Towards a Hybrid Algorithm: Reverse Annealing in Closed System Optimisation Problems

Charles Lillywhite
Level 4 Project, MPhys Physics
Supervisor: Dr N. Chancellor
Department of Physics, Durham University

Submitted: April 2022

I present an extended analysis of reverse annealing in closed systems, focusing on the problem of portfolio optimisation. Initial results show that, without an intrinsic mechanism identifying low energy states, significant diabatic dynamics are required to recover the ground state of the system and thus solve the problem. I demonstrate that biasing the initial state towards low energy bitstring configurations significantly improves success outcomes, such that for small-size examples, repeated runs of reverse annealing can outperform single-run forward annealing. By considering problem instances of increasing computational complexity I show that energy separation is in fact the strict regulator of success outcomes, and that energy distributions become increasingly isotropic for instances resembling random energy models. I propose an algorithm which uses iterative reverse annealing to provide quantum updates; the speed of this algorithm is found to be highly dependent on start-state energy, but even at its slowest it offers a 45% speed-up over random guessing. By comparing the range of local search to the distribution of energies, I demonstrate that this algorithm can get stuck in certain states for an arbitrarily large number of iterations. I propose a second, hybrid algorithm, which combines reverse annealing with classical greedy search; this is found to decrease run-times by a factor of 3.5, extending speed-up over random guessing to 78%. Finally I propose a third algorithm using biased forward annealing, and show that the presence of an intrinsic mechanism for lowering the energy results in a much more reliable performance, capable of offering yet further speed-up still.

Contents

1. Introduction 2

2. Quantum Annealing 3

3. Portfolio Optimisation & Markowitz Theory 5

4. Formulating Portfolio Optimisation 6

5. Methodology 7

6. Parameter Setting 8

7. **Biasing** 11

8. **Iterative Reverse Annealing** 15

9. **Greedy Search & Hybrid Algorithms** 19

10. **Biased Forward Annealing** 23

11. **Conclusions & Future Scope** 27

References 29

A. **Problem Instances** 31

B. **Errors and Uncertainty** 31

1. INTRODUCTION

Quantum computation by continuous-time evolution, as opposed to traditional gate-based models, is a field of research which has garnered significant attention in recent years. Many techniques fall under this umbrella: adiabatic quantum computing, continuous time quantum walks, and quantum annealing – the focus of this investigation. Quantum annealing is an approach whereby the solution to a computational problem is mapped to the ground state of a Hamiltonian, and solved by varying the amount of quantum fluctuations in the system. Traditional forward annealing recovers the ground state from an equal superposition of all possible solutions, by adiabtaically reducing these quantum fluctuations over the duration of the anneal. Reverse annealing is a novel technique which was designed to incorporate prior knowledge of potentially optimal solutions, perhaps obtained by one of the myriad classical algorithms which exist for a broad range of computational problems. In reverse annealing the system is initialised in a classical state and the quantum fluctuations are initially increased from zero. Optimisation problems are a natural choice of computational task to which quantum annealing can be applied. A broad and diverse range of fields, both academic and applied, require efficient solutions to complex optimisation problems; these include, but are certainly not limited to: finance[1], computer science[2,3], traffic management[4], scheduling[5], and even computational biology[6].

Quantum annealers, the physical hardware on which quantum annealing is conducted, contain quantum processing units (QPUs) made up of superconducting flux qubits, cooled to less than one tenth of one degree Kelvin [7]. These devices, first constructed by D-Wave Systems Inc. in 2011, are available commercially, and at the time of writing remain the only operational devices which possess sufficient amounts of quantum memory to be applied to non-trivial optimisation problems. For this reason they are the subject of many empirical investigations by a broad range of groups from both industry and academia. When first proposed, reverse annealing was expected to conduct a local search around the state in which the system was initialised[8], a result which has since been verified experimentally[9]. For this local search to identify states of lower energy under adiabatic conditions, reverse annealing fundamentally relies on dissipation to the environment, and is therefore a technique thought to be best suited for open system conditions, such as those on current D-Wave devices. In this report, however, I focus on reverse annealing in closed systems. I investigate how the introduction of significant diabatic dynamics can improve success outcomes in systems with little or no dissipation, particularly when the system is initialised in a state which is biased towards the solution, and how these principles can be used to construct hybrid algorithms involving reverse annealing. Even leaving aside the nature of the dynamics, investigating reverse annealing in closed systems is not a futile endeavour, since open questions still remain about how dissipation and noise from thermal fluctuations affects the performance of both forward and reverse annealing on D-wave devices. For slow enough run times with no noise, forward annealing is guaranteed to recover the solution to a given optimisation problem, yet recent findings[10] have shown that thermal fluctuations can yield improved results over a purely adiabatic run, on shorter time scales. Moreover, whilst noise is intrinsic to the ability of reverse annealing to find low energy states, if the thermal efficiency of D-Wave devices improves significantly in the coming years, then real-life reverse annealing will be conducted in environments which increasingly resemble closed systems.

The rest of this report is structured as follows: in Section 2 I present a more formal introduction to quantum annealing, and explain how the system is initialised and evolved in both forward and reverse annealing. In section 3 I summarise the problem of portfolio optimisation according to a theory of risk minimisation, before showing how this can be encoded into the ground state of a Hamiltonian in section 4. In section 7, I conduct a detailed analysis of how biasing the initial state can affect the success outcomes of reverse annealing; I describe two ways in which such a bias can be achieved, and qualitatively compare them in terms of the probability of solving the portfolio problem. Motivated by the results on biasing, I propose two distinct algorithms for solving closed system problems; one which relies solely on reverse annealing, and a second hybrid algorithm which combines reverse annealing with the classical technique of greedy search. I compare the speed and robustness of each algorithm, and in each case benchmark performance against random guessing. Finally, in section 10 I propose an alternative hybrid algorithm involving forward annealing, for which there is an intrinsic mechanism of lowering the energy, even under adiabatic conditions. I contrast the performance of this algorithm with the previous two, and demonstrate how biased forward annealing is more robust when dealing with problem instances which have highly anisotropic energy structures.

2. QUANTUM ANNEALING

Under quantum annealing, the state of an ideal quantum system is evolved according to the time-dependent Schrodinger equation:

$$i\hbar\frac{\partial}{\partial t}\left|\Psi(t)\right\rangle = H(t)\left|\Psi(t)\right\rangle \tag{1}$$

Where $H(t)$ is the total, time-dependent Hamiltonian which describes the system. By convention the total Hamiltonian is split into two parts: the first, H_p, has a ground state which encodes the solution to the specific optimisation problem under consideration, whilst a second term H_0 has a ground state which is easy to both find, and to prepare experimentally.

$$H(t) = A(s)H_0 + B(s)H_p \tag{2}$$

In order to solve an optimisation problem by quantum annealing, the problem must first be mapped to a physical model which can be encoded on a real quantum device (or classical device which simulates quantum dynamics). As per convention, this is achieved using an Ising spin model, where the problem Hamiltonian H_p is constructed classically as a quadratic sum over spins $s_i \in \{-1, +1\}$:

$$H_p = \sum_{k,j=1}^{n} J_{kj} s_k s_j + \sum_{j=1}^{n} h_j s_j \tag{3}$$

Where $J_{k,j}$ gives the coupling strength between the k^{th} and j^{th} spins, and h_j is the field strength applied to the j^{th} spin. In order to consider quantum effects, classical spins in Eq. (3) are replaced by the operator σ_j^z, which applies the single-qubit Pauli-Z operator to the j^{th} qubit, and is formulated explicitly in terms of a repeated tensor product with the 2 × 2 identity matrix:

$$\sigma_j^z = \left(\bigotimes_{r=1}^{j-1} I_2 \right) \otimes \sigma^z \otimes \left(\bigotimes_{r=j+1}^{n} I_2 \right) \tag{4}$$

In this formulation the term H_0 represents a transverse field, which drives transitions between states by flipping adjacent qubits. This driver Hamiltonian is constructed as a sum of Pauli-X operators, analogous to the σ_j^z terms described above:

$$H_0 = -\sum_{j=1}^{n} \sigma_j^x \tag{5}$$

The Ising model is a particularly useful method of encoding optimisation problems, due to the natural tendency of spin configurations to converge towards low energy configurations at low temperatures[11]. In a quantum mechanical model, the transverse field strength of the driver Hamiltonian is analogous to temperature. The way in which the control functions A and B evolve as functions of $s(t)$ determines the type of annealing implemented on the system, namely forward (FA) or reverse (RA). Since the terminology is not consistent across the available literature, the annealing schedules used in this investigation are defined explicitly across the rest of this section.

Forward annealing (sometimes described elsewhere as Adiabatic Quantum Computing[12] or the Quantum Adiabatic Algorithm) makes use of the adiabatic theory of quantum mechanics: A system initialised in the ground state of one Hamiltonian may remain in the ground state even as the Hamiltonian is changed, provided the change occurs slowly enough. If the control functions are defined such that $A(0) > B(0) = 0$, then the initial ground state is simply the lowest energy state of each individual σ_j^x. If, as the function $s(t)$ sweeps slowly from 0 to 1, we end with $B(1) > A(1) = 0$ then the ground state of H_p is recovered, which is by definition the solution to the problem. This outcome is achieved by defining the function $s(t)$ as follows:

$$s(t) = \tau = \frac{t}{T} \tag{6}$$

Where T is the total forward annealing time. Control functions are then linear in s:

$$A(t) = 1 - \tau \ , \ B(t) = \tau \tag{7}$$

The system is initialised in an equal superposition of the classical states – each corresponding to a particular bitstring – which form the computational basis. This is the ground state of the driver Hamiltonian, H_0.

In reverse annealing the system is initialised with $B(0) > A(0) = 0$, and the initial spin configuration is that of a single classical bitstring. The control functions initially undergo an inverse evolution, relative to forward annealing, such that A is increased and B decreased for a predetermined reverse duration ω_R, until the time-dependence of the Hamiltonian is temporarily paused. After some chosen pause duration, ω_p, evolution is resumed, this time in the 'forward' direction until $B > A = 0$ once more. To achieve this desired evolution, the control function $s(t)$ is defined as follows:

$$s(t) = \begin{cases} 1 + (s_p - 1)\frac{t}{t_1}, & \text{if } t < t_1 \\ s_p, & \text{if } t_1 < t < t_2 \\ s_p + \frac{(1-s_p)}{(T-t_2)}(t - t_2), & \text{if } t_2 < t < T \end{cases} \tag{8}$$

Where s_p is the value of s at which the evolution of the control functions is paused, and $\omega_p = t_2 - t_1$ is the duration of the pause. A and B have the same linear form as in Eq. (7).

In Fig. 1, below, a cartoon schematic provides a visual representation of how forward and reverse annealing solve optimisation problems. An arbitrary energy landscape is sketched, with the pink shaded region symbolising the wavefunction. In the case of forward annealing (top), the wave function starts as an equal superposition of all classical states, and localises into a trough in the energy as the quantum fluctuations are turned down. Whilst in general this is an effective means of identifying the true ground state, standard forward annealing has the potential to get stuck in local false minima, where the energy landscape is particularly broad and smooth, as shown here. For reverse annealing (bottom), the system is initialised in a single classical state corresponding to a local energy minimum. In the reverse phase, a delicate interplay between quantum and thermal fluctuations may allow the system to tunnel through the broad energy barrier. Then the pause and forward phases allow the system to thermalise and relax, within the new energy neighbourhood, into the true ground state [13].

