Benchmarking Quantum Annealing Controls with Portfolio Optimization

Erica Grant,[1,2,*] Travis S. Humble,[1,2,†] and Benjamin Stump[3,‡]

[1]*Quantum Computing Institute, Oak Ridge National Laboratory, Oak Ridge, Tennessee 37830, USA*

[2]*Bredesen Center, University of Tennessee, Knoxville, Tennessee 37996, USA*

[3]*National Transportation Research Center, Oak Ridge National Laboratory, Knoxville, Tennessee 37932, USA*

Quantum annealing offers an approach to finding the optimal solutions for a variety of computational problems, where the quantum annealing controls influence the observed performance and error mechanisms by tuning the underlying quantum dynamics. However, the influence of the available controls is often poorly understood, and methods for evaluating the effects of these controls are necessary to tune quantum computational performance. Here we use portfolio optimization as a case study by which to benchmark quantum annealing controls and their relative effects on computational accuracy. We compare empirical results from the D-Wave 2000Q[TM] quantum annealer to the computational ground truth for a variety of portfolio optimization instances. We evaluate both forward and reverse annealing methods and we identify control variations that yield optimal performance in terms of probability of success and probability of chain breaks.

I. INTRODUCTION

Optimization is integral to many scientific and industrial applications of applied mathematics including verification and validation, operations research, data analytics, and logistics, among others [1,2]. In many cases, exact methods of solution, including stochastic optimization and quadratic programming, are computationally intractable and new heuristics are frequently used to solve problems in practice [3]. Quantum annealing (QA) offers a meta-heuristic that uses quantum mechanics for unconstrained optimization by encoding the problem cost function in a Hamiltonian [4,5]. Recovery of the Hamiltonian ground state solves the original optimization problem and this approach has been mapped to a variety of application areas [6–9]. Several experimental efforts have realized quantum annealers [10–12], and application benchmarking of these systems has shown QA is capable of finding the correct result with varying probabilities of success [13–19].

There are ongoing efforts to evaluate quantum annealing hardware and algorithms to identify resource advantages over conventional computation for real-world applications [20–22]. Simulated quantum annealing (SQA) has been demonstrated to be exponentially faster than classical simulated annealing unless the Hamiltonians are nonstoquastic [23,24]. However, QA has not been shown to outperform SQA for stoquastic Hamiltonians [22]. QA performance depends implicitly on the complexity of the underlying problem instance as well as the controls that implement the heuristic [25,26]. Presently, there are multiple controls available to program quantum annealers that may each impact the observed probability of success. Notionally, the controls may be categorized as preprocessing, annealing, and postprocessing methods. Whereas preprocessing controls define the encoded Hamiltonian and embedding onto the quantum annealer [27,28], the annealing controls drive the time-dependent physics of the device and the underlying quantum state [25,29] while postprocessing controls influence the readout and decoding of the observed results [30,31]. Collectively, the choice for each type of control may either enhance or impede the probability of reaching the encoded ground state and, therefore, impact the resulting solution state.

Here we benchmark a selection of preprocessing and annealing controls available in a programmable quantum annealer [10] using a well-defined class of unconstrained optimization problems derived from the application of Markowitz portfolio theory [32]. As a variant of binary optimization, Markowitz portfolio optimization selects the subset of investment assets expected to yield the highest return value and minimal risk while staying within a

*egrant8@vols.utk.edu

†humblets@ornl.gov

‡stumpbc@ornl.gov

total budget constraint [32,33]. We cast this problem that forms a complete graph as unconstrained optimization and benchmark the probability of success for QA to recover the global optimum. In particular, we benchmark the preprocessing and annealing controls available in the 2000Q, a programmable quantum annealer from D-Wave Systems [10]. This includes controls for mapping the logical problem onto hardware and scheduling the annealing process. We gather insight into the underlying dynamics using multiple measures of success tested across an ensemble of randomly generated instances of portfolio optimization.

Previous research has benchmarked QA in comparison to classical heuristics for solving various optimization problems [14,34,35]. In particular, several variations of portfolio optimization have been used to benchmark QA performance [25,36,37]. Rosenberg *et al.* [37] demonstrated several encodings of a multiperiod Markowitz portfolio optimization formulation to be solvable using a quantum annealer and found promising initial results in probability to find the optimal result. Venturelli and Kondratyev [25] benchmarked a similar mean-variance model of portfolio optimization using a hybrid solver that couples quantum annealing with a genetic algorithm. This hybrid algorithm was found to be 100 times faster than forward annealing alone. In this work, we present a formulation of portfolio optimization to benchmark the behavior of QA controls. We present studies focused on the variability in success with respect to available quantum annealing controls in an attempt to establish a methodology for finding an optimal set of controls that yield the highest solution quality [38,39].

The presentation is organized as follows. In Sec. II, we review quantum annealing and the available controls. In Sec. III, we provide an overview of the benchmarking methods and the use of Markowitz portfolio selection for problem specification. In Sec. IV, we present the results from experimental testing using different quantum annealing controls with the 2000Q. We draw our conclusions in Sec. V.

II. QUANTUM ANNEALING

Under ideal conditions, forward annealing evolves a quantum state $|\Psi(t)\rangle$ under the time-dependent Schrödinger equation

$$i\hbar \frac{d}{dt}|\Psi(t)\rangle = H(t)|\Psi(t)\rangle, \qquad t \in [0, T], \quad (1)$$

where T is the total forward annealing time and the time-dependent Hamiltonian is

$$H(t) = A(s(t))H_0 + B(s(t))H_1, \quad (2)$$

where $s(t) \in [0, 1]$ is the control schedule and time-dependent amplitudes $A(s)$ and $B(s)$ satisfy the conditions

$A(0) \gg B(0)$ and $A(1) \ll B(1)$. We consider the initial Hamiltonian $H_0 = -\sum_i^n \sigma_i^x$ as a sum of Pauli-X operators σ_i^x over n spins. The final Hamiltonian H_1 represents the unconstrained optimization problem with a corresponding ground state that encodes the computational solution. We consider below only problems represented using the Ising Hamiltonian

$$H_1 = \sum_i h_i \sigma_i^z + \sum_{i,j} J_{i,j} \sigma_i^z \sigma_j^z + \beta, \quad (3)$$

where h_i is the bias on the ith spin, $J_{i,j}$ is the coupling strength between the ith and jth spin, σ_i^z is the Pauli-Z operator for the ith spin, and β is a problem-specific constant. The Ising Hamiltonian is well known for representing a variety of unconstrained optimization problems [40].

Instantaneous eigenstates at time t are defined as

$$H(t)|\Phi_j(t)\rangle = E_j(t)|\Phi_j(t)\rangle, \quad (4)$$

where j ranges from 0 to $N - 1$ with $N = 2^n$ the dimension of the Hilbert space. For an initial quantum state prepared in the lowest-energy eigenstate at time $t = 0$, i.e., $|\Psi(0)\rangle = |\Phi_0(0)\rangle$, adiabatic evolution under the Hamiltonian in Eq. (2) to time T will prepare the final state $|\Psi(T)\rangle = |\Phi_0(T)\rangle$ with high probability provided T is sufficiently large. In particular, the evolution must be much longer than the inverse square of the minimum energy gap between the ground and first excited states [4]. At time T, the prepared quantum state is measured in the computational basis to generate a candidate solution for the encoded problem.

Another variation of quantum annealing reverses the time-evolution process by beginning in an eigenstate of H_1. Known as reverse annealing, the initial quantum state evolves under Eq. (2) in the reverse direction. The Hamiltonian starts as H_1 at time $t = 0$ and evolves backward to a point s_p in the control schedule that corresponds to a ramp time of $t_r = t_1$. The Hamiltonian then optionally pauses for a time $t_p = t_2 - t_1$ before evolving in the forward direction from the value s_p at time t_2 back to the final Hamiltonian at time T' for a total quench time of $t_q = T' - t_2$, where the latter time represents the reverse annealing time. Reverse annealing typically starts in the excited state of the final Hamiltonian, such that the annealing process may lead to nonadiabatic dynamics in regions of small energy gaps. However, thermal relaxation during the pause time t_p has been shown to be a leading factor in settling to the ground state [29]. The control schedule for reverse annealing is then defined as [41,42]

$$s'(t) = \begin{cases} 1 + \dfrac{s_p - 1}{t_1}t, & 0 \le t \le t_1, \\ s_p, & t_1, \le t \le t_2, \\ s_p + \dfrac{1 - s_p}{T' - t_2}(t - t_2), & t_2 \le t \le T'. \end{cases} \quad (5)$$
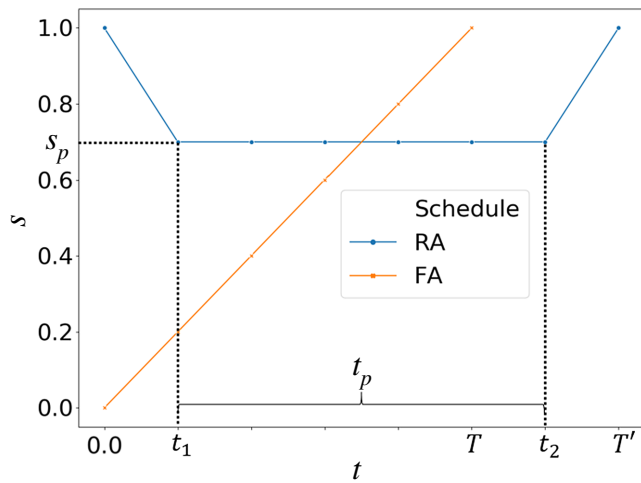
FIG. 1. The control schedule for reverse annealing (RA) compared to forward annealing (FA) plotted with respect to time. The control schedule for forward annealing starts at $t = 0, s = 0$ and anneals at a constant rate to $t = T, s = 1$, while the control schedule for reverse annealing starts at $t = 0$ with $s = 1$, decreases to a value s_p at time t_1, pauses for time $t_p = t_2 - t_1$, and then increases to $s = 1$ at time T'.

