Quantum Machine Intelligence (2019) 1:17–30
https://doi.org/10.1007/s42484-019-00001-w

RESEARCH ARTICLE

Reverse quantum annealing approach to portfolio optimization problems

Davide Venturelli[1] · **Alexei Kondratyev**[2]

Received: 22 November 2018 / Accepted: 28 February 2019 / Published online: 26 April 2019

Abstract

We investigate a hybrid quantum-classical solution method to the mean-variance portfolio optimization problems. Starting from real financial data statistics and following the principles of the Modern Portfolio Theory, we generate parametrized samples of portfolio optimization problems that can be related to quadratic binary optimization forms programmable in the analog D-Wave Quantum Annealer 2000Q^{TM}. The instances are also solvable by an industry-established genetic algorithm approach, which we use as a classical benchmark. We investigate several options to run the quantum computation optimally, ultimately discovering that the best results in terms of expected time-to-solution as a function of number of variables for the hardest instances set are obtained by seeding the quantum annealer with a solution candidate found by a greedy local search and then performing a reverse annealing protocol. The optimized reverse annealing protocol is found to be more than 100 times faster than the corresponding forward quantum annealing on average.

Keywords Portfolio optimization · Quadratic unconstrained binary · Optimization · Quantum annealing · Genetic algorithm · Reverse quantum annealing

1 Introduction

One reason behind the current massive investment in quantum computing is that this computational paradigm has access to resources and techniques that are able to circumvent several bottlenecks of state-of-the-art digital algorithms. Much research is today devoted to understanding how to exploit this power to deliver high quality solutions to discrete optimization problems at a fraction of the time required using the best classical methods on high-end hardware.

Quantum annealers are special-purpose machines inspired by the adiabatic quantum computing paradigm. These machines are manufactured by D-Wave Systems and appeared on the market in 2011 and, while being limited in programmability with respect to other experimental devices under testing by other companies, are still the only available quantum devices that feature a sufficient amount of quantum memory (qubits) to be applied to non-trivial problems at the time of writing. For this reason, they are subject to extensive empirical investigation by several groups around the world, not only for scientific and research purposes (Rønnow et al. 2014; Denchev et al. 2016) but also for performance evaluation on structured real-world optimization challenges (Venturelli et al. 2015b; Stollenwerk et al. 2019; Mott et al. 2017). It is expected that as quantum computing devices mature, they'd become available to enterprises for use through dedicated cloud services (Johnson et al. 2016).

Optimization problems comprise a large class of hard to solve financial problems, not to mention the fact that many supervised and reinforcement learning tools used in finance are trained via solving optimization problems (minimization of a cost function, maximization of reward). Several proposed applications of the D-Wave machine to portfolio optimization (Rosenberg et al. 2016; Marzec 2016), dealt with portfolios that were too small to evaluate the scaling of the chosen approach with the problem size. In this paper, we go beyond these early approaches and provide an analysis on sufficient data points to infer a scaling and

✉ Davide Venturelli
dventurelli@usra.edu

[1] USRA Research Institute for Advanced Computer Science, Mountain View, CA 94035, USA

[2] Data Analytics Group, Financial Markets, Standard Chartered Bank, London, EC2V 5DD, UK

measure a limited speedup with respect a state-of-the-art numerical method based on genetic algorithms. Moreover, we extensively tune the D-Wave machine's runs using the hybrid method of reverse annealing which is available only on the most advanced models to date, and whose usage has never been published before on structured optimization problems.

Portfolio optimization The optimal portfolio construction problem is one of the most extensively studied problems in quantitative finance. The Modern Portfolio Theory (MPT) as formulated by Markowitz (1952) has laid the foundation for highly influential mean-variance portfolio optimization approach. According to the MPT, a typical portfolio optimization problem can be formulated as follows. Let N be the number of assets, μ_i be the expected return of asset i, σ_{ij} be the covariance between the returns for assets i and j, and R be the target portfolio return. Then, the decision variables are the weights w_i, i.e., the investment associated with asset i ($w_i \in \mathbb{R}$). The standard Markowitz mean-variance approach consists in the constrained, quadratic optimization problem:

$$\min \sum_{i=1}^{N} \sum_{j=1}^{N} w_i w_j \sigma_{ij}; \ \sum_{i=1}^{N} w_i = 1; \ \sum_{i=1}^{N} w_i \mu_i = R. \quad (1)$$

Quadratic problems of the form (1) are efficiently solvable by standard computational means (e.g., quadratic programming with linear constraints (Fernando 2000)) if the covariance matrix is positive definite. Interestingly, the problem can also be cast into an unconstrained quadratic optimization problem (QUBO) which is a suitable formulation for quantum annealers. This observation spurred in the last few years a couple of proof-of-principle papers that were performing runs on the first-generation D-Wave machines solving discrete portfolio optimization problems (Rosenberg et al. 2016; Marzec 2016). The discrete portfolio optimization problem is known to be NP-complete (Kellerer et al. 2000) and is much harder to solve than continuous mean-variance portfolio optimization problem, which can be solved efficiently by interior point algorithms for convex optimization.

In this paper, we propose an original approach to portfolio optimization that extends the mean-variance model to a general dependence structure and allows portfolio managers to express discretionary views on relative attractiveness of different assets and their combinations. The manuscript is structured as follows: in Section 2, the formulation of the portfolio optimization problem is constructively described starting from real market data. Section 3 describes the hybrid quantum annealing solver and its setup. Section 4 is devoted to the experimental results, including results obtained with the classical genetic algorithm (GA) benchmark (following Kondratyev and Giorgidze (2017)). In Section 5, we conclude with a discussion and considerations for future work.

2 Portfolio optimization beyond Markowitz

The problem we are trying to solve here is construction of the optimal portfolio from the assets with known characteristics such as asset returns, volatilities, and pairwise correlations. A typical portfolio optimization problem consists of selecting M assets from the universe of N investible assets. These M assets should ideally be the best possible choice according to some criteria. The scenario we target is a Fund of Funds portfolio manager who is facing a task of selecting the best funds that follow particular trading strategies (e.g., EM Macro Funds) in order to maximize the risk-adjusted returns according to some model.

A realistic case occurs when the assets are selected with equal preference weights (a classical example of exposure to equally weighted assets is the CSO[1] portfolio). Should we want to generalize the portfolio with larger allocation to a given asset, we could allow for multiples of the reference weight by cloning an asset and treating it as a new one.

The task of encoding the relationship among the choices of M funds (*without replacement*) from the universe of N funds can then be formulated as a quadratic form:

$$O(\mathbf{q}) = \sum_{i=1}^{N} a_i q_i + \sum_{i=1}^{N} \sum_{j=i+1}^{N} b_{ij} q_i q_j, \quad (2)$$

where $q_i = 1$ means that asset i is selected and $q_i = 0$ means that asset i is not selected. Coefficients a_i reflect asset attractiveness on a standalone basis and can be derived from the individual asset's expected risk-adjusted returns. Coefficients b_{ij} reflect pairwise diversification penalties (positive values) and rewards (negative values). These coefficients can be derived from the pairwise correlations.

The minimization of the QUBO objective function given by expression (2) should optimize the risk-adjusted returns by the use of the metrics of Sharpe ratio (Sharpe 1966) for the a_i parameters (measuring expected excess asset return in the asset volatility units) and correlation between assets (as a measure of diversification) for the b_{ij} coefficients.

The Sharpe ratio is calculated as $(r - r_0)/\sigma$ where r is the asset's expected annualized return, r_0 is the applicable risk-free interest rate and σ is asset's volatility (annualized

[1] Collateralized Synthetic Obligation (CSO) is a type of Collateralized Debt Obligation (CDO) where credit exposure to the reference names is provided in synthetic form via single name Credit Default Swaps (CDS). A typical CSO references between 100 and 125 equally weighted names.

standard deviation of the asset's log-returns). The higher fund's Sharpe ratio, the better fund's returns have been relative to the risk it has taken on. Volatility can be estimated as historical annualized standard deviation of the Net Asset Value (NAV) per share log-returns. Expected return can be either estimated as historical return on fund investment or derived independently by the analyst/portfolio manager taking into account different considerations about the future fund performance. The asset correlation matrix ρ_{ij} is constructed from the time series of log-returns (Hull and Basu 2016).