3. PORTFOLIO OPTIMISATION & MARKOWITZ THEORY

Given a collection of n different assets, there are a possible 2^n unique portfolios, of varying size, that one could construct – each with a different level of associated risk and reward. Markowitz Portfolio Theory (MPT) argues that diversification is key – the optimal portfolio is the one which minimises the financial risk for a pre-determined minimum acceptable level of reward [15]. In general, most assets are either high-risk-high-reward, or low-risk-low-reward; MPT says that an asset should should not be evaluated solely by its own characteristics, but rather by

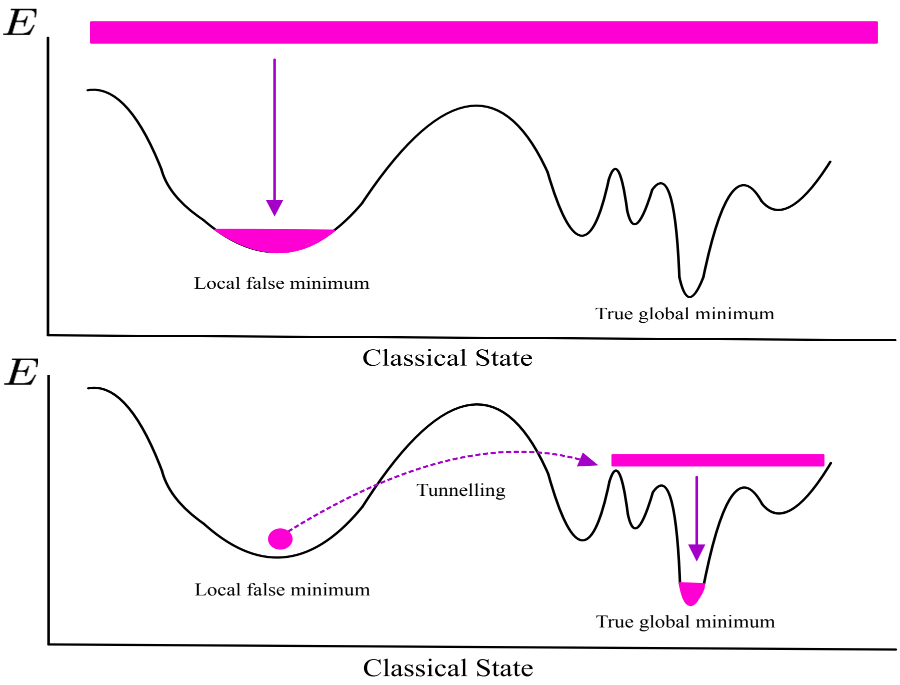


FIG. 1: Cartoon schematic of how forward and reverse annealing (top & bottom respectively) can be used to solve optimisation problems. In forward annealing, quantum fluctuations are gradually decreased such that the wavefunction localises into an energy minimum. If the energy landscape is particularly broad and smooth, this minimum may not correspond to the true grand state. In reverse annealing, quantum and thermal fluctuations may enable the system to tunnel through broad energy barriers; in the pause and forward durations the system then thermalises and relaxes in the new energy neighbourhood, into the true minimum.

the effect that its inclusion would have on the overall risk and returns of the portfolio as a whole. Portfolio optimisation is a closed-system problem which lends itself nicely to this investigation; it has shown to be in general NP-Hard[14], and thus provides a good test for any potential hybrid algorithm constructed using reverse annealing. In this investigation I use historical price data to quantify risk and reward. Reward is given by expected returns, calculated by looking at the average annual logarithmic change in the price of an asset. Covariance between pairs of stock prices serves as a proxy measure of risk; if two particular assets are highly correlated then a fall in the market value of one asset will be accompanied by a falling value of the other. In this sense, a diverse portfolio is one with low correlation between its constituent assets.

4. FORMULATING PORTFOLIO OPTIMISATION

According to MPT, portfolio optimisation can be cast as a quadratic programming problem whereby we must determine the fraction z_i, of the total budget b, to allocate to asset i. The expected return of each potential asset is given by r_i, whilst $c_{i,j}$ represents the covariance between the returns of assets i and j. The problem is thus cast as:

$$\begin{aligned} \max_{z} \quad & \sum_{i=1}^{n} r_i z_i \\ s.t. \quad & \sum_{i=1}^{n} b z_i = b \\ & \sum_{i,j=1}^{n} c_{i,j}\; z_i z_j \leq c \end{aligned} \tag{9}$$

Here, c is a correlation threshold which should not be exceeded. An individual covariance term $c_{i,j}$ will be positive if an adverse change in the returns of asset i is accompanied by an adverse change in the returns of asset j, or negative if the accompanying change is favourable. Throughout this investigation the correlation threshold is therefore set to be $c = 0$, such that losses due to one asset may be offset by increased returns elsewhere in the portfolio. The goal of the continuous mean-variance portfolio problem is to optimise the weightings z_i such that total returns are maximised for an acceptable level of risk. In this investigation I choose to simplify the problem by replacing weightings with binary variables $x_i \in \{0, 1\}$. In this discrete formulation, optimisation only selects whether or not to include an asset within the portfolio (x_i is 1 if the i^{th} asset is included, and 0 if not) , and the budgetary constraint in Eq. (9) is removed. This formulation is known as a quadratic unconstrained binary optimisation (QUBO) problem, which is solved by minimising an objective cost function:

$$\min_{x} \left(-\sum_{i}^{n} r_i x_i + \sum_{i,j}^{n} c_{i,j} x_i x_j \right) \tag{10}$$

As stated in section 2, in order to execute quantum annealing the problem must be expressed as a Hamiltonian whose ground state encodes the solution. To convert from QUBO to Ising models, a simple mapping of $s_i = 2x_i - 1$ is made, which relates the binary variables of QUBO to the spin variables of Ising. By inspection of the objective cost function, defining $h_j = -\theta_1 \frac{r_j}{2} + \sum_k J_{j,k}$ and $J_{kj} = \frac{1}{4}\theta_2 c_{k,j}$ formulates the problem as a Hamiltonian with the form of Eq. (3). The terms $\theta_{1,2}$ are Lagrange multipliers which respectively set the relative weighting of returns and risk in the problem Hamiltonian. Lastly, to consider quantum mechanical effects, classical spin variables are replaced by the Pauli spin operators σ_i^z, and the problem is now fully formulated such that quantum annealing can be implemented.

5. METHODOLOGY

As outlined in section 3, the reward of an asset is quantified by expected returns, calculated by looking at the average annual logarithmic change in the price of an asset. Pairwise correlation between asset prices serves as a proxy measure of risk. Data is obtained from Yahoo Finance using the Pandas Datareader package[16] available within Python3[17]. Portfolios of different size and contents are used at different points in this report; these are stated explicitly where necessary, and in all other cases are referenced in Appendix A, along with accompanying sections of Python code.

In this investigation, quantum annealing is simulated numerically using Python3. The instantaneous state vector of the system, in relation to the Hamiltonian, is given exactly by Eq. (1). The evolution of this Hamiltonian is simulated numerically by integrating an infinite product of exponential terms:

$$\int_0^{t_{max}} \mathcal{T} \exp\left\{-iH(t)\right\} \mathcal{D}t = \lim_{q\to\infty} \mathcal{T} \prod_{j=1}^{q} \exp\left\{-i\frac{t_{max}}{q} H\left(j\frac{t_{max}}{q}\right)\right\} \tag{11}$$

Where $\mathcal{T}$ indicates that the integral or product must be appropriately time-ordered, due to the non-commutative property of matrices. By accepting a small degree of numerical error, q is set to be large but strictly finite, and the instantaneous state vector is thus approximated by:

$$\left|\psi\left(t = \frac{kt_{max}}{q}\right)\right\rangle \simeq \mathcal{T} \prod_{j=1}^{k} \exp\left\{-i\frac{t_{max}}{q} H\left(j\frac{t_{max}}{q}\right)\right\} \tag{12}$$

In each problem instance considered, the Hamiltonian and control functions are defined appropriately for the type of annealing under consideration. The state vector at the end of an anneal of length t_{max} is then calculated using Eq. (12), and from this the probability of recovering the ground state ϕ_0 (hereafter referred to as *success probability*) is calculated as:

$$P = \left|\langle\Psi(t_{max})|\phi_0\rangle\right|^2 \tag{13}$$

6. PARAMETER SETTING

Before conducting a more detailed analysis of reverse annealing, the weighting terms θ_1, θ_2 must be appropriately selected such that the solution is non-trivial. As described in section 4, these terms set the relative weighting of returns to risk in the problem Hamiltonian. If θ_1 is much larger than θ_2 then the optimal solution will simply correspond to selecting all of the assets for the portfolio. In the opposite case, the optimal solution is simply to buy no stocks. Neither of these scenarios correspond to practically useful real-world solutions; in the rest of this report the weightings will always be set, for each portfolio considered, such that the solutions are non-trivial.

What is less obvious is the magnitude that these terms should have, given an appropriate ratio between them. To investigate this I have considered a small size problem instance including just 5 assets, and plotted heat maps of the success probability obtained by reverse annealing for different combinations of forward and reverse duration. This is shown in Fig. 2, where weighting terms take the form $[\theta_1, \theta_2] = M \times [1, 500]$ for different magnitudes *M*. In the case of a particularly small magnitude (*M* $= 0.1$, top left) an obvious structure is observed where success probabilities appear to be oscillatory for increasing forward or reverse durations, and broadly symmetric about the line $\omega_R = \omega_F$. As *M* is increased, this structure deteriorates markedly, but success probabilities are increased within certain localised regions of the parameter space. Eventually, for particularly large magnitudes, success outcomes are only non-zero for very short reverse or forward durations. This behaviour is best understood by consideration of the energy expectations of the problem and driver Hamiltonians. The magnitude of the weightings $\theta_{1,2}$ (for a given ratio between them) determines the magnitude of the eigenvalues of the problem Hamiltonian, and therefore the energy scale of the system. If *M* is too large, then the energy expectation of the problem Hamiltonian will dominate throughout the evolution, and the driver Hamiltonian is not strong enough to generate significant transitions out of the starting state. This explains why the bottom right panel of Fig. 2 is mostly dark – except in regions where ω_R or ω_F are close to zero, and strong diabatic dynamics can drive some transitions. In the opposite case of *M* being too small, it is the driver Hamiltonian which dominates the total energy expectation throughout the anneal. In this case the evolution actually proceeds by random bit flipping, analogous to an unstructured quantum walk through the solution space. This explains the oscillatory pattern in the top left panel of Fig. 2 – oscillatory success probability is a key feature of quantum walks – and also the fact that the plot is symmetric about $\omega_R = \omega_F$, since it is only total annealing time which matters, not the specific combination of durations. Anecdotally, by comparing several problem instances of sizes *n* = 3 up to *n* = 10, I find that setting the weightings such that the eigenvalues of the driver and problem Hamiltonians are approximately equal – at least up to an order of magnitude – is the optimal approach. This is applied to all problem instances referenced hereafter.

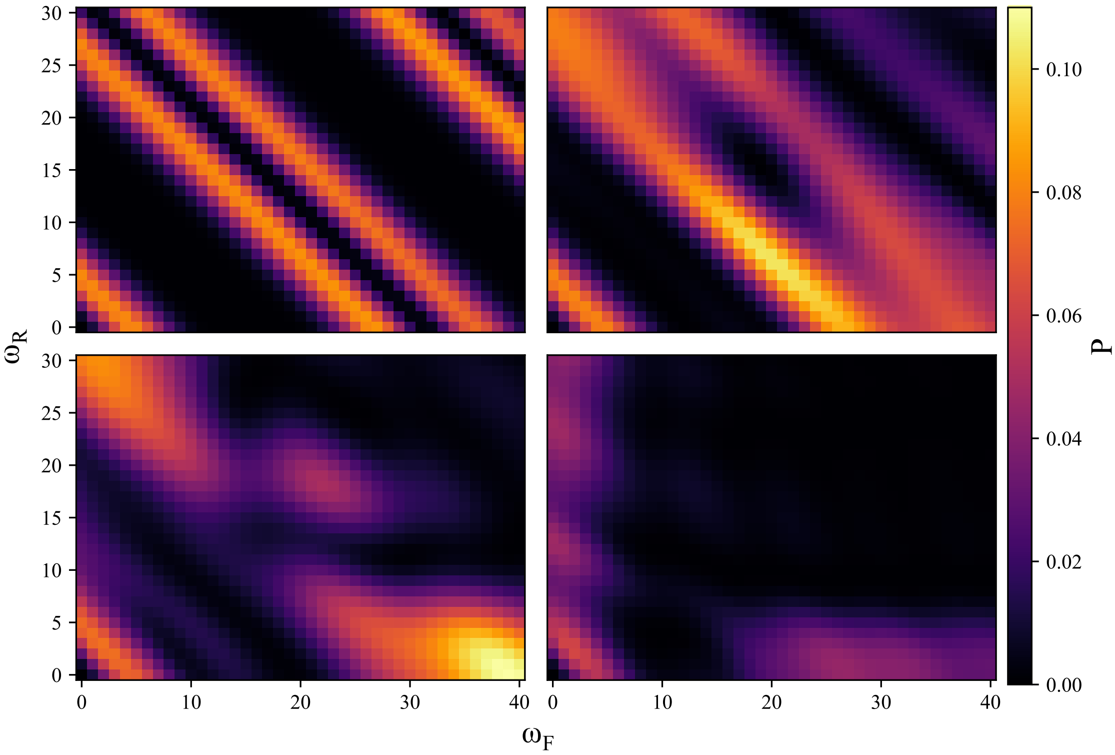


FIG. 2: Success probabilities obtained by reverse annealing on problem instance 1, for different combinations of reverse and forward durations (ω_R, ω_F respectively). Durations are in relative units where $\hbar$ = 1. In each panel the weighting terms have the form $[\theta_1, \theta_2] = M \times [1, 500]$ for magnitudes *M* $= 0.1$ (top left), 2 (top right), 4 (bottom left), 10 (bottom right). In all cases the pause duration has been set to $\omega_p = 10$, and evolution is paused when $s_p = 0.6$.

The second set of reverse annealing parameters which should be investigated are the durations $\omega_{R,p,F}$. Determining the specific combinations which yield the greatest success probabilities will improve the performance of an algorithm which uses reverse annealing to recover the ground state. Fig. 3 shows heat maps of success probability obtained by reverse annealing for combinations of reverse and pause duration; each panel corresponds to a different forward duration. In each case the distribution of probability across the parameter space is similar, but as ω_F is increased heat maps appear dimmer, with larger darker regions appearing, particularly for longer pause durations. This means that maximum obtainable probability is decreasing as the forward duration is increased, and poorer success outcomes are likely to be obtained across a greater range of the parameter space.

