Genetic Algorithms Assisted Adiabatic Quantum Computing

by

Pratibha Raghupati Hegde

Submitted in Partial Fulfillment of the

Requirements for the Degree

Doctor of Philosophy

"Ettore Pancini" Department of Physics

University of Napoli Federico II

Napoli, Italy

2022

Acknowledgments

I am thankful to "Ettore Pancini" Department of Physics, University of Napoli Federico II for giving me an opportunity to pursue Ph.D. I offer my sincere gratitude to my advisor Dr. Procolo Lucignano for training me to do research as well as for being very kind and supportive during my PhD. I am thankful to my advisor Prof. Vittorio Cataudella for his mentorship and support. I thank my colleague Dr. Gianluca Passarelli who has been very kind in teaching me various research techniques and for being always available to discussions and moral support. I thank my collaborators Prof. Giovanni Acampora and Dr. Autilia Vitiello for their valuable guidance in computational intelligence and machine learning techniques. I thank my colleague Ms. Annarita Scocco for her useful collaboration.

I am very thankful to Mr. Guido Celentano for helping me to settle in Naples by helping in bureaucracy related works and for being always there in innumerable occasions. I thank Pegah, Zahra, Manjot and all other Ph.D colleagues and friends for their moral support and who have made these three years of Ph.D memorable. I

am thankful to the city of Naples, where the sea, the sun and the warm people have kept me peaceful during these academically rigorous years.

I am thankful to Prof. Swarnamala Sirsi because of whom I got introduced to research in Quantum Physics. She has supported me both morally and academically through all these years and whom I look up to as a woman in science. I am thankful to Prof. Anil Shaji from whom I have learnt a lot of Physics during my summer project and Master's thesis and for guiding me to obtain a Ph.D position outside the country. I thank Dr. Aditi Sen(De) for being an inspiring woman in science and guiding me. I thank my Prof. Rajeev Joshi, Prof. Deepak Samuel, and Prof. Bharat Kumar who have taught me and guided me during my Master's programme. I thank Subramanya Hegde who has taught me a lot of Physics and motivated me to continue in Physics during my undergraduate days. I thank my friends Nidhi, Sahamatha, Dinesh, Guru, Raghunandan who have been always there for me back in India and in Italy a phone call away, especially during the pandemic period.

Lastly and most importantly, I thank my parents Daxayani Hegde and Raghupati Hegde for providing me education, comfort and a safe place to rely upon. I thank my sister Chandraprabha for always encouraging me and accompanying me in uncountable travels for academic reasons throughout India. I thank my partner Sreenath for being there always and motivating me everyday to continue what I am doing.

Abstract

In the current Noisy Intermediate Scale Quantum (NISQ) era, quantum computational resources can be utilized efficiently by optimizing them with suitable classical algorithms. In the same spirit, this thesis addresses the relevant problems in the paradigm of adiabatic quantum computation and in turn quantum annealing. We mainly resort to heuristic optimization techniques of evolutionary algorithms as a numerical tool and demonstrate their effectiveness in finding the solutions to the problems considered.

The first problem we focus on is to find an equivalent 2-body interactions of a p-body Hamiltonian as it is a necessity for embedding optimization problems with p-local interactions in the current quantum annealer hardware architectures. We use genetic algorithms to optimize a function which minimizes the energy difference between the lower spectrum of the original and the mapped 2-body Hamiltonian. We consider the two analytically solvable cases of a ferromagnetic p-spin model to discuss our results. We also show further improvements by implementing memetic

algorithms which enforces additional local searches.

As the second problem of this thesis, we propose an effective approach to shortcuts to adiabaticity using a numerical approach based on genetic algorithms. The hard optimization problems often have small spectral gaps which make the system to undergo diabatic transitions in the finite time quantum annealing. In this thesis, we tackle this problem by engineering the annealing schedules starting from the polynomial ansatz by treating their coefficients as chromosomes of a genetic algorithm. We also explore shortcuts to adiabaticity by computing a practically feasible k-local optimal driving operator, showing that even for k = 1 we achieve substantial improvement of the fidelity over the standard annealing solution. With these genetically optimized annealing schedules and/or optimal driving operators, we are able to perform quantum annealing in relatively short time-scales and with higher fidelity compared to traditional approaches.

Contents

List of Tables

List of Figures

1. Introduction

1.1 Introduction to quantum computation

The formulation of quantum mechanics in the early 1900's changed the way we perceive the world. Rigorous theories have been proposed ever since to understand the dynamics of the systems which are small enough to be in quantum mechanical regime. The implications of quantum mechanics such as quantization of energy levels, quantum entanglement, quantum superposition, the possibility of tunneling the system through a large potential and other implications of quantum mechanics became the backbone of quantum computation (Steane 1998; Preskill 2018). Quantum mechanical systems can have a high degree of correlation among their components which cannot be created by local operations and classical communications (LOCC) (Vedral and Plenio 1998), which play a key role in formulating quantum algorithms. Quantum systems also provide exponentially large memory units to store information when compared with classical systems. Thanks to experimental advances, quantum computation is very relevant and highly researched now than ever before. With the parallel advancements in the computer science,

especially in optimization tools and machine learning, quantum computation is well aided to show promising results even with the preliminary quantum hardware resources available (Arute et al. 2019; Y. Wu et al. 2021).

Quantum computation aims at building devices which with suitable quantum algorithms can solve hard classical problems faster which otherwise would have taken a ridiculously long time using classical computers. In addition, quantum computers in principle can simulate quantum systems, which opens the door to new possible physics especially to better understanding of many-body systems (Feynman 1982; A. Smith et al. 2019). Further, the proposed theories have also indicated in the substantial technological advances in terms of secure communications harnessing the quantum physical nature of substances, all of which are studied under the name of quantum technologies (Dowling and Milburn 2003). These reasons make it worthwhile for the scientific community to invest themselves in research in the field of quantum computation.

So far, there have been multiple constructions of frameworks to build a quantum computer (Nielsen and Chuang 2002). This thesis mainly addresses adiabatic quantum computation. However, in the following part of this section, we shall also briefly review the advancements and applications of quantum computation in a broader picture. The supposedly most versatile approach of quantum computation is the standard model quantum computation or gate model quantum computation. In this paradigm of quantum computation, there is a system of particles or "qubits" which are initiated in a particular state and are subjected to discrete unitary evolutions implemented by quantum gates until the qubits evolve to the final states

which are subsequently measured. At present, most quantum initiatives that are involved in building quantum computers use this model of quantum computation. Further, the qubits can be prepared using Josephson junction of superconducting systems (Shnirman, Schön, and Hermon 1997), polarisation states of light (James et al. 2001), quantum dots (Loss and DiVincenzo 1998; Trauzettel et al. 2007) and other systems. In general, qubits can be constructed using quantum systems for which two distinct states or their superposition state can be defined and can be controlled to be in one of the states by subjecting to an external perturbation. Be it in any kind of quantum computational model, it is important to find suitable algorithms, which are enhanced in their performance due to quantum physical implications. To claim a quantum speedup in any quantum algorithm, one should prove that the given quantum algorithm can solve a given problem faster than any existing classical algorithm. The least criteria to claim quantum speedup in a quantum algorithm could be to prove that the algorithm outperforms a similar approach, but implemented on a classical hardware (Rønnow et al. 2014). There are multiple quantum algorithms formulated for circuit model quantum computation that outperform classical algorithms in time to solution. For example, Grover's search algorithm which aims at finding a particular element in an unsorted data set of size N can successfully solve the problem in $\sqrt{N}$ iterations, as opposed to the classical algorithm which requires at least N number of queries (Lov K Grover 1996). Shor's algorithm (Shor 1997), and Deutsch–Jozsa algorithm (Deutsch and Jozsa 1992) are some of the other famous examples to mention.

Adiabatic Quantum Computation (AQC) is an approach aimed mainly at solv-

ing optimization problems (Farhi, Goldstone, Gutmann, and Sipser 2000; Farhi, Goldstone, Gutmann, Lapan, et al. 2001; Dam, Mosca, and Vazirani 2001; Albash and Daniel A. Lidar 2018; Aharonov et al. 2008; Hogg 2003; McGeoch 2014; G. E. Santoro et al. 2002). The basis of AQC is the adiabatic theorem of quantum mechanics. The adiabatic theorem of quantum mechanics deals with the slow evolution of a quantum mechanical system driven by a given Hamiltonian (Kato 1950; Nenciu 1980). It hints at keeping the system in the given eigenstate throughout the evolution if the external perturbations are slow enough. Further, it also gives a quantitative description of how “slow" the process should be for a given system in order to avoid excitations. The system is initially in an easy to prepare ground state of a Hamiltonian. The solution to the optimization problem is encoded in the ground state of a final Hamiltonian. The system is evolved slowly from the ground state of an initial Hamiltonian and according to the adiabatic theorem, if this process is slow enough one can hope to find the system in the ground state of the final Hamiltonian with a high probability. In this paradigm of computation, the study of adiabatic theorem, essentially the adiabatic time scales in which adiabatic quantum computation can be achieved becomes important. These timescales depend on the instantaneous energy spectra of the system driven by the time-dependent Hamiltonian according to the Schrödinger equation. Further, adiabatic quantum computation and standard model quantum computation can simulate each other as shown in (Aharonov et al. 2008). The quantum algorithms like Grover’s search, Deutsch-Jozsa algorithm, Bernstein-Vazirani algorithm are formulated in the paradigm of adiabatic quantum computation (Albash and Daniel

A. Lidar 2018).

During the course of adiabatic evolution, the instantaneous eigen spectrum of the Hamiltonian can show energy crossing or avoided crossing, at a particular value of induced quantum fluctuations, and at this point, the system which is prepared initially in the ground state can excite to the next level or in other words the system would go through quantum phase transition in the thermodynamic limit (Sachdev 2011). The second-order phase transitions are characterised by polynomial closing of energy gaps, while first-order phase transitions are characterized by exponential closing of energy gaps. This is reflected in the adiabatic time scales as well, with systems showing first order phase transition is harder to solve than the systems showing second order phase transition.

In AQC, the final Hamiltonian is usually classical, with the ceasing of transverse fields at the final time. For example, in the case of the ferromagnetic Ising model, at the final time, the transverse fields are absent and hence they do not have any flips in spins. The ground state corresponds to the case of all spins up or all spins down. By switching on the transverse field for a short time, the spins can be flipped to one of these degenerate states and in the absence of transverse field, the system acquires one of the ground states breaking $\mathscr{Z}_2$ symmetry. When studying adiabatic quantum computation, it is crucial to analyze its performance for different models (the final Hamiltonian) so that the technique can be benchmarked for a broader class of optimization problems. In this work, we have mainly considered the model of fully connected ferromagnetic p-spin model as the case study of adiabatic quantum computation. More details of this model are given in the further section of this

chapter.