QUBO instances Instead of using the raw real numbers obtained from financial data for the QUBO coefficients, we opt to coarse-grain the individual funds Sharpe ratios and their mutual correlation values down to integer values by grouping intervals in buckets according to Table 1. By using bucketed values, we define a scorecard, which is loosely based on the past fund performances but can be easily adjusted by portfolio manager according to his/her personal views and any new information not yet reflected in the funds reports. Indeed, in real-world scenarios, optimization needs to be performed with respect to the discretionary views about the future held by portfolio manager, not necessarily with respect to automatically fetched market data. Moreover, if we assume that funds report their NAV per share on a monthly basis and we have compa rable funds data for the previous year, then, we only have 12 NAV observations in our time series from which we want to construct a correlation matrix for N assets ($N \gg 12$). It is likely that the resulting correlation matrix will not be positive definite due to the number of observations being much smaller than the number of assets, ruling out the traditional solvers of mean-variance portfolio optimization problems which require a positive definite correlation matrix.

The instance set used for our case study is obtained by simulating asset values with the help of correlated geometric Brownian motion (GBM) processes with uniform constant asset correlation ρ, drift μ and log-normal asset volatility σ. The specific values of these parameters were derived from a wide range of fund industry researches (see, e.g., Darolles and Gourieroux 2010 for the Sharpe ratio distributions) and, therefore, can be viewed as representative for the industry. The simulation time horizon was chosen to be 1 year and the time step was set at 1 month—this setup corresponds to the situation where a Fund of Funds portfolio manager is dealing with the short time series of comparable monthly fund performance reports. Detailed description of the simulation scheme can be found in Appendix 1. Every simulated (or, "realized") portfolio scenario consists of 12 monthly returns for each asset. From these returns, we calculated total realized return and realized volatility for each asset (which, obviously, differ from the their expected values μ and σ) and for the portfolio as a whole. We also calculated realized pairwise correlations between all assets according to the input uniform correlation ρ. Finally, we calculated individual assets and portfolio Sharpe ratios. For instance, with $\rho = 0.1$, $\mu = 0.075$, $\sigma = 0.15$, and the constant risk-free interest rate set at $r_0 = 0.015$, the expected Sharpe ratio for each asset in a portfolio is 0.4. The expected Sharpe ratio for the portfolio of N assets is significantly larger due to the diversification and low correlation between the assets, e.g., for a 48-asset portfolio, we would expect Sharpe ratio values from 0.5 (25th percentile) to 2.1 (75th percentile) with a mean around 1.4.

Constraint on the number of selected assets In QUBOs, the standard way to deal with the cardinality constraint (i.e., selection of only M assets) would be to add a term $O_{\text{penalty}}(\mathbf{q})$ to the objective function given by expression (2)

Table 1 Specification of the sample QUBO coefficients from NAV time series market data

SR bucket (asset i) Equally spaced buckets, from worst to best	Coefficient a_i Sample Mapping Scheme	Correlation bucket	Coefficient b_{ij} Sample Mapping Scheme
1st	15	$-1.00 \le \rho_{ij} < -0.25$	− 5
2nd	12	$-0.25 \le \rho_{ij} < -0.15$	− 3
3rd	9	$-0.15 \le \rho_{ij} < -0.05$	− 1
4th	6	$-0.05 \le \rho_{ij} < 0.05$	0
5th	3	$0.05 \le \rho_{ij} < 0.15$	1
6th	0	$0.15 \le \rho_{ij} < 0.25$	3
7th	− 3	$0.25 \le \rho_{ij} \le 1.00$	5
8th	− 6		
9th	− 9		
10th	− 12		
11th	− 15		

such that the unsatisfying selections would be penalized by a large value $P \gg 1$, which would force the global minimum to be such that $\sum_{i=1}^{N} q_i = M$:

$$O_{\text{penalty}}(\mathbf{q}) = P\left(M - \sum_{i=1}^{N} q_i\right)^2. \tag{3}$$

Unfortunately, the introduction of such a large energy scale P is typically associated with precision issues connected to the analog nature of the machine and the fact that there is a physical maximum to the energy that can be controllably programmed on local elements of the D-Wave chip (Johnson et al. 2011). However, several hybrid quantum-classical strategies can be put in place to overcome the limitation. For instance, we observe that shifting artificially the Sharpe ratio values by a constant amount $\pm\Delta$ (and adding buckets according to the prescription chosen, e.g., Table 1) will essentially amount to forcing the ground state solution of the unconstrained problem to have more or less desired number of assets selected. Hence, while not solving the same problem, we could imagine a solver of a similarly constrained problem that will enclose the quantum annealing runs in a classical loop which checks for the number of selected assets $m(\Delta)$ in the best found solution with $\Delta = 0$, then increases or decreases the individual desirability of the assets according to whether m is larger or smaller than M and runs again until $m(\Delta) = M$ for $\Delta = \Delta^\star$. This sort of hybridization scheme is not uncommon for quantum-assisted solvers (Tran et al. 2016; Venturelli et al. 2015b) and the number of expected rounds of runs should scale as $\propto \log_2(\Delta^\star)$ as per a binary search introducing a prefactor over the time-to-solution complexity which should stay manageable. Other hybrid approaches could also be put forward to deal with the constraint such as fixing some asset selections in pre-processing via sample persistence (Karimi and Rosenberg 2017).

Per above arguments, in our benchmark case study in this work, we will focus on running unconstrained problems, setting $\Delta = 0$. By design, the instance ensambles are such that the average number of assets in the ground state should be around $N/2$ (see Table 2) corresponding to the largest combinatorial space by exhaustive enumeration of the possible solutions, so our tests will arguably be targeting among the most challenging instances of our parametrized model.

3 Quantum annealing hybrid solver

A QUBO problem can be easily translated into a corresponding Ising problem of N variables s_i, $i = 1, \ldots, N$ with $s_i \in \{+1, -1\}$ given by the following expression:

$$O_{\text{Ising}}(\mathbf{s}) = \sum_{i=1}^{N} h_i s_i + \sum_{i=1}^{N} \sum_{j=i+1}^{N} J_{ij} s_i s_j. \tag{4}$$

Ising and QUBO models are related through the transformation $s_i = 2q_i - 1$; hence, the relationship with Eq. (2) is $J_{ij} = \frac{1}{4} b_{ij}$ and $h_i = (\frac{1}{2} a_i + \sum_j b_{ij})$, disregarding an unimportant constant offset.

Chimera-embedded Ising representation In the standard programming practices of the analog D-Wave Quantum Annealer 2000Q$^{\text{TM}}$ (DW2000Q), each spin-variable s_i should be ideally assigned to a specific chip element, a superconducting flux qubit, modeled by a quantum two level system that could represent the quantum Hamiltonian $\mathcal{H}_{\text{local}} = \sum_i h_i \sigma_i^z$, with σ_i^z being the usual Pauli matrix representation for an Ising quantum spin-$\frac{1}{2}$. Each qubit supports the programming of the h_i terms. Instead, J_{ij} parameters can then be implemented energetically through inductive elements, meant to represent $\mathcal{H}_{\text{couplers}} =$

Table 2 Benchmark Instance Set characterization. For the number of assets in the optimal portfolio, we report the median and the maximum and minimum in parentheses, over 30 instances. The parameter search space for optimization of quantum annealing is reported in the last three columns, in the format min–max (step)

Problem size	% of instances solved		Number of assets in the optimal	Parameters search space		
	Greedy	Forward				
N	Search	Annealing	portfolio	J_F	s_p	ρ (μs)
24	80%	100%	10 (−3, +4)	3–7 (0.5)	–	–
30	90%	100%	12 (−5, +7)	3–7 (0.5)	–	–
36	93%	97%	13 (−7, +7)	3–7 (0.5)	–	–
42	70%	90%	16 (−7, +6)	5–8 (0.5)	0.32–0.5 (0.02)	1–15 (7)
48	47%	87%	17 (−6, +5)	5–8 (0.5)	0.32–0.5 (0.02)	1–15 (7)
54	60%	60%	19 (−7, +12)	5–8 (0.5)	0.32–0.5 (0.02)	1–15 (7)
60	57%	30%	23 (−13, +15)	6–9 (0.5)	0.32–0.5 (0.02)	1–15 (7)