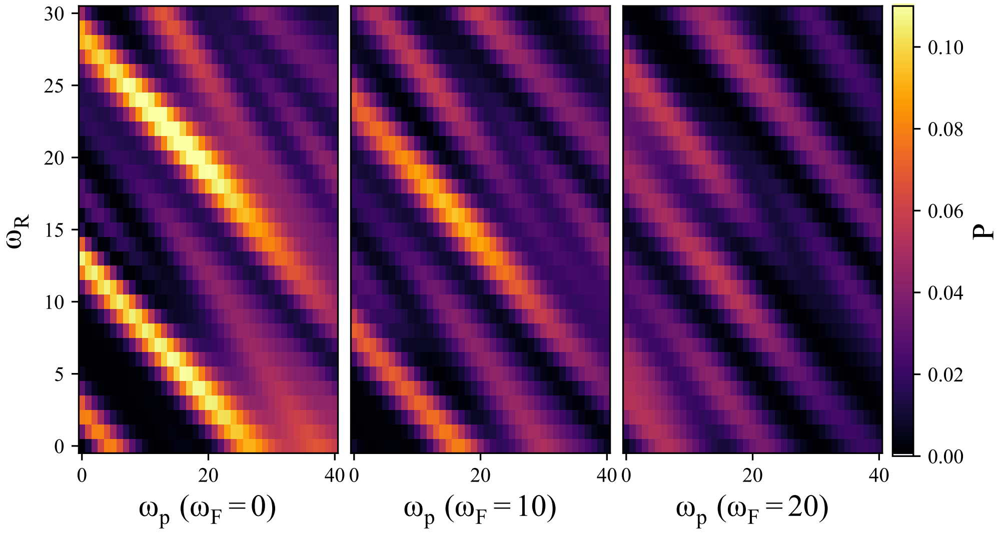


FIG. 3: Success probabilities obtained by reverse annealing on problem instance 1, for different combinations of reverse and pause duration. Each panel corresponds to a different value of forward duration.

The most likely explanation for this observation is that significant diabatic transitions are required to generate transitions in a closed system, and setting arbitrarily short forward durations is the best means of achieving this. In a system which is not coupled with the environment, there is no mechanism by which the system's energy is lowered during a single annealing run. In an open system, dissipation to the environment causes thermal relaxation between energy levels, such that there is a significant population transfer towards lower energy states even under adiabatic conditions. In a closed system however, the system always remains in an instantaneous eigenstate of the time-dependant Hamiltonian, such that there is no transfer unless a significant degree of diabatic transitions are introduced by the annealing schedule. This concept is visualised in Fig. 4. For the rest of this investigation, the forward duration is always taken to be $\omega_F = 0$.

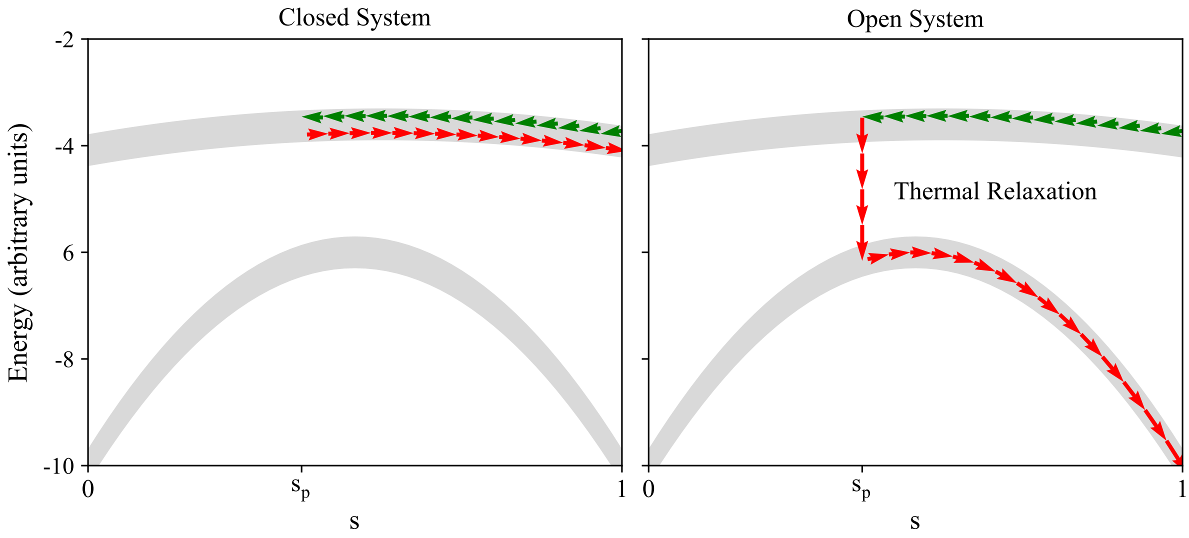


FIG. 4: A cartoon schematic showing state occupation as s is evolved. In an open system, dissipation to the environment causes a thermal relaxation of the energy levels, such that there is a significant transfer of the wavefunction from an excited state to the instantaneous ground state, even under adiabatic conditions. As s is increased after the pause phase, this enables the ground state to be recovered and the problem solved. In a closed system, transitions between energy levels is not possible without significant diabatic dynamics.

7. BIASING

The probability heat maps in section 6 were produced from a single, specific starting state. However, the original motivation for reverse annealing as an optimisation technique is that it is a method which can exploit prior knowledge of "good" solutions by searching a local region of solution space around the initial state. Success outcomes are greatly improved when the system on which reverse annealing is conducted is initially biased towards one of these potential solutions. To verify that this local search is indeed happening within the closed system portfolio problem, I have executed repeated reverse anneals across a range of different classical starting states, and measured the energy difference between input and output states. As plotted in Fig. 5, this confirms that reverse annealing is conducting a local search of the energy landscape, as expected. The most obvious way to achieve bias is therefore to prepare the system in a low energy state. The solution is encoded by the ground state of the problem Hamiltonian; we would therefore expect other low energy classical states in the computational basis (which are eigenvectors of the problem Hamiltonian) to perform well when used as the starting point for a reverse anneal.

An alternative way of biasing the initial state, however, is in terms of bit-flips. Each classical state corresponds to an n-bit binary string; the number of bits by which this string differs from that of the ground state is quantified by the *Hamming Distance*, D_H (so-called because it refers to a distance in bit-space). As an explicit example: the ground state of the Hamiltonian in problem instance 1 is given by the $|12\rangle$ state (a column vector with a 1 in the

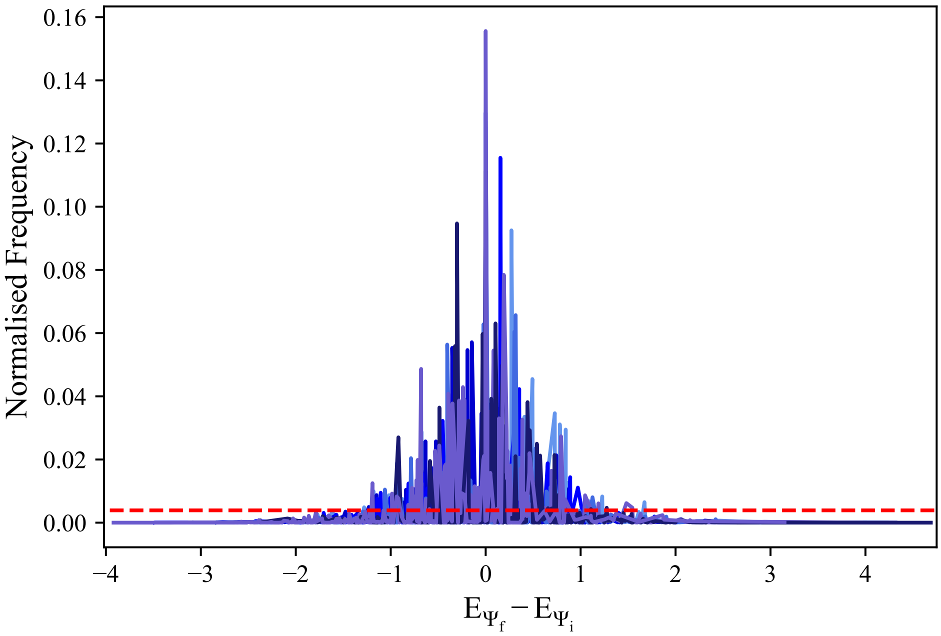


FIG. 5: Energy difference between input and output states for reverse annealing. A starting state is randomly selected and an arbitrary reverse anneal is executed. The output state is selected according to the probability distribution of the final-state-vector, and the energy difference between input and output states is measured. This process is applied to twenty states of different energies (corresponding to the different shades of blue) and 100 repeats taken in each case. The normalised frequency of the repeats is highly clustered around an energy difference of zero in all cases, showing that a local search of the energy landscape is being conducted. The red dotted line corresponds to the probability picking the output states at random, for comparison.

thirteenth entry, counting from zero, and 0 elsewhere), which can be written as the 5-bit string $|01100\rangle$. If the system is initialised in the $|7\rangle = |00111\rangle$ state, then the Hamming distance between initial and ground states is D_H = 3. Fig. 6 shows the effect of biasing the initial state when reverse annealing is used to solve problem instance 1. Each panel corresponds to a different value of D_H, where the heat map shows success probabilities which have been averaged over all of the possible starting states at that Hamming distance. Fig. 6 clearly shows that biasing the start state towards a smaller Hamming distance will increase the success probabilities obtained by reverse annealing. As D_H is increased, the maximum obtainable success probability decreases, whilst the heat maps become less brightly coloured with a greater area of dark regions, showing a decrease in the average probability over all combinations of reverse and pause durations.

The distribution of states according to Hamming distance follows a combinatorial relation, such that the number of states at a given hamming distance d from the solution is given by $\binom{n}{d}$. From this, one can ascertain that the majority of states in a 5-asset problem will be either 2 or 3 bit-flips away from the ground state; as such, a Hamming distance of 2 would correspond to a weak bias, whilst a value of 1 would denote a strong bias. The performance of reverse annealing, according to the degree of bias, can therefore be benchmarked by comparison

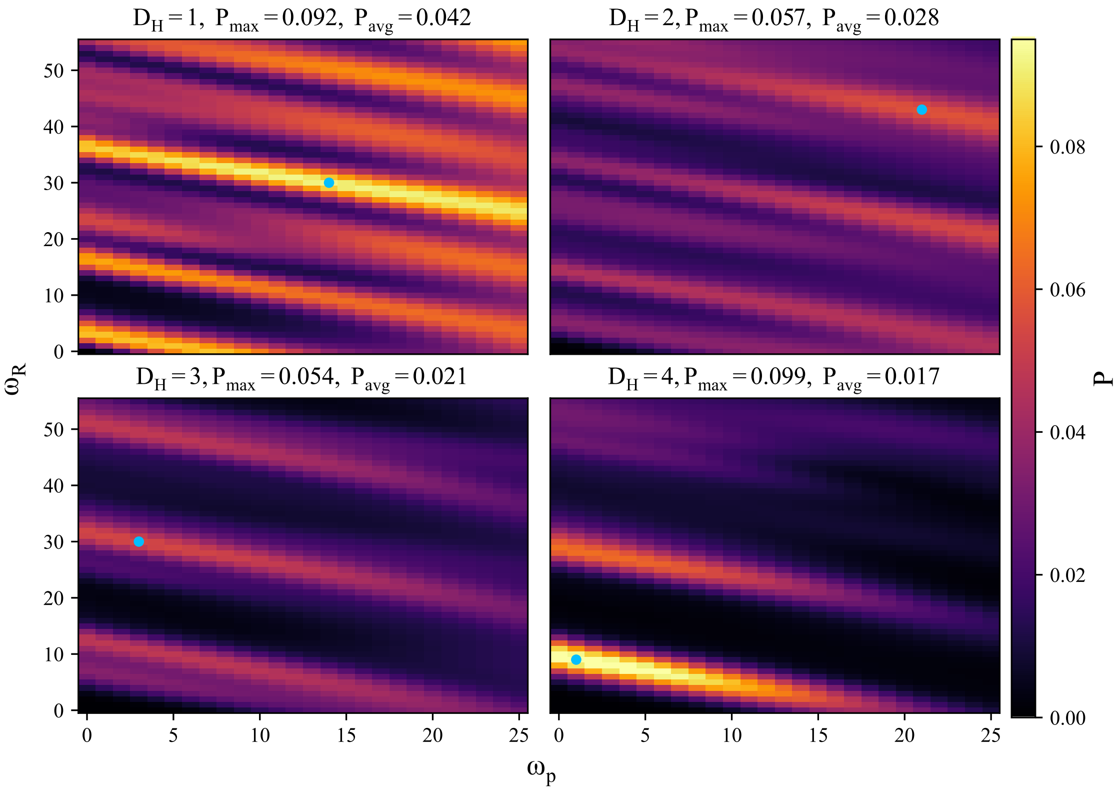


FIG. 6: Average success probabilities obtained by reverse annealing from starting states at different Hamming distances, D$_H$, in problem instance 1. In each panel the heat map is produced by averaging over all of the starting states at a given value of D$_H$. Forward duration is set to be instantaneous in all cases, as per the arguments of section 2. The value of D$_H$ is stated explicitly above each panel. The values of the average probability and the maximum probability obtainable in this region of parameter space are also stated. The blue data point in each panel shows the specific combination of durations for which the maximum probability is obtained.