The differences in the control schedules of forward and reverse annealing are demonstrated in Fig. 1, where a linear reverse annealing schedule is compared to a linear forward annealing schedule using the amplitudes $A(s) = (1 - s)$ and $B(s) = s$. Notably, forward annealing controls increase monotonically with time whereas reverse annealing controls include a change in the direction of the control schedule.

A. Quantum annealing controls

In practice, there are nonideal behaviors that arise in practical implementations of quantum annealing. Equations (1)–(5) represent quantum annealing under ideal adiabatic conditions that are difficult to realize in actual quantum devices. Real-world quantum annealers have limits in the ability to control the Hamiltonian and quantum dynamics [43]. In addition, the presence of ill-characterized environmental couplings give rise to flux noise [44]. The imperfect setting of the Hamiltonian parameters $(h, J_{i,j})$ by the analog control circuits gives rise to a small intrinsic control error [39]. These errors undermine the accuracy of the physical hardware [27,43]. Finally, annealing too quickly may violate the essential adiabatic condition [4], while annealing too slowly may lead to undesired thermal excitations of the quantum state due to nonzero temperature fluctuations [45]. This multitude of effects complicates both the description of quantum annealing as well as the assessment of its performance.

Given the implicit dependence on several competing factors, a variety of strategies have emerged for controlling quantum annealing to maximize probability of success in recovering the ground state and minimizing errors in the quantum computational solution. These control strategies include efficiently mapping the problem Hamiltonian onto the physical hardware Hamiltonian, tuning the annealing schedule, applying variable transformations to mitigate control biases, and using reverse annealing to refine initial solutions [39,41].

We investigate a subset of controls available in the D-Wave 2000Q, a programmable quantum annealer composed from an array of superconducting flux qubits operated at cryogenic temperatures [46]. The 2000Q consists of up to 2048 physical qubits arranged in a sparsely connected array whose governing Hamiltonian is described by a time-dependent, transverse Ising Hamiltonian for which, with the Hamiltonian parameters in the device, can be programmed individually [47,48]. This enables a broad variety of computational problems, including portfolio optimization, to be realized. We briefly review some of the controls available to influence the success of solving these problems using quantum annealing.

1. Problem embedding

The Hamiltonian encoding the computational problem must be mapped into the physical hardware while satisfying the constraints of limited connectivity. The 2000Q hardware supports a sparse Chimera graph in which physical qubits are not fully connected but have average degree 6. A fully connected graph, like in Fig. 2, must be mapped onto the more sparse Chimera graph. A single spin from the input Hamiltonian may be realized in hardware using multiple physical qubits that form a strongly interacting representative chain of spins. By judiciously choosing these chains and their interactions, the original input Hamiltonian may be constructed. This process, known as embedding, depends on the input problem as well as the target hardware connectivity. In general, embedding is NP-hard for arbitrary input graphs [49], and there are upper limits on the maximum graph that can be embedded [50]. For example, the largest fully connected problem that can be embedded onto the 2000Q has approximately 60 spins, while the limit in practice depends on the number of faulty physical qubits in the device.

Embedding algorithms that optimize chain length may greatly reduce the number of physical qubits required by considering problem symmetry as well as the location of faults in the hardware. We highlight two embedding algorithms widely used in programming the 2000Q. The first method by Cai *et al.* [51] is based on randomized placement and search for the weighted shortest path between spin chains. This method, which we denote as Cai, Maccready, and Roy (CMR), applies to arbitrary input graphs, but typically creates a distribution of chain lengths. By contrast, a second method by Boothby *et al.* [52] based on a

clique embedding typically generates shorter and uniform chain lengths of size

$$l_c = \frac{n}{4} + 1 \qquad (6)$$

for n logical spins. A representative example of the output from these different methods is shown in Fig. 2 using a fully connected problem with 20 logical spins. Both methods are available in the D-Wave Ocean software library [53].

Ensuring an embedded chain of qubits that collectively represents a single logical variable requires an intrachain coupling that is larger in magnitude than the interchain couplings between chains. In other words, the chain of physical qubits must be strongly coupled to remain a single
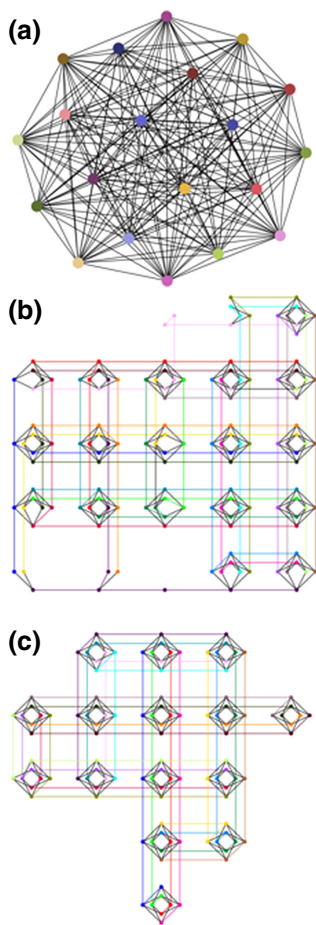

FIG. 2. The embedding of a 20 logical spin complete graph onto a Chimera graph structure. (a) The complete K_{20} graph that is fully connected with 20 nodes and 190 edges where each node represents a logical spin and each edge is a coupling between spins. (b) The CMR algorithm that requires the allocation of 23 unit cells and (c) the clique embedding algorithm that requires the allocation of 15 unit cells. The nodes represent physical qubits, lines are the couplings between physical qubits, and each color is a different physical spin chain corresponding to a logic spin.

logical spin. However, it is possible for chains to become "broken" in so far as individual physical spins within the chain differ in their final state. In general, chain breaks arise from nonadiabatic dynamics that lead to local excitation out of the lowest energy state with longer chains more susceptible to these effects [39,54].

An additional control is required for decoding embedded chains to recover the computed logical spin state. In the absence of chain breaks, the logical value is inferred directly from the unanimous selection of a single spin state by every physical qubit. In the presence of chain breaks, several strategies may be employed to decide the logical value including majority vote [39], which selects the logical spin value as the value that occurs with the highest frequency in a chain.

2. Spin reversal

Interactions between embedded chains arise from the required coupling between the logical spins. However, imperfections in the control of these spins lead to small biases that can become non-negligible for larger qubit chains and contribute to the complex dynamics describing the device. In turn, the probability of finding the expected ground state solution can decrease due to these bias errors. The influence of these errors on the computational result may be mitigated by using spin reversal transforms to average out biases.

As a gauge transformation, spin reversal redefines the Hamiltonian by replacing the biases and couplings for a subset of spins with their negated value [38,39]. This transformation maintains the ground state of the logical problem. However, this transformation flips the sign of randomly selected qubits so that on average their bias is reduced. This strategy mitigates errors on individual spins by balancing the noise on the device prior to annealing [31]. The control g dictates the number of times to perform the transformation.

3. Annealing schedules

Tailoring the annealing amplitudes $A(s)$ and $B(s)$ is perhaps the most direct method to control forward annealing. The annealing schedules control the rate of change of $H(t)$, which must be sufficiently slow to approximate the adiabatic condition [55]. An example of the amplitudes in a D-Wave 2000Q is shown in Fig. 3. While forward annealing on the D-Wave 2000Q, $A[s(t)] \gg B[s(t)]$ at $t = 0$, $A[s(t)]$ decreases and $B[s(t)]$ increases for $0 < t < T$, and $B[s(t)] \gg A[s(t)]$ at $t = T$.

The optimal annealing time is problem dependent and inversely proportional to the minimum energy gap [4], and, in general, the value and position of the minimum energy gap for a given $H(t)$ is typically unknown and hard to identify. Extending the annealing time T arbitrarily long may not only be limited by hardware parameters
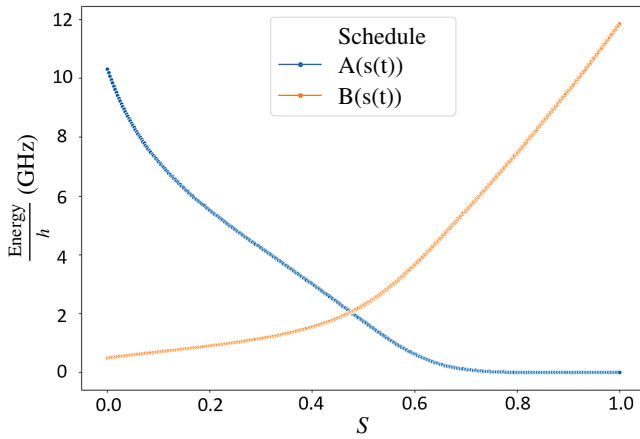
FIG. 3. The amplitudes of the D-Wave 2000Q over the range of control schedules as measured from $s = 0$ to $s = 1$ in increments of 0.001.

but also be counterproductive due to competing thermal processes that depopulate the ground state [30,56]. There is an upper limit to the total job time ($N_s T \leq 1$ s) as well as the total annealing time ($T \leq 2$ s) on the D-Wave 2000Q.

When reverse annealing, the primary parameters for control are the initial state e_i, the pause point s_p, the pause duration t_p, and the ramp and quench times t_r and t_q. We choose to set t_r and t_q to be constant and symmetric at 5 μs for our experiments. Reverse annealing uses e_i to set the initial quantum state, which may be based on the output of forward annealing, a heuristically computed candidate, a random state, or other methods. Our experiments use a precomputed initial state, e.g., using forward annealing. An iterative procedure is then used that replaces the e_i of each subsequent reverse annealing sample with the output from the previous reverse annealing iteration.