$\sum_{ij} J_{ij}\sigma_i^z\sigma_j^z$, if and only if the required circuitry exists between qubits i and j. Couplers cannot be manufactured connecting qubits too far apart in the spatial layout of the processor due to engineering considerations. In other words, $J_{ij} = 0$ unless $(i, j) \in \chi_{16}$, where $\chi_{16} = (V, E)$ is the *Chimera* graph of DW2000Q, featuring 16×16 unit cells of 8 qubits each, in the ideal case, for a total of 2048 qubits.[2]

To get around this restriction, we employ the minor-embedding compilation technique for fully connected graphs (Boothby et al. 2016; Venturelli et al. 2015a). By means of this procedure, we obtain another Ising form, where qubits are arranged in ordered 1D chains interlaced on the Chimera graph:

$$\mathcal{H}_{\chi\text{-Ising}} = -\sum_{i=1}^{N} |J_F| \left[\sum_{c=1}^{N_c-1} \boldsymbol{\sigma}^{\mathbf{z}}_{\mathbf{ic}} \sigma^z_{i(c+1)} \right] \tag{5}$$

$$+ \sum_{i=1}^{N} \frac{h_i}{N_c} \left[\sum_{c=1}^{N_c} \boldsymbol{\sigma}^{\mathbf{z}}_{\mathbf{ic}} \right]$$

$$+ \sum_{i,j=1}^{N} J_{ij} \sum_{c_i,c_j=1}^{N_c} \delta^{\chi}_{ij}(c_i, c_j) \boldsymbol{\sigma}^z_{\mathbf{ic_i}} \boldsymbol{\sigma}^{\mathbf{z}}_{\mathbf{jc_j}}. \tag{6}$$

In Eq. (5), we explicitly isolate the encoding of the *logical* quantum variable: the classical binary variable s_i is associated with $N_c = (N/4 + 1)$ Ising spins σ^z_{ic}, ferromagnetically coupled directly by strength J_F, forming an ordered 1D chain subgraph of χ_{16}. The value of J_F should be strong enough to correlate the value of the magnetization of each individual spin if measured in the computational basis ($\langle \boldsymbol{\sigma}^{\mathbf{z}}_{\mathbf{ic}} \rangle = \langle \boldsymbol{\sigma}^{\mathbf{z}}_{\mathbf{ic'}} \rangle$). In the term (6), we encode the objective function (4) through our extended set of variables: the local field h_i is evenly distributed across all qubits belonging to the logical chain i, and each couplers J_{ij} is active only between one specific pair of qubits $\boldsymbol{\sigma}^{\mathbf{z}}_{\mathbf{ic}^\star_\mathbf{i}}$, $\boldsymbol{\sigma}^{\mathbf{z}}_{\mathbf{jc}^\star_\mathbf{j}}$ which is specified by the adjacency check function $\delta^{\chi}_{ij}(c_i, c_j)$ which assumes unit value only if $(c_i = c_i^\star)$ and $(c_j = c_j^\star)$ and is zero otherwise.[3]

Forward quantum annealing The quantum annealing protocol, inspired by the adiabatic principle of quantum mechanics, dictates to drive the system from an initial (easy to prepare) ground state of a quantum Hamiltonian $\mathcal{H}_{\text{initial}}$ to the unknown low energy subspace of states of the problem Hamiltonian (6), ideally to the lowest energy states corresponding to the global minima of the objective function (4). This *forward* quantum annealing procedure on DW2000Q can be ideally described as attempting to drive the evolution of the following time-dependent Hamiltonian:

$$\mathcal{H}_{\text{QA}}(t) = A[t] \sum_{i=1}^{N} \left(\sum_{c=1}^{N_c} \boldsymbol{\sigma}^{\mathbf{x}}_{\mathbf{ic}} \right) + B[t] \mathcal{H}_{\chi\text{-Ising}}, \tag{7}$$

where the term multiplying $A[t]$ is a Hamiltonian describing an independent collection of local transverse fields for each spin of the system (Johnson et al. 2011). Figure 1a shows how $A[t]$ and $B[t]$ are varying over the scale of total *annealing time* τ on DW2000Q in units of the typical energy scale of the nominal temperature measured on the chip, 12 milliKelvin. At the end of the annealing run, $A[t] = 0$ and the system is projected on the computational basis by a measurement of each qubit magnetization. The duration of the anneal, τ, is a free parameter; hence, it is often useful to define the fractional completion of the annealing schedule, $s = t/\tau$.

We note that the real time-dependent evolution of the spin-system programmed in the DW2000Q chip is only loosely modeled correctly by the Schrödinger evolution of $\mathcal{H}_{\text{QA}}$, mostly because of the dominance of open-system dynamics effects (Job and Lidar 2018), the existence of unknown static and dynamic misspecification on the programmed parameters h_i and J_{ij}, and cross-talk couplings (King and McGeoch 2014). Recent research attempts to model accurately the forward annealing dynamics of DW2000Q and earlier models of D-Wave machines by means of ab initio reasoning have been moderately successful on carefully crafted problems (Boixo et al. 2016). However, for the purpose of benchmarking, the details of the evolution are not important: once specified $\mathcal{H}_{\chi-\text{Ising}}$, a single machine run could essentially be treated as a black-box solver dependent on the parameters J_F, and τ, returning a bitstring at every run.

Reverse quantum annealing Figure 1b illustrates the quantum annealing protocol when the DW2000Q is set to operate as a reverse annealer. The system is initialized with $B[t] = \max B$ and $A[t] = 0$, with spins set to a classical bit-string $\mathcal{S}$. The evolution then undergoes an inverse schedule illustrated in Fig. 1a up to a point where the Hamiltonian time-dependence is temporarily paused. With reference to the Hamiltonian evolution in Eq. 7, the transverse field evo-

[2] Note that the graph is not ideal, there is a set of 17 qubits that have not been calibrated successfully and are unoperable (see Fig. 4 in Appendix 2).

[3] In the actual embedding employed, it might happen that some pairs of logical variables i, j could have two pairs that can be coupled, instead of one. In that case, we activate both couplings at a strength $J_{ij}/2$ to preserve the classical value of the objective function.

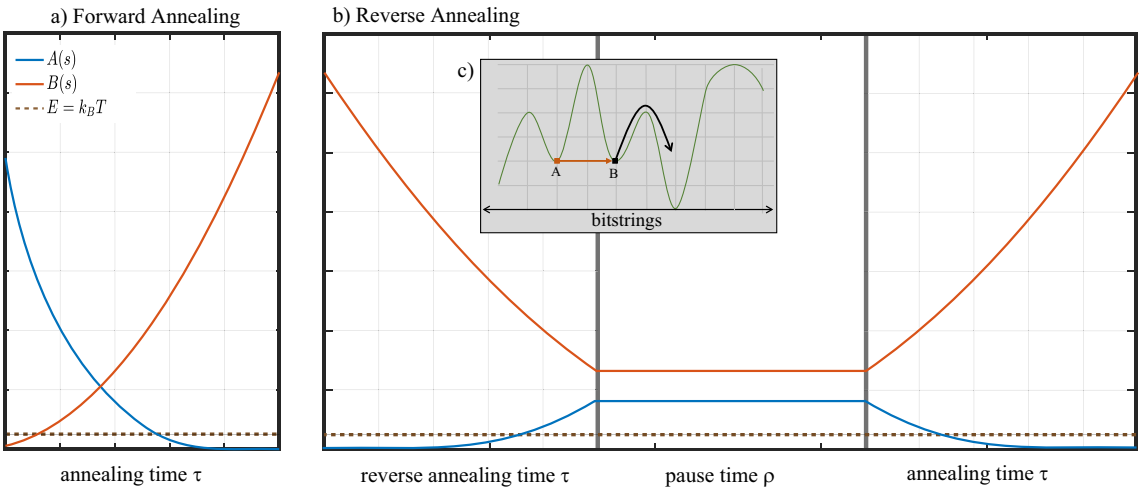

Fig. 1 Quantum annealing schedules on DW2000Q used in our investigations, with reference to Eq. (7). **a** Forward quantum annealing schedule. **b** Reverse quantum annealing schedule where the three steps of the protocol are all set to the same duration of $\tau = \rho$. **c** Pictorial view of what should be facilitated by reverse quantum annealing: when the initial bitstring is a local minimum (A), the reverse protocol should allow tunneling between valleys (A→B) while the pause should facilitate gradient descent as well as allow for local exploration through thermal hopping (escape from B towards the global minimum)