with the success probability obtained by forward annealing in the same total run time. In the case of strong bias (as exhibited in the top-left panel of Fig. 6), fine-tuning the annealing durations $[\omega_R,\ \omega_p,\ \omega_F] = [14, 0, 32]$ yields the maximum obtainable success probability of P$_{max}$ $= 0.092$ in a single run time of $t_{run} = 46$ (in units where $\hbar = 1$). Success probability of $\mathcal{O}(1)$ would therefore be achieved after 1/P$_{max}$ $\simeq 11$ repeats, corresponding to a total annealing time of T$_{anneal}$ $\simeq 500$. A forward anneal of the same total annealing time, starting from an equal superposition of classical states, recovers the ground state with a probability of only P$_{FA}$ $= 0.86$. Even if a fine-tuning of the duration parameters is not possible, the average success probability for a strong bias is P$_{avg}$ $= 0.042$, meaning 23 repeats for $\mathcal{O}(1)$ success. Even in a “worst-of-worst” case scenario in which each repeat took the maximum single run time extractable from Fig. 6, this would require a total annealing time of approximately 1800 units. A forward anneal can yield unitary success probabilities after approximately 1600 units, but in physical terms this equates to only a very small difference. We therefore see that, even with sub-optimal specification of parameters, repeated runs of reverse annealing from a strongly biased state can yield results which are comparable with forward annealing, and a fine-tuning of annealing durations in fact gives a slight improvement.

Even with a weak bias applied to the starting state (as exhibited in the top-right panel of Fig. 6), a similar effect is observed. With a fine-tuning of duration parameters, a single run can yield a maximum success probability P$_{max}$ in t$_{run}$ $= 65$, such that 17 repeats recovers unitary success probability in a total annealing time T$_{anneal}$ $\simeq 1100$. Again this is comparable with the performance of forward annealing in the same total annealing time. In practical terms, it may prove experimentally difficult to fine-tune the parameters to the level of precision expressed so far, since in physical units the time scales correspond to a very small fraction of a second. It is perfectly reasonable to assume however, that it is possible to constrain the durations to within a local interval around the optimal combination. For example, with reference to the top-right panel of Fig. 6, the region described by $[20 < \omega_p < 25, 40 < \omega_R < 50]$ has a local average probability almost 20% than across the subplot as a whole. The result is that, even without a high precision tuning of the duration parameters, repeated runs of a reverse annealing from a biased starting state are capable of producing unitary success probability on time scales equivalent (or better) than forward annealing.

The results shown in Fig. 6 show the effect of biasing in terms of Hamming distance. However, as previously mentioned it is possible, and perhaps more logical, to bias the initial state in terms of energy, given that the solution is encoded in the ground state of a Hamiltonian. Indeed, it has already been shown, in Fig. 5, that the process of reverse annealing conducts a local search of the energy landscape. This then begs the question of which factor – Hamming distance, or energy separation from the ground state – is dominant in determining the success outcomes obtained by reverse annealing from a given starting state. To answer this question I consider a slightly larger problem in which there are a possible *n* = 8 assets to choose from. The details of this problem are given explicitly by problem instance 2, in appendix A. Fig. 7 shows the energy separation of each of the $2^8 = 256$ classical states, grouped by Hamming distance D$_H$. We observe that, in general, states at a greater Hamming distance will exist at a greater energy separation from the ground state. However, it is clear that there is a large amount of overlap between the range of energies of states at different Hamming distances. To quantify this explicitly, the blue shaded region in Fig. 7 has been drawn such that its height corresponds to exactly 10% of the total energy range. Within this small energy interval, one can find states at six of the nine different values of Hamming distance possible, namely from D$_H$ $= 2$ up to D$_H$ $= 7$.

To understand how this interplay between ΔE and D$_H$ affects success outcomes, I have executed repeated reverse anneals from starting states with different values of energy and Hamming distance, and recorded the probability of recovering the ground state in each case. The results are shown in Fig. 8, where success probability is given as a function of energy separation, and each panel groups the results according to the Hamming distance of the initial state. In each subplot the general trend is the same: for a given value of D$_H$, the smaller the energy separation from the ground state, the higher the success probability obtained by reverse annealing. This dependence on energy appears to be particularly strict, with the drop-off in probability appearing to be approximately exponential as ΔE is increased. The range of energies and success probabilities however is different in each panel of Fig. 8; the subplots are

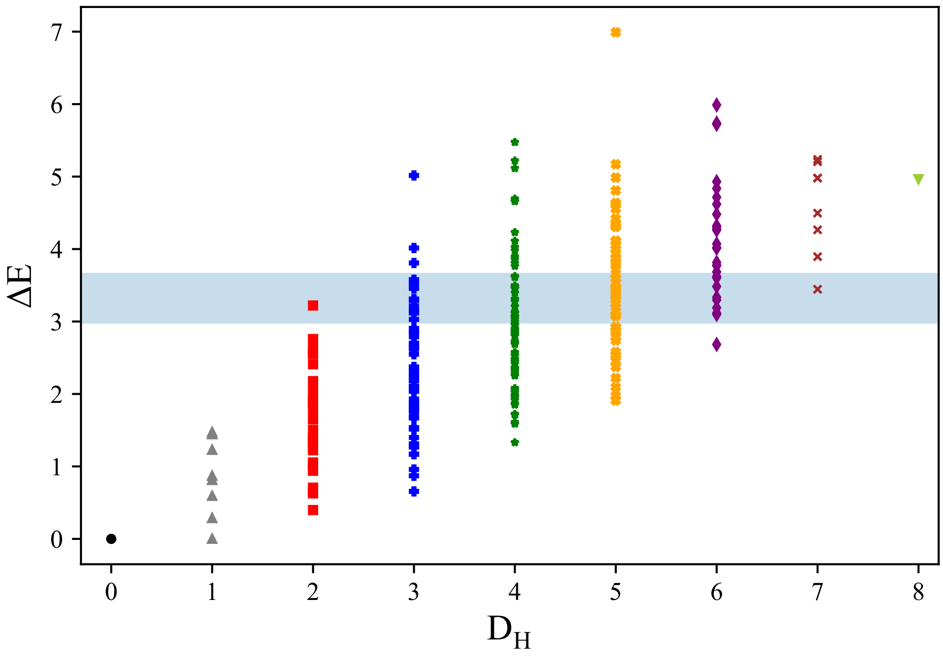


FIG. 7: The energy landscape for problem instance 2. The energy separation, from the ground state, for each of the 256 classical states is shown grouped according to Hamming distance D_H. The blue shaded region represents 10% of the total energy range, highlighting that – although median energy increases with Hamming distance – there is a significant amount of overlap.

therefore combined into a single set of axes in Fig. 9. By looking at the distribution of the data points along the x-axis in Fig. 9, one can infer that states at smaller values of D_H generally exist at lower energies; for example the majority of the red data points ($D_H = 2$) are found at $\Delta E < 2$, whilst the majority yellow data points ($D_H = 5$) are found at $\Delta E \geq 2$. There is however a large degree of energy overlap between the states closest to each other in bit-flips, as predicted by Fig. 7. The exponential decay of probability with increasing energy is particularly apparent, regardless of the Hamming distance. For example one may choose to look at the tight clustering success probabilities for states corresponding to $\Delta E \simeq 2$, regardless of the vale of D_H. From Figs. 7 and 8 it can be concluded that, whilst probability is loosely related to the Hamming distance, it is the energy bias of the initial state which strictly regulates the success outcomes obtained by reverse annealing.

8. ITERATIVE REVERSE ANNEALING

Repeated iterations of reverse annealing are one means by which bias can be successively introduced, as part of an algorithm. Starting from an arbitrary classical state – which in general may not be a state of low energy – a reverse anneal is executed for a predetermined combination of reverse, pause and forward durations. The state vector recovered at the end of the anneal then provides a probability distribution – where the square modulus of each entry gives the probability of the system being in the corresponding classical state – from which a new starting state is selected for the next iteration. This simulates the collapsing of the wave function when a

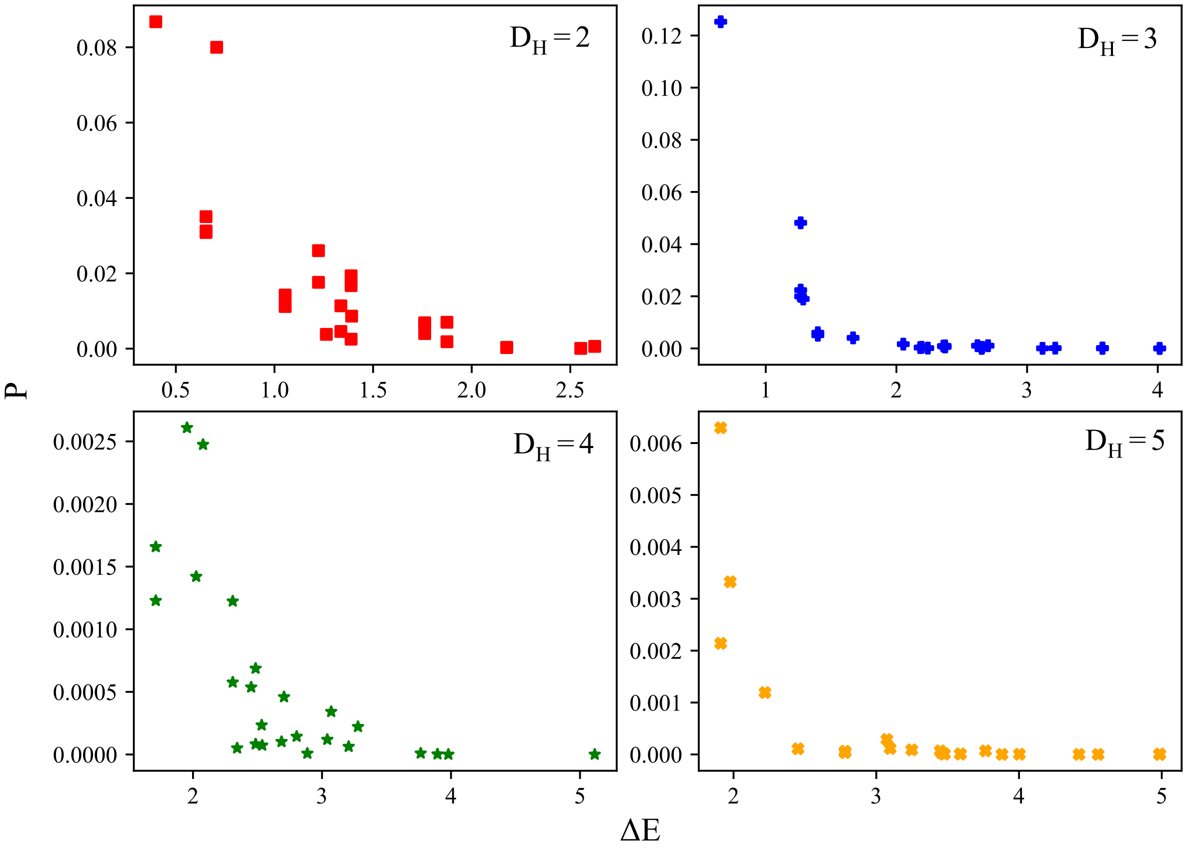


FIG. 8: Success probabilities as a function of the energy of the initial state, for problem instance 2. To produce each data point a reverse anneal is executed, with durations randomly selected from the set of values [$\omega_R \leq 15$, $\omega_p \leq 10$, $\omega_F = 0$]. Each panel corresponds to states at different Hamming distances. In all four cases the drop-off in probability with increasing energy appears to be exponential, although sampling effects are also present due to the random nature in which the duration of the anneal is selected.

physical measurement is made, following an anneal, on a real device. This process is iterated, each time updating the start state with the result of the previous run, until the ground state is recovered. This method outlined above is hereafter referred to as Algorithm 1. Without introducing an additional layer of complexity however, this framework alone will not efficiently recover the ground state with absolute certainty. This is because the system is not coupled with the environment, so no mechanism exists for preferentially identifying states of lower energy. Furthermore, it has already been shown that a single run of reverse annealing conducts a local search of the energy landscape which is broadly symmetric about the current energy level. As such we would expect that, after many repeats in which the initial state is iteratively updated, we end up back at the original starting point. Without an intrinsic mechanism, some notion of lowering the energy must be built-in to any hybrid algorithm involving reverse annealing on closed systems. To achieve this in a rudimentary way, Algorithm 1 can be modified slightly, such that the output state of a given iteration is only accepted as an update if it has a lower energy than the input state. If the output state is of a higher, energy then the initial state for the next iteration remains unchanged. To test the performance of Algorithm 1, I have considered