B. Quantum annealing metrics

We characterize quantum annealing using the probability of success

$$p_s = \sum_{i=1}^{N_g} \langle \Phi_0(T) | \rho_i | \Phi_0(T) \rangle, \tag{7}$$

defined as the overlap of the final, potentially mixed quantum state ρ prepared by QA with the pure state describing the expected computational outcome $\Phi_0(T)$. Empirically, the probability of success is estimated from the frequency with which the observed solution state matches the expected outcome. When the expected ground state solution is known, we define the statistic $\delta_i = 1$ if the ith sample matches the known ground state and $\delta_i = 0$ if it does not. For the kth problem Hamiltonian instance, the

estimated probability of success is then defined as

$$\tilde{p}_s^{(k)} = \frac{1}{N_s} \sum_{i=1}^{N_s} \delta_i, \tag{8}$$

where N_s is the total number of samples. The average over an ensemble of N_p problem instances is defined as

$$\tilde{p}_s = \frac{1}{N_p} \sum_{k}^{N_p} \tilde{p}_s^{(k)}. \tag{9}$$

A second metric for characterizing quantum annealing performance, and especially the nonadiabatic dynamics, is the number of chain breaks observed in the recovered solution samples. As noted above, a chain break is observed when the chain of physical qubits embedding a logical spin has more than one unique spin value. We estimate the probability of chain breaks for a problem instance as

$$\tilde{p}_b^{(k)} = \frac{1}{N_s} \sum_{i=1}^{N_s} \epsilon_i, \tag{10}$$

where the statistic $\epsilon_i = 1$ when the ith sample solution contains at least one broken chain for any of the logical spins and $\epsilon_i = 0$ when no embedded chain is broken. The average probability of chain breaks over an ensemble of N_p problem instances is then defined as

$$\tilde{p}_b = \frac{1}{N_p} \sum_{k}^{N_p} \tilde{p}_b^{(k)}. \tag{11}$$

It is important to note that the effects of chain breaks can be mitigated by postprocessing methods, such as majority vote, which make hard decisions on the logical spin value.

While the above metrics quantify the probability with which quantum annealing recovers the correct solution, additional information about computational performance comes from the distribution of all solution samples obtained. In particular, the distribution over sample energies provides a representation for the weight of errors in the solution samples. A distribution concentrated around the lowest energy indicates a small number of errors in the computed solutions, while a broad or shifted distribution hints at a larger number of errors. We denote the energy computed from the ith solution sample as $E(i)$ and we define the jth energy bin as h_j. The bin h_j counts the number of samples with an energy in the range $[j, j+1]\Delta$, where Δ controls the granularity of binning the energies. The resulting set $\{(j\Delta, h_j)\}$ approximates the energy distribution of the sampled solutions.

III. BENCHMARKING METHODS

We benchmark performance of the quantum annealing controls presented in Sec. II using a variant of constrained optimization derived from Markowitz portfolio theory. We recast this problem as an unconstrained optimization before reducing to a quadratic unconstrained binary optimization (QUBO) form. The latter form is easily translated to the classical Ising spin Hamiltonian and, subsequently, to the problem Hamiltonian defined by Eq. (3).

A. Markowitz portfolio selection

Portfolio optimization selects the best allocation of assets to maximize expected returns while staying within the budget and minimizing financial risk. The Markowitz theory for portfolio selection focuses on diversification of the portfolio for risk mitigation [32]. Instead of allocating high percentages of a budget toward assets with the highest projected returns, the budget is distributed over assets that minimize correlation between the asset's historical prices. In this model, the covariance between purchasing prices serves as a proxy for risk in which positively correlated assets are considered to be more risky. We review the methods by which the benchmark problems are generated and solved in this section.

We consider Markowitz portfolio optimization as a quadratic programming problem that determines the fraction of available budget b to allocate toward purchasing assets with the goal of maximizing returns while minimizing risk. By selecting a partition number w, the fraction $p_w = 1/2^{w-1}$ represents the granularity of the partition. The portfolio optimization problem selects how many of those partitions to allocate toward each asset with an integer z_u. Thus, the fraction of b to invest in each uth asset is given by $p_w b z_u$, and portfolio optimization identifies how much of the m assets to select given the budget b and a risk threshold c. Thus, portfolio selection is cast as

$$\max_z \sum_{u=1}^m r_u z_u$$

$$\text{such that} \quad \sum_{u=1}^m p_w b z_u = b \quad \text{and} \quad \sum_{u,v=1}^m c_{u,v} z_u z_v \leq c, \tag{12}$$

where, for the uth asset, r_u is the expected return and $c_{u,v}$ is the historical price correlation between assets u, v.

In Eq. (12), the first term represents maximization of the expected returns over the available assets. There are many methods for forecasting expected returns, e.g., based on market price, expert judgement, and historical price data [57,58]. For simplicity, we model expected returns as

$$r_u = p_w \bar{a}_u, \tag{13}$$

where $\bar{a}_u$ is the average of a_u, the history of price data for the uth asset. The first constraint in Eq. (12) places a hard constraint on the total allocation of assets to sum to b. We emphasize that this constraint penalizes portfolios that do not allocate the entire budget as well as those that over commit. Finally, the second constraint accounts for diversification by asserting that the sum of the covariance between asset prices $c_{u,v}$ be less than or equal to the risk threshold c. The historical price covariance is calculated as the correlation between pairs of assets by comparing the p_w fraction of each asset's historical price data. Here covariance is defined as

$$c_{u,v} = \frac{p_w^2 \sum_{l=1}^{N_f} (a_{u,l} - \bar{a}_u)(a_{v,l} - \bar{a}_v)}{N_f - 1}, \tag{14}$$

where $a_{u,l}$ is the lth historical price value for asset u and N_f is the number of price points in the historical data.

We solve this variation of Markowitz portfolio selection using quantum annealing by casting the formulation in Eq. (12) into QUBO. We express the integer variable z_u as a w-bit binary expansion

$$z_u = \sum_{k=1}^w 2^{k-1} x_{i(u,k)} \tag{15}$$

with $x_i \in \{0,1\}$ and the composite index $i(u,k) = (u-1)w + k$. The expected returns are then expressed as

$$r_u z_u = \sum_{k=1}^w 2^{k-1} r_u x_{i(u,k)}, \tag{16}$$

while the allocation constraint becomes the penalty term

$$-\left(\sum_{u=1}^m \sum_{k=1}^w 2^{k-1} p_w b x_{i(u,k)} - b \right)^2. \tag{17}$$

We consider a correlation threshold $c = 0$ such that the correlation constraint becomes

$$\sum_{u,v}^m c_{u,v} z_u z_v = \sum_{u,v}^m \sum_{k,k'}^w 2^{k-1} 2^{k'-1} c_{u,v} x_{i(u,k)} x_{j(v,k')}. \tag{18}$$

Our formulation of Markowitz portfolio selection as an unconstrained optimization problem then becomes

$$\max_x \quad \theta_1 \sum_i^n r_i x_i,$$

$$-\theta_2 \left(\sum_i^n 2^{k-1} b p_w x_i - b \right)^2, \quad -\theta_3 \sum_{i,j}^n c_{i,j} x_i x_j, \tag{19}$$

where the problem size $n = mw$, $r_i = 2^{k-1} r_u$, $c_{i,j} = 2^{k-1} 2^{k'-1} c_{u,v}$, and $\theta_1, \theta_2,$ and θ_3 are Lagrange multipliers used to weight each term for maximization or penalization.

For purposes of benchmarking, we generate an ensemble of problem instances by sampling from uniform random price data with a seed of $b/5$. A random number is drawn as the initial price $a_{u,1}$ and every subsequent historical price point up to the purchasing price is -25% to $+25\%$ of the previous price $a_{u,l}$. The price range is set to be between $b/10$ and b with $N_f = 100$ historical price points per asset. In addition, we normalize all $a_{u,l}$ by a_{u,N_f} to keep all asset prices to a similar range.

We set $\theta_1 = 0.3, \theta_2 = 0.5, \theta_3 = 0.2$ in the problem instances where θ_2 is set higher to enforce the budget constraint. These weights are chosen after testing which combination stayed on budget and gave some diversity. By keeping θ_2 constant and increasing θ_3 while decreasing θ_1, an investor could increase the diversity relative to the potential returns and vice versa when decreasing θ_3 relative to θ_1. We generate 1000 problems for each problem size with $m = 2, 3, 4, 5$ assets and $w = 4$ slices. The value of $w = 4$ yields smallest coupling magnitudes of the order of $1/2^{w-1}$; however, larger values of w may exceed the precision of the D-Wave 2000Q hardware controls [59,60].

B. QUBO to Ising Hamiltonian

We formalize the unconstrained portfolio optimization problem in Eq. (19) to QUBO as

$$\min_x \left(\sum_i^n q_i x_i + \sum_{i,j}^n Q_{i,j} x_i x_j + \gamma \right), \quad (20)$$

where q_i is the linear weight for the ith spin, $Q_{i,j}$ is the quadratic weight for interactions between the ith and jth bits, and γ is a constant. Note that our definition of QUBO expresses optimization as minimization by switching the sign of Eq. (19) to be consistent with the use of quantum annealing to recover the lowest-energy state. The corresponding relationships with the original problem instance are given as

$$q_i = -\theta_1 r_i - 2\theta_2 b^2 p_w, \quad (21a)$$

$$Q_{i,j} = \theta_2 b^2 p_w^2 + \theta_3 c_{i,j}, \quad (21b)$$

$$\gamma = \theta_2 b^2. \quad (21c)$$

Similarly, the quadratic binary form may be reduced to a classical Ising Hamiltonian

$$H(s) = \sum_i s_i h_i + \sum_{i,j} s_i s_j J_{ij} + \beta, \quad (22)$$

where spin $s_i \in \{-1, 1\}$ is defined by $s_i = 2x_1 - 1$ with $s = (s_1, s_2, \ldots, s_n)$ while h_i is the spin weight, J_{ij} is the coupling strength, and β is a problem-specific constant.

