acknowledgements
We acknowledge the support of the iDesignRES (Integrated Design of the Components of the Energy System to Plan the Uptake of Renewable Energy Sources)1 project which has received funding from the Horizon Europe research and innovation program under Grant Agreement No. 101095849.

Author contributions
F.A.Q, M.B., P.A.S.M., and P.C.G. conceived the work and writing-editing the revision; F.A.Q and P.A.S.M. conducted the simulation; F.A.Q, P.A.S.M, and M.B. analysed the results; F.A.Q., P.A.S.M, M.B., P.C.G and H.Z. wrote or edited the original draft; P.C.G. funding acquisition; All authors reviewed the manuscript.

Funding
Open access funding provided by NTNU Norwegian University of Science and Technology (incl St. Olavs Hospital - Trondheim University Hospital)

Declarations

Competing interests
The authors declare that they have no known competing financial interests or personal relationships that could influence the work reported in this paper.

Additional information
Correspondence and requests for materials should be addressed to F.A.Q. or P.C.d.G.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.