1.2 Quantum Annealing

An experimental realization of adiabatic quantum computation is quantum annealing (QA) (G. Santoro and Tosatti 2006; Bian et al. 2020; Kadowaki and Nishimori 1998; Jörg et al. 2010; Finnila et al. 1994; P. Hauke et al. 2019; Seoane and Nishimori 2012; Seki and Nishimori 2012; Ohkuwa, Nishimori, and Daniel A. Lidar 2018a; Matsuura, Nishimori, Vinci, Albash, et al. 2017; Susa, Yamashiro, et al. 2018; Matsuura, Nishimori, Vinci, and Daniel A. Lidar 2019; Yamashiro et al. 2019; Inack and Pilati 2015). Quantum annealing is a process wherein a quantum mechanical system, initially a large transverse field is applied such that it is in a superposition of all states possible and eventually this transverse field is ceased slowly and according to the adiabatic theorem we can hope to find the system in the lowest energy configuration at the end of evolution. This is a protocol formulated analogously to thermal annealing or simulated annealing (SA). In SA, the system is subjected to thermal fluctuations in a slow manner (Laarhoven and Aarts 1987). In other words, the system is cooled down starting from a high temperature slowly such that at every instant of time it is in thermal equilibrium and at the lowest temperature the system is found in the lowest energy state of the energy cost function. SA finds a setback when the energy cost function contour has local minimums and the algorithm is trapped in this local minimum for a long time subsequently leading to sub-optimal solutions (Farhi, Goldstone, and Gutmann 2002). QA, on

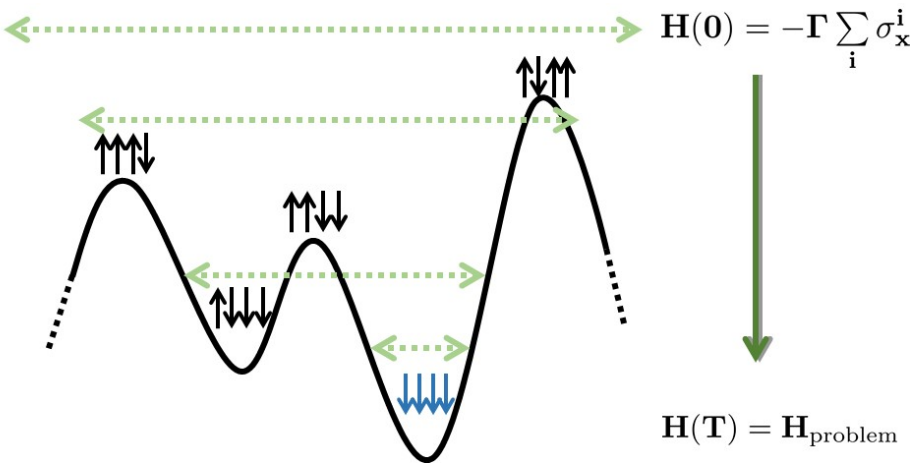


Figure 1.1: Process of quantum annealing. The cartoon represents the energy contour of a problem Hamiltonian, whose ground state corresponds to all spins down. The system is prepared in an initial Hamiltonian whose ground state is the superposition of all spin configurations possible, and it can traverse between these states by the process of tunneling because of the induced quantum fluctuations Γ. As this fluctuation is ceased slowly, the system follows the instantaneous ground state of the time-dependent Hamiltonian. Finally, at the annealing time T, it is in the configuration of all spins down which is the ground state of the final Hamiltonian.

the other hand, benefits from quantum tunneling, where the system can tunnel between the states separated by a large potential and essentially speeding up the optimization (Kadowaki and Nishimori 1998). The study of time scales and theoretical study of quantum annealing can be done using adiabatic theorem (Messiah 1962). The process of quantum annealing can be modelled using a Hamiltonian of the form

$$H(t) = A(t)H_x + B(t)H_z, \tag{1.1}$$

where usually $H_x = -\Gamma \sum_i \sigma_x^i$, whose ground state is easy to prepare. Here, Γ is the tunneling strength. H_z is the problem many body Hamiltonian. $\sigma_{x,y,z}^i$ are the

Pauli spin operators acting on the i^{th} spin in the respective direction. $A(t)$ and $B(t)$ are annealing schedules which satisfy the boundary conditions $A(0) \gg 1, B(0) = 0$ and $A(T) = 0$ and $B(T) \gg 1$. T is the annealing time. The annealing time highly depends on the properties of the problem Hamiltonian. The adiabatic theorem gives the quantitative bound on the annealing time,

$$T_{\mathrm{AD}} = \max_{\lambda \in [0,1]} \frac{|\langle E_0(\lambda)|\partial_\lambda H(\lambda)|E_1(\lambda)\rangle|}{|E_1(\lambda) - E_0(\lambda)|^2}, \qquad \lambda = t/T. \tag{1.2}$$

The derivation of the above equation is given in Messiah 1962. The efficiency of the quantum annealing is computed using the fidelity of the final state with the exact groundstate probability of the problem Hamiltonian H_z, i. e.,

$$\text{Fidelity} = |\langle \psi_0 | E_0(T) \rangle|^2. \tag{1.3}$$

It is to be noted that when solving the Hamiltonians which have degenerate ground states by quantum annealing, the total ground state probability has to be considered. In fact, it is shown in Matsuda, Nishimori, and Katzgraber 2009 that quantum annealing can not always lead to all the ground states with equal probabilities. Alternatively, one can also compute residual energy which is the difference between the ground state energy obtained by quantum annealing and the true ground state energy (Suzuki and M. Okada 2005).

Quantum annealing finds applications in solving a number of combinatorial optimization problems including some of the NP-hard (non-deterministic polynomial-time hardness) problems designed in computational complexity theories such as

Boolean satisfiability Problems (SAT problems) (Bian et al. 2020), traveling salesman problem (H. Chen et al. 2011), max-cut problem (Qiu, Zoller, and X. Li 2020) etc. We come across these kinds of combinatorial optimization problems in many different fields such as studying DNA sequencing (Boev et al. 2021), protein folding (Babbush et al. 2014), machine learning (Adachi and Henderson 2015), so on and so forth (Ikeda, Nakamura, and Humble 2019).

There are a number of problems to be addressed in the field of quantum annealing in order to improve the final solution obtained. An important one among them is to address small energy gaps of the systems during the evolution which can lead the system to transition to a higher level. The small energy gaps can be modeled using Landau-Zener problem (Kadowaki and Nishimori 1998), where the diabatic transitions occur with a probability $e^{\frac{-\pi|b|^2}{\hbar a}}$, where b is proportional to the energy gap at the avoided crossing (in the adiabatic evolution) and a is a constant. This has led to the research area of transitionless driving by counterdiabatic driving (Del Campo 2013). This involves adding an additional potential that compensates for the small gaps in the instantaneous energy spectrum sufficiently during the evolution. More discussion on the techniques of counterdiabatic driving is done later in this chapter. Moreover, small energy gaps mean a long annealing time as predicted by the adiabatic theorem. But at the moment we cannot afford to protect qubits from dissipation for a longer time. Therefore, it is essential to find an alternative fast path of evolution for the annealing process at the same time being adiabatic. In short, these techniques fall under the category of optimal control and shortcuts to adiabaticity. Further, one should bear in mind the practical

feasibility of the techniques introduced. For example, it is desirable that the counterdiabatic potential introduced is local and can be easily implementable on a quantum hardware architecture. Another technique to design an efficient quantum annealing is to monitor the pace of quantum evolution. It may not be essential to maintain a monotonous annealing schedule in quantum annealing. By finding optimal annealing schedules, the performance of the quantum annealing can be improved drastically. In fact, in Roland and Cerf 2002, the authors have shown that by resorting to non monotonous annealing schedules computed by implementing local adiabaticity condition, one can obtain a quadratic speedup in adibatic search problem (equivalent to Grover's search problem in gate model computation). This is a remarkable result in showing quantum speedup using adiabatic quantum computation or quantum annealing.

There are also several other formalisms and techniques which deviate from the traditional protocol of quantum annealing in order to improve its efficiency. One of them is applying inhomogeneous transverse field in the Hilbert space, which modulate the interactions, especially at the point small energy gaps to avoid transitions to higher energy level. However, it is difficult to control the field inhomogenously at the moment from the practical point of view (Philipp Hauke et al. 2019). Another approach is reverse annealing. In this strategy, the system is initialized in a particular state usually close to solution state instead of a standard approach of starting from a superposition state (Ohkuwa, Nishimori, and Daniel A. Lidar 2018b; Gianluca Passarelli, Yip, et al. 2020; Philipp Hauke et al. 2019). Such an initial state can be found by a classical simulation. The system is prepared in

this candidate state with zero transverse field and the transverse field is increased and decreased gradually, so that system searches for ground state configuration in a smaller search space close to the candidate solution. The standard approach of quantum annealing is implemented on stoquastic Hamiltonians only. Stoquastic Hamiltonians are those where the non-diagonal elements are only non-positive. There are some efforts in implementing quantum annealing in non-stoquastic Hamiltonians (Nishimori and Takada 2017; Philipp Hauke et al. 2019). It is also shown that non-stoquastic quantum annealing can simulate universal model of quantum computation (Jacob D. Biamonte and Love 2008).

In this thesis, the simulation of adiabatic quantum computation or quantum annealing is done using QuTip python library (Johansson, Nation, and Nori 2012; Johansson, Nation, and Nori 2013). Specifically, we use "mesolve" function which simulates master equation evolution for a given list of discreet time points. By setting the collapse operators variable "c_ops" to 0, this function simulates unitary dynamics generated by the given Hamiltonian. The Hamiltonian operators with their respective time dependent coefficients can be passed as a nested operator. The expectation values of the observables on the evolved state can also be obtained by defining the observable as the variable "e_ops". The typical computational time to simulate the unitary dynamics driven by the Hamiltonian in Eq. 1.1 with H_z being the ferromagnetic p-spin Hamiltonian, with 15 spins for a final time $T = 3$, sampled at 100 points during evolution is approximately equal to 0.03s

The experimental test bed for the results obtained in this thesis can be offered by D-Wave systems quantum annealer (O'Malley et al. 2018; Harris et al. 2011). In

the most recent D-Wave systems *Advantage* quantum annealer, there are more than 5000 superconducting flux qubits with 1,000,000 Josephson junctions (Orlando et al. 1999). In each qubit, the basis states are defined by the clock-wise and the anti-clockwise current in the superconducting current loop. The qubits can be controlled to be in one of these states by applying microwave pulse with the energy same as that of the energy difference between these two states of the flux qubit. Further, the qubits are coupled pairwise (but not all-to-all connected) via switchable coupling loops. There are more than 35,000 of such couplers in the present hardware of D-Wave. These Quantum Processing Units (QPUs) can be accessed using the open-source cloud service, *Leap*. Programming in the D-Wave can be done using python based software *Ocean*. This thesis focuses on providing useful classical-quantum hybrid algorithms, specifically combining classical genetic algorithms with quantum annealing. The *D-Wave Hybrid* offers a platform to contribute and run our own hybrid algorithms on their computing systems which can provide experimental test-bed to verify our results.

1.3 Ferromagnetic p-spin model

Ferromagnetic p-spin model is widely studied in the context of quantum annealing (Bapst and Semerjian 2012; Gross and Mezard 1984; Seoane and Nishimori 2012; Seki and Nishimori 2012; Ohkuwa, Nishimori, and Daniel A. Lidar 2018a; Matsuura, Nishimori, Vinci, Albash, et al. 2017; Susa, Yamashiro, et al. 2018).

The Hamiltonian of this model composed of N spins is,

$$H_p = -JN\left(\frac{1}{N}\sum_i^N \sigma_i^z\right)^p, \tag{1.4}$$

where $J > 0$, is the coupling strength. In the finite time adiabatic quantum computation, this model shows quantum phase transitions. For $p = 2$, the phase transition is of second order and for $p \geq 3$, the phase transition is of first order in the thermodynamic limit. The detailed derivation of thermodynamic properties and phase transitions of the model is given in Bapst and Semerjian 2012.

Furthermore, the ferromagnetic p-spin model with even p exhibits time reversal symmetry or Z_2 symmetry and hence the ground state is 2-fold degenerate(all spins up and all spin down) and for odd p, the ground state corresponds to the case where all spins are up. For odd p, there is only one state with lowest energy among 2^N states and hence forms a harder problem. It can be shown that in the thermodynamic limit, for odd p and $p \to \infty$, the ferromagnetic p-spin model corresponds to Grover's search algorithm (L. K. Grover 1996; Jörg et al. 2010).

The Hamiltonian of this model commutes with total spin operator S^2, and therefore this model can be effectively represented in the total spin bases of S operator. Therefore, the dimension of Hilbert space to describe a system of N spins system reduces from 2^N to $N+1$. This offers a huge advantage in simulating the system using classical machines. The procedure of basis change is described in appendix B.