The parameters for the Ising Hamiltonian are given as

$$J_{i,j} = \frac{1}{4} Q_{i,j}, \quad (23a)$$

$$h_i = \frac{q_i}{2} + \sum_j J_{i,j}, \quad (23b)$$

$$\beta = \frac{1}{4} \sum_{i,j} Q_{i,j} + \frac{1}{2} \sum_i q_i + \gamma. \quad (23c)$$

The classical Ising formulation is then converted into a corresponding quantum Ising Hamiltonian given by Eq. (3) using the correspondence $s_i \rightarrow \sigma_i^z$.

C. Computational methods

We use a D-Wave 2000Q quantum annealer for our experiments. We calculate the probability of success, the probability of chain breaks, and the energy distribution across each problem instance. For each instance, we estimate these metrics by collecting $N_s = 1000$ samples of the computed solution. We use D-Wave's solver API (SAPI) with PYTHON 2.7 to solve each instance of Markowitz portfolio selection using the hardware controls outlined in Sec. II A. We run 1000 samples per problem over a set of 1000 problems for forward annealing examples and 100 problems for reverse annealing examples. We implement the majority vote postprocessing technique for any broken chains to interpret raw solutions returned by the 2000Q. The program implementation and data sets collected from these experiments are available online [61].

For benchmarking purposes, we also solve each problem instance using brute force search for the minimal energy solutions of the QUBO formulation. We compute the complete energy spectrum for each portfolio instance. These energy spectra and the corresponding states are then used as ground truth for testing the accuracy of results obtained from quantum annealing. By sorting the spectrum, we benchmark the success of reverse annealing using initial states e_i sampled from these different parts of the spectrum.

IV. RESULTS

We benchmark quantum annealing controls by evaluating their influence on the probability of success and probability of chain breaks across problem instances. We first characterize how problem parameters influence the baseline performance by estimating the probability of success for forward annealing using $T = 15 \mu s$, $g = 0$, and a randomized embedding strategy. As shown in Fig. 4, we compare $\tilde{p}_s$ for the two cases $w = 1$ and $w = 4$ across increasing n. The estimated probability of success for problems with $w = 4$ is consistently higher for problems with no slicing.
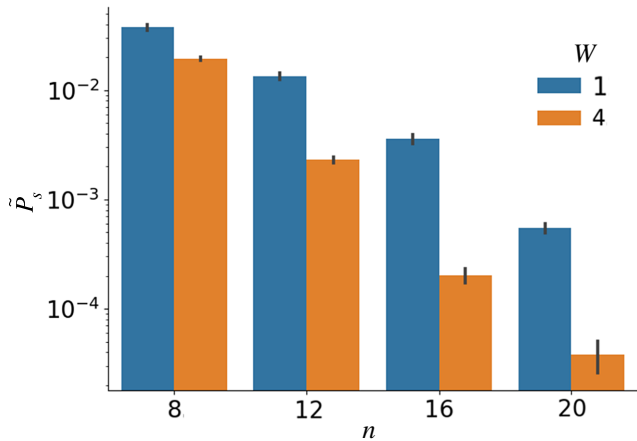
FIG. 4. The average probability of success over 1000 problems each with 1000 samples using CMR, $g = 0$, and $T = 15\,\mu$s. The comparison is between a set of problems from problem sizes 8 to 20 for $w = 1$ (blue) and $w = 4$ (orange). The problems set to slices $w = 1$ are much less complex and therefore have a much higher probability of success.

FIG. 5. Probability histogram (100 bins) of all possible energies for a problem of size 20 where (a) $w = 1$ and (b) $w = 4$. There is a higher density of states close to e_0 in (b) and therefore more opportunities to jump to an excited state throughout the sample.

These results are explained by the energy spectra for the different problem parameters, which indicate sharp differences in the density of states. As shown in Fig. 5, a typical problem instance with $w = 4$ has a much higher density of states than those with no slicing ($w = 1$). Intuitively, the single-slice behavior results from the specification that the price for each asset is proportional to budget, and, therefore, only a single asset may be selected without penalty when $w = 1$. However, the number of satisfying solutions v increases for arbitrary w combinatorially and, as shown in the Appendix,

$$v = \frac{(2^{w-1} + m - 1)!}{(2^{w-1})!\,(m-1)!}. \tag{24}$$

Consequently, the probability to recover the lowest-energy state competes with these closely spaced, higher-energy solutions, which leads to a corresponding decrease in the probability of success. For the remaining benchmark tests below, we choose $w = 4$ as it represents a more challenging test for the quantum annealer as well as a greater interest to real-world financial applications.

A. Benchmarking forward annealing controls

1. Embedding

Embedding generates and places the physical spin chains for each logical spin on the quantum annealing hardware. We evaluated the CMR and clique embedding algorithms described in Sec. IV A 1 by estimating the probability of success across problem sizes of $m = 8, 12, 16$, and 20 logical spins. For all problem instances of the same problem size, we use the same embedding because they require the same number of fully connected logical spins.
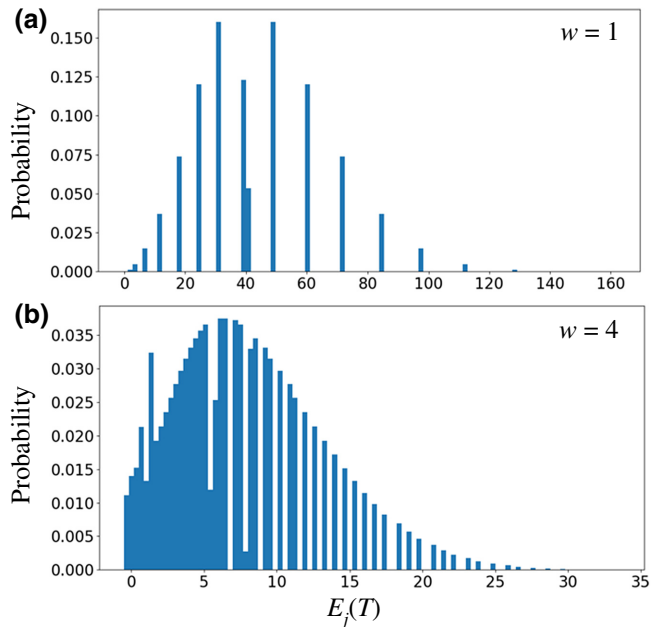
We set the parameters of the embedded Ising Hamiltonian by scaling the interchain couplings $J_{i,j}$ to lie in the range $[-1, +1]$. We scale all $J_{i,j}$ using a rescale factor of $1/j_{\max}$, where $j_{\max}$ is the largest $J_{i,j}$ so all embedded $J_{i,j} = (1/j_{\max})J_{i,j}$. This scales all $J_{i,j}$ to be between ± 1. The intrachain coupling strength is set to -1 to have a negative bias stronger than the $J_{i,j}$ values whose range is $-10^{-1} \leq J_{i,j} \leq 10^{-1}$ due to our data generation and normalization techniques.

The average chain length $\langle l_c \rangle$ from CMR and clique embedding methods grows with the number of logical spins n. The average is computed with respect to all chains in an embedding and plotted with respect to n in Fig. 6. As expected from Eq. (6), the clique embedding method has a uniform chain length for each n. By contrast, the CMR method generates chains of variable length, as indicated by the average chain length and variance shown in the plot.

From each of the embedding methods, we estimate the probability of success and probability of broken chains. As shown in Fig. 7, we observe very small differences in both metrics with increasing problem size. From fitting the resulting point to an exponential form, we find that $\tilde{p}_s$ decays exponentially with respect to n with rate $r = -0.523$ for the CMR embedding and $r = -0.528$ for the clique embedding according to the observed fit $\tilde{p}_s = ce^{rn}$. We find that $\tilde{p}_b$ grows at an exponential rate of $r = 0.1824$ for CMR embedding and $r = 0.1656$ for clique embedding as n increases. There is not a significant difference in the
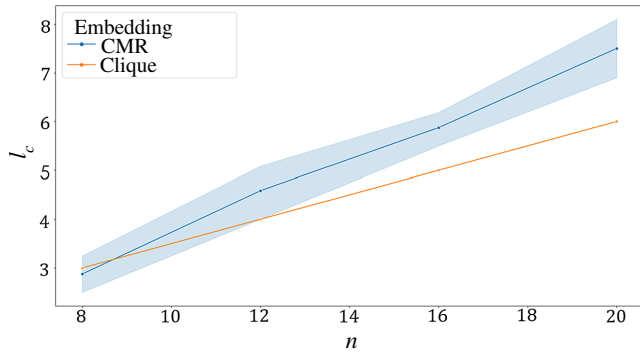
FIG. 6. The average chain length over all chains for a given embedding clique and CMR embedding as n increases.

$\tilde{p}_s$ performance between CMR and clique embedding, but clique embedding requires a fewer number of spins as n increases and shows a slight improvement in $\tilde{p}_b$. Therefore, we choose to use clique embedding for subsequent benchmarks.