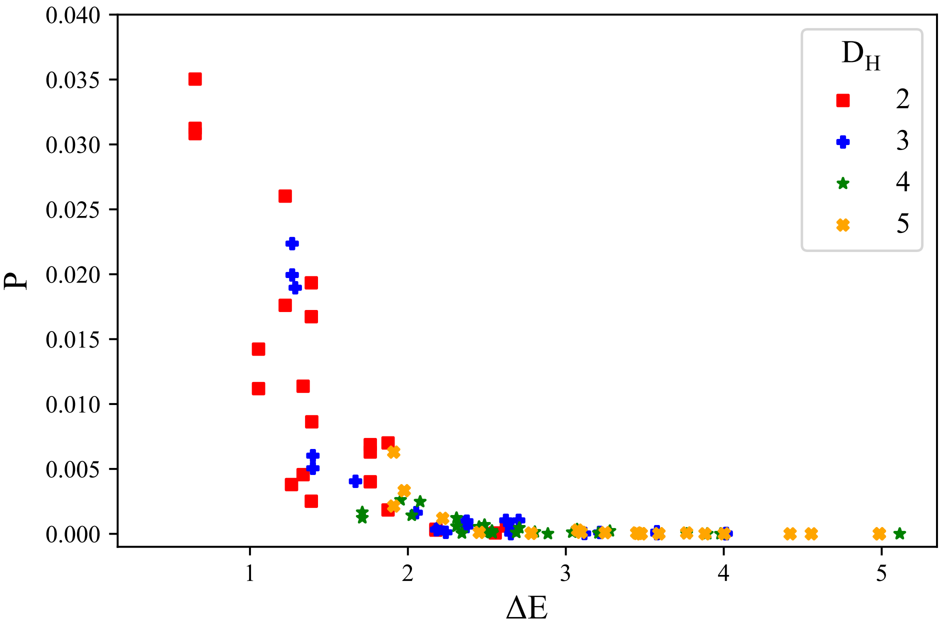


FIG. 9: The data points from Fig 7 are reproduced here on a single set of axes, to show the relative scales of energy and probability for states at different values of D_H. All data points appear to lie closely on a line which shows an exponential drop-off in probability with increasing energy. Again, some sampling effects are present in the data due to the random selection of annealing durations.

problem instance 4, which employs a semi-random energy model. In general, such models are constructed by taking a given problem Hamiltonian and randomly selecting a number of states, η_{rand}, whose energies are then randomised (with the range of the original energies preserved). Problem instances with this structure are in general more computationally complex; in the limit $\eta_{rand} \to 2^n$, a fully random energy model is recovered, which is known to be very difficult to solve with even the best available classical algorithms. Fig. 10 shows the performance of Algorithm 1 on problem instance 4, with $\eta_{rand} = 150$, for three different choices of initial starting states. Dot-dashed lines represent the average energy – relative to the ground state – after each iteration, whilst the shaded regions show the range of energy values at each iteration, in each case across twenty five repeats.

Fig. 10 shows two main features of the iterative approach. The first is that the success of the algorithm, in terms of how quickly the ground state is recovered, is highly dependent on the energy of the initial starting state. To quantify the average number of iterations required to reach the ground state in each case, we can look at the midpoint of each shaded region along the line $\Delta E = 0$. The uncertainty on this value is then given by half of the range of the shaded region along the same line. For example, for an initial starting state with $\Delta E = 1.76$ (blue shaded region), the average number of iterations required to reach the ground state is $\overline{N}_{it} = 20 \pm 12$. If the initial starting state has a higher energy however ($\Delta E = 3.53$, pink shaded region), then the average number of iterations increases to $\overline{N}_{it} = 35 \pm 25$. For an initial starting state energy which is yet higher still (green shaded region, $\Delta E = 7.07$), the average number of iterations increases further to $\overline{N}_{it} = 71 \pm 27$. This in fact is the highest energy state,

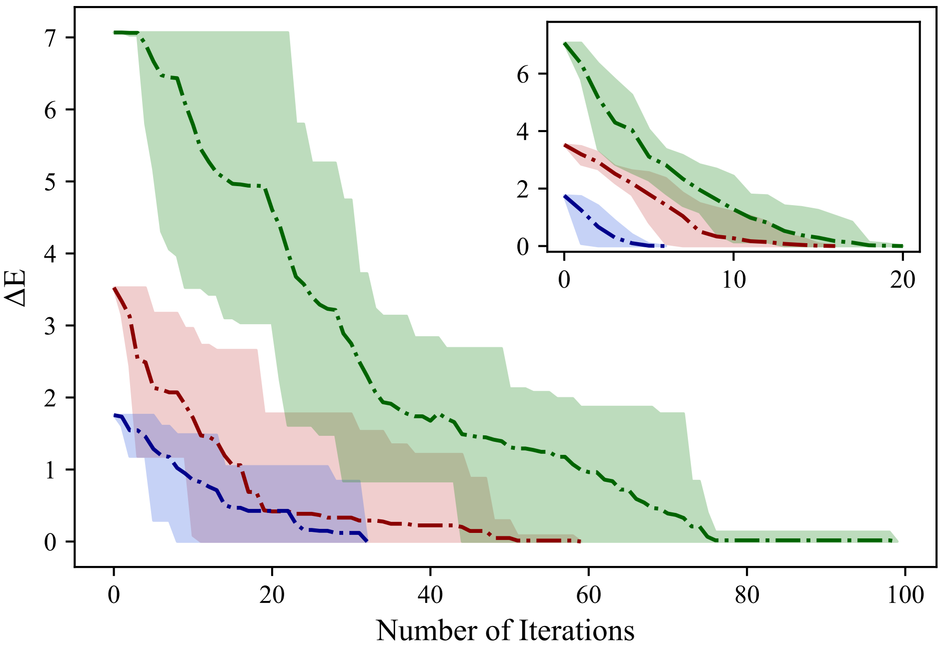


FIG. 10: MAIN FIGURE: The performance of an algorithm which uses reverse annealing to iteratively update the system state is shown for three different energies of initial starting state. Dot-dashed lines correspond to the average energy (relative to the ground state) at each iteration, across 25 repeat runs. Shaded regions correspond to the range of energies, at each iteration, across the repeats. **INSET:** The same information as the main plot, but with only successful iterations shown. A successful iteration is defined as one in which the output state has a lower energy than the input state.

and thus $\overline{N}_{it} = 20$ is the slowest that an iterative algorithm can be, for this problem instance. We can therefore benchmark the performance of the iterative approach by comparing this to random guessing. On average it would take $N/2 = 128$ guesses (where $N = 2^n$ is the size of the solution space), to correctly identify the ground state by random guessing. We therefore see that, even at its slowest, a process of iterative reverse annealing offers an improvement of approximately 45% on random guessing.

The second key feature of Fig. 10 is that failed iterations have a significant effect on how quickly an iterative approach can recover the ground state. By failed iterations, I refer to those in which the output state is of a greater energy than the input state, and as such the update is rejected. To quantify this explicitly we can look at the inset figure within Fig. 10. The subplot shows the same information as the main plot, but only accounts for "successful" iterations in which the energy is lowered. By comparing the scales of the x-axes, we can see that the successful iterations only account for a small proportion of the total amount required to recover the ground state. In a best-case scenario, we would expect successful iterations to account for approximately 50% of the total, since the local search of reverse annealing is broadly symmetric and thus each iteration is equally likely to lower or raise the energy (corresponding to success and failure respectively). This assumes a broadly uniform distribution of energy levels within the system however, which is not in general true for realistic problem instances, which have been demonstrated to have anisotropic energy structure in Fig 6. In particular, it is observed when running the iterative algorithm that the system often remains in certain states much longer than others. The local energy landscape around one such state is shown in Fig. 11. We see that, within this particular section of the energy landscape, the energy levels form two clusters, separated by a larger energy gap. The size of this gap is large enough such that the probability of the system transferring to the next available lower energy state is small at the end of a single iteration. The larger an energy gap is, the smaller the probability becomes, and the system has the potential to get stuck for an arbitrarily long period of time. The distribution of the system's energy levels is an inherent characteristic of the specific problem instance being considered, rather than a feature of the dynamics, and we are therefore forced to conclude that a simple algorithm which only uses iterative reverse annealing cannot be the most effective. Using an iterative approach in tandem with another technique for lowering the energy may prove more useful, as explored in the following sections.

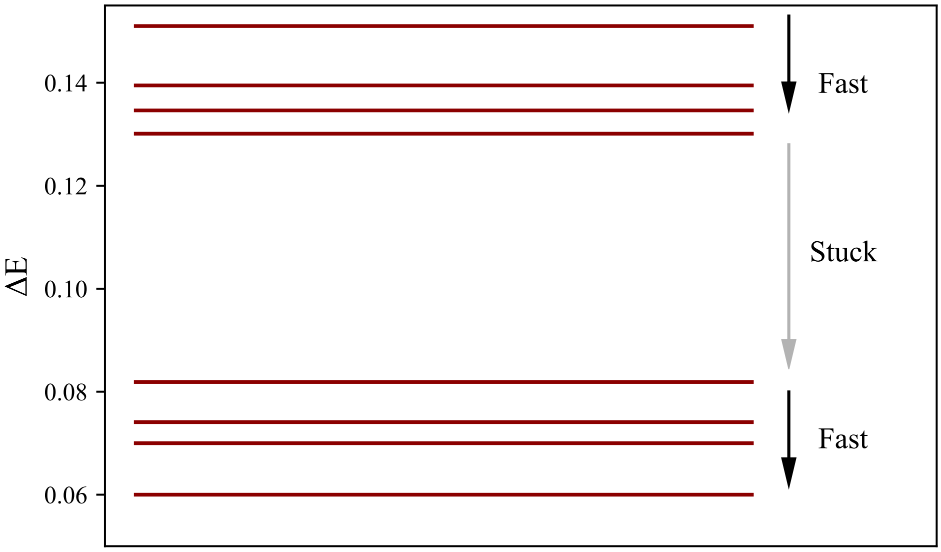


FIG. 11: Energy level spacing in the local region of a state in which the iterative reverse annealing algorithm frequently gets stuck, and is unable to output a state of lower energy, even across many iterations. The energy gap shown here is larger than the energy range of the local search conducted by reverse annealing, such that the probability of transitioning to a lower energy state in a single iteration is particularly small. The local clustering of adjacent energy levels, leaving large gaps, is a typical feature of highly structured problem instances.

9. GREEDY SEARCH & HYBRID ALGORITHMS

A greedy algorithm is one which follows the problem-solving heuristic of making the locally optimum choice choice at each stage[18]. A greedy search is implemented on a given problem instance as follows – first, for a given starting state, the state vector is expressed as an n-bit binary expansion. Next, each bit is flipped independently (such that no more than one bit is flipped at once), and the energy of the resulting state is measured in each case. Finally, the starting state is updated to whichever bit-flip lowered the energy by the greatest amount, and the process repeated until there are no possible single-bit-flip updates which lower the energy. Table 1 shows the results of greedy search applied to four different problem instances of increasing computational complexity. Each instance is created by applying the semi-random-energy model, as described in section 8. The number of randomised energies are $\eta_{rand} = 0, 75, 150, 225$ respectively. For an 8-asset problem there are 256 possible states from which we could start a greedy search; if the ground state is recovered then this constitutes a success, but if a state is reached in which no single-bit-flip update lowers the energy then the search ends in failure. Table 1 shows that, in each of the four problem instances, greedy search is capable of solving the problem. For $\eta_{rand} = 0$, greedy searching from any of the classical starting states resulted in the ground state being recovered. As η_{rand} is increased however, the frequency with which greedy search fails increases, and a greater number of potential failure states exist. The range of energies also increases, with states at increasingly high energies causing the search to fail. Since η_{rand} corresponds to the computational complexity of a problem, we can conclude that greedy search becomes less effective for increasingly complex problem instances. To explain why greedy search fails for increasingly complex problem instances, a consideration of the relationship between the distribution of states according to energy and hamming distance is needed, as these are the two factors which determine the path of a greedy search. Fig. 12 shows this relation for three of the four problem instances given in Table 1. The three panels correspond to $\eta_{rand} = 0, 75, 225$ respectively. For $\eta_{rand} = 0$ the problem is highly structured; states which exist at greater Hamming distances from the solution will, on average, be of higher energy. As η_{rand} is increased and the problem tends towards a random energy model, this structure breaks down, with the energy distribution becoming increasingly isotropic in relation to Hamming distance. For a highly structured problem, it is much more likely to be able to find a lower energy state by making a single bit-flip – in this case by bit-flipping to a smaller Hamming distance, which is seen by moving from right to left in the first panel of Fig. 12. As computational complexity increases however, the spreading of energy levels across all values of D_H means that a single bit-flip from any given state is less likely to lower the energy, and so the number of failure states increases.