In this thesis, we have studied the quantum annealing of ferromagnetic p-spin model in two contexts. First, to demonstrate mapping of many-body Hamiltonian

with p-local interaction into a Hamiltonian with 2-local interactions so that the optimization problems can be solved on quantum annealer architecture. Second, we propose a method of shortcuts to adiabaticity using the same model by engineering the annealing schedules and by introducing an optimal driving operator using genetic algorithms.

1.4 Embedding optimization problems into quantum annealer architectures

At present, D-Wave systems has built quantum annealer based on superconducting qubits and are commercially available (Harris et al. 2011). In a D-Wave quantum annealer, the qubits are arranged in chimera graphs or more recently Pegasus graphs with 5000 qubits available for computation. Therefore, from the practical point of view, all the combinatorial optimization problems have to be translated into a form which can be embedded into these architectures. In particular, these architectures allow at most 2-local interactions with sparse connectivity, with Pegasus graph having better connectivity than that of Chimera architecture (S. Okada et al. 2019). These machines can solve quadratic unconstrained binary optimization (QUBO) problems, which can be modelled as Ising model interaction among the qubits (Date et al. 2019; Philipp Hauke et al. 2019). Therefore, it is an important topic of research to map a given optimization problems or the Hamiltonians of whose ground state is to be sought into Ising like Hamiltonian. This requires two crucial steps. One is to find an equivalent system with new

interactions, but they simulate the same lower energy spectrum as the original system. This requires to introduce some additional degrees of freedom into the new system and which when introduced satisfy a suitable penalty condition so that they do not alter the lower spectral properties of the new system when compared to our problem Hamiltonian (Choi 2011; Choi 2008; J. D. Biamonte 2008). In this thesis, we have attempted to solve this problem resorting to heuristic methods of evolutionary algorithms. Second step of mapping the problem Hamiltonian is to find a graphical embedding scheme which satisfy the interactions of the new Hamiltonian and can be implementable in the presently available quantum annealer hardware. In addition to representing the 2-local interactions between the qubits, the interactions between the physical qubits which represent the same logical qubits have to be comparatively very large.

In this thesis, the first problem that we have addressed in the paradigm of quantum annealing is to reduce a k-local Hamiltonian into a 2-local Hamiltonian, in particular to solve the parameter setting problem. This problem facilitates minor embedding process by finding the strengths of interactions between the qubits. A k-local Hamiltonian is one, where each term of the Hamiltonian acts on at most k qubits. The motive of our research is to find a new equivalent interaction among the qubits where at most 2 qubits interact with each other, at the same time have their eigen spectrum closely resembling the system with original k-local interactions. In these kinds of problems the qubits or spins in the original system are referred to as logical qubits and the qubits in the new equivalent system are referred to as physical qubits. The algebraic reduction of such k-local Hamiltonians into 2-local Hamiltonians

has been done in J. D. Biamonte 2008 by the introduction of auxiliary degrees of freedom or introducing additional ancilla qubits. The new 2-local Hamiltonians are supposed to represent the original problem by having the same lower energy spectrum of the original k-local Hamiltonian. This constraint is taken care of by introducing a Boolean expression that the ancilla qubits have to satisfy. In this work, we have considered AND embedding, where each ancilla introduced has to satisfy an AND condition between a pair of logical states. All possible configurations of the physical qubits are generated. A penalty function is used to make sure if each configuration satisfies the AND embedding rule and whenever a configuration fails to satisfy this condition, a large penalty is added to the energy of this configuration of states of the mapped system, such that they are separated from the lower energy spectrum. A choice of penalty function for such AND embedding is,

$$E_{pen} = \delta(3\tilde{x_3} + x_1x_2 - 2x_1\tilde{x_3} - 2x_2\tilde{x_3}), \tag{1.5}$$

where $\tilde{x_3}$ is the ancilla qubit. x_1 and x_2 are the logical qubits. By choosing a large value for δ, the unphysical qubit configurations can be assigned a large energy so that they are separated form the lower energy spectrum.

To be precise, if the original system contains N qubits, we match the lower 2^N states of the mapped 2-local Hamiltonian with the eigen spectrum of the original system. We formulate an optimization problem, which minimizes the differences between the corresponding eigen energies between the two Hamiltonians.

Another strategy of embedding technique which has gained recent attention is

Lechner-Hauke-Zoller (LHZ) Scheme (Leib, Zoller, and Lechner 2016; Hartmann and Lechner 2019). In this case, the physical qubits represent the interaction between two logical qubits, rather than the logical qubits themselves. There have also been developments in finding an effective LHZ embedding for counterdiabatic potentials.

A specimen model to represent the many body interaction is the ferromagnetic p-spin model. We have considered the same model to find an equivalent mapping to it by using computational intelligence technique of evolutionary algorithms. Evolutionary algorithms are heuristic approaches inspired by the evolution process of living organisms. The detailed overview and comments on these evolutionary algorithms are given later in this chapter.

1.5 Shortcuts to adiabaticity

The main hurdle of adiabatic quantum computation and in turn, quantum annealing is small energy gaps in the instantaneous eigen spectrum of the driving Hamiltonian. A common and the most researched way of tackling this problem is to add a suitable counterdiabatic potential to the driving Hamiltonian which overcomes the transition to higher energy state during the path of evolution (Del Campo 2013; Funo et al. 2017). Also, it is of practical interest to find evolution paths which are faster than the decoherence of the qubits by controlling the paramaters of the quantum annealing or to find shortcuts to adiabaticity (Berry 2009; Del Campo 2013; Torrontegui et al. 2013; Campbell and Deffner 2017; Guéry-Odelin et al.

2019; Campbell, De Chiara, et al. 2015; Mukherjee, Montangero, and Rosario Fazio 2016; G. Passarelli, V. Cataudella, R. Fazio, et al. 2020; Abah and Lutz 2018; Alan C. Santos and Marcelo S. Sarandy 2015; Coulamy et al. 2016; Alan C Santos and Marcelo S Sarandy 2017; Hu et al. 2018; Alan C. Santos, Nicotina, et al. 2020). These techniques are collectively studied under the names of transitionless driving, counterdiabatic driving, short cuts to adiabaticity(STA), quantum optimal control and others.

The exact solution for a counterdiabatic potential was given by M. V. Berry in Berry 2009. He derived a Hamiltonian which leads to transitionless driving using reverse engineering technique. Which means, given a Hamiltonian $H_0(t)$, a new time dependent Hamiltonian $H(t)$ is found such that the instantaneous spectrum shows no transitions. The derivation lead to the counterdiabatic potential of the form,

$$H(t) = H_0(t) + H_{\mathrm{cd}}(t), \tag{1.6}$$

where,

$$H_{\mathrm{cd}}(t) = \mathrm{i}\hbar \sum_{m \neq n} \sum \frac{|m\rangle \langle m| \partial_t H_0 |n\rangle \langle n|}{E_n - E_m}. \tag{1.7}$$

This solution of transitionless driving, albeit exact, is problematic at the point of level crossing or avoided level crossing. At these points, the small denominator, leads to a large potential, not possibly implementable. In addition, the counter-diabatic potential H_{cd} in Eq. 1.6 requires the exact diagonalization of the driving Hamiltonian. In principle, the motive of the quantum annealing is to find the ground state of the Hamiltonian, therefore in general we do not have access to the

knowledge of instantaneous eigen spectrum of the driving Hamiltonian.

A variational approach to find the counterdiabatic potential was invented by Sels and Polkovnikov (Sels and Polkovnikov 2017). The phase transitions during the evolution can be modelled using a Hamiltonian in the moving frame of reference as

$$H_0^{\text{eff}} = H_0 - \dot{\lambda} A_\lambda, \tag{1.8}$$

where A_λ is the adiabatic gauge potential defined in Sels and Polkovnikov 2017. The excitations are caused due to the second term. Therefore, if this term is added to the total Hamiltonian, the system is driven by the Hamiltonian H_0 which can be diagonalised exactly throughout, without any phase transitions. It can be seen that when the system evolution is very slow which means, $\dot{\lambda} \to 0$, the system is driven by the Hamiltonian H_0. It can be shown that the exact adiabatic potential obeys the equation,

$$[i\hbar\partial_\lambda H_0 - [A_\lambda, H_0], H_0] = 0 \tag{1.9}$$

In Sels and Polkovnikov 2017, the authors propose a variational method to compute A_λ. In particular, instead of solving Eq. 1.9, the authors equivalently resort to the variational principle of minimizing an action,

$$\frac{\delta S(A_\lambda)}{\delta(A_\lambda)} = 0, \tag{1.10}$$

$$S(A_\lambda) = \text{Tr}[G_\lambda^2(A_\lambda)], \tag{1.11}$$

$$G_\lambda(A_\lambda) = \partial_\lambda H_0 + \frac{i}{\hbar}[A_\lambda, H_0]. \tag{1.12}$$

There have been multiple proposals for a suitable ansatz for finding A_λ. One of the successful ansatz is the nested commutator operator proposed in Claeys et al. 2019, which reads,

$$A_\lambda^{(l)} = i\hbar \sum_{k-1}^{l} \alpha_k [H_0, [H_0, ...[H_0, \partial_\lambda H_0]]] \tag{1.13}$$

This operator is used to construct a potential $G_\lambda(A_\lambda)$ as in Eq. 1.12, and the Hilbert-Schmidt norm in Eq. 1.10 is minimized. A different ansatz using cyclic local operators is introduced in (G. Passarelli, V. Cataudella, R. Fazio, et al. 2020). This ansatz outperforms the nested commutator ansatz for the adiabatic quantum computation of ferromagnetic p-spin model. The cyclic ansatz can be explicitly written as,

$$A_\lambda^{\mathrm{CA}} = \sum_{i=1}^{p'} \alpha_i S_y^i + \sum_{abc} \alpha_{abc} \varepsilon_{abc} S_a S_b S_c. \tag{1.14}$$

In this thesis, we approach the shortcuts to adiabaticity by finding an optimal driving potential using genetic algorithms. In contrast to the variational approach given by Sels and Polkovnikov, we optimize the local ansatz operator using genetic algorithms with the condition to gain higher fidelity at the end of adiabatic evolution.

In particular, we consider the practically implementable local ansatz of the form,

$$
\begin{aligned}
H_{\mathrm{od}}(s,\gamma)_{d=3} &= C(s)\sum_{i=1}^{3}\gamma_i S_i, \\
H_{\mathrm{od}}(s,\gamma)_{d=9} &= H_{\mathrm{od}}(s)_{d=3} + C(s)\sum_{i,j=1}^{3}\gamma_{i,j} S_i S_j, \\
H_{\mathrm{od}}(s,\gamma)_{d=21} &= H_{\mathrm{od}}(s)_{d=9} + C(s)\sum_{i,j,k=1}^{3}\gamma_{i,j,k} S_i S_j S_k.
\end{aligned}
\tag{1.15}
$$

In our work we show that by optimizing few parameters of $H_{\mathrm{od}}(s,\gamma)_{d=3}$, we are able to achieve a high fidelity. Using other advanced techniques like multi-objective optimization we are able to maintain high groundstate probability throughout the evolution.

In addition to finding an optimal driving path by computing a potential, it is also possible to engineer annealing schedules to increase the fidelity of the solution obtained in quantum annealing. The exact optimal schedule was derived by Roland and Cerf in (Roland and Cerf 2002). Here, the authors have imposed local adiabatic condition throughout the evolution to obtain an optimal annealing schedule. The authors have proven this method for Grover search algorithm in adiabatic quantum computation regime and retrieve the quadratic speed up with the exact optimal annealing schedule. In our work, we have used genetic algorithms to find an optimal annealing schedule starting from a polynomial ansatz, for quantum annealing without optimal driving. The results are promising and improved drastically when compared with the traditional linear schedules. However with large number of spins, this ansatz is not sufficient to obtain high fidelity. Fortunately, when time

schedules are optimized in the optimal driving picture, quantum annealing provides high fidelity even for large number of spins.