2. Forward annealing time

According to the adiabatic theorem, forward annealing more slowly should increase the probability of the system remaining in the ground state and thus increase the probability of success. We vary the forward annealing time T from 1 to 999 μs, which is the broadest range accessible on the D-Wave 2000Q. As shown in the upper panel of Fig. 8, we observe statistically insignificant changes in the
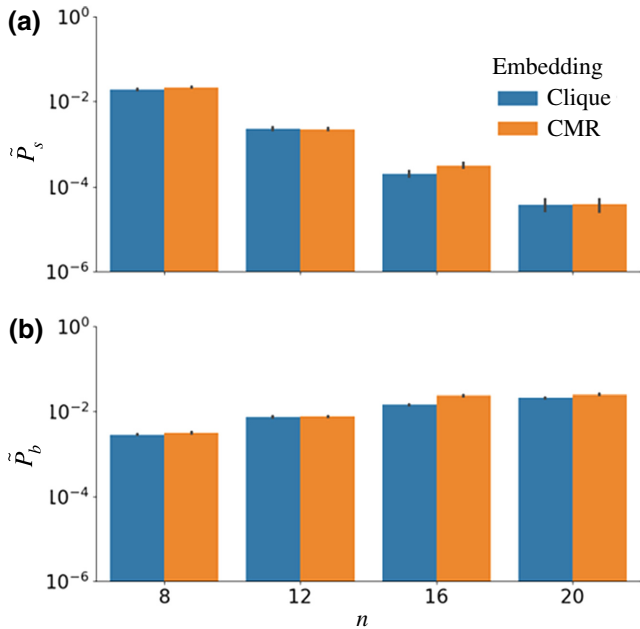

FIG. 7. The $\tilde{p}_s$ (a) and $\tilde{p}_b$ (b) on a log scale over 1000 samples for 1000 problems comparing CMR to clique embedding for parameter settings of $g = 0$ and $T = 100\ \mu$s.
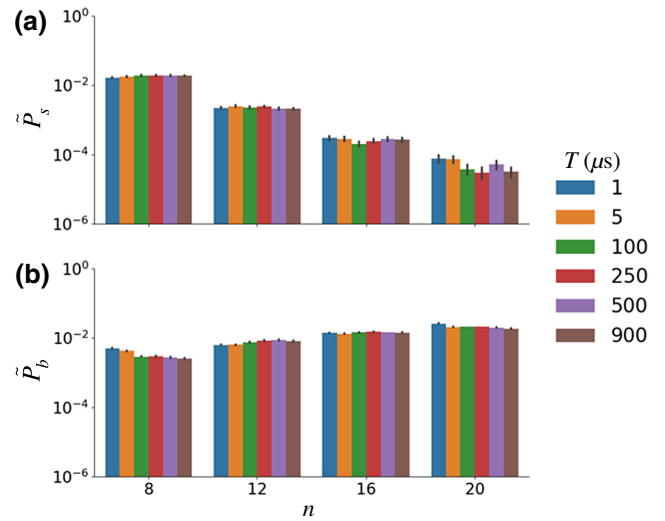

FIG. 8. The average $\tilde{p}_s$ (a) and $\tilde{p}_b$ (b) on a log scale over 1000 samples for 1000 problems at various annealing times for a parameter setting of $g = 0$ and clique embedding.

probability of success as the annealing time increases at each problem size. Fitting the average probability of success with respect to problem size for the annealing time $T = 100\ \mu$s yields a subexponential decay rate for $\tilde{p}_s$ given by $r = -0.528$ and a subexponential growth rate for $\tilde{p}_b$ given by $r = 0.1628$ as n increases. We do observe a statistically significant difference in the estimated probability of chain breaks $\tilde{p}_b$ with respect to the forward annealing time, as shown in the lower panel of Fig. 8. For $T = 100\ \mu$s, we recover a growth rate of $r = 0.1656$ for the probability of chain breaks with respect to problem size.

3. Spin reversal

As discussed in Sec. IV A 1, embedding maps a logical spin to many physical spins by creating strongly coupled chains. Coupling of these embedded spins via $J_{i,j}$ in Eq. (3) can lead to small biases that may be amplified by imperfections in the hardware. A spin reversal transform mitigates against bias errors by reversing the sign of a spin in the Ising Hamiltonian. This transform preserves the logical problem but reverses the bias error on the embedded spin chain. By randomly selecting a subset of spins to revise, we evaluate the influence of spin-reversal transform on the probability of success. We use g transforms when estimating the probability of success for a given problem instance, such that there are N_s/g samples per transform. We observed nominal improvements in Fig. 9 by using at least $g = 2$ with no advantage to using $g > 2$. For $g = 2$, we observe an subexponential decay rate of $r = -0.505$ for $\tilde{p}_s$ and a subexponential growth rate of $r = 0.146$ for $\tilde{p}_b$ as problem size increases.
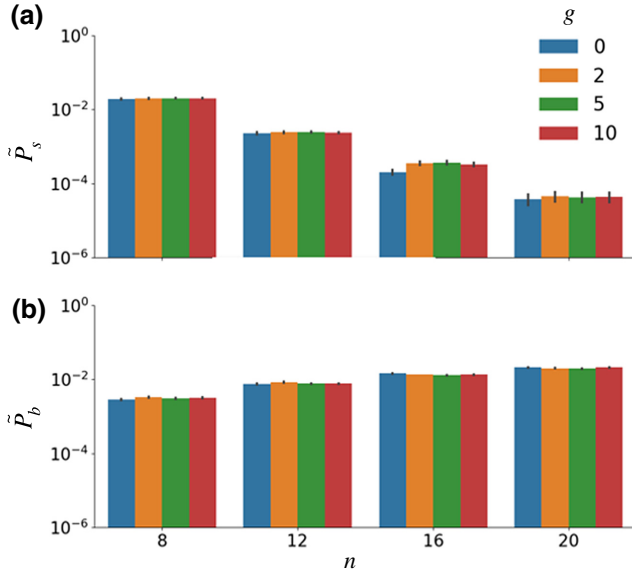
(a)

(b)

FIG. 9. The $\tilde{p}_s$ (a) and $\tilde{p}_b$ (b) on a log scale over $N_s = 1000$ samples for $N_p = 1000$ problems at $g = 0 \to 10$ for a parameter setting of $T = 100 \,\mu$s and clique embedding.

B. Benchmarking reverse annealing controls

From the reverse annealing controls listed in Sec. IV A 3, we design three experiments based on the e_i for the reverse annealing heuristic that include (i) starting in the known ground state e_0, (ii) starting in the known first excited state e_1, and (iii) starting in the lowest-energy state obtained from 1000 forward annealing samples e_f. We then sweep over various schedules to find the optimal s_p with a range of $[0.1, 0.9]$ and t_p with a range of $[15 \to 800]\,\mu$s. The t_r and t_q parameters are set to be constant and symmetric at $5\,\mu$s each. Thus, the total anneal time is $T' = t_r + t_p + t_q$, where t_p is the time parameter that we choose to analyze. For all experiments, we ran the reverse annealing iterative heuristic with 1000 samples for 100 of the random problems used in the forward annealing experiments. We estimate the probability of success for reverse annealing with respect to different choices for e_i, s_p, and t_p. We compare the combined heuristic of reverse annealing with forward annealing to forward annealing alone with $\tilde{p}_s$, $\tilde{p}_b$, as well as the frequency of finding energies in excited states to forward annealing alone [62].

By setting e_i to the ground state, we test for parameters s_p and t_p that decrease $\tilde{p}_s$ when the quantum annealer is fed the correct solution. For this experiment, $\tilde{p}_s$ can be thought of as the probability of staying in e_0

$$\tilde{p}_s(e_0 \to e_0) = p_f \tilde{p}_s, \tag{25}$$

$$p_f \tilde{p}_s = \frac{\sum_i^{N_p} \alpha_i}{N_p} \frac{\sum_i^{N_p} \sum_j^{N_s} \delta_{ij}}{N_s}, \tag{26}$$

where p_f is the probability that forward annealing found the ground state, $\alpha_i \in \{0, 1\}$ indicates whether forward annealing found the ground state for the ith problem prior to reverse annealing, and $\delta_j \in \{0, 1\}$ is a variable indicating whether the jth sample of the ith problem is measured to be the ground state with reverse annealing. By setting $e_i = e_1$, we test whether reverse annealing enhances the probability to populate the ground state. For these tests, $\tilde{p}_s$ estimates the probability of moving from an excited state to the ground state

$$\tilde{p}_s(e_e \to e_0) = (1 - p_f)\tilde{p}_s, \tag{27}$$

$$(1 - p_f)\tilde{p}_s = \frac{\sum_i^{N_p}(1 - \alpha_i)}{N_p} \frac{\sum_i^{N_p} \sum_j^{N_s} \delta_{ij}}{N_s}. \tag{28}$$

In addition to testing reverse annealing at $e_i = e_0$ and e_1, we test reverse annealing in combination with forward annealing for which $\tilde{p}_s$ estimates the cumulative probability of finding the correct solution state:

$$\tilde{p}_s(R) = \tilde{p}(e_0 \to e_0) + \tilde{p}(e_e \to e_0). \tag{29}$$

For these experiments, we find it useful to primarily analyze $\tilde{p}_s(R) - \tilde{p}(e_0 \to e_0) = \tilde{p}(e_e \to e_0)$ to determine if reverse annealing improves upon the $\tilde{p}_s$ of forward annealing.

The results from setting $e_i = e_0$ for each problem with a problem size of $n = 20$ where $m = 5$ and $w = 4$ are shown in Fig. 10. Because the computation begins in the correct solution state, this test measures the probability by which reverse annealing introduces errors into the correct solution. Ideally, $\tilde{p}_s$ will remain near unity for all s_p and t_p. We observe that reverse annealing causes the system to leave the ground state with $\tilde{p}_s$ reducing to the order of 10^{-5} by annealing back to at least $s = 0.6$ and increasing $t_p \geq 200\,\mu$s

The results from setting $e_i = e_1$ with a problem size of $n = 20$ where $m = 5$ and $w = 4$ for each problem are shown in Fig. 11. A maximal value of 4.8×10^{-4} for $\tilde{p}_s$ is found with parameters $s = 0.7$ and $t_p = 800\,\mu$s. This is a $\tilde{p}_s$ one order of magnitude higher than what is observed with forward annealing. This suggests that if e_i is very close to e_0, there may be some benefit to choosing reverse annealing over forward annealing.