lution that we program[4] for this protocol is the following three-phase function (identical equations for $B^R[t]$, $B^p[t]$, $B^F[t]$ in terms of $B[t]$):

$$\begin{aligned} A^R[t] &= A[\tau + (s_p - 1)t] && \text{(Reverse Annealing)} \\ & && t \in [0, \tau] \\ A^p[t] &= A[s_p \tau] && \text{(Annealing Pause)} \\ & && t \in [\tau, \tau + \rho] \\ A^F[t] &= A[(1 - s_p)(t - \rho) - (1 - 2s_p)\tau] && \text{(Forward Annealing)} \\ & && t \in [\tau + \rho, 2\tau + \rho] \end{aligned} \tag{8}$$

where ρ is the duration of the pause and $s_p \in [0, 1]$ indicates the location of the forward schedule where the pause is implemented. The total duration of the selected reverse anneal protocol is $2\tau + \rho$ as opposed to τ for the forward anneal.

While the theory of reverse annealing is just starting to be investigated (Marshall et al. 2018; Ohkuwa et al. 2018; Kechedzhi et al. 2018), the physics rationale of reverse annealing is to be found in the oversimplified idea that, if the system is initialized in a state $\mathcal{S}$ corresponding to a local minimum of the objective function (4), the interplay of quantum and thermal fluctuations might help the state *tunnel* out of the energy trap during the reverse annealing. The annealing pause allows the system to thermalize (by means of dissipative quantum interaction with the bath) to a Gibbs state corresponding to the equilibrium state of the quantum Hamiltonian at s_p, and ultimately the final forward annealing and measurement allows the sampling of a neighborhood of a new minimum (see Fig. 1c for a pictorial view of the concept, and see Ref. Chancellor (2017) for a generic discussion on hybridizing local searches and initialization with quantum annealing).

The quality of the initial state $\mathcal{S}$ is likely to influence dramatically the reverse annealing process. For our experiments, we use a classical greedy algorithm to set $\mathcal{S}$, as described in the next section.

4 Experimental results

We aim to solve representative portfolio instances at the limit of programmability of the most advanced DW2000Q device, and to benchmark the results by means of a competitive heuristics running on classical computers. Note that due to the embedding overhead on DW2000Q, we can embed a maximum of 64 logical binary variables on a fully connected graph, which means that the largest search space for our benchmarks is around $60!/(30!)^2 \simeq$ hundred thousand trillion if $M = N/2$. Our instance set consists of

[4]Many options are possible, since the duration of the three phases can be chosen arbitrarily within limited but wide ranges.

30 randomly generated instances for each of the following number of assets $N = \{24, 30, 36, 42, 48, 54, 60\}$. In Table 2, we show the expected number of assets in the optimal portfolio of our unconstrained problem sets, computed by using all methods in this study, showing that the search space is non-trivial for all tested cases. Since exhaustive search is out of question for the largest problems, we provide two algorithms below that will be used to have a reference benchmark of the DW2000Q results.

A common metric to benchmark performance of non-deterministic iterative heuristics against quantum annealing is Time-to-Solution (TTS). The TTS is defined as the expected number of independent runs of the method in order to find the ground state with probability (confidence level) α:

$$\mathrm{TTS} = t_{\mathrm{run}} \frac{\ln(1-\alpha)}{\ln(1-p)}, \tag{9}$$

where t_{run} is the running time elapsed for a single run (either τ for forward or $2\tau + \rho$ for reverse), and p is the probability of finding the optimum of the objective function in that single shot (Rønnow et al. 2014; Mandra and Katzgraber 2018).

Greedy search The simplest reference benchmark to consider is the portfolio obtained with the best M assets, neglecting correlations ($b_{ij} = 0$).[5] The second simplest approach is to consider an iterative greedy search where correlations are taken into account progressively starting from the selection/exclusion of the most desirable/undesirable assets. This is implemented for instance by the routine shown in the pseudocode (Algorithm 1) below.[6]

Note that this greedy search is the procedure we use to set the initial state $\mathcal{S}$ for reverse annealing. Table 2 displays results obtained by applying the greedy search algorithm to the unconstrained problem Eq. (2) for the instance set, as a reference point, highlighting the percentage of instances solved with the greedy search heuristic as a function of problem size.

[5]We note that optimal portfolios constructed through minimization of objective function (2) and the number of asset constraints with QUBO coefficients given by Table 1 have typically better Sharpe ratios than alternative portfolios constructed from the individually best assets where the a_i coefficients have not been coarse-grained in buckets.

[6]This algorithm is inspired by the routine provided by D-Wave Systems to decode the binary value of a set of qubit measurements that are originally associated to a single logical variable s_i (i.e., the N_c spins ferromagnetically coupled during embedding—see Eq. (5) and Ref. King and McGeoch 2014).

Algorithm 1 Greedy search heuristic.

```
# Initialize energy tuples with the local magnetic fields, h:
for i from 0 to number of qubits − 1 do
    energy[i] = {−|h(i)|, h(i), i}
end for
# Reorder energy tuples into a heap according to absolute magnitude:
Energies = heap(energy)
# Initialize solutions:
while Energies do
    # Energy tuple elements (largest magnitude energy tuple first):
    {x, e, i} = heappop(Energies)
    if e > 0 then
        Solution[i] = −1
    else
        Solution[i] = +1
    end if
    # Update the rest of the heap:
    for z in Energies do
        n = z[2] # Qubit number
        # Energy update with the coupling strength, J:
        z[1] = z[1] + Solution[i] ∗ (J(i, n) + J(n, i))
        z[0] = −|z[1]|
    end for
end while
```

Genetic algorithm There is extensive literature covering classical algorithms used for the quantum annealing benchmarking purposes such as simulated annealing, parallel tempering and cluster Monte Carlo methods, and many others (Denchev et al. 2016; Mandra and Katzgraber 2018; King et al. 2015). Algorithms such as semidefinite programming (SDP) and particle swarm optimization (PSO) are widely used in Finance for solving continuous portfolio optimization problems (Cornuejols and Tütüncü 2006). PSO in particular can be a powerful tool due to its ability to solve non-linear problems, the feature it shares with other evolutionary algorithms (Kondratyev and Giorgidze 2017). Some of these methods, such as SDP, are recently receiving attention from the gate model quantum computing perspective (Brandao and Svore 2017)—but are still far away from being practical. In this paper, we choose genetic algorithms (GA) from the wide range of well-studied evolutionary search heuristics. Genetic algorithms are adaptive methods of searching a solution space by applying operators modeled after the natural genetic inheritance and simulating the Darwinian struggle for survival. There

is a rich history of GA applications to solving portfolio optimization problems (Lin et al. 2005; Oh et al. 2005), including recent researches (Kondratyev and Giorgidze 2017; Kshatriya and Prasanna 2018) that explored a range of new portfolio optimization use cases.

A GA performs a multi-directional search by maintaining a population of proposed solutions (called chromosomes) for a given problem. Each solution is represented in a fixed alphabet with an established meaning (genes). The population undergoes a simulated evolution with relatively good solutions producing offsprings, which subsequently replace the worse ones. The estimate of the quality of a solution is based on a fitness function, which plays role of an environment. The simulation cycle is performed in three basic steps. During the selection step, a new population is formed by stochastic sampling (with replacement). Then, some of the members of the newly selected populations recombine. Finally, all new individuals are re-evaluated. The mating process (recombination) is based on the application of two operators: mutation and crossover. Mutation introduces random variability into the population, and crossover exchanges random pieces of two chromosomes in the hope of propagating partial solutions.