1.6 Evolutionary algorithms

Evolutionary algorithms are meta-heuristic techniques inspired by the Darwinian theory of evolution (Fonseca and Fleming 1995; Bäck and Schwefel 1993; Rothlauf 2006). The living organisms undergo reproduction, mutation, natural selection, and the fittest individuals survive along this continuous evolution process. Evolutionary algorithms are built on these concepts, where the candidate solutions represent the individuals which undergo optimization. It is a population-based algorithm, where a set of candidate solutions are considered and replaced in every generation. The "strongest" solutions survive through generations and these solutions are used to create off-springs by the process of cross-over. Further, the population is diversified by the process of mutation. There are different classes of evolutionary algorithms, all of which are inspired from the dynamics of evolution of living organisms. The standard and typical algorithm is the genetic algorithm (D. E. Goldberg and Holland 1988; Giovanni Acampora, Vittorio Cataudella, Pratibha R Hegde, et al. 2019; Sastry, D. Goldberg, and Kendall 2005). In this thesis, we mainly use genetic algorithms as a numerical tool for optimization problems in the paradigm of adiabatic quantum computation and quantum annealing. We also implement other variants of genetic algorithms such as memetic algorithm (Giovanni Acampora, Cadenas, et al. 2011; H. Wang, Yang, et al. 2010; Giovanni Acampora, Vittorio Cataudella,

Pratibha Raghupati Hegde, et al. 2021), multi-objective genetic algorithm (Deb et al. 2002; Chivilikhin et al. 2020) in this thesis. The implementation of genetic algorithms is done using the python library Deap (Fortin et al. 2012).

Recently, genetic algorithms and quantum computation have been used to improve each other. Genetic algorithms have been used to learn quantum operators and circuit designs. On the other hand, there have been proposals of quantum-inspired genetic algorithms. The review of these two aspects is given in Giraldi, Portugal, and Thess 2004. In Venturelli and Kondratyev 2019, authors have used genetic algorithms to obtain a candidate solution to an optimization problem which is further used to perform reverse quantum annealing and have shown remarkable results. In Flynn et al. 2021, authors have used genetic algorithms to find an appropriate Hamiltonian model which best describes a given quantum system, along with finding which of the lattice interactions are present. In R. Li et al. 2017, genetic algorithms have been used to find approximate quantum adders. Further, multi-objective genetic algorithms are used to optimize variational quantum eigensolver in Chivilikhin et al. 2020. In Las Heras et al. 2016, genetic algorithms have been proven to be effective in finding equivalent discrete quantum operations of a continuous evolution generated by a given Hamiltonian and in implementing error correction protocols in quantum gates.

The key definitions and steps involved in genetic algorithms are as follows.

Fitness function Genetic algorithms compute optimal solutions by minimizing or maximizing a function called the fitness function. It is a function of parameters

to be optimized and describes the “motive" for genetic algorithms. Each chromosome is assigned fitness values evaluated from fitness function. The quality of a chromosome is determined by this fitness value and is a criterion for the selection process. For example, consider one-max problem (Fortin et al. 2012). It is a very simple problem where, given a set of m integers, which can take the values between 0 and 1, we want each of them to be equal to 1. The fitness function for this problem can be simply the sum of all m integers and direct the genetic algorithm to maximize this fitness function. Further, a fitness function can have single or multiple objectives.

Population initialization The algorithm is initialized by creating a string of random real numbers in the range $[\mathrm{pop}_{min}, \mathrm{pop}_{max}]$ called the chromosomes. These real numbers are individually termed genes. $N_{\mathrm{pop}}$of such chromosomes compose a population. Note that the range of the real numbers is only an initial condition. The numbers can go beyond this initial range throughout the process of optimization, by the process of mutation.

Selection Once the population is created, the best individuals are selected to be parents for crossover. The selection process can be done in different ways like Roulette wheel selection, selecting the best chromosomes, tournament selection and others. Roulette wheel selection is to consider the chromosomes from N_{roul} spins of a roulette wheel which has pie chart of chromosomes weighted by their fitness values. Tournament selection is running a tournament among every N_{T}

individuals and selecting the individuals. In this thesis, we have mainly used tournament selection and we have made a comparison performance of the algorithm by changing the tourn size N_{T}.

Cross-over In this process, among every two selected individuals, a segment is interchanged between the parent chromosomes to give a new offspring chromosome. These new chromosomes replace the parent chromosomes in the population. This process occurs with a probability p_c. There are a number of ways of performing this process. In this thesis, we have mainly resorted to the comparative study of one point cross-over and two-point cross-over. A higher value of p_c leads to premature convergence of the algorithm because similar solutions dominate in the population set. Therefore, it is necessary to vary p_c and choose the appropriate value suitable for the problems chosen.

Mutation Mutation is responsible for creating diversity among living organisms in the evolution process. It has the same role in genetic algorithms as well. Mutation is implemented by varying the values of chromosomes according to a probability distribution. The process of mutation occurs with a probability p_m and each gene of the chromosome changes with the probability p_{ind}. So the total probability of this process is $p_m \times p_{\mathrm{ind}}$. Once again, there are many ways of implementing mutation. In this thesis, we mainly use Gaussian mutation with the mean value μ and standard deviation σ. We vary σ to suit the problem at hand and fix the best value which gives the best fitness value at the end of algorithm.

Termination criteria Genetic algorithm repeatedly applies generations of selection, crossover, and mutation operators to the population of chromosomes. We can choose to stop the algorithm after a certain number of such generations or we can terminate the algorithm once it succeeds to reach a predefined value of fitness function for a chromosome. One can make a qualitative analysis for a problem to estimate the number of generations for which the fitness values converge. Alternatively, one can also choose the number of fitness evaluations to be the termination criteria. This is particularly useful when comparing two different algorithms, such as genetic and memetic algorithms. For example, memetic algorithms implement additional generations of local searches after each generation of the standard genetic algorithm. In this case, comparing the performances for a fixed number of generations is not meaningful.

Hall of fame This is a list that gets updated in every generation with the best chromosome found in that generation. At the end of the algorithm, the parameters stored in this hall of fame are the optimal parameters given by genetic algorithms and are used for analyzing the results. Note that it can sometimes happen that the best chromosome in the current generation is slightly worse than the best chromosome is in the previous generation. It is because in the crossover process, even though the best chromosomes are used for mating, the child chromosome may not be better than either of the parent chromosome. To avoid this and not to lose better solutions, one can use elitism. One way to implement elitism is to replace the worst chromosome in a given generation with the best chromosome of

the previous generation.

In this thesis, we also explore a variant of genetic algorithms called memetic algorithms (Giovanni Acampora, Cadenas, et al. 2011; H. Wang, Yang, et al. 2010). It is inspired from Richard Dawkin's idea of a meme (analogous to the gene), where a quality or technique is shared among a portion of the population from brain to brain rather genetically. Similarly, in the memetic algorithm (MA), a portion of the population is subjected to local optimization in every generation. We have used this technique in solving parameter setting problem of embedding many-body interactions into 2-body interactions (Giovanni Acampora, Vittorio Cataudella, Pratibha Raghupati Hegde, et al. 2021).

We have also implemented the technique of multi-objective genetic algorithms for finding optimal annealing schedules and computing an optimal driving operator. We have used Non-domination Sorted Genetic Algorithms-II (NSGA-II) (Deb et al. 2002; Chivilikhin et al. 2020). This algorithm deviates from the standard genetic algorithms in many ways. To begin with, the fitness function returns two values which have to maximized or minimized. The population is distinguished into different fronts based on if the given chromosome dominates in terms of the trade-off between the two fitness values. And the parents are selected from these fronts, the best front is called the pareto optimal front. Naturally, instead of hall of fame, in every generation the pareto optimal front is stored which consists of a set of chromosomes that have the best trade-off values. In addition, the chromosomes undergo mutation process as in the standard genetic algorithms. The optimal chromosomes in the end of evolution are used to compute and analyze further

results.

In this thesis, we have used the python library DEAP to implement genetic algorithms (Fortin et al. 2012). This library offers flexible operators for selection, crossover and mutation, which can be chosen according to our problem and stored in a "toolbox". In addition, the toolbox is already equipped with module for statistical operations such as min, max, median, standard deviation and others which are computed in each generation for the chromosome distribution. It also has a logbook which stores the history of statistical data of each generation and hall of fame which stores the best chromosome found in each generation. A simple genetic algorithm can be implemented in a straightforward way using a single function "eaSimple". DEAP package is also advantaged by the in-built algorithm of NSGA-II algorithm. This functionality can be accessed by choosing the selection operator as "selNSGA2". In addition, DEAP also has an in-built multithreaded programming tool called "multiprocessing", that allows to parallelize the fitness evaluations.

This thesis is organized as follows. In Chapter 2 we map k-local Hamiltonians into Hamiltonians with 2-local interactions to facilitate embedding optimization problems into quantum annealer architectures. We implement genetic algorithms to achieve the same. We show further improvement in the results by using memetic algorithms. In Chapter 3, we find shortcuts to adiabaticity by engineering the annealing schedules and by finding an optimal driving operator which suppresses Landau-Zener-like transitions. We use single objective genetic algorithms and multi-objective genetic algorithms to optimize quantum annealing. We conclude

by discussing our results in Chapter 4.

2. An evolutionary strategy for finding effective quantum 2-body Hamiltonians of p-body interacting systems[1]

Finding the solution of NP-hard problems requires a time-to-solution increasing exponentially as a function of the system size (Cook 1971). NP-hard tasks can be studied with adiabatic quantum computation (Farhi, Goldstone, Gutmann, and Sipser 2000; Albash and Daniel A. Lidar 2018), a heuristic tool for finding the optimal solution to this kind of problems. The D-Wave quantum machines (Harris et al. 2011) can perform finite-time adiabatic quantum computation, or quantum

[1]This chapter is reprinted by permission from [Springer Nature Customer Service Centre GmbH]: [Springer Nature] [Quantum Machine Intelligence] [An evolutionary strategy for finding effective quantum 2-body Hamiltonians of p-body interacting systems, G. Acampora et al), [License Number: 5233550881398, License Date : Jan 21, 2022] (2019) (Giovanni Acampora, Vittorio Cataudella, Pratibha R Hegde, et al. 2019) and
Giovanni Acampora, Vittorio Cataudella, Pratibha Raghupati Hegde, et al. 2021

annealing. The superconducting architecture of D-Wave processors is built on the Chimera graph (Choi 2008; Choi 2011), a sparsely connected graph that can host $N \leq 2048$ qubits, with at most 2-body interactions. The recent quantum processing unit of D-Wave has upto 5000 qubits and the architecture is a Pegasus graph which has more connectivity. However, many interesting problems, including the ferromagnetic p-spin model (Derrida 1981; Gross and Mezard 1984; Bapst and Semerjian 2012), can be mapped on fully-connected qubit systems with p-body interactions ($p \geq 2$). In order to exploit the available quantum hardware, these problem have to be mapped to effective Hamiltonians (Lucas 2014), containing at most 2-body interactions. This necessarily implies the introduction of auxiliary degrees of freedom, or ancillae (J. D. Biamonte 2008). The major challenge in this problem is to find the free parameters in the 2-body Hamiltonian, corresponding to the p-body one, such that the two Hamiltonians share the same spectral properties, at least at lower energy levels.