When solving optimization problems for applications in practice, the ground state and excited state will be unknown. However, one approach is to use reverse annealing in addition to forward annealing by using the lowest energy state found with 1000 forward annealing samples e_f as e_i for another 1000 samples of reverse annealing. The next experiment tests whether reverse annealing used in combination with forward annealing increases $\tilde{p}_s$ with a problem size of $n = 20$ where $m = 5$ and $w = 4$. The

(a) $\tilde{P}_s$

s \ t_p	15.0	200.0	400.0	600.0	800.0
0.1	0.00012	0.00003	0.00006	0.00004	0.00003
0.2	0.00003	0.00003	0.00005	0.00002	0.00001
0.3	0.00009	0.00001	0.00007	0.00009	0.00005
0.4	0.00011	0.00007	0.00007	0.00008	0.00008
0.5	0.00018	0.00001	0.00002	0.00001	0.00004
0.6	0.0073	0.00017	0.00005	0.00005	0.00005
0.7	0.39	0.033	0.015	0.01	0.011
0.8	0.98	0.68	0.52	0.48	0.49
0.9	1	0.98	0.99	0.99	0.99

(b) $\tilde{P}_b$

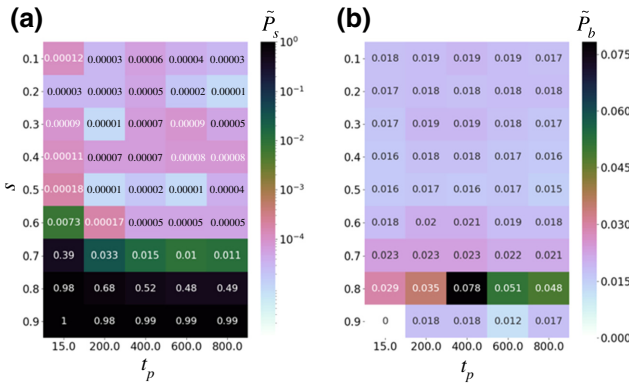| s \ t_p	15.0	200.0	400.0	600.0	800.0
0.1	0.018	0.019	0.019	0.019	0.017
0.2	0.017	0.018	0.018	0.018	0.018
0.3	0.017	0.019	0.019	0.018	0.017
0.4	0.016	0.018	0.018	0.017	0.016
0.5	0.016	0.017	0.016	0.017	0.015
0.6	0.018	0.02	0.021	0.019	0.018
0.7	0.023	0.023	0.023	0.022	0.021
0.8	0.029	0.035	0.078	0.051	0.048
0.9	0	0.018	0.018	0.012	0.017

FIG. 10. The $\tilde{p}_s$ (a) and $\tilde{p}_b$ (b) for reverse annealing where $e_i = e_0$ and as $s = [0.1 \rightarrow 0.9]$ and $t_p = [15 \rightarrow 800]$ μs for $n = 20$ with $m = 5$ assets and $w = 4$.

experimental results from setting $e_i = e_f$ are shown in Fig. 12. These tests are constructed to determine when combining reverse annealing with forward annealing can improve upon forward annealing. Therefore, we remove the six problems forward annealing provided an $e_i = e_0$ and thus $\tilde{p}_s$ for this experiment is given by $\tilde{p}_s(R) - p(e_0 \rightarrow e_0)$ in this analysis. Similar to the previous experiment in Fig. 11, the $\tilde{p}_s$ is at best of the order of 10^{-4} at parameters $s = 0.7$ and $t_p = 400$ μs, which is one order of magnitude greater than the forward annealing experiments.

In Fig. 12 we show a potential for reverse annealing to improve upon results found with forward annealing in $\tilde{p}_s$. Therefore, we take a set of 100 problems solved with reverse annealing and forward annealing and compare the $\tilde{p}_s$ of forward annealing (orange) alone to the $\tilde{p}_s$ of reverse annealing alone (blue) to the $\tilde{p}_s$ with a selection of either forward annealing or reverse annealing (green). If a problem forward annealing finds at least one ground
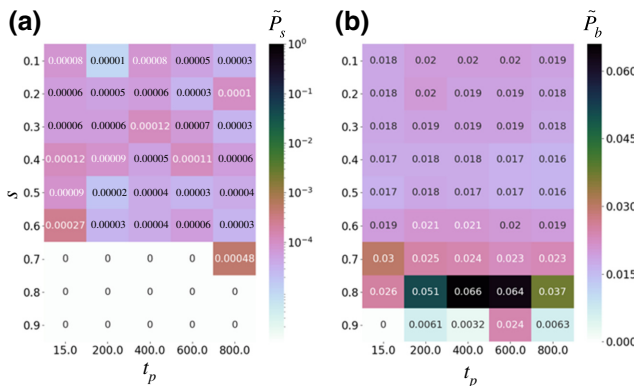
(a) $\tilde{P}_s$

s \ t_p	15.0	200.0	400.0	600.0	800.0
0.1	0.00008	0.00001	0.00008	0.00005	0.00003
0.2	0.00006	0.00005	0.00006	0.00003	0.0001
0.3	0.00006	0.00006	0.00012	0.00007	0.00003
0.4	0.00012	0.00009	0.00005	0.00011	0.00006
0.5	0.00009	0.00002	0.00004	0.00003	0.00004
0.6	0.00027	0.00003	0.00004	0.00006	0.00003
0.7	0	0	0	0	0.00048
0.8	0	0	0	0	0
0.9	0	0	0	0	0

(b) $\tilde{P}_b$

s \ t_p	15.0	200.0	400.0	600.0	800.0
0.1	0.018	0.02	0.02	0.02	0.019
0.2	0.018	0.02	0.019	0.019	0.018
0.3	0.018	0.019	0.019	0.019	0.018
0.4	0.017	0.018	0.018	0.017	0.016
0.5	0.017	0.018	0.017	0.017	0.016
0.6	0.019	0.021	0.021	0.02	0.019
0.7	0.03	0.025	0.024	0.023	0.023
0.8	0.026	0.051	0.066	0.064	0.037
0.9	0	0.0061	0.0032	0.024	0.0063

FIG. 11. The $\tilde{p}_s$ (a) and $\tilde{p}_b$ (b) for reverse annealing where $e_i = e_1$ for each problem, $s = [0.1 \rightarrow 0.9]$, and $t_p = [15 \rightarrow 800]$ μs for a problem size of 20 with $m = 5$ assets and $w = 4$ slices.
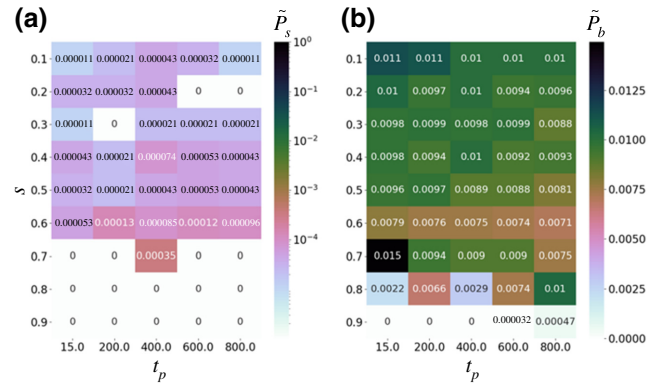
(a) $\tilde{P}_s$

s \ t_p	15.0	200.0	400.0	600.0	800.0
0.1	0.000011	0.000021	0.000043	0.000032	0.000011
0.2	0.000032	0.000032	0.000043	0	0
0.3	0.000011	0	0.000021	0.000021	0.000021
0.4	0.000043	0.000021	0.000074	0.000053	0.000043
0.5	0.000032	0.000021	0.000043	0.000053	0.000043
0.6	0.000065	0.00013	0.000085	0.00012	0.000096
0.7	0	0	0.00035	0	0
0.8	0	0	0	0	0
0.9	0	0	0	0	0

(b) $\tilde{P}_b$

s \ t_p	15.0	200.0	400.0	600.0	800.0
0.1	0.011	0.011	0.01	0.01	0.01
0.2	0.01	0.0097	0.01	0.0094	0.0096
0.3	0.0098	0.0099	0.0098	0.0099	0.0088
0.4	0.0098	0.0094	0.01	0.0092	0.0093
0.5	0.0096	0.0097	0.0089	0.0088	0.0081
0.6	0.0079	0.0076	0.0075	0.0074	0.0071
0.7	0.015	0.0094	0.009	0.009	0.0075
0.8	0.0022	0.0066	0.0029	0.0074	0.01
0.9	0	0	0	0.000032	0.00047

FIG. 12. The $\tilde{p}_s$ (a) and $\tilde{p}_b$ (b) for reverse annealing where $e_i = e_f$ for each problem, $s = [0.1 \rightarrow 0.9]$, and $t_p = [15 \rightarrow 800]$ μs for a problem size of 20 with $m = 5$ assets and $w = 4$ slices. The six problems where $e_f = e_0$ are excluded. Thus, $\tilde{p}_s = p(e_e \rightarrow e_0)$.