In the case of portfolio optimization problem the solution (chromosome) is a vector $\mathbf{q} = (q_1, \ldots, q_N)$ consisting of N elements (genes) that can take binary values $q_i \in \{0, 1\}$. Our task is to find the combination of genes that minimizes the objective (fitness) function $O(\mathbf{q})$. Due to the relatively short string of genes we do not use the crossover recombination mechanism as it provides very little value in improving the algorithm convergence (see Algorithm 2 for details).

Algorithm 2 Genetic algorithm—unconstrained portfolio optimization.

1: Generation of L initial solutions by populating the chromosomes through the random draw from the pool of possible gene values {0, 1}.
2: Evaluation of the objective (fitness) function for each solution.
3: Ranking of solutions from 'best' to 'worst' according to the objective function evaluation results.
4: **for** i **from** 0 **to** *number of iterations* − 1 **do**
5: Selection of K best solutions from the previous generation and production of L new solutions by randomly changing the values of one or more genes. With $L = mK$ every one of the 'best' solutions will be used to produce m new solutions.
6: Evaluation of the objective (fitness) function for each solution.
7: Ranking of solutions from 'best' to 'worst' according to the objective function evaluation results.
8: **end for**

The best values of parameters L and K depend on the problem size and specific QUBO coefficient mapping scheme and can be found through the trial and error process. The objective here is to achieve the target convergence with the smallest number of objective function calls. The computational time of a single objective function call on the test machine (Intel(R) Xenon(R) CPU E5-1620 v4 processor run at 20% CPU utilization) has been measured to be proportional to N^2 up to a cost of about 30 μs for $N = 60$.

Note that, as with all heuristics, the GA approach does not guarantee optimality. Hence the scaling measured is not necessarily representative of the complexity of the problem—but rather a measure of the ability of the algorithmic approach to find the best solution.

Optimization of forward and reverse quantum annealing As discussed in Section 3, our runs on DW2000Q are fully specified by the following: an embedded Ising model (5) where the free embedding parameter J_F has been set, an annealing time τ, and the pause time ρ and location s_p (for reverse annealing). Moreover, in order to obtain meaningful results on the values of the logical variables, a *majority voting decoding* procedure has been applied to the returned measurements of the physical qubits of the embedded Ising Eqs. (5–6) for each 1D chain, as customary (Venturelli et al. 2015a). Finally, runs are separated in batches of 1500 anneals each with a different spin-reversal transormation (or "gauge") in order to average out systematic errors during the anneals (King and McGeoch 2014).

In order to determine the best parameters, we bruteforce over a fixed set of values (see Table 2), and we measure the TTS for all those parameters, singling-out the lowest found value, instance by instance (see Appendix 3). This simple procedure represents a statistical pre-characterization of the instance ensemble sensitivity to the parameters, and it is common for quantum annealing benchmarking initial studies of a problem (Venturelli et al. 2015a, b; Hamerly et al. 2018). Due to the post-selection of the best embedding and pause location parameters, the reported results then will represent a "best case scenario"; however, previous empirical research shows that once the distribution of typical optimal parameters has been estimated, it is possible to set up a lightweight on-the-fly procedure to set the parameters dynamically while performing the runs (Perdomo-Ortiz et al. 2015). The results of the parameter setting optimization are shown in Fig. 2 for forward and reverse annealing. It should be immediately noted that we don't optimize over τ since our experiments (as well as prior research on similar instances (Hamerly et al. 2018)) indicates that the best TTS is obtained for the fastest annealing time, $\tau = 1$ μs.[7] What is shown is that the median best choice for

[7] For the largest problems studied, according to Hamerly et al. (2018), there might be an advantage in varying τ, but this is usually a small prefactor. See also Appendix 3 for results on a limited set of instances.

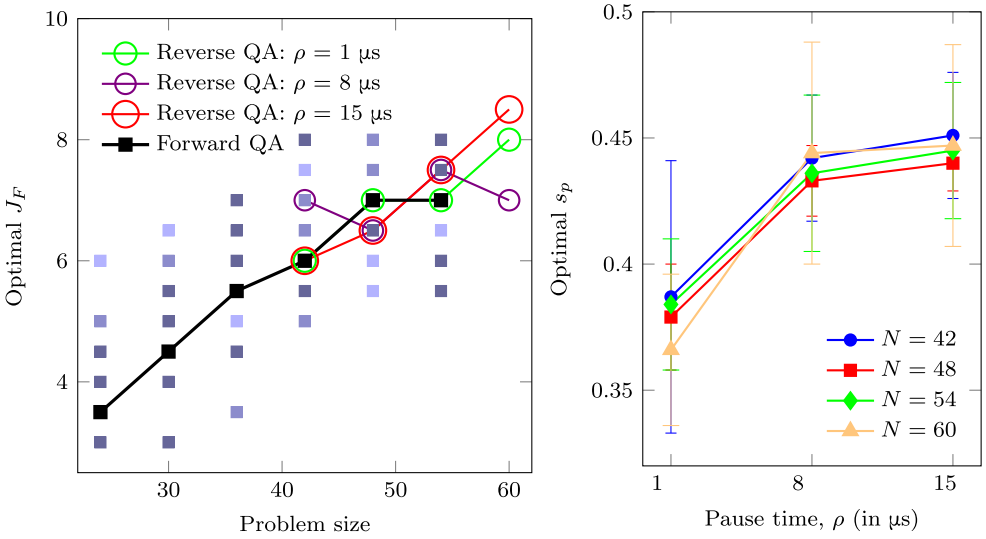

Fig. 2 Left: Median and distribution of optimal J_F as a function of problem size (see Table 2 for ranges of search). Darker colors indicate higher probability (forward quantum annealing results). Right: Mean optimal s_p as a function of pause time, ρ. Error bars show 1 standard deviation confidence bands. Annealing time τ was fixed at 1 µs

$|J_F|$ is increasing with the problem size (as expected Venturelli et al. 2015a) and the mean best choice for s_p increases with the pause period. With regard to reverse annealing, we should note that we don't pre-characterize all 30 instances, but only those that have not been solved by the greedy heuristics (Algorithm 1) that is used to initialize the reverse annealing procedure. For those instances, we are facing a three-parameter optimization ($|J_F|$, ρ, s_p).

What is clear from the results on reverse annealing is that (i) a good guess for the optimal J_F parameter is accurately predicted by the value obtained for forward annealing, and (ii) the optimal pause location is consistently later in the anneal for larger ρ. The first observation considerably simplifies the parameter setting strategy when considering operational scenarios with a new unknown instance, while the second observation is consistent with recent results and interpretation of the annealing pause tested on native Chimera problems (Marshall et al. 2018).

Results Figure 3 displays the TTS results for the GA algorithm, the forward QA solver and the reverse QA solver results. Of particular interest of this study is the probability