In this chapter, we show that genetic algorithms can be a powerful tool to optimize the free parameters in the effective 2-body model, focusing on the ferromagnetic p-spin system described in Sec. 1.3. Genetic algorithms are stochastic meta-heuristics for finding solutions to optimization problems, inspired by the Darwinian theory of evolution (D. E. Goldberg and Holland 1988). The (real) free parameters to optimize, or genes, are arranged in a chromosome. Many such chromosomes, or individuals, compose a population. The fitness of each individual represents its chances of survival along generations. Choosing an appropriate fitness function is the core of genetic algorithms. As shown with more in-depth in the following

Sections, we use the mean square error of the effective spectrum from that of the original Hamiltonian as our fitness function. The idea to apply genetic algorithms is motivated by recent works (O'Driscoll, Nichols, and Knott 2019; Hardy and Steeb 2010), where this kind of evolutionary algorithms have been successfully exploited to solve optimization problems in quantum computing domain.

The ferromagnetic p-spin model is equivalent to the Grover search algorithm in the limit of large and odd p. However, in this chapter we focus on the very simple cases involving small p ($p = 3$) that can be also analytically addressed. As shown by a set of preliminary experiments involving two simple configurations of ferromagnetic p-spin model, the analytic solutions are well-reproduced by the designed genetic algorithm. Moreover, to ensure the validity of our approach, we also simulate a quantum annealing and study the time evolution of the ground state probability for the p-spin system and its effective 2-body counterpart.

The rest of the chapter is organized as follows. In Section 2.1, our model Hamiltonian is introduced. In Section 2.2, the details about the proposed genetic algorithm including chromosome structure and fitness function are given. Section 2.3 presents the settings and the results of a set of preliminary experiments related to the application of the proposed genetic algorithm to two small instances of the p-spin model, which we use as benchmarks for the accuracy of our scheme. In Section 2.4 we introduce and perform memetic algorithms in order to further improve the optimization of mapping. Conclusions and improvements to be performed in the future are reported in Section 2.5.

2.1 Problem definition

We consider a system of N qubits. The two logical states in the computational basis of qubit i can be equivalently labeled as $|\sigma_i\rangle$, with $\sigma_i = \pm 1$, or $|x_i\rangle$, with $x_i = 0, 1$. The two choices are related by $\sigma_i = 1 - 2x_i$. In the following, we will use the x_i representation to express the energy of the system, unless stated otherwise. We denote by σ_i^k, with $k = x, y, z$, the Pauli matrices acting on the ith qubit. Moreover, we work in natural units and fix $\hbar = 1$.

We focus on the ferromagnetic p-spin model (Derrida 1981; Gross and Mezard 1984), whose dimensionless classical Hamiltonian reads

$$E_{\mathrm{p}} = -N \left[\frac{1}{N} \sum_{i=1}^{N} (1 - 2x_i) \right]^p . \tag{2.1}$$

The quantum version of this Hamiltonian reads

$$H_{\mathrm{p}} = -N \left(\frac{1}{N} \sum_{i=1}^{N} \sigma_i^z \right)^p . \tag{2.2}$$

For even p, there are two degenerate ground states due to the Z_2 symmetry of this model, while for odd p the ground state is nondegenerate. For $N \to \infty$ and $p \to \infty$ ($p \leq N$, p odd), this model can implement a Grover-like search in adiabatic quantum computation (L. K. Grover 1996).

In adiabatic quantum computation, one usually employs the parametric Hamiltonian

$$H(s) = A(s) H_0 + B(s) H_{\mathrm{p}}, \tag{2.3}$$

where $s = t/T$ is a dimensionless time and ranges in $[0,1]$, T being the annealing time, and the two functions $A(s)$ and $B(s)$ satisfy $A(0) \gg B(0)$ and $A(1) \ll B(1)$. H_0 is the transverse field Hamiltonian:

$$H_0 = -\sum_{i=1}^{N} \sigma_i^x. \tag{2.4}$$

The qubit system is prepared in the ground state of $H(0)$ and is evolved by slowly changing the parameter s towards $s = 1$. If the evolution is adiabatic compared to the inverse of the minimal gap Δ between the instantaneous ground state and the first excited state, the system is found at $s = 1$ in the ground state of H_p with large probability. In this chapter, we will use a linear annealing schedule, i. e., $A(s) = 1 - s$ and $B(s) = s$.

Despite the fact that it is analytically solvable, the p-spin model is heavily studied in the context of quantum optimization (Seoane and Nishimori 2012; Susa, Yamashiro, et al. 2018; Ohkuwa, Nishimori, and Daniel A. Lidar 2018b; G. Passarelli, De Filippis, et al. 2018; Gianluca Passarelli, De Filippis, et al. 2019), due to its ability to capture the essential feature of NP-hard problems, i. e., the exponentially growing time-to-solution as a function of N. In fact, when $p > 2$ and in the thermodynamic limit, the p-spin system undergoes a first-order quantum phase transition that makes its spectral gap Δ close exponentially fast as a function of N (Bapst and Semerjian 2012).

However, due to its full-connectivity and the presence of p-body interactions, this model can be hardly embedded in the available quantum hardware. The Chimera

graph of latest D-Wave machines only allow to study sparse models with at most 2-body interactions (Choi 2008; Choi 2011). In order to use D-Wave machines to perform the quantum annealing of the p-spin model, first we have to map its Hamiltonian (2.2) into an effective one, containing only 2-body interactions, yet still fully connected. Then, using minor embedding (Choi 2008), this fully-connected effective 2-body Hamiltonian is mapped onto a sparse model, respecting the topology of the Chimera graph. Both these two steps require the introduction of a certain number N_{a} of ancillary degrees of freedom. In this chapter, we will address only the first question and discuss the mapping of the p-spin Hamiltonian with p-body interactions onto the effective fully-connected 2-body Hamiltonian

$$H'_{\mathrm{p}} = K + \sum_{i=1}^{M} h_i \sigma_i^z + \sum_{i=1}^{M} \sum_{j=i+1}^{M} J_{i,j} \sigma_i^z \sigma_j^z, \tag{2.5}$$

where $M = N + N_{\mathrm{a}}$ is the total number of qubits, K is a constant energy shift, h_i are local longitudinal fields and $J_{i,j}$ couples qubits i and j ($j > i$). The corresponding classical effective energy reads

$$E'_{\mathrm{p}} = c_0 + \sum_{i=1}^{M} c_i x_i + \sum_{i=1}^{M} \sum_{j=i+1}^{M} d_{i,j} x_i x_j. \tag{2.6}$$

Parameters in the two Hamiltonians (2.5) and (2.6) are related by (Philipp Hauke

et al. 2019)

$$K = c_0 + \frac{1}{2}\sum_{i=1}^{M} c_i + \frac{1}{4}\sum_{i=1}^{M}\sum_{j=i+1}^{M} d_{i,j}. \tag{2.7}$$

$$h_i = -\frac{1}{2}c_i - \frac{1}{4}\sum_{j=1}^{M} d_{j,i} - \frac{1}{4}\sum_{j=1}^{M} d_{i,j}, \tag{2.8}$$

$$J_{i,j} = \frac{1}{4}d_{i,j}. \tag{2.9}$$

All these free parameters are real-valued.

To map the Hamiltonian (2.2) to the Hamiltonian (2.5) means that the low part of the spectrum of H'_{p} has to match the spectrum of H_{p}, and all other energy levels must be separated by a large energy gap from the original eigenvalues. Indeed, for the purpose of adiabatic quantum computation, only the ground state and the first excited subspace have to be matched in the purely adiabatic limit. However, in this chapter we will always aim at matching the first $L = 2^N$ eigenvalues of H'_{p} and all the original spectrum. We stress that even if the low-energy subspace of H'_{p} correctly reproduces the spectrum of H_{p}, the quantum dynamics could be different. However, this mapping allows to solve the original optimization problem, through an experimentally viable effective model.

Multiple-body interactions can be turned into 2-body interactions using AND embedding. Pairs of binary variables (x_i, x_j) are encoded in an ancillary degree of freedom $\tilde{x}_{i,j} = x_i \wedge x_j$. Of course, allowed configurations for the triple $(x_i, x_j, \tilde{x}_{i,j})$ are those where the logical AND is satisfied. A penalty function $E_{\mathrm{pen}}(x_i, x_j, \tilde{x}_{i,j})$ penalizes nonphysical configurations through a large cost $\delta > 0$. We will use the

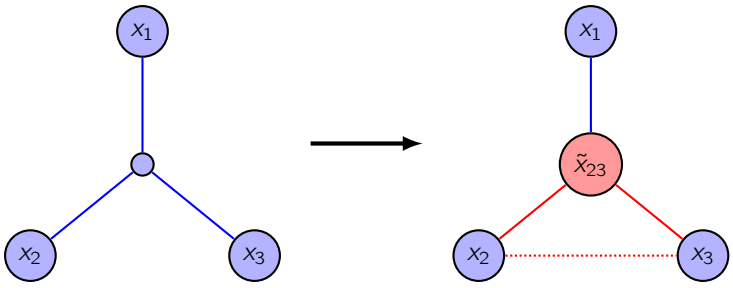


Figure 2.1: Visual representation of the AND embedding of Eq. (2.11). On the left, the graph representing the original 3-body term $Jx_1x_2x_3$. On the right, the graph corresponding to the decomposed Hamiltonian with maximum 2-body interactions, including penalties. Blue and red circles represent the original and the ancillary qubits, respectively. Blue lines represent coupling strength J, solid red lines represent -2δ and dashed red lines δ.

penalty function

$$E_{\text{pen}}(x_i, x_j, \tilde{x}_{i,j}) = \delta(3\tilde{x}_{i,j} + x_i x_j - 2\tilde{x}_{i,j}x_i - 2\tilde{x}_{i,j}x_j). \tag{2.10}$$

It is easy to see that $E_{\text{pen}} = 0$ if $\tilde{x}_{i,j} = x_i \wedge x_j$, while $E_{\text{pen}} \geq \delta$ if $\tilde{x}_{i,j} \neq x_i \wedge x_j$ (J. D. Biamonte 2008; Leib, Zoller, and Lechner 2016).

To be specific, consider a 3-body term as $Jx_1x_2x_3$. Using the previously introduced AND embedding, this term can be rewritten using an ancillary qubit $\tilde{x}_{23}$ as

$$Jx_1x_2x_3 \equiv Jx_1\tilde{x}_{23} + \delta(3\tilde{x}_{23} + x_2x_3 - 2\tilde{x}_{23}x_2 - 2\tilde{x}_{23}x_3), \tag{2.11}$$

where the equivalence is intended as equality between corresponding $L = 8$ lowest energy levels. This is pictorially represented in Fig. 2.1.

2.2 A genetic algorithm for optimizing Hamiltonian free parameters

This section is devoted to present the application of genetic algorithms for finding the free parameters useful for mapping p-body interacting systems in the 2-body Hamiltonian. Genetic algorithms are population-based meta-heuristics which try to solve an optimization (or search) problem by manipulating a multi-set of potential solutions and reproducing the natural selection process involving human individuals. In detail, as natural selection process leads to the survival of only the fittest human individuals (i. e., those capable of adapting to the changing environment), so the genetic algorithms perform an evolution process that leads to the survival of only the fittest solutions (i. e., those that better solve the optimization problem). Specifically, genetic algorithms operate on encoded representations of the solutions, called *chromosomes*. To determine how good a solution is, a *fitness function* is used to reflect the capability of the solution to solve the problem. In general, the workflow of a genetic algorithm includes the following steps. Firstly, a population of chromosomes is generated randomly and evaluated by using the fitness function. Successively, the algorithm performs a set of generations until some termination criteria are satisfied. In each generation, a set of chromosomes is selected to survive (*parent selection mechanism*) and reproduce by means of the crossover operator. Generally, this operator takes in input two chromosomes (*parent*) and gives in output two new chromosomes (*offspring*) by exchanging portions of the parents. As in the natural evolution process, some mutations can

occur. The mutation operator performs by randomly changing some of the genes in the chromosomes. Both mutation and crossover operators are stochastic procedures that are applied according to a probability, named *mutation probability* p_{mut} and *crossover probability* p_{cx}, respectively. As for the termination criteria, the most common one is the achievement of a maximum number of generations. Therefore, in this chapter, we use this termination criterion.