state, the forward annealing $\tilde{p}_s$ is plotted for that problem (six problems); otherwise, the reverse annealing $\tilde{p}_s$ is plotted (94 problems). The $\tilde{p}_s$ is measured over n in the range [8, 20]. The reverse annealing parameters are set to have an $e_i = e_f$, $s = 0.7$, and $t_p = 400$ μs. As shown in Fig. 13, we observe that, when taking the combination of best results from forward annealing and reverse annealing with $e_i = e_f$, we get a $\tilde{p}_s$ that improves by an order of magnitude over forward annealing alone for $n = [16, 20]$ with a subexponential decay at a rate of -0.309. Note that, although the blue reverse annealing trend looks to perform the best, this trend is artificially inflated because six of the problems have $e_i = e_0$, which has been demonstrated in Fig. 10 to yield a $\tilde{p}_s$ of the order of 10^{-2} at $s = 0.7$
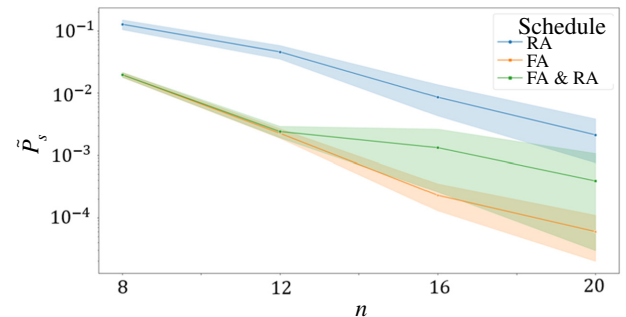

FIG. 13. The $\tilde{p}_s$ as a function of n over a set of 100 problems each with 1000 samples. We compare reverse annealing (blue) with $e_i = e_f$, $s = 0.7$, and $t_p = 400$ μs to forward annealing (orange) with clique embedding, $g = 0$, and an annealing time of 100 μs. We also compare the combination of forward annealing and reverse annealing where the $\tilde{p}_s$ is chosen by problem (green). In this green trend, the $\tilde{p}_s$ are calculated using the forward annealing $\tilde{p}_s^{(k)}$ for the six problems where forward annealing would have provided reverse annealing with an $e_i = e_0$ and the reverse annealing $\tilde{p}_s^{(k)}$ for the 94 problems where $e_i \neq e_0$.
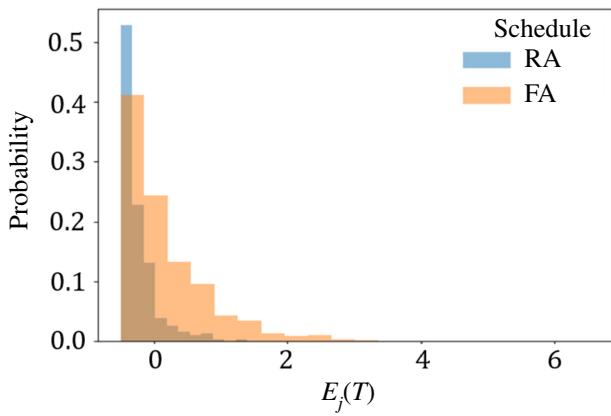
FIG. 14. A probability histogram (20 bins) comparing all energies found with forward annealing and reverse annealing from all 1000 samples for the 94 problems where $e_i \neq e_0$ for problems with $m = 5$ assets and $w = 4$.

and $t_p = 400$ μs. We next visualize a histogram, as seen in Fig. 14, of all energies recorded from 1000 samples returned for a set of 94 problems where forward annealing did not find e_0 with $n = 20$. We compare forward annealing to reverse annealing where $e_i = e_f$. We observe that even for problems where neither reverse annealing or forward annealing found e_0 reverse annealing still on average finds a lower-energy solution than forward annealing.

V. CONCLUSIONS

We benchmark quantum annealing using Markowitz portfolio selection to evaluate the effects of various controls on probability of success and chain breaks. We explore a variety of quantum annealing controls, including the embedding algorithm, the forward annealing time T, and the number of spin reversal transforms g. When comparing clique embedding against CMR embedding, we find little difference in the estimated probability of success $\tilde{p}_s$ as both techniques yield a subexponential decay for $\tilde{p}_s$ with exponents of -0.528 and -0.523, respectively. We do observe that CMR demonstrates a slightly higher probability of chain breaks $\tilde{p}_b$, and we consider this a sufficient justification to use the clique embedding for studying fully connected problems like the Markowitz Portfolio section problem.

When varying the forward annealing time $T \in [1, 999]$ μs, we find that $\tilde{p}_b$ is slightly higher in the range $T = [1, 5]$ μs, while increasing the annealing time further yields little to no improvement. For this reason, we chose to continue all future forward annealing experiments using $T = 100$ μs where the exponential decay rate in p_s is -0.528. When varying $g = [0, 10]$, we find small improvements in $\tilde{p}_s$ between $g = 0$ and $g = 2$ where the exponential decay rate becomes -0.505 without much change from increasing the value of g further, and there is no consistent difference in $\tilde{p}_b$.

We benchmark reverse annealing controls with respect to the parameters e_i, s, and t_p. We begin by observing the results in $\tilde{p}_s$ and $\tilde{p}_b$ at $n = 20$. We consistently observe that $\tilde{p}_b$ is the same order of magnitude as with the forward annealing experiments and $\tilde{p}_b$ is consistently highest for $s = 0.8$. By setting $e_i = e_0$, we observe that $\tilde{p}_s$ decreases exponentially as s increases. By setting $e_i = e_1$, we observe that reverse annealing has a $\tilde{p}_s$ an order of magnitude higher than forward annealing. From these results, we conclude that, when e_i is close to the ground state, reverse annealing provides some advantage over forward annealing. Because in general the ground state will not be known for a problem, we develop a heuristic that sets $e_i = e_f$ where we again observe $\tilde{p}_s$ to be an order of magnitude higher than using forward annealing alone.

We further evaluate $\tilde{p}_s$ as a function of n to compare reverse annealing with $e_i = e_f$, $s = 0.7$, and $t_p = 400$ μs to forward annealing with clique embedding, $T = 100$ μs, and $g = 0$ alone. In particular, we use the $\tilde{p}_s^{(k)}$ of forward annealing for the six problem instances in which $e_i = e_0$ and the $\tilde{p}_s^{(k)}$ of reverse annealing for the 94 problems where $e_i \neq e_0$. We continue to observe reverse annealing demonstrate an order of magnitude increase in p_s over forward annealing alone. Lastly, by creating a histogram that plots the lowest energies found across 1000 samples for the 94 problems where $e_i \neq e_0$, we find that reverse annealing ($e_i = e_f$) on average finds lower-energy solutions as compared to forward annealing.

In summary, the benchmarks presented here evaluate a variety of quantum annealing controls with respect to the baseline ground truth for portfolio selection. By comparing the observed influence of these controls on the performance of solution accuracy, we have developed insights into the best selections of controls for solving these problems with the highest accuracy that may help guide the future use of quantum annealing as a metaheuristics for optimization.

published form of this manuscript or allows others to do so for U.S. Government purposes.

APPENDIX: NUMBER OF COMBINATIONS CONSTRAINED TO THE BUDGET

Assuming that the optimal solution lies where the total value of assets bought equals the budget, the number of solutions that need to be checked is drastically reduced. If we have one asset, the only solution is buying the slice equal to 1. If we have two assets, the slice of the second asset is dictated by whichever slice is chosen from the first asset. If the number of slices chosen is w then we know that the slices correspond to $1, 1/2, 1/4, 1/8, \ldots, 1/2^{w-1}$. This gives a total of $2^{w-1} + 1$ (since we can also buy 0 for all slices), which is less than or equal to the budget. Mathematically, this can be expressed as

$$\text{number of solutions} = \sum_{a_1=1}^{2^{w-1}} \sum_{a_2=1}^{2^{w-1}-a_1} 1 = 2^{w-1} + 1. \quad (A1)$$

This is an equivalent problem to stating how many distinct terms are in the binomial $(a_1 + a_2)^{2^{w-1}}$.

Extending this to an arbitrary amount of assets (m), this equates to finding how many distinct terms are in the multinomial expansion $(a_1 + a_2 + \cdots + a_m)^{2^{w-1}}$, which can be found using the equation

$$\text{number of solutions} = \prod_{a=1}^{m-1} \frac{2^{w-1} + a}{a} = \frac{(2^{w-1} + m - 1)!}{(2^{w-1})! \, (m-1)!}. \quad (A2)$$

[1] P. M. Pardalos and J. B. Rosen, *Constrained Global Optimization: Algorithms and Applications* (Springer, Berlin, 1987), Vol. 268.

[2] J.-F. Tsai, J. G. Carlsson, D. Ge, Y.-C. Hu, and J. Shi, Optimization theory, methods, and applications in engineering 2013, Math. Probl. Eng. **2014** (2014).

[3] M. W. Krentel, in *Proceedings of the Eighteenth Annual ACM Symposium on Theory of Computing* (Association for Computing Machinery, New York, NY, United States, Berkely, CA US, 1986), p. 69.

[4] E. Farhi, J. Goldstone, S. Gutmann, and M. Sipser, Quantum computation by adiabatic evolution. arXiv preprint quant-ph/0001106 (2000).

[5] S. Morita and H. Nishimori, Mathematical foundation of quantum annealing, J. Math. Phys. **49,** 125210 (2008).

[6] H. N. Djidjev, G. Chapuis, G. Hahn, and G. Rizk, Efficient combinatorial optimization using quantum annealing. arXiv preprint arXiv:1801.08653 (2018).

[7] F. Neukart *et al.*, Traffic flow optimization using a quantum annealer, Frontiers in ICT **4,** 29 (2017).

[8] T. Stollenwerk *et al.*, Quantum annealing applied to de-conflicting optimal trajectories for air traffic management, IEEE Trans. Intell. Transp. Syst. **21,** 285 (2019).

[9] R. Martoňák, G. E. Santoro, and E. Tosatti, Quantum annealing of the traveling-salesman problem, Phys. Rev. E **70,** 057701 (2004).

[10] M. W. Johnson *et al.*, Quantum annealing with manufactured spins, Nature **473,** 194 (2011).

[11] T. Lanting *et al.*, Entanglement in a Quantum Annealing Processor, Phys. Rev. X **4,** 021041 (2014).