η_{rand}	# Failure States (/256)	Failure Frequency	Failure State Energy Range [$\hbar$ = 1]
0	0	0%	0.00
75	14	42%	0.00 – 1.89
150	19	73%	0.00 – 2.44
225	26	82%	0.00 – 2.78

TABLE I: The number and energy range of greedy search failure states for four different problem instances using a semi-random energy model. Failure frequency denotes the proportion of the 2^n classical starting states from which greedy search terminates in a failure state rather than the ground state.

To construct a hybrid algorithm, greedy search can be combined with iterative reverse annealing

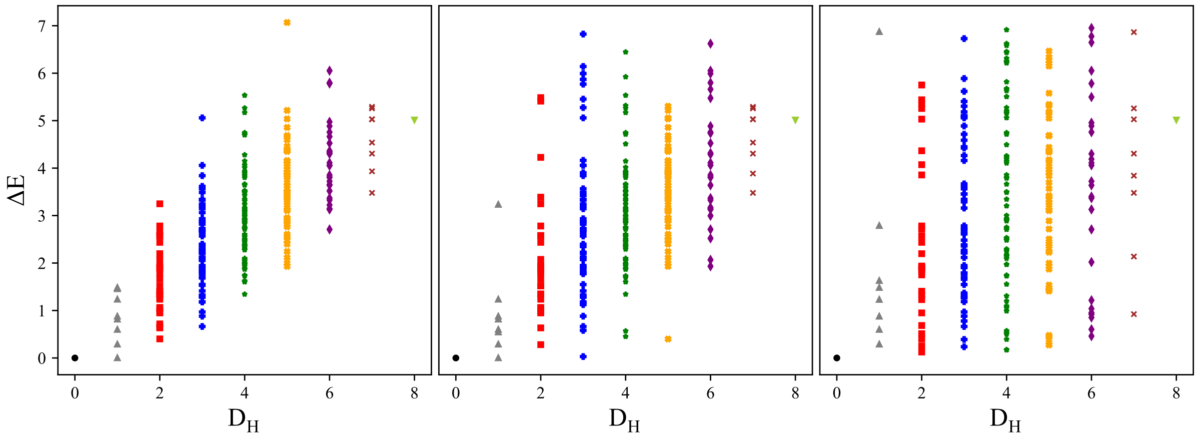


FIG. 12: The energy landscape for three of the semi-random-energy problem instances (left: $\eta_{rand} = 0$, middle: $\eta_{rand} = 150$, right: $\eta_{rand} = 225$). Data points give the energy of each of the 256 classical states, and have been grouped by Hamming distance in each case.

of the kind described in section 8. In this framework, the greedy search is a mechanism for pre-preparing the system in a low-energy state from which iterative reverse annealing can be more effectively implemented. To benchmark the performance of such a hybrid algorithm, a direct comparison can be made with Algorithm 1. For each of the three starting states, a greedy search is first implemented, before reverse annealing is then iteratively executed on the failure state. The results are shown separately in Fig. 13, where the label of the x-axis has been amended to use the term "steps", to emphasise that an iteration of greedy search is fundamentally different from an iteration of reverse annealing.

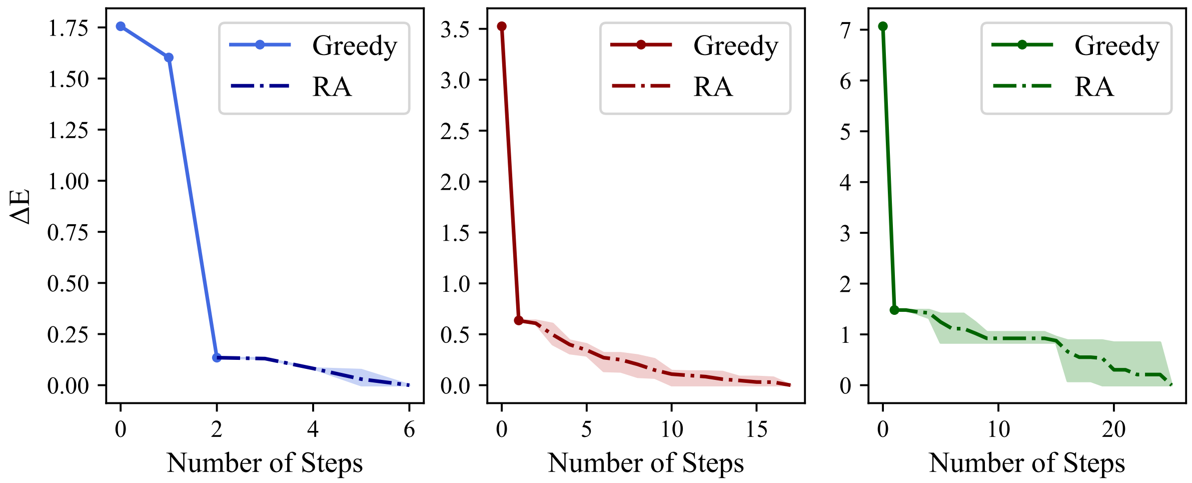


FIG. 13: Energy paths for a hybrid algorithm which combines greedy search and iterative reverse annealing. The three panels correspond to the three states shown in Fig. 10, and are colour-coded accordingly. As in Fig. 10, dot-dashed lines represent the average energy at each iteration of reverse annealing, across many repeats, with the shaded regions showing the range. Solid lines show the greedy search iterations. The x-axis is labelled as "steps" to emphasise that an iteration of greedy search is fundamentally different from an iteration of reverse annealing.

Fig. 13 clearly demonstrates how greedy search can significantly lower the energy in a small number of steps, and therefore the iterative reverse annealing which follows is able to recover the ground state much faster when compared with Algorithm 1. To quantify this we can compare the scales of the x-axes in Figs. 13 and 12 (main plot). In the case of no greedy search, the maximum number of iterations observed – across the repeats taken – were 32, 58, and 98 respectively for each starting state. When greedy search is used to pre-bias the energy, the maximum number of steps required (again across many repeats, and including greedy iterations) are 6, 18, and 28 respectively. In the case of the latter, this is three-and a-half times faster than algorithm 1, such that the overall speed-up over random guessing is increased to 78%. It is clear then that incorporating greedy search into a hybrid algorithm with reverse annealing significantly improves the speed with which the problem can be solved, compared to an approach which only uses iterations of reverse annealing.

It is worth stating explicitly that, with the hybrid algorithm (hereafter reffered to as Algorithm 2), the starting states furthest away in energy no longer necessarily correspond to the slowest performance. The speed of the second portion of the algorithm is still determined by the energy of the state from which it begins. For the greedy search portion of the algorithm however, there does not appear to be a clear relationship between the starting state energy and the number of iterations of greedy search we expect to see. This is visualised explicitly in Fig. 14, where greedy search paths of lengths 2, 3, and 4 are represented by colours green, red and blue, with start and end points shown by crossed data points. All energies are lowered by greedy searching, but the amount by which the energy is reduced does not seem to depend on the starting energy, nor the number of iterations in the process. For example there are starting states of high energy for which greedy search lowers the system energy by a large amount over a small number of iterations, but also states of a lower energy for which the system energy is lowered by a small amount across a greater number of iterations. What is clear, however, is that the range of energies of the greedy search end-states is much narrower than the energy range of the start-states. This is significant because it is the end-state energy of a greedy search which then determines the speed of the iterative reverse annealing which follows (as per the arguments put forth in section 8). Since the energy range of greedy end-states is comparatively narrow, we would expect the hybrid algorithm to perform at a broadly similar speed, regardless of the choice of start state. This has significant practical implications because it reduces the dependence of performance upon a careful selection of start state. Expressed alternatively, by combining iterative reverse annealing with greedy search, the speed at which the algorithm performs is less dependent on a “good choice” of starting state.

Although a hybrid algorithm which incorporates greedy searching offers significant speed up over an algorithm which only uses reverse annealing, the issue of arbitrarily long run times still persists. Whilst the effect of greedy searching is such that the reverse annealing section of Algorithm 2 has to search a smaller portion of the solution space, at lower state energies, than Algorithm 1, there is still every possibility that an energy gap may exist which exceeds the range of local search in a single run of reverse annealing. As such, without an intrinsic mechanism for lowering the energy, the reverse annealing in Algorithm 2 still has the potential to get stuck in a particular state for an arbitrarily large number of iterations, as per the arguments put forth at the

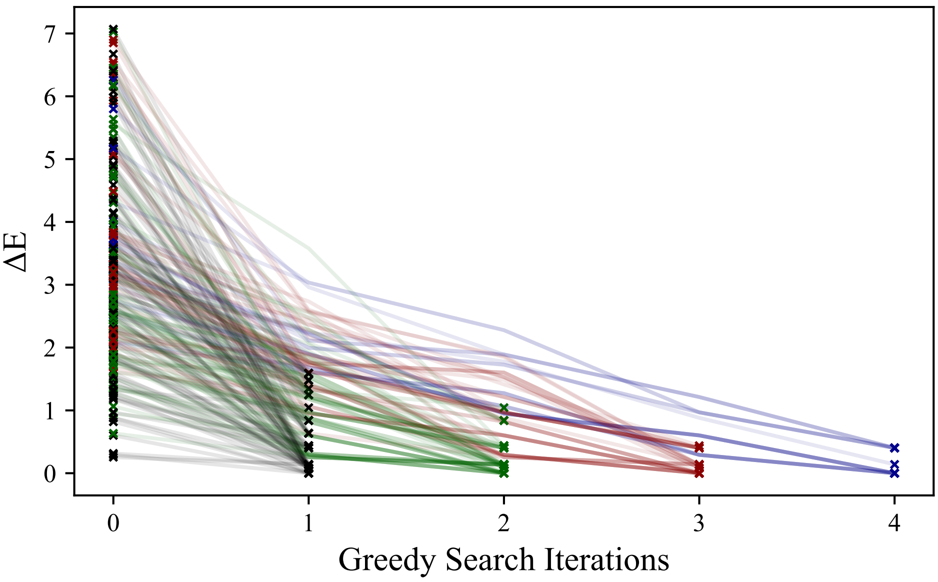


FIG. 14: Greedy search paths of lengths 2 (black), 3 (red), and 4 (blue). In all cases the system energy is lowered by greedy searching, but the final state energy does not in general depend on the starting energy, nor the number of iterations required to do so. End-state energies are marked by crossed data points.

end of section 8. This then poses the question of whether the reverse annealing in Algorithm 2 should be replaced with an alternative optimisation technique. One such possibility is explored below.

10. BIASED FORWARD ANNEALING

With a modification to Eq. (5), standard forward annealing can in fact also be used to incorporate prior knowledge of potential solutions. Motivated by the work of [19], I introduce bias into the dynamics by modifying the driver Hamiltonian to include longitudinal field components, alongside the transverse fields applied in traditional forward annealing:

$$H_0 = \sum_{m=1}^{n} \left(-\cos{(\chi)}\sigma_m^x + \sin{(\chi)} h_m \sigma_m^z\right) \tag{14}$$

Where $h_m \in \{-1, +1\}$ are the components of a trial (or "guess") solution, taking the value of +1 if the m^{th} bit of the trial solution's bit-string is 0, and -1 if it is 1. As per the formulation of [20], χ parameterises the certainty of the guess; if $\chi = 0$ then the driver resorts to the unbiased case of Eq. (5), if $\chi = \pi/2$ then the ground state of the biased driver is the purely classical state corresponding to the trial solution, and no dynamics occur. For intermediate values of χ, the ground state of the driver Hamiltonian is an unequal superposition state, with the probability density of the wave vector skewed towards the trial solution. Although not a purely classical state, the superposition is un-entangled, and moreover can be relatively easily prepared on a real device, such that we can think of this state as being "pseudo-classical". The system is initialised in the ground state of the biased driver, and dynamics applied according to the control functions in Eqs. (6) and (7). Fig. 15, below, shows the effect of biasing the driver Hamiltonian on the success outcomes obtained by forward annealing, on problem instance 4. End-of-run success probabilities are shown for three different cases of guess certainty, χ, where the true ground state is used as the trial solution. Whilst the shape of each curve is not dissimilar, the initial success probability at $t_{run} = 0$ is much higher as the degree of bias is increased. It is therefore apparent that the effect of biasing the driver, and thus the initial ground state, is to effectively give the anneal a "head start" in terms of success probability.