Starting from this description, in order to implement a genetic algorithm for our problem, it is necessary to define the chromosome structure, the fitness function and the used genetic operators. Hereafter, a detailed description of the genetic algorithm components is given.

2.2.1 Chromosome structure

The chromosome must encode the solution of our problem, that is the set of Hamiltonian free parameters (2.6). In order to achieve this aim, the chromosome structure has been defined as follows:

$$\vec{v} \equiv (c_0, c_1, \ldots, c_M, d_{1,2}, d_{1,3}, \ldots, d_{M-1,M}). \tag{2.12}$$

The length of the defined chromosome is $D = (M^2 + M + 2)/2$. The values for the genes belong to the range $[-10, 10]$. This choice is motivated by the fact that, in the analyzed cases, the genes of the chromosome $\vec{v}$ are strictly included within these bounds, except for the penalties that are not subject to the same constrictions.

2.2.2 Fitness function

The fitness function is used to evaluate the quality of the candidate solutions encoded in the chromosomes. It is implemented by taking into account E_{p} and E'_{p} reported in Eq. (2.1) and Eq. (2.6), respectively. In detail, firstly, we list all possible configurations with N qubits, for the starting model, and with M qubits, for the effective one. Conventionally, we arrange qubits of the effective model so that ancillae are at the beginning of the sequence. Secondly, we apply E_{p} and E'_{p} for each combination, sort the corresponding energies in ascending order and perform the differences. Formally, the fitness function F is defined as follows:

$$F = \frac{1}{L}\sum_{i=1}^{L}\left[(E_{\mathrm{p}})_i - (E'_{\mathrm{p}})_i\right]^2 + E^{\mathrm{vec}}_{\mathrm{pen}}, \tag{2.13}$$

with $L = 2^N$. The first term enforces equality between corresponding eigenvalues, while the second one is a penalty cost to be applied when the eigenvectors of the effective Hamiltonian are ordered differently than the original ones. We do not apply penalties when eigenvectors are ordered differently within symmetry subspaces of the original Hamiltonian. In our code, $E^{\mathrm{vec}}_{\mathrm{pen}} = l\delta$, where l is the number of unsorted configurations.

2.2.3 Genetic operators

Once defined the chromosome structure and the fitness function, it is necessary to discuss about the genetic operators, that is, crossover, mutation and selection mechanism. In the literature, different kinds of crossover, mutation and selection

operators have been defined (Yao 1993) (Herrera, Lozano, and Sánchez 2003). However, when a new problem is addressed with genetic algorithms, it is necessary to select the most opportune configuration for these operators. For this reason, in this chapter, we perform a design study of the implemented genetic algorithm aimed at selecting the most opportune configuration for the problem at issue. In detail, this study has involved the investigation of two different crossover operators, that is, the one-point crossover and the two point-crossover, different Gaussian distributions for mutation operator, and different values for tournament size for the selection mechanism. The results of this design study are reported in the next section. To conclude, in this section, we give more details about the investigated genetic operators.

Crossover operators Generally, the crossover operator works by combining portions of two chromosomes, denoted as parents. In this work, we investigate two different strategies, i. e., one- and two-point crossover. In detail, the one point crossover chooses a random number r in the range $[1, D-1]$ (with D the length of the chromosome), and then splits both parents at this point by creating the two children by exchanging the tails. Instead, the two-point crossover chooses two random numbers r_1 and r_2 in the range $[1, D-1]$, breaks parents in these two points by creating the children by taking alternative segments from the parents.

Mutation Generally, the mutation operator works by changing values of chromosome genes randomly. The Gaussian mutation chooses values drawn from a Gaussian distribution with zero mean and standard deviation σ. In this work,

we investigate several values for σ, i. e., $\sigma = 0.2$, 0.4, 0.6, 0.8 and 1.0.

Selection Selection mechanism is devoted to select the chromosomes that will become parents of the next generation. One of the most known selection operators is the tournament mechanism which selects each parent by performing a tournament among N_T chromosomes, randomly selected, where the chromosome that wins is the fittest one. In this work, we investigate $N_T = 2$, 3 and 5.

2.3 Preliminary experiments and results

This section is devoted to show the results of some preliminary experiments carried out to demonstrate the feasibility of the proposed approach. In detail, the designed genetic algorithm is applied to solve two simple configurations of the ferromagnetic p-spin model. This choice is due to the possibility to analytically solve these configurations and perform a comparison with the output of the genetic algorithm. The configuration of the applied genetic algorithm is the result of a design study involving the genetic operators described in Section 2.2.3. The comparison between the solution obtained by the designed genetic algorithm and that computed analytically is carried out by considering the energy eigenvectors and eigenvalues of the first 2^N Hamiltonian states, as well as the Hamiltonian free parameters. Moreover, the use of the solution obtained by the genetic algorithm is investigated for the adiabatic quantum computation with respect to the original p-spin model. Hereafter, more details about the considered configurations of the

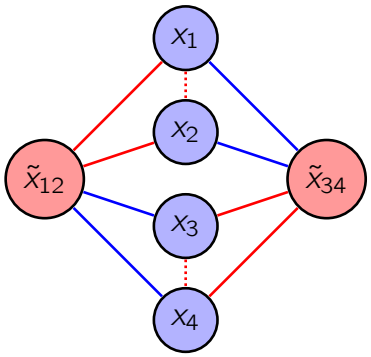


Figure 2.2: Graph representing the effective 2-body model for the p-spin Hamiltonian with $N = 4$ and $p = 3$, originally containing four 3-body terms. Blue and red circles represent the original and the ancillary qubits, respectively. Blue lines represent coupling strength J, solid red lines represent -2δ and dashed red lines δ.

ferromagnetic p-spin model, the design study, the comparison results and the exploitation of genetic solutions in the adiabatic quantum computation are given.

2.3.1 Experimental set-up

To perform our experimentation, we consider the simplest configurations of the ferromagnetic p-spin model which require the minimum number of ancillary qubits for embedding, e. g., $N = 3$, $p = 3$ and $N = 4$, $p = 3$. The Hamiltonian of the former one only contains a single 3-body term, which can be decomposed as described in Eq. (2.11) with a single ancilla, i. e., $\tilde{x}_{23} = x_2 \wedge x_3$ and hence $M = 4$. By contrast, the Hamiltonian for the $N = 4$ case contains four 3-body terms, which require two ancillae, i. e., $\tilde{x}_{12} = x_1 \wedge x_2$ and $\tilde{x}_{34} = x_3 \wedge x_4$, to be decomposed as described in Section 2.1, i. e., $M = 6$. The reduction process leads to the graph represented in Fig. 2.2.

These two cases are selected because it is possible to work out by hand the analytic solution for these settings with little effort and, this is useful for carrying out the comparison study with the designed genetic algorithm. We report the analytic solutions below.

$$\vec{v}_{M=4} \equiv (-3, -3\delta, \frac{26}{9}, \frac{26}{9}, \frac{26}{9}, 2\delta, 2\delta, \frac{16}{3}, -\frac{8}{3} - \delta, -\frac{8}{3}, -\frac{8}{3}); \tag{2.14}$$

$$\vec{v}_{M=6} \equiv (-4, -3\delta, -3\delta, \frac{7}{2}, \frac{7}{2}, \frac{7}{2}, \frac{7}{2}, 0, 2\delta, 2\delta, 3, 3, 3, 3, 2\delta, 2\delta, -3-\delta, -3, -3, -3, -3, -3-\delta). \tag{2.15}$$

In what follows, we will fix $\delta = 50$ as this number provides a large separation between the largest eigenvalue of the target subspace and the smallest eigenvalue of the nonphysical one, in both cases.

2.3.2 Design study

In order to select the best configuration for genetic operators, we perform a design study by considering the operators described in Section 2.2.3. By using 2 different crossover operators, 5 different mutation operators and 3 different selection operators, our design study involves the assessment of 30 different combinations. Table 2.1 gives an index to the different combinations. As for the other parameters of the genetic algorithm, in our experimentation, we set $N_{\text{pop}} = 20$ chromosomes, the crossover probability $p_{\text{cx}} = 0.4$, the mutation probability $p_{\text{mut}} = 0.7$. This choice is not typical, as usually $p_{\text{mut}} < p_{\text{cx}}$. However, the results we discuss below are qualitatively independent on these two parameters. The termination criterion

is the achievement of a number of generations, i. e., $N_g = 25\,000$. Genetic algorithms are stochastic procedures, thus we repeat the simulation 100 times for every combinations.

The comparison among all the different combinations of genetic operators is shown in the boxplot of Fig. 2.3. In detail, boxplots show the minimum, the maximum, the median and the likely range of variation of the fitness values over the 100 runs. However, in order to select the most opportune combination, the median fitness values are compared.

By analyzing Fig. 2.3, for $N = 3$, $M = 4$, the best median of the fitness values (the minimum one) is the combination 18, i. e., the combination involving the two-point crossover, the Gaussian mutation with $\sigma = 0.2$ and tournament selection with $N_T = 5$. Instead, for $N = 4$ and $M = 6$, the configuration 2 is the one yielding the smallest median fitness value, i. e., the combination involving one-point crossover, $\sigma = 0.2$ and $N_T = 3$.

2.3.3 Results

Once performed 100 runs of the genetic algorithm with configuration 18 for $N = 3$, $M = 4$ and 100 runs with configuration 2 for $N = 4$, $M = 6$, we obtain 100 solutions for $N = 3$, $M = 4$ and 100 solutions for $N = 4$, $M = 6$. As an example, Fig. 2.4 shows the fitness values against the number of generations for the genetic algorithm with configuration 18 used to address $N = 3$, $M = 4$ problem.

To compare the solutions obtained by the genetic algorithm and the analytically computed ones for both considered configurations of the ferromagnetic p-spin

Table 2.1: Combinations of genetic operators investigated in the design study. 1P (2P) stands for one-point (two-point) crossover.