[12] S. H. W. van der Ploeg *et al.*, Adiabatic quantum computation with flux qubits, first experimental results, IEEE Trans. Appl. Supercond. **17,** 113 (2007).

[13] H. G. Katzgraber, F. Hamze, and R. S. Andrist, Glassy Chimeras Could be Blind to Quantum Speedup Designing Better Benchmarks for Quantum Annealing Machines, Phys. Rev. X **4,** 021008 (2014).

[14] J. King, S. Yarkoni, M. M. Nevisi, J. P. Hilton, and C. C. McGeoch, Benchmarking a quantum annealing processor with the time-to-target metric. arXiv preprint arXiv:1508.05087 (2015).

[15] Z. Zhu, A. J. Ochoa, S. Schnabel, F. Hamze, and H. G. Katzgraber, Best-case performance of quantum annealers on native spin-glass benchmarks: How chaos can affect success probabilities, Phys. Rev. A **93,** 012317 (2016).

[16] M. Jarret, S. P. Jordan, and B. Lackey, Adiabatic optimization versus diffusion monte carlo methods, Phys. Rev. A **94,** 042318 (2016).

[17] D. O'Malley, V. V. Vesselinov, B. S. Alexandrov, and L. B. Alexandrov, Nonnegative/binary matrix factorization with a d-wave quantum annealer, PloS one **13,** e0206653 (2018).

[18] T. Albash and D. A. Lidar, Demonstration of a Scaling Advantage for a Quantum Annealer Over Simulated Annealing, Phys. Rev. X **8,** 031016 (2018).

[19] A. Ajagekar, T. Humble, and F. You, Quantum computing based hybrid solution strategies for large-scale discrete-continuous optimization problems, Comput. Chem. Eng. **132,** 106630 (2020).

[20] G. E. Santoro, R. Martoňák, E. Tosatti, and R. Car, Theory of quantum annealing of an ising spin glass, Science **295,** 2427 (2002).

[21] T. F. Rønnow *et al.*, Defining and detecting quantum speedup, science **345,** 420 (2014).

[22] S. V. Isakov *et al.*, Understanding Quantum Tunneling through Quantum Monte Carlo Simulations, Phys. Rev. Lett. **117,** 180402 (2016).

[23] E. Crosson and A. W. Harrow, in *2016 IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS)* (IEEE Computer Society, New Brunswick, NJ, 2016). http://dx.doi.org/10.1109/FOCS.2016.81.

[24] L. Hormozi, E. W. Brown, G. Carleo, and M. Troyer, Nonstoquastic hamiltonians and quantum annealing of an ising spin glass, Phys. Rev. B **95,** 184416 (2017).

[25] D. Venturelli and A. Kondratyev, Reverse quantum annealing approach to portfolio optimization problems, Quantum Mach. Intell. **1,** 17 (2019).

[26] G. Quiroz, Robust quantum control for adiabatic quantum computation, Phys. Rev. A **99,** 062306 (2019).

[27] W. Vinci, T. Albash, G. Paz-Silva, I. Hen, and D. A. Lidar, Quantum annealing correction with minor embedding, Phys. Rev. A **92**, 042310 (2015).

[28] Z. Bian *et al.*, Mapping constrained optimization problems to quantum annealing with application to fault diagnosis, Front. ICT **3**, 14 (2016).

[29] J. Marshall, D. Venturelli, I. Hen, and E. G. Rieffel, Power of Pausing: Advancing Understanding of Thermalization in Experimental Quantum Annealers, Phys. Rev. Appl. **11**, 044083 (2019).

[30] K. L. Pudenz, T. Albash, and D. A. Lidar, Error-corrected quantum annealing with hundreds of qubits, Nat. Commun. **5**, 1 (2014).

[31] K. L. Pudenz, in *2016 IEEE high performance extreme computing conference (HPEC)* (IEEE, Waltham, MA US, 2016), p. 1.

[32] H. Markowitz, Portfolio selection, J. Finance. **7**, 77 (1952).

[33] N. Elsokkary, F. S. Khan, D. La Torre, T. S. Humble, and J. Gottlieb, Financial portfolio management using d-wave quantum optimizer: The case of abu dhabi securities exchange. Tech. Rep., (Oak Ridge National Lab.(ORNL), Oak Ridge, TN, USA, 2017).

[34] C. C. McGeoch and C. Wang, in *Proceedings of the ACM International Conference on Computing Frontiers* (Association for Computing Machinery, New York, NY, United States, Ischia, Italy, 2013), p. 1.

[35] D. S. Steiger, T. F. Rønnow, and M. Troyer, Heavy Tails in the Distribution of Time to Solution for Classical and Quantum Annealing, Phys. Rev. Lett. **115**, 230501 (2015).

[36] M. Marzec, in *Handbook of High-Frequency Trading and Modeling in Finance* (John Wiley & Sons, Inc., Hoboken, NJ US, 2016), p. 73.

[37] G. Rosenberg *et al.*, Solving the optimal trading trajectory problem using a quantum annealer, IEEE. J. Sel. Top. Signal Process. **10**, 1053 (2016).

[38] E. Pelofske, G. Hahn, and H. Djidjev, in *2019 IEEE International Conference on Rebooting Computing (ICRC)* (IEEE, San Francisco Bay Area, California, 2019), p. 1.

[39] A. D. King and C. C. McGeoch, Algorithm engineering for a quantum annealing platform. arXiv preprint arXiv:1410.2628 (2014).

[40] A. Lucas, Ising formulations of many np problems, Front. Phys. **2**, 5 (2014).

[41] Y. Yamashiro, M. Ohkuwa, H. Nishimori, and D. A. Lidar, Dynamics of reverse annealing for the fully connected p-spin model, Phys. Rev. A **100**, 052321 (2019).

[42] G. Passarelli, K.-W. Yip, D. A. Lidar, H. Nishimori, and P. Lucignano, Reverse quantum annealing of the p-spin model with relaxation, Phys. Rev. A **101**, 022331 (2020).

[43] A. Pearson, A. Mishra, I. Hen, and D. A. Lidar, Analog errors in quantum annealing: Doom and hope, NPJ Quantum Inf. **5**, 1 (2019).

[44] J. M. Martinis, S. Nam, J. Aumentado, K. Lang, and C. Urbina, Decoherence of a superconducting qubit due to bias noise, Phys. Rev. B **67**, 094510 (2003).

[45] S. Novikov *et al.*, in *2018 IEEE International Conference on Rebooting Computing (ICRC)* (IEEE, Tysons, Virginia, USA, 2018), p. 1.

[46] P. I. Bunyk *et al.*, Architectural considerations in the design of a superconducting quantum annealing processor, IEEE Trans. Appl. Supercond. **24**, 1 (2014).

[47] W. Tichy, Is quantum computing for real? an interview with catherine mcgeoch of d-wave systems, Ubiquity **2017**, 1 (2017).

[48] D-Wave Systems, I. D-wave system documentation. https://docs.dwavesys.com/docs/latest/doc˙qpu.html.

[49] V. Choi, Minor-embedding in adiabatic quantum computation: I. the parameter setting problem, Quantum Inf. Process **7**, 193 (2008).

[50] C. Klymko, B. D. Sullivan, and T. S. Humble, Adiabatic quantum programming: Minor embedding with hard faults, Quantum Inf. Process **13**, 709 (2014).

[51] J. Cai, W. G. Macready, and A. Roy, A practical heuristic for finding graph minors, arXiv preprint arXiv:1406.2741 (2014).

[52] T. Boothby, A. D. King, and A. Roy, Fast clique minor generation in chimera qubit connectivity graphs, Quantum Inf. Process **15**, 495 (2016).

[53] D-Wave Systems, I. Ocean tools library embedding documentation. https://docs.ocean.dwavesys.com/projects/system/en/latest/reference/embedding.html.

[54] J. Dziarmaga, Dynamics of a Quantum Phase Transition: Exact Solution of the Quantum Ising Model, Phys. Rev. Lett. **95**, 245701 (2005).

[55] A. M. Childs, E. Farhi, and J. Preskill, Robustness of adiabatic quantum computation, Phys. Rev. A **65**, 012322 (2001).

[56] T. Albash and D. A. Lidar, Decoherence in adiabatic quantum computation, Phys. Rev. A **91**, 062320 (2015).

[57] X. Huang, Mean–variance models for portfolio selection subject to experts estimations, Expert Syst. Appl. **39**, 5887 (2012).

[58] I. Martin, What is the expected return on the market?, Q. J. Econ. **132**, 367 (2017).

[59] D-Wave Systems and I. Ice, Dynamic ranges in h and j values. https://docs.dwavesys.com/docs/latest/c˙qpu˙1.html.

[60] J. E. Dorband, Extending the d-wave with support for higher precision coefficients, arXiv preprint arXiv:1807.05244 (2018).

[61] E. Grant and T. Humble, Qa controls for portfolio optimization. https://code.ornl.gov/egy/qa-controls-for-portfolio-optimization/. Data is available upon request.

[62] After completing the majority of experiments on the D-Wave processor DW_2000Q_2_1, the remaining experiments are performed on the D-Wave processor DW_2000Q_5. This includes the parametric tests of reverse annealing with respect to s and t_p. Prior to testing, we confirm computational consistency between the results generated using the first device and those using the second. We evaluate differences in $\tilde{p}_s$ and a standard deviation between the processors by comparing a previous reverse annealing experiment on the DW_2000Q_2_1 to the same experiment on the DW_2000Q_5. We find that the same $\tilde{p}_s$ using both devices and a standard deviation that is within 10^{-5} of the measurements on the previous D-Wave processor.

[63] E. Grant, T.S. Humble, and B. Stump, Data for: "Benchmarking quantum annealing controls with portfolio optimization," http://energy.gov/downloads/doe-public-access-plan (2020).