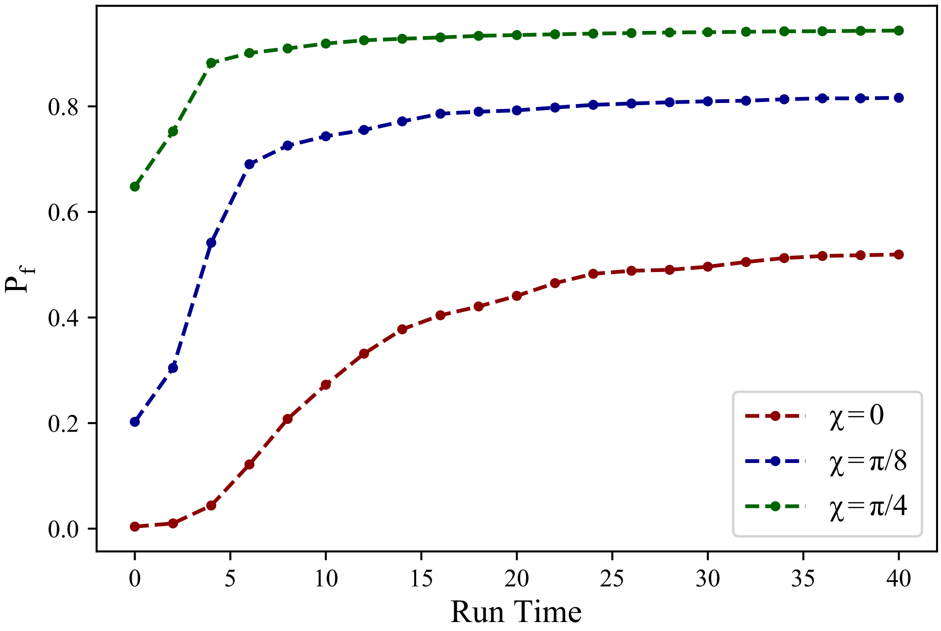


FIG. 15: End-of-run success probabilities obtained by biased forward annealing, as a function of increasing run time. Problem instance 4 is used, and the true ground state has been used as the trial solution, with three different values of guess certainty χ.

To show explicitly why biased forward annealing does not get stuck in large energy gaps, like reverse annealing, we can examine the probability distribution given by the end-of-run wave function. Fig. 16 shows the end-of-run wave function when biased forward annealing is executed on a classical state, in which reverse annealing has been found to frequently get stuck. The driver is initialised with a bias – quantified by $\chi = \pi/16$ – towards this classical state with energy E_i. The x-axis lists the classical states in the local energy region of the initial state, and E_G denotes the ground state. The y-axis corresponds to the probability of each state being occupied at the end of a single run; solid lines correspond to bias forward anneals of three different run times, and the dashed line shows the case of reverse annealing, for comparison. It is clear from inspection of Fig. 16 why reverse annealing has the potential to get stuck; the wave vector is highly peaked around the initialised state, such that any single run is unlikely to change the measured state of the system. Indeed, for this particular example, the likelihood of a single run of reverse annealing identifying **any** state of a lower energy is only approximately 5%. For biased forward annealing however, this is clearly not the case. In each of the three examples shown, the probability of obtaining a lower energy state by biased forward annealing is significantly higher than the case of reverse annealing; the likelihoods of lowering the energy are 37%, 54%, and 69% for run times of 15, 30, and 45 units respectively. We therefore conclude that biased forward annealing is much less likely to get stuck in a region where the energy gap between adjacent states is particularly large. Fig. 16 also demonstrates how biased forward annealing is capable of solving the problem in a single iteration, even if the target state is not the ground state. In this example, the target state used to bias the driver Hamiltonian is an excited state with energy E_i ($> E_G$), yet the probability of recovering the ground state is almost 0.4 for a single iteration with t_{run} = 45. Success probability increases with run time, such that even for a poor choice of target state the problem can be solved with a slow enough anneal.

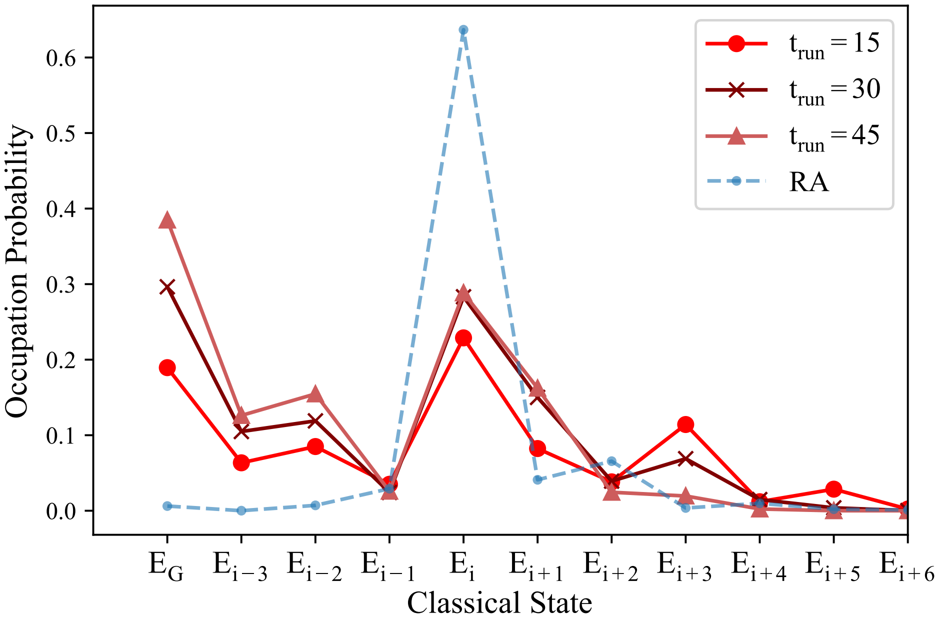


FIG. 16: The end-of-run wave functions for reverse annealing (dashed line) and biased forward annealing (solid lines), implemented on a classical state with energy E_i, in which iterative reverse annealing of Algorithm 2 is found to frequently get stuck. The square-modulus of each entry in the end-of-run wave function gives the probability of the corresponding classical state being occupied. The biased forward annealing is shown for three different run times; in each case the driver is biased towards the classical state with energy E_i, with a guess certainty of $\chi = \pi/16$. The dashed line is averaged over 25 repeats, where reverse annealing durations have been chosen randomly from the interval $[0 < \omega_R < 15, 0 < \omega_p < 10, \omega_F = 0]$ in each case.

Much like reverse annealing, biased forward annealing can also be combined with greedy search and used iteratively as part of a hybrid algorithm. In this framework, starting from an arbitrary classical state, greedy search is executed to identify states of lower energy. The end-state of the greedy search then serves as the trial solution for the first iteration of biased forward annealing. At the end of the run, the wave function is collapsed, and the system is measured to be in a state which then provides the trial solution for the next iteration. This process of updating the trial solution is repeated until the ground state is recovered. An algorithm with this formulation (hereafter referred to as Algorithm 3), is exactly analogous to Algorithm 2, but with biased forward annealing taking the place of reverse annealing. To quantify how such an algorithm performs, I have applied it to the three classical states in problem instance 4, which have been discussed several times already in this report. The path of Algorithm 3 through the energy landscape is shown below in Fig. 17, where the reverse annealing iterations of Algorithm 2 are also reproduced to enable a direct comparison. Dashed lines represent the average energy after each iteration of biased forward annealing, with surrounding shaded regions showing the range of energies, across 25 repeats. For starting states of higher energies, shown in the central and right hand panels of Fig. 17, it is clear that Algorithm 3 recovers the ground state significantly faster than Algorithm 2, as the dashed lines are steeper, and intersect the x-axis much earlier than the dot-dashed lines. Narrower shaded regions also show that biased forward annealing is exploring a more concentrated region of the energy landscape across successive iterations. Interestingly, in the case of starting states which are already low in energy (left panel), there is very little difference in the speeds of the two algorithms, with Algorithm 3 in fact requiring slightly more iterations on average to solve the problem. Furthermore, in this case it is actually reverse annealing which explores a narrower range of state energies. Together this suggests that the local search of reverse annealing is perhaps a more direct strategy for starting states which are already very close in energy to the solution. What can be said of the biased forward annealing in Algorithm 3, however, is that the number of iterations required to reach the ground state is broadly similar, regardless of the starting state's energy (total steps, including greedy iterations, are 7, 6, and 5 respectively for each panel). Given that biased forward annealing is also much less likely than reverse annealing to get stuck in large energy gaps, Algorithm 3 provides a faster and more robust method with which to solve the problem.

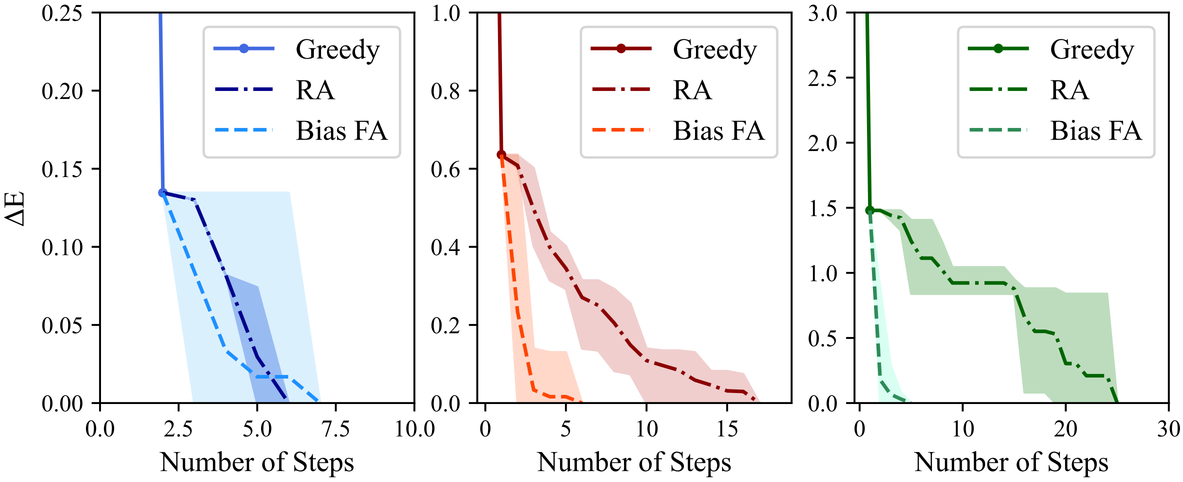


FIG. 17: Energy paths for Algorithm 3 (dashed), shown alongside those of Algorithm 2 (dot-dashed). Starting states in each panel match those in Figs. 10 and 13; The greedy search portions of each algorithm are identical, and are not shown in full here as they remain unchanged form Fig. 13. As was the case for Algorithm 2, the dashed lines show the average energy after each iteration, across 25 repeats. Shaded regions correspond to the range of energies obtained across the repeated iterations.

It is worth emphasising that the biased forward annealing parameters chosen to produce Fig. 17, $\chi = \pi/16$ and $t_{run} = 30$, have been chosen arbitrarily. It is reasonable to assume that a more careful selection of parameters could yield further improvements in the performance of

Algorithm 3. Indeed, Fig. 16 has already shown that the probability of solving the problem in a single run increases with run time, which poses the question of whether fewer, longer iterations will yield better results than a large number of shorter iterations. It is also feasible that updating the parameters iteratively with each run may increase the speed with which Algorithm 3 localises on the solution. Such ideas go well beyond the scope of this report, but provide an interesting groundwork for future investigations. What is of key importance here, however, is that the results in this section show clearly that biased forward annealing provides a more reliable framework for an iterative algorithm, and even with arbitrarily set parameters is capable of out-performing reverse annealing as the mechanism for iterative quantum updates.

11. CONCLUSIONS & FUTURE SCOPE

In this report I have presented a detailed investigation into reverse annealing as an optimisation technique for the portfolio problem in closed systems. Initial results in section 6 verify a theoretical prediction that reverse annealing conducts a local search of the energy space in a single run, but that no intrinsic mechanism exists for preferentially finding states of lower energy, without dissipation to the environment. As such, significant diabatic dynamics are required in order that the probability of recovering lower energy states is not exponentially small. In section 7 I have conducted a qualitative analysis of how biasing the initial state of a single run of reverse annealing can significantly improve success outcomes. For a specific 8-asset portfolio, I have shown that strongly biasing towards small Hamming distances yields success probabilities three times larger than random guessing, such that repeated repeated runs of reverse annealing can solve the problem faster than a single forward anneal. Even in the case of a weaker bias and a limited tuning of duration parameters, single-run success probabilities are approximately equal to random guessing, but again repeated reverse anneals outperform a single run of forward annealing. This result is certainly non-trivial, as it shows that even without preferentially selecting lower energy states, the local search conducted by reverse annealing explores the solution space quickly enough that repeated runs recover the solution in a time frame which is not exponentially long. By graphing the energy distribution of states I am able to show that a typical real-world problem instance is structured such that states at greater Hamming distances are generally of higher energies, but with a significant amount of overlap. By executing many runs of reverse annealing on a wide range of initial starting states, I show that whilst success outcomes are broadly proportional to decreasing Hamming distance, it is in fact energy which is the strict regulator of success outcomes. The closer in energy a given state is to the ground state, the greater the success probability obtainable from a single run, with a drop-off which appears exponential.