#	Crossover	σ	N_{T}	#	Crossover	σ	N_{T}
1	1P	0.2	2	16	2P	0.2	2
2	1P	0.2	3	17	2P	0.2	3
3	1P	0.2	5	18	2P	0.2	5
4	1P	0.4	2	19	2P	0.4	2
5	1P	0.4	3	20	2P	0.4	3
6	1P	0.4	5	21	2P	0.4	5
7	1P	0.6	2	22	2P	0.6	2
8	1P	0.6	3	23	2P	0.6	3
9	1P	0.6	5	24	2P	0.6	5
10	1P	0.8	2	25	2P	0.8	2
11	1P	0.8	3	26	2P	0.8	3
12	1P	0.8	5	27	2P	0.8	5
13	1P	1.0	2	28	2P	1.0	2
14	1P	1.0	3	29	2P	1.0	3
15	1P	1.0	5	30	2P	1.0	5

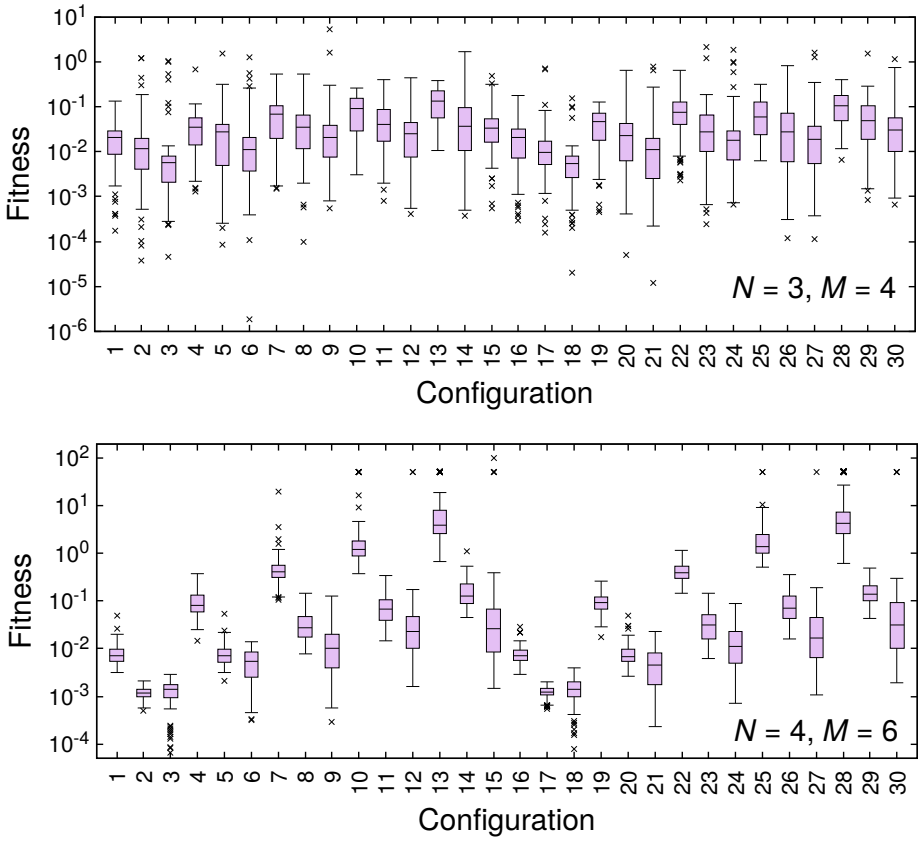


Figure 2.3: Box graphs depicting the distribution of fitness values for different combinations of the genetic operators. Here, N is the number of qubits in the original Hamiltonian and M is the number of qubits in the mapped Hamiltonian. For each combination, the black line inside the box corresponds to the median over 100 runs of the genetic algorithm. Outliers are explicitly indicated using black crosses. The indexes of the combinations on the x-axis are tabulated in Table 2.1.

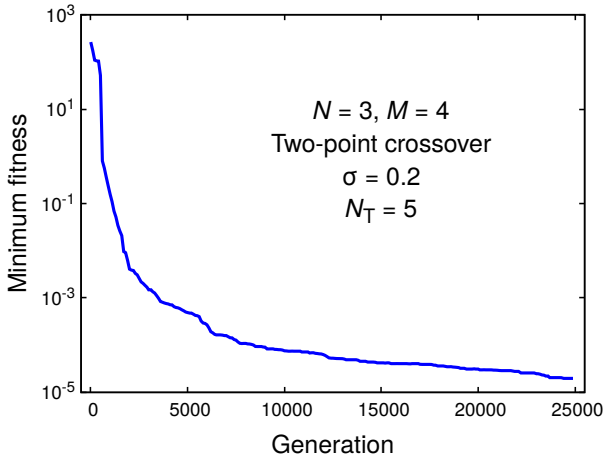


Figure 2.4: Updating of the fitness values through 25.000 generations, for the case $N = 3$, $M = 4$, and for the best combination of genetic operators, i. e., combination 18 in Table 2.1. The final fitness value at the last generation is $F = 9.8799 \times 10^{-8}$.

model, we select the best chromosome (i. e., the one with the least fitness value) among all solutions over 100 runs. The comparison is carried out by considering the computed Hamiltonian free parameters, but also the energy eigenvectors and eigenvalues of the first 2^N states of the embedded Hamiltonian generated by the genetic algorithm with that of the original Hamiltonian in Eq. (2.1). Table 2.2 and Table 2.3 show the results of this comparison for $N = 3$, $M = 4$ and $N = 4$, $M = 6$, respectively. For $N = 3$, $M = 4$, the first qubit in the genetic eigenvector is the ancilla qubit defined as $q_0 = q_1 \wedge q_2$. For the case $N = 4$ and $M = 6$, the first two qubits of the sequence are the two ancillae, defined as $q_0 = q_2 \wedge q_3$ and $q_1 = q_4 \wedge q_5$. In addition, the first 2^N genetic eigenvectors always respect the original degeneracies of the starting spectrum. Moreover, we also observe the sign-flip pattern in the spectrum, as predicted by the Z_2 anti-symmetry of this model for odd p.

Table 2.2: The results of the comparison between the best chromosome obtained by the genetic algorithm and the analytically computed solution for $N = 3$, $M = 4$ problem. We fixed $\delta = 50$.

Free parameters		Eigenvectors		Eigenvalues	
Analytic	Genetic	Analytic	Genetic	Analytic	Genetic
-3	-2.99919	[0, 0, 0]	[0, 0, 0, 0]	-3.00000	-2.99919
-150	-150.853	[0, 0, 1]	[0, 1, 0, 0]	-0.11111	-0.11138
$26/9$	2.88781	[0, 1, 0]	[0, 0, 0, 1]	-0.11111	-0.11129
$26/9$	2.88795	[1, 0, 0]	[0, 0, 1, 0]	-0.11111	-0.11124
$26/9$	2.88790	[0, 1, 1]	[1, 1, 1, 0]	0.11111	0.11111
100	100.720	[1, 0, 1]	[0, 1, 0, 1]	0.11111	0.11120
100	101.118	[1, 1, 0]	[0, 0, 1, 1]	0.11111	0.11120
$16/3$	5.33174	[1, 1, 1]	[1, 1, 1, 1]	3.00000	2.99999
$-158/3$	-53.6496				
$-8/3$	-2.66531				
$-8/3$	-2.66545				

An indicator of the accuracy of the returned solution is the root mean square

$$\text{rms} \equiv \sqrt{\frac{1}{D}\sum_{i=1}^{D}\left(\frac{v_i^{\text{analytic}} - v_i^{\text{genetic}}}{v_i^{\text{analytic}}}\right)^2}. \tag{2.16}$$

In the $N = 3$ case, the best solution yields rms $\approx 7.2 \times 10^{-3}$, while for $N = 4$ we have rms ≈ 0.22. The scaling of the accuracy of the best returned solution as a function of the input size is a serious question that will be addressed in a forthcoming paper. However, we observe that the analytic solution is qualitatively well-reproduced by the genetic algorithm in both cases.

Table 2.3: The results of the comparison between the best chromosome obtained by the genetic algorithm and the analytically computed solution for $N = 4$, $M = 6$ problem. We fixed $\delta = 50$.

Free parameters		Eigenvectors		Eigenvalues	
Analytic	Genetic	Analytic	Genetic	Analytic	Genetic
-4	-3.99450	[0, 0, 0, 0]	[0, 0, 0, 0, 0, 0]	-4.0	-3.99450
-150	-148.165	[0, 0, 0, 1]	[0, 0, 0, 0, 1, 0]	-0.5	-0.53947
-150	-144.833	[0, 0, 1, 0]	[0, 0, 1, 0, 0, 0]	-0.5	-0.52837
$7/2$	3.46613	[0, 1, 0, 0]	[0, 0, 0, 0, 0, 1]	-0.5	-0.47565
$7/2$	3.54304	[1, 0, 0, 0]	[0, 0, 0, 1, 0, 0]	-0.5	-0.45146
$7/2$	3.45503	[0, 0, 1, 1]	[1, 0, 1, 1, 0, 0]	-0.0	-0.03157
$7/2$	3.51886	[0, 1, 0, 1]	[0, 0, 0, 1, 0, 1]	-0.0	-0.02561
0	-0.02015	[0, 1, 1, 0]	[0, 0, 1, 0, 0, 1]	-0.0	-0.01985
100	96.397	[1, 0, 0, 1]	[0, 0, 1, 0, 1, 0]	-0.0	-0.01965
100	95.7088	[1, 0, 1, 0]	[0, 0, 0, 1, 1, 0]	-0.0	0.00748
3	2.98789	[1, 1, 0, 0]	[0, 1, 0, 0, 1, 1]	-0.0	0.04105
3	3.12228	[0, 1, 1, 1]	[1, 0, 1, 1, 1, 0]	0.5	0.46894
3	2.91879	[1, 0, 1, 1]	[0, 1, 1, 0, 1, 1]	0.5	0.46932
3	3.04070	[1, 1, 0, 1]	[1, 0, 1, 1, 0, 1]	0.5	0.50621
100	97.8836	[1, 1, 1, 0]	[0, 1, 0, 1, 1, 1]	0.5	0.53568
100	97.9453	[1, 1, 1, 1]	[1, 1, 1, 1, 1, 1]	4.0	4.00773
-53	-58.5698				
-3	-2.94631				
-3	-3.01034				
-3	-2.99610				
-3	-3.09301				
-53	-56.9343				

2.3.4 Discussion for adiabatic quantum computation

The genetic 2-body model can be used for adiabatic quantum computation, with the time-dependent Hamiltonian of Eq. (2.3), and compared with the original p-spin model, or with the analytic 2-body model. In this last part, we focus on $N = 3$, $M = 4$ for computational ease. We performed the same analysis also for $N = 4$, $M = 6$ with similar results. In each of the cases, the number of ancillae qubits utilized is $M - N$. For the purpose of quantum optimization, it is paramount that the fidelity Φ, i. e., the ground state occupation probability at the end of the annealing ($s = 1$), is high. Of course, due to the larger number of degrees of freedom of the effective model with ancillae, we expect that a slower annealing is needed to reach the target ground state, compared with the original p-spin model.

First, we compare the low part of the instantaneous spectra of the two models in Fig. 2.5, using $\delta = 11$ for visual clarity. We observe that the first $2^N = 8$ states match at $s = 1$, though they differ for $0 < s < 1$. Higher excited states, subjected to penalty, are significantly separated from the lower ones.

Second, we simulate a quantum evolution of annealing time $T = 100$, and compare the evolution of the ground state occupation probability of the genetic model with those of the original p-spin model and of the analytic 2-body model of Eq. (C.1). Results are shown in Fig. 2.6. The ground state population of the effective model evolves differently than the original one. This is not surprising, as the goal of our genetic algorithm is to match the final spectrum, irrespective of the instantaneous dynamics. By contrast, the evolution of the genetic model closely resembles that of the analytic 2-body model. The fidelity at the end of the evolution is large

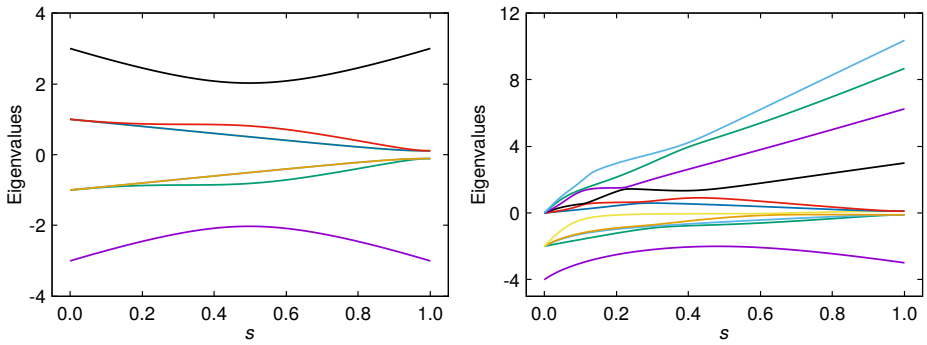


Figure 2.5: Instantaneous eigenvalues of the time-dependent Hamiltonian (2.3), for $s \in [0,1]$, with $\delta = 11$ for sake of clarity. Panel (a) is for the p-body ferromagnetic p-spin model in Eq. (2.2), panel (b) is for the genetic 2-body Hamiltonian in Eq. (2.5), for the first $L = 11$ states. At $s = 1$, the first 2^N states are the same for the two models, whereas higher (penalized) energy levels are separated from the low part of the spectrum. Increasing δ will increase this separation.