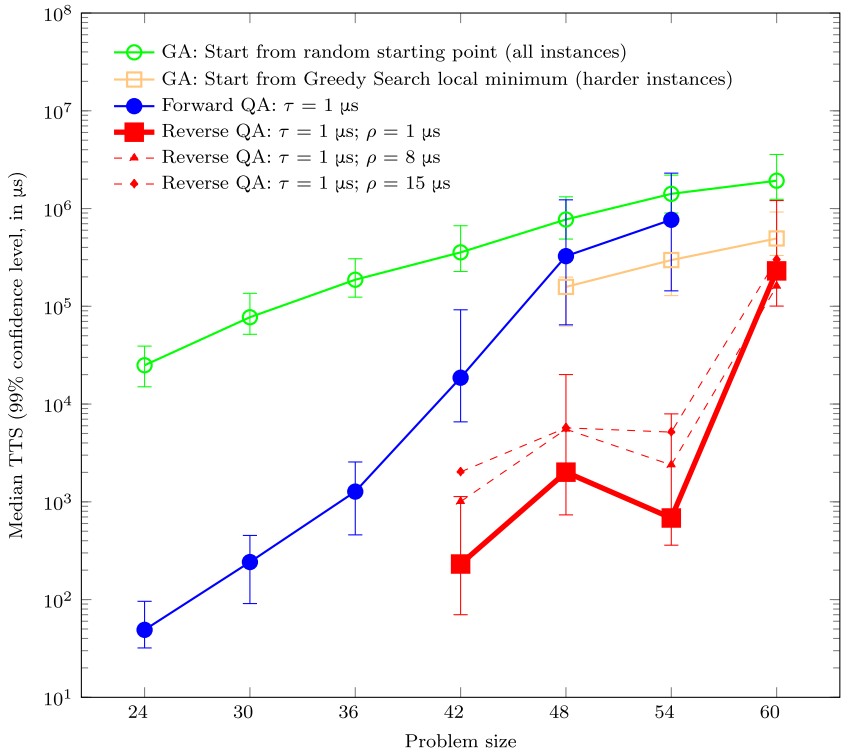

Fig. 3 Time-to-solution (99% confidence level): GA, Forward and Reverse Quantum Annealing. Curves follow the median and bars indicate 30% and 70% percentile over the 30 instances. The TTS curve for Forward QA is limited to $N = 54$ since for $N = 60$ the median is not well defined, as only 30% of instances solve to the best known objective value. Dashed curves indicate median TTS obtained with higher ρ (no error bars displayed for clarity). All TTS are measured not counting the time required to run the greedy descent that initializes the initial ansatz $\mathcal{S}$, nor the overhead times for operating the DW2000Q (see discussion in Section 5)

of success obtained for 42-, 48-, 54-, and 60-asset portfolios against the forward annealing results obtained for all portfolio sizes. We report also on the comparison with GA—to have a baseline reference on how comparatively perfomant is the quantum solver against a classical CPU. GA can also be initialized by the greedy search heuristics, and this also decreases the TTS required for GA to find the global minimum (orange curve in Fig. 3).

From the comparisons, the best solver appears to be reverse quantum annealing at minimum annealing time and pause time. In the median case, we observe one to three orders of magnitude speedup when applying reverse quantum annealing with respect to quantum annealing.[8]

5 Conclusion and next steps

In this paper, we did investigate a combinatorial optimization problem in Finance, identifying parameters and mappings that are related to real-world portfolio optimization approaches and that are compatible with the precision requirements of current D-Wave machines. The resulting QUBO has integer coefficients and is very densely connected. Hence, we leveraged prior research on fully ;connected graphs by employing the state-of-the-art methods to compile the problem on the Chimera architecture.

While for general coefficients the quadratic problem is NP-hard, the instances that we randomly generate are not necessarily hard. As a matter of fact, using the simple greedy pre-processing approach (Algorithm 1), we can solve a large part of them as shown in Table 2. We did not filter on difficulty since we don't expect all portfolio optimization problems of interests to be hard necessarily, and it is difficult to classify the instances on their expected difficulty beforehand. However, as the number of variables N increases, while QUBO parameters are kept the same, we expect greedy approaches and classical heuristics to fail on typical instances.

Consistently with the rest of the literature, all reported times for runs on the DW2000Q have been calculated in terms of the annealing times τ (or ρ), ignoring all other necessary setup and iteration times, since they don't scale with the problem size and they could be reduced in the future as system integration of the chips improves. However, once one considers the additional overhead times required to complete a full run on the D-Wave machine,[9] the advantage disappears, and the TTS results are on par with our classical benchmark. It is important to note that the reported scaling of TTS with problem size has not a fundamental character as the parameters that influence TTS (J_F, τ, ρ, s_p) are optimized only within a limited range. In particular, the annealing time is expected to be suboptimal for $N \leq 50$, leading to a measurement of a flatter slope for the median TTS (Albash and Lidar 2018). While asymptotic scaling analysis can be extrapolated only on future devices that support more variables and parameter values, we believe there is value in reporting on the complexity according to the current operational scenarios.

We note that improvements can be made over our analysis and approach and engineering advances are being planned in the next few years on quantum annealers. For instance, research papers on reverse annealing and paused annealing are just beginning to appear (Marshall et al. 2018; Ottaviani and Amendola 2018), and it seems that for hard problems it might be beneficial to implement a longer pause, larger than 100 ms, to gain a sizeable TTS advantage. Moreover, we do not expect difference in probability of success if we were to decrease the forward/reverse anneal time τ beyond 1 μs, since the process is dominated by thermalization during the pause. Based on these considerations, it is reasonable to believe that more than an order of magnitude of performance can be gained by further tuning the quench time τ and the pause time ρ.

A second very promising development that could also lead to order of magnitudes improvements is the transition of the D-Wave Chimera architecture to the Pegasus architecture. According to forecasts, the next chip will be able to embed almost 400 logical variables, reducing embedding overhead of at least a factor of 3. According to recent research (Hamerly et al. 2018), the embedding overhead is responsible for a large part of the performance; hence, we expect conservatively to gain at least an order of magnitude from this improvement. Another possible performance enhancement could be coming from inhomogeneous driving (Adame and McMahon 2018). As the number of physical qubits increases, we could also leverage error-suppression encodings nested with embedding (Vinci et al. 2016) that while reducing the total number of logical variables they provably improve the probability of success. Additional knobs available that

[8] We believe that the non-monotonic behavior for $N = 54$ is not of fundamental significance but it is due to the finite small size of our instance set for reverse annealing.

[9] Programming time, post-programming thermalization time, readout time; respectively 7.575 ms, 1 ms, 124.98 μs for the current experiments.

could improve upon the current TTS are as follows: the use of multiple pauses and the additional precision on the J couplings offered by the *extended* J-range feature.

Bearing all these considerations in mind, while it is not clear if quantum annealing is going to be the most compelling solver for portfolio optimization, our results indicate that as technology and theory progresses it could represent a viable choice. Our conclusion is indeed that these first results on the use of quantum annealing in reverse mode look very promising, especially in light of the large room for improvement that is possible to achieve by future hardware design.

It should be noted that while empirical investigations with quantum annealing are possible today, gate model quantum computers are reaching a level of maturity for which it will become possible in the next few years to experiment with digital quantum heuristics such as the Quantum Alternating Operator Ansatz (QAOA) (Hadfield et al. 2019). Moreover, when error-corrected gate model quantum computers will become viable, quantum algorithms with proven speedup such as HHL and Grover will be applicable to portfolio optimization problems (Rebentrost and Lloyd 2018).

From a Finance perspective, another direction of future research is to explore alternative portfolio optimization approaches. For example, we can mention here work done Sortino and Van Der Meer (1991) on skewed return distributions. Sortino ratio that normalizes excess return by the downside deviation often provides more valuable information about relative portfolio performances than widely used Sharpe ratio, where excess return is normalized by the standard deviation of both positive and negative portfolio returns.

Acknowledgements The collaboration between USRA and Standard Chartered Bank has been supported by the USRA Cycle 3 Program that allowed the use of the D-Wave Quantum Annealer 2000Q^{TM}, and by funding provided by NSF award no. 1648832 obtained in collaboration with QC-Ware. We acknowledge QC-Ware and specifically thank Eric Berger, for facilitating the collaboration and contributing to the runs on the D-Wave machine. D.V. acknowledges general support from NASA Ames Research Center and useful discussions with QuAIL research team. A.K. would like to thank David Bell and USRA for the opportunity to conduct research on the quantum annealer at QuAIL.

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest.

Disclaimer The authors alone are responsible for the content and writing of the paper. The opinions expressed are those of the authors and do not necessarily reflect the views and policies of Standard Chartered Bank or the Universities Space Research Association. All figures are based on own calculations.

This paper is for information and discussion purposes only and does not constitute either an offer to sell or the solicitation of the offer to buy any security or any financial instrument or enter into any transaction or recommendation to acquire or dispose of any investment.

Appendix 1. Geometric Brownian motion

A geometric Brownian motion (GBM) is a stochastic process $S(t)$ that satisfies the following stochastic differential equation (SDE):

$$dS(t) = \mu S(t)dt + \sigma S(t)dB(t), \tag{10}$$

where t is continuous time and $B(t)$ is a Brownian motion. GBM is widely used to model asset prices. If a unit of time is 1 year, then σ is interpreted as an annualized volatility (standard deviation) of asset's log-returns, which are assumed to be normally distributed. The drift coefficient μ controls deterministic component of the asset price process.