Motivated by the results on the effects of biasing, I have proposed an algorithm which applies reverse annealing iteratively, updating the starting state from the output of each previous run. Fig. 10 demonstrates that the performance of this algorithm is highly dependent on the energy of the initial starting state, but that even at its slowest the ground state is able to be recovered almost twice as quickly as random guessing. However, failed iterations – in which the output state is not of a lower energy – account for the majority of the algorithm's run time. The primary reason for this is the anisotropic energy landscape of problem instances with a high degree of structure; clustering of energy levels leaves energy gaps, between adjacent states, which exceed the range of the local search conducted by reverse annealing. This can result in an arbitrarily small probability of identifying lower energy states in a single run, such that on average many runs are required to proceed, and the algorithm effectively becomes stuck. In this sense, without an intrinsic mechanism for lowering the energy, the algorithm's progress is essentially driven probabilistically. This result is significant in the context of the development of future quantum annealers, and the reduction of noise; any algorithm involving reverse annealing in such a setting must have some mechanism for selecting low energy states built-in by design.

To this end, I have combined iterative reverse annealing with the classical technique of greedy search, to create a hybrid algorithm. For any given starting state the effect of greedy search is to identify lower energy states on which iterative reverse annealing is carried out. The energy range which reverse annealing has to explore in the hybrid algorithm is significantly reduced, such that the ground state was identified up to 3.5 times faster than Algorithm 1, increasing the speed-up over random guessing to approximately 78%. By graphing the greedy search paths I have been able to infer that the energy range of greedy search end-states is much narrower than the total energy range of the problem. This result has significant practical implications because it relaxes the strict dependence of performance on initial state energy, compared to Algorithm 1. However, whilst greedy search reduces the size of the energy region over which iterative reverse annealing has to search, there remains the possibility of the algorithm getting stuck in states around which there are large energy gaps. In this investigation, the value of s_p was a fixed parameter in all instances. However, early proof-of-principle results [23] have suggested that this parameter determines the range of the local search conducted by reverse annealing. It would certainly be worth conducting further research to determine an optimal value of s_p when using iterative reverse annealing. It may well be possible that a more intuitive selection of the pause value will increase the range of the local search, beyond the energy gaps seen in the problem instances discussed here, such that iterative reverse annealing no longer gets stuck. To this end, further investigation is required to determine whether the pitfalls of iterative reverse annealing outlined in this report are general, or an artefact of the specific paramaters chosen.

At several points in this report I have made reference to the semi-random-energy model, and used this as the basis with which to compare the performance of the three algorithms considered. It is likely that there are a vast number of real-life instances which have such a structure. This model represents problems of increasing complexity, with a more homogeneous spacing of energy levels for states at different Hamming distances. The results presented in Table I show that, as the computational complexity of semi-random-energy problem instances increases, greedy search fails more frequently, across a greater number of states at increasing energies. The knock-on effect for Algorithm 2, which also uses reverse annealing, is that an increasing number of total iterations are required to solve the problem. Interestingly however, a consideration of the arguments put forth in section 8 suggests that the reverse annealing is also less likely to get stuck in large energy gaps, owing to the increasingly isotropic energy distribution. Algorithm 2 is likely therefore to be slower, but more reliable. Testing this hypothesis would be a particularly interesting direction in which to pursue further investigation.

Finally, the results presented in section 11 show that, even with a rudimentary selection of parameters, an algorithm which uses biased forward annealing to provide the quantum updates can yield yet further speed-up compared to reverse annealing. Most critically however, the results of Fig. 16 demonstrate that biased forward annealing is far less likely to get stuck in a state around which there are large energy gaps. Indeed, for a long enough single run time the probability of recovering the ground state tends to unity even for an arbitrarily poor selection of trial solution. In this sense, biased forward annealing provides a much more robust technique in systems with little or no dissipation, when incorporated into a hybrid algorithm with greedy search. Further investigation of biased forward annealing parameters also sets a broad scope for future work. In the first instance it would be beneficial to compare algorithms which employ a large number of shorter anneals, compared to those which use a smaller number of long anneals. It may further be possible to construct more sophisticated algorithms in which the guess certainty and run time are also updated iteratively. A schedule could be imagined which begins with short anneals with low guess certainty, before moving to longer anneals with higher guess certainty after the best-known classically-obtainable solution is recovered, for example.

Whether or not more imaginative schedules are formulated, further research is required, particularly on larger problem instances, to establish the extent to which the phenomena observed here are general characteristics of reverse annealing, and the resulting algorithms constructed. In particular, establishing how the performance of Algorithms 1 and 2 scales with problem size will be a crucial step in determining whether or not reverse annealing can truly pave the way for successful optimisation in closed system problems.

References

[1] Davide Venturelli and Alexei Kondratyev. Reverse quantum annealing approach to portfolio optimization problems. Quantum Machine Intelligence, 1(1):17–30, May 2019. doi: 10.1007/s42484-019-00001-w.

[2] Vicky Choi. Adiabatic quantum algorithms for the NP-complete Maximum-Weight Independent set, Exact Cover and 3SAT problems, 2010.

[3] Vicky Choi. Different Adiabatic Quantum Optimization Algorithms for the NP-Complete Exact Cover and 3SAT Problems. Quantum Information and Computation, Vol. 11, No. 7&8 (2011) 0638-0648, 2011

[4] T. Stollenwerk, et. al, Quantum annealing applied to de-conflicting optimal trajectories for air traffic management. IEEE Transactions on Intelligent Transportation Systems, 21(1):285–297, 2020. doi: 10.1109/TITS.2019.2891235.

[5] Tony T. Tran, et. al, A hybrid quantum-classical approach to solving scheduling problems. In SOCS, 2016.

[6] Alejandro Perdomo-Ortiz, Neil Dickson, Marshall Drew-Brook, Geordie Rose, and Alán Aspuru-Guzik. Finding low-energy conformations of lattice protein models by quantum annealing. Scien-

tific Reports, 2(1):571, 2012. ISSN 2045-2322.

[7] Mishra, Anurag et. al, Finite temperature quantum annealing solving exponentially small gap problem with non-monotonic success probability. (2018) Nature, Comm. 9. 10.1038

[8] Nicholas Chancellor. Modernizing quantum annealing using local searches. New Journal of Physics, 19(2):023024, 2017.

[9] Nicholas Chancellor and Viv Kendon, Search range in experimental quantum annealing . Phys. Rev. A 104, 012604 (2021)

[10] N. G. Dickson et al. Thermally assisted quantum annealing of a 16-qubit problem. Nature Communications, 4(1):1903 2013. ISSN 2041-1723.

[11] Tadashi Kadowaki and Hidetoshi Nishimori, Quantum annealing in the transverse Ising model, Phys. Rev. E 58, 5355–5363 (1998).

[12] E. Farhi, et. al, Quantum computation by adiabatic evolution. arXiv preprint quant-ph/0001106. 2000 Jan 28.

[13] Jeffrey Marshall, Davide Venturelli, Itay Hen, Eleanor G., Power of Pausing: Advancing Understanding of Thermalization in Experimental Quantum Annealers., Phys. Rev. Applied 11, 044083 (2019)

[14] O. Goldreich,et. al, "P, NP, and NP-Completeness: The Basics of Computational Complexity", Cambridge University Press,2010.

[15] Harry Markowitz. Portfolio selection. The journal of finance, 7(1):77–91, 1952.

[16] https://pydata.github.io/pandas-datareader/. Accessed 05/11/2021.

[17] Travis E Oliphant. A guide to NumPy, volume 1. Trelgol Publishing USA, 2006.

[18] Thomas H. Cormen et. al,Introduction To Algorithms (2001). MIT Press. pp. 370–. ISBN 978-0-262-03293-3.

[19] Tobias Graß. Quantum annealing with longitudinal bias fields. Phys. Rev. Lett., 123:120501, Sep 2019. doi: 10.1103/ PhysRevLett.123.120501.

[20] Adam Callison, Max Festenstein, Jie Chen, Laurentiu Nita, Viv Kendon, and Nicholas Chancellor. An energetic perspective on rapid quenches in quantum annealing. ariv:2007.11599, 2020.

[21] Erica Grant, Travis S. Humble, and Benjamin Stump, Benchmarking Quantum Annealing Controls with Portfolio Optimization Phys. Rev. Applied , Vol. 15, Iss. 1, January 2021

[22] Michael Marzec. Portfolio optimization: applications in quantum computing. Handbook of High-Frequency Trading and Modeling in Finance (John Wiley Sons, Inc., 2016) pp, pages 73–106, 2016.

[23] D-Wave White paper, Reverse Quantum Annealing for Local Refinement of Solutions, available at https://www.dwavesys.com/resources/white-paper/, Accessed 21/02/2022

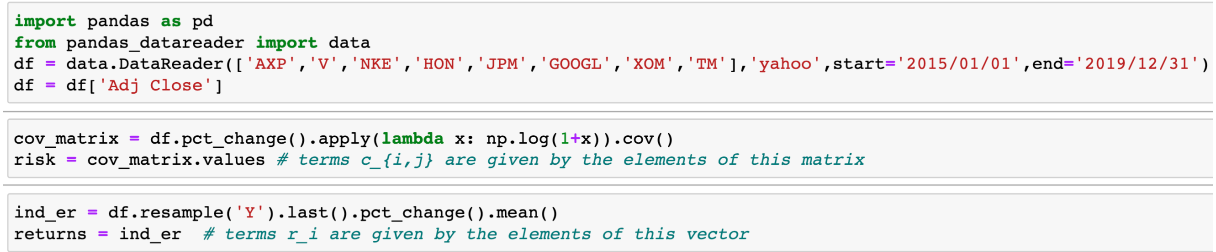
[24] E. J. Crosson and D. A. Lidar. Prospects for quantum enhancement with diabatic quantum annealing, 2020. arXiv preprint quant-ph/2008.09913.

Appendix A: Problem Instances

A summary of the problem instances considered in this investigation is shown below in Table II. To construct a given problem instance I have used the *"Datareader"* function, available within the Pandas library in Python3. This function is able to extract asset price data from Yahoo Finance, across a specified time period. In the interests of reproducibility I have included the relevant section of code for constructing the problem instance, below in Fig. 18.

Problem Instance	Size	Possible Assets	Weightings	η_{rand}
1	5	AXP, V, HON, GOOGL, TM	$\theta_1 = 2, \theta_2 = 1000$	0
2	8	AXP, V, NKE, HON, JPM, GOOGL, XOM, TM	$\theta_1 = 11, \theta_2 = 3000$	0
3	8	AXP, V, NKE, HON, JPM, GOOGL, XOM, TM	$\theta_1 = 11, \theta_2 = 3000$	75
4	8	AXP, V, NKE, HON, JPM, GOOGL, XOM, TM	$\theta_1 = 11, \theta_2 = 3000$	150
5	8	AXP, V, NKE, HON, JPM, GOOGL, XOM, TM	$\theta_1 = 11, \theta_2 = 3000$	225

TABLE II: A summary of the problem instances considered in this investigation. As well as listing the Ticker (stock symbol) of the assets considered in each instance, the weightings of returns and risk are also given in column four. Instances 3, 4, and 5 correspond to the semi-random energy model described in Section 9, Table I.



```
import pandas as pd
from pandas_datareader import data
df = data.DataReader(['AXP','V','NKE','HON','JPM','GOOGL','XOM','TM'],'yahoo',start='2015/01/01',end='2019/12/31')
df = df['Adj Close']
```

```
cov_matrix = df.pct_change().apply(lambda x: np.log(1+x)).cov()
risk = cov_matrix.values # terms c_{i,j} are given by the elements of this matrix
```

```
ind_er = df.resample('Y').last().pct_change().mean()
returns = ind_er  # terms r_i are given by the elements of this vector
```

FIG. 18: An example of the code which is used to construct each problem instance. The Datareader function, available within the Pandas library in Python3, is used to extract historical price data from Yahoo Finance.

Appendix B: Errors and Uncertainty

In sections 6 and 7, all of the probabilities quoted are those which have been obtained by a single run of reverse annealing. The largest source of uncertainty in these results will arise from the way in which time-evolution is simulated across the anneal, given by Eq. 12 and repeated here for clarity:

Under this formulation, error arises from the fact that q is taken to be large but strictly finite, and continuous evolution is therefore being approximated by a discrete product. As the total run time for a given anneal increases, the size of q must be increased also, although need not necessarily scale linearly with run length. In order to mitigate for the uncertainty introduced by discrete approximation, I have set q at all times to be large enough such that increasing it further does not produce a change in the success probability up to the fourth decimal place. Since all values are quoted to 3 decimal places (as is standard practice for research works in this field), this ensures that all of the uncertainty which arises due to discretization is included in the precision to which values are quoted.

In sections 8, 9 and 10, the speed of the algorithms is quantified by the average number of iterations required to reach the ground state. The uncertainty on these quantities is quantified by taking many repeats, and obtaining a range of iteration lengths; the uncertainty on the average speed is then quoted as half of the range.