($\Phi \approx 0.994$ and $\Phi \approx 0.993$ for the analytic and the genetic 2-body Hamiltonians, respectively), although not as large as that of the original model ($\Phi \approx 0.99998$) for this choice of T. This can be justified by the adiabatic condition.

In fact, a common adiabatic criterion states that the evolution time must satisfy the following condition,

$$T_{\mathrm{AD}} = \max_{t\in[0,T]} \frac{|\langle \varepsilon_0(t)|\partial_t H(t)|\varepsilon_1(t)\rangle|}{|\varepsilon_1(t)-\varepsilon_0(t)|^2}, \tag{2.17}$$

where $|\varepsilon_i(t)\rangle$ are the instantaneous eigenstates of $H(s)$ (Albash and Daniel A. Lidar 2018). The introduction of ancillary qubits with large energy penalties δ for unphysical configuration makes the numerator of the right-hand side of Eq. (2.17) larger for the effective model than for the original p-spin model, while the minimal gap is similar for both models. For the cases we analyzed, the adiabatic time scale

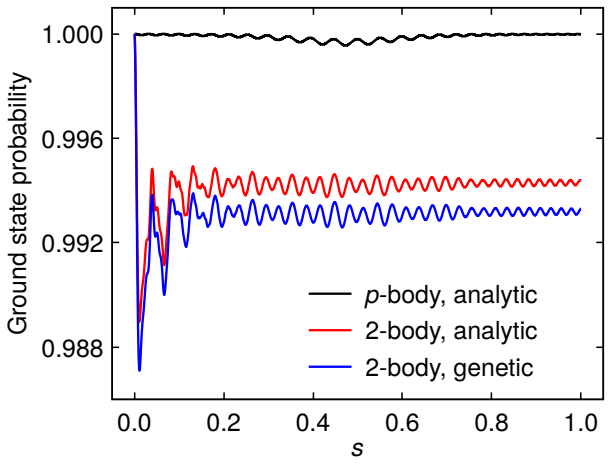


Figure 2.6: Evolution of the ground state occupation probability as a function of the dimensionless time s, for the original p-spin model with $N = 3$ (black line), the analytic 2-body model of Eq. (C.1) (red line), and the best genetic solution (blue line).

of the effective model is $\sim \delta$ times longer than for the original model. Thus, it is natural to expect that, for fixed T, the original model is closer to the adiabatic limit than the effective one, thus the corresponding fidelity is larger.

2.4 Improvement in the mapping by using Memetic algorithms

In this section we solve the problem of finding a mapping of p-body Hamiltonian into 2-body Hamiltonian using a class of evolutionary algorithms called memetic algorithms(MA). This algorithm gives importance to local search after every generation of global search of genetic algorithms. Here, we have used hill-climbing local search procedure where a portion of candidate solutions are repeatedly mutated for

some generations and these candidate solutions are replaced in the population with the best chromosome obtained in this procedure (Renders and Bersini 1994; H. Wang, D. Wang, and Yang 2009). This process leads to greedy search for solutions around the region of the intial solution. In this section, this heuristic technique is shown to be more efficient than the simple genetic algorithm in optimizing the chromosome, for the problem considered. Results of this section are published in the article Giovanni Acampora, Vittorio Cataudella, Pratibha Raghupati Hegde, et al. 2021.

Since we are considering the same problem of mapping k-body Hamiltonian into 2-body Hamiltonian, the fitness function remains the same as in Eq. 3.11. Memetic algorithm involves the cyclic steps of selection, crossover, mutation, and local search procedure of hill climbing. In order to design an efficient memetic algorithm, it is necessary to deal with an important design issue: what is the best tradeoff between local search and the global search provided by evolution? (Krasnogor and J. Smith 2005). This issue leads naturally to questions such as the following:

- *Local search frequency*: How often should local search be applied within the evolutionary cycle?
- *Local search intensity*: How much computational effort should be allocated to each local search?

In this work, the *frequency* hyper-parameter l_f refers to the first question and represents the number of individuals that will be undergone to the local search procedure. Instead, the *intensity* hyper-parameter l_i refers to the second question and, in this work, it will be represented by the number of fitness function evaluations

computed during the application of the local search procedure on one of the selected individuals. These two hyper-parameters, frequency and intensity, strongly affect the performance of the memetic algorithms, and, as a consequence, in the experimental session, an exhaustive study is performed to tune them (see appendix C).

2.4.1 Local search procedure

To complete the description of the proposed MA, it is necessary to discuss the integrated local search procedure. In this work, we use a variant of the Hill Climbing, whose pseudo-code is reported in Table 2.4. The choice of integrating this local search with genetic algorithms is tied to the fact that this hybridization represents the most popular one for memetic algorithms (Renders and Bersini 1994) and it is suitable for continuous search space (Shahamatnia et al. 2011). In general, the Hill Climbing Search is a greedy strategy which performs iterative search for optimum solution in the *neighborhood* of a candidate (G. Acampora and Vitiello 2012). The typical steps of the Hill Climbing Search are:

- to generate an arbitrary candidate solution,
- to change the current solution at each iteration by using, typically, a mutation operator
- if the change improves the current candidate, then the new one becomes the current one. The algorithm ends when a termination criterion is satisfied.

In literature, there are several variants of Hill Climbing search depending on how the next solution is tried (G. Acampora and Vitiello 2012). Among them, the

Table 2.4: Stochastic Hill Climbing pseudo-code

Input: *initial_solution* representing the initial solution, maximum number of evaluations of fitness function *max_evals*, l_p is the local mutation probability.

Output: *sol* which represents the solution optimized by means of local search.

```
1: sol ← initial_solution;
2: evals ← 0;
3: while (evals < max_evals ) do
4:     new_sol ← getRandomNeighborWithGaussianDistribution(sol, l_p);
5:     if (evaluate(new_sol) < evaluate(sol)) then
6:         sol ← new_sol;
7:     end if
8:     evals ← evals + 1;
9: end while
10: return sol;
```

stochastic version of the Hill Climbing search selects a neighbor at random. In our work, we use this variant equipped with the same Gaussian distribution used in the genetic mutation to change the variables of the current solution. Moreover, each solution variable will be mutated according to a probability, denoted as *local mutation probability* l_p. This numerical hyper-parameter will be tuned in the systematic design study. We have considered two instances of the ferromagnetic p-spin model – $N = 3, M = 4$ and $N = 4, M = 6$.

The details of choosing the optimal parameters of memetic algorithms for these instances is given in Appendix. C. The results obtained using our memetic algorithms is compared with analytical results and other heuristics in the following subsections.

2.4.2 Comparison with the analytical approach

The solution obtained by the memetic algorithm, run with the best configuration obtained thanks to the aforementioned tuning process, is assessed by comparing it with the reported known analytical solution. The comparison is performed by taking into account the energy eigenvectors and eigenvalues of the first 2^N Hamiltonian states, as well as the D Hamiltonian free parameters. Since memetic algorithms are stochastic methods, we compare the analytical solution with the memetic solution characterized by the median fitness value on 100 runs. Tables 2.5 and 2.6 show the comparison between the solutions computed by the proposed memetic algorithm and by the analytical approach for the considered problem instances, respectively.

Table 2.5: The free parameters, eigenvalues and eigenvectors obtained from the memetic algorithm and their comparison with the analytical values, for the instance of $N = 3$ and $M = 4$ case, with $\delta = 50$. The first qubit of the memetic approach is the ancilla $\tilde{x}_{23} = x_2 \wedge x_3$.

Free parameters		Eigenvectors		Eigenvalues	
Analytic	Memetic	Analytic	Memetic	Analytic	Memetic
-3	-2.99875	[0, 0, 0]	[0, 0, 0, 0]	-3.00000	-2.99875
-150	-151.71271	[0, 0, 1]	[0, 0, 1, 0]	-0.11111	-0.11296
$26/9$	2.88847	[0, 1, 0]	[0, 0, 0, 1]	-0.11111	-0.11141
$26/9$	2.88578	[1, 0, 0]	[0, 1, 0, 0]	-0.11111	-0.11027
$26/9$	2.88733	[0, 1, 1]	[0, 0, 1, 1]	0.11111	0.11081
100	101.02709	[1, 0, 1]	[0, 1, 0, 1]	0.11111	0.11093
100	105.91522	[1, 1, 0]	[1, 1, 1, 0]	0.11111	0.11099
$16/3$	5.33107	[1, 1, 1]	[1, 1, 1, 1]	3.00000	2.99970
$-158/3$	-57.89412				
$-8/3$	-2.66613				
$-8/3$	-2.66356				

Table 2.6: The free parameters, eigenvalues and eigenvectors obtained from the memetic algorithm and their comparison with the analytical values, for the instance of $N = 4$ and $M = 6$ case, with $\delta = 50$. The first two qubits of the memetic approach are the two ancillae $\tilde{x}_{12} = x_1 \wedge x_2$ and $\tilde{x}_{34} = x_3 \wedge x_4$.

Free parameters		Eigenvectors		Eigenvalues	
Analytic	Memetic	Analytic	Memetic	Analytic	Memetic
-4	-4.00015	[0, 0, 0, 0]	[0, 0, 0, 0, 0, 0]	-4.0	-4.00015
-150	-156.82967	[0, 0, 0, 1]	[0, 0, 0, 0, 1, 0]	-0.5	-0.52681
-150	-145.30849	[0, 0, 1, 0]	[0, 0, 1, 0, 0, 0]	-0.5	-0.522034
$7/2$	3.47812	[0, 1, 0, 0]	[0, 0, 0, 1, 0, 0]	-0.5	-0.49854
$7/2$	3.50162	[1, 0, 0, 0]	[0, 0, 0, 0, 0, 1]	-0.5	-0.49541
$7/2$	3.47334	[0, 0, 1, 1]	[0, 0, 0, 1, 0, 1]	-0.0	-0.00737
$7/2$	3.50473	[0, 1, 0, 1]	[0, 0, 1, 0, 0, 1]	-0.0	-0.00623
0	0.04262	[0, 1, 1, 0]	[0, 0, 0, 1, 1, 0]	-0.0	-0.00367
100	105.23229	[1, 0, 0, 1]	[1, 0, 1, 1, 0, 0]	-0.0	0.00553
100	99.95697	[1, 0, 1, 0]	[0, 1, 0, 0, 1, 1]	-0.0	0.01108
3	2.94584	[1, 1, 0, 0]	[0, 0, 1, 0, 1, 0]	-0.0	0.01597
3	2.99295	[0, 1, 1, 1]	[0, 1, 0, 1, 1, 1]	0.5	0.49935
3	2.93736	[1, 0, 1, 1]	[1, 0, 1, 1, 0, 1]	0.5	0.50072
3	2.97869	[1, 1, 0, 1]	[0, 1, 1, 0, 1, 1]	0.5	0.50228
100	105.10894	[1, 1, 1, 0]	[1, 0, 1, 1, 1, 0]	0.5	0.51089
100	93.38913	[1, 1, 1, 1]	[1, 1, 1, 1, 1, 1]	4.0	3.99791
-53	-51.33364				
-3	-2.93534				
-3	-2.98893				
-3	-2.97847				
-3	-3.01356				
-53	-56.15643				

By analysing Tables 2.5 and 2.6, it is possible to observe that the first 2^N memetic eigenvectors always respect the original degeneracies of the starting spectrum. Moreover, it is possible to say that the analytic solution is qualitatively well-reproduced by the memetic algorithm in both considered cases. However, in order

to quantify the precision of the produced solution, we compute the *Normalized Root