Integrating the process, we obtain:

$$S(t) = S(0)\exp\left(\left(\mu - \frac{1}{2}\sigma^2\right)t + \sigma B(t)\right). \tag{11}$$

Although GBM SDE can be used directly to simulate an asset process, it is better to use its solution to ensure that simulated asset prices do not turn negative—this may be the case for large enough time step. In our portfolio optimization example $\Delta t = 1$ month and we use the following discretization scheme for a single asset price process:

$$S(t_n) = S(t_{n-1})\exp\left(\left(\mu - \frac{1}{2}\sigma^2\right)\Delta t + \sigma z_n\sqrt{\Delta t}\right), \tag{12}$$

where $t_n = t_{n-1} + \Delta t$ and z_n is a standard normal random variable. Asset prices from the N-asset portfolio are jointly simulated using the same scheme but correlated standard normal random variables $(z^{(1)}, \ldots, z^{(N)})$ are constructed via Cholesky decomposition of the correlation matrix ρ.

Appendix 2. Chimera graph of DW2000Q and embedding

In Fig. 4, we show the layout of the chip used for the experiments, belonging to the machine D-Wave 2000Q hosted at NASA Ames Research Center.

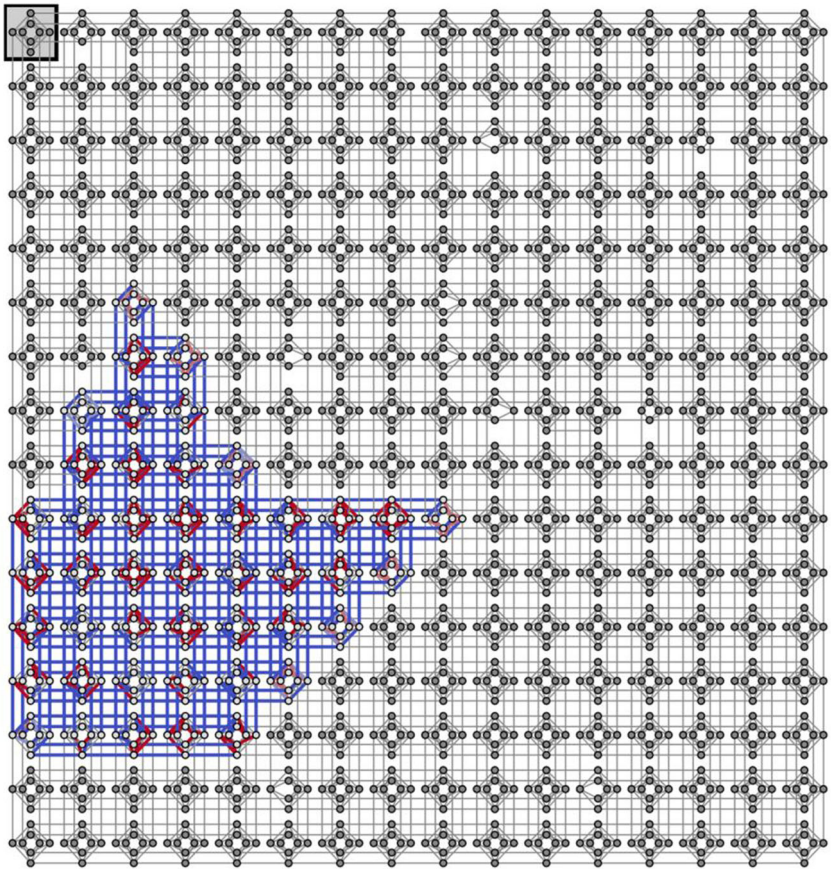

Fig. 4 Chimera Chip of DW2000Q. Each gray dot represents an active qubit (missing dots are broken qubits), the black shaded square is representative of one unit cell. The embedding for an instance ($N = 42$) is highlighted: blue bonds are ferromagnetic couplings set to J_F, while red and pink bonds represent logical couplings (J_{ij} in Eq. (6))

Appendix 3. More details on parameter setting for reverse annealing

Figure 5 displays median TTS results obtained for the mapping schemes provided by Table 1 and annealing times 1 µs and 10 µs, obtained for the first 10 instances of the benchmark ensamble on an independent set of runs with

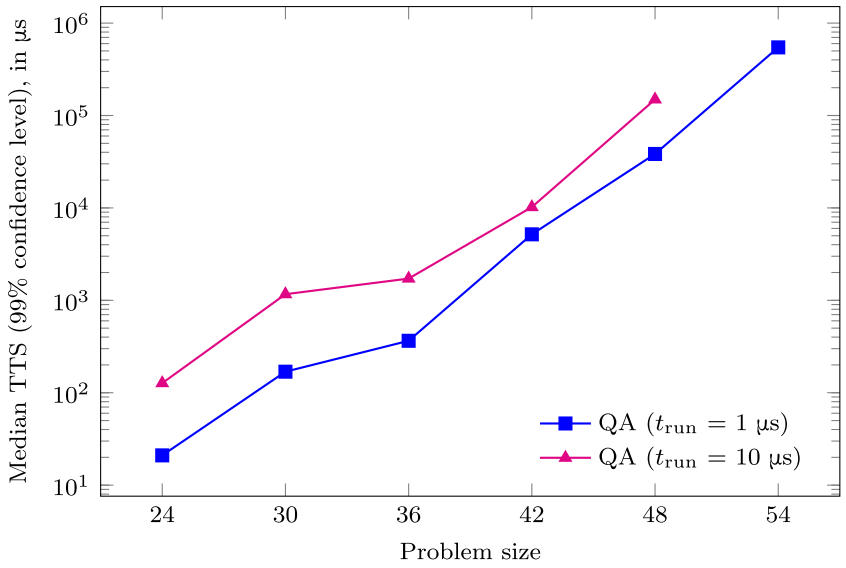

Fig. 5 Time-to-solution (99% confidence level) for different t_{run} (median over 10 instances)

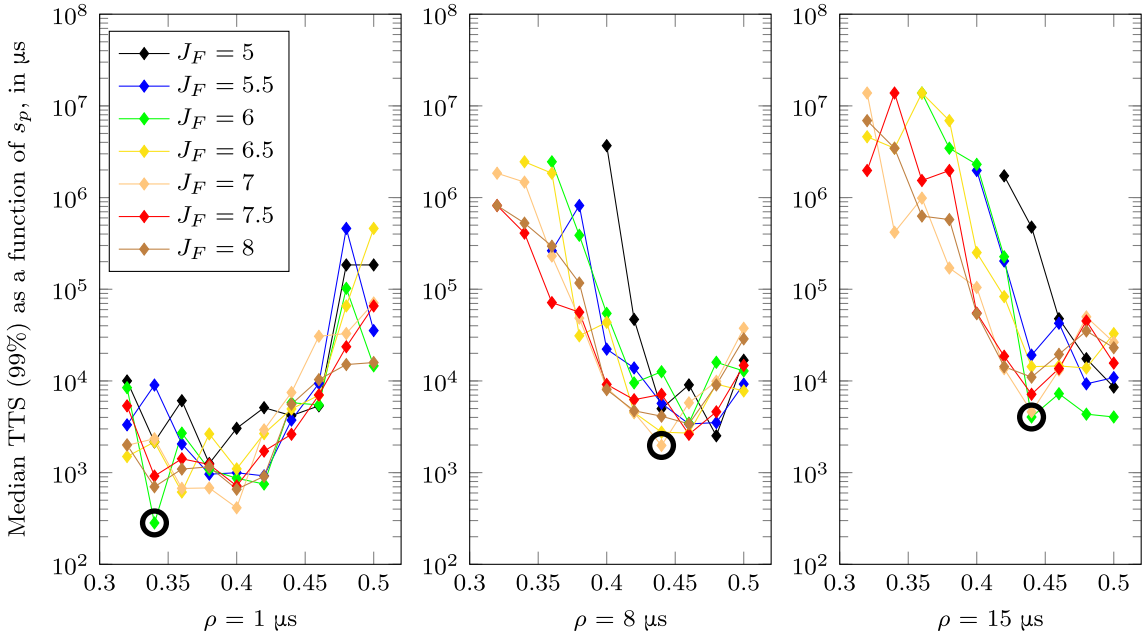

Fig. 6 Time-to-solution (99% confidence level) as a function of annealing parameter s_p. Results for a single instance, $N = 42$, $\tau = 1$ µs. Circles point out to the best found (J_F, s_p) for these three illustrative cases

respect to the results presented in Fig. 3.[10] It is clear that the choice of $\tau = 1$ µs is the most advantageous.

In Fig. 6, we show on an example how the optimal parameter setting is performed to generate data in Figs. 2 and 3. Scans are performed for different J_F and s_p and the best TTS is selected, instance by instance.

References

Adame J, McMahon PL (2018) Inhomogeneous driving in quantum annealers can result in orders-of-magnitude improvements in performance. arXiv:1806.11091

Albash T, Lidar D (2018) Demonstration of a scaling advantage for a quantum annealer over simulated annealing. Phys Rev X 8(3):031016

Boixo S, Smelyanskiy VN, Shabani A, Isakov S, Dykman M, Denchev VS, Amin MH, Smirnov AY, Mohseni M, Hv Neven (2016) Computational multiqubit tunnelling in programmable quantum annealers. Nat Commun 7:10327

Boothby T, King AD, Roy A (2016) Fast clique minor generation in chimera qubit connectivity graphs. Quantum Inf Process 15(1):495–508

[10]The reported median TTS on these runs seems to be in general faster than the results in the main paper. This could be due to finite statistics effect or to general drift in performance of the machine over time, since the runs relative to Fig. 5 were performed more than a month earlier when the machine was under low utilization. The effective temperature of the machine can vary of few milliKelvins over time for uncontrollable factors, and this is known to affect the performance of quantum annealing (Boixo et al. 2016).

Brandao FGSL, Svore KM (2017) Quantum speed-ups for solving semidefinite programs. In: 2017 IEEE 58th annual symposium on foundations of computer science (FOCS). IEEE

Chancellor N (2017) Modernizing quantum annealing using local searches. New J Phys 19(2):023024

Cornuejols G, Tütüncü R (2006) Optimization methods in finance. Cambridge University Press, Cambridge

Darolles S, Gourieroux C (2010) Conditionally fitted sharpe performance with an application to hedge fund rating. J Bank Financ 34(3):578–593

Denchev VS, Boixo S, Isakov SV, Ding N, Babbush R, Smelyanskiy V, Martinis J, Neven H (2016) What is the computational value of finite-range tunneling Phys Rev X 6(3):031015

Fernando KV (2000) Practical portfolio optimization. The Numerical Algorithms Group, Ltd White Paper

Hadfield S, Wang Z, O'Gorman B, Rieffel EG, Venturelli D, Biswas R (2019) From the quantum approximate optimization algorithm to a quantum alternating operator ansatz. Algorithms 12(2):34

Hamerly R, Inagaki T, McMahon PL, Venturelli D, Marandi A, Onodera T, Ng E, Langrock C, Inaba K, Honjo T et al (2018) Experimental investigation of performance differences between coherent ising machines and a quantum annealer. arXiv:1805.05217

Hull JC, Basu S (2016) Options, futures, and other derivatives. Pearson Education India

Job J, Lidar D (2018) Test-driving 1000 qubits. Quantum Sci Technol 3(3):030501

Johnson MW, Amin MHS, Gildert S, Lanting T, Hamze F, Dickson N, Harris R, Berkley AJ, Johansson J, Bunyk P et al (2011) Quantum annealing with manufactured spins. Nature 473(7346):194

Johnson MC, Giesecke R, McMahon K, Su PV (2016) Quantum-annealing computer method for financial portfolio optimization. US20170372427A1 QC-Ware Corp. Patent application

Karimi H, Rosenberg G (2017) Boosting quantum annealer performance via sample persistence. Quantum Inf Process 16(7):166

Kechedzhi K, Smelyanskiy V, McClean JR, Denchev VS, Mohseni M, Isakov S, Boixo S, Altshuler B, Neven H (2018) Efficient

population transfer via non-ergodic extended states in quantum spin glass. arXiv:1807.04792
Kellerer H, Mansini R, Speranza MG (2000) Selecting portfolios with fixed costs and minimum transaction lots. Ann Oper Res 99(1–4):287–304
King AD, McGeoch CC (2014) Algorithm engineering for a quantum annealing platform. arXiv:1410.2628
King J, Yarkoni S, Nevisi MM, Hilton JP, McGeoch CC (2015) Benchmarking a quantum annealing processor with the time-to-target metric. arXiv:1508.05087
Kondratyev A, Giorgidze G (2017) Evolutionary algos for mva optimisation. Risk 30(12):136–141
Kshatriya S, Prasanna PK (2018) Genetic algorithm-based portfolio optimization with higher moments in global stock markets. Risk 20(4):1–26
Lin D, Li X, Li M (2005) A genetic algorithm for solving portfolio optimization problems with transaction costs and minimum transaction lots. In: International Conference on Natural Computation. Springer, pp 808–811
Mandra S, Katzgraber HG (2018) A deceptive step towards quantum speedup detection. Quantum Science and Technology
Markowitz H (1952) Portfolio selection. J Financ 7(1):77–91
Marshall J, Venturelli D, Hen I, Rieffel EG (2018) The power of pausing: advancing understanding of thermalization in experimental quantum annealers. arXiv:1810.05881
Marzec M (2016) Portfolio optimization: applications in quantum computing. In: Handbook of high-frequency trading and modeling in Finance. Wiley, pp 73–106
Mott A, Job J, Vlimant J-R, Lidar D, Spiropulu M (2017) Solving a higgs optimization problem with quantum annealing for machine learning. Nature 550(7676):375
Oh KJ, Kim TY, Min S (2005) Using genetic algorithm to support portfolio optimization for index fund management. Expert Syst Appl 28(2):371–379
Ohkuwa M, Nishimori H, Lidar D (2018) Reverse annealing for the fully connected *p*-spin model. Phys Rev A 98:022314
Ottaviani D, Amendola A (2018) Low rank non-negative matrix factorization with d-wave 2000q. arXiv:1808.08721
Perdomo-Ortiz A, Fluegemann J, Biswas R, Smelyanskiy VN (2015) A performance estimator for quantum annealers: gauge selection and parameter setting. arXiv:1503.01083
Rebentrost P, Lloyd S (2018) Quantum computational finance: quantum algorithm for portfolio optimization. arXiv:1811.03975
Rønnow TF, Wang Z, Job J, Boixo S, Isakov S, Wecker D, Martinis J, Lidar D, Troyer M (2014) Defining and detecting quantum speedup. Science 345(6195):420–424
Rosenberg G, Haghnegahdar P, Goddard P, Carr P, Wu K, López De Prado M (2016) Solving the optimal trading trajectory problem using a quantum annealer. IEEE J Sel Top Sign Proces 10(6):1053–1060
Sharpe WF (1966) Mutual fund performance. J Bus 39(1):119–138
Sortino FA, Van Der Meer R (1991) Downside risk. J Portf Manag 17(4):27–31
Stollenwerk T, O'Gorman B, Venturelli D, Mandrà S, Rodionova O, Ng H, Sridhar B, Rieffel EG, Biswas R (2019) Quantum annealing applied to de-conflicting optimal trajectories for air traffic management. In: IEEE transactions on intelligent transportation systems
Tran TT, Do M, Rieffel EG, Frank J, Wang Z, O'Gorman B, Venturelli D, Beck JC (2016) A hybrid quantum-classical approach to solving scheduling problems. In: Ninth annual symposium on combinatorial search
Venturelli D, Mandra S, Knysh S, OǴorman B, Biswas R, Smelyanskiy V (2015a) Quantum optimization of fully connected spin glasses. Phys Rev X 5(3):031040
Venturelli D, Marchand DJJ, Rojo G (2015b) Quantum annealing implementation of job-shop scheduling. arXiv:1506.08479
Vinci W, Albash T, Lidar DA (2016) Nested quantum annealing correction. NPJ Quantum Inf 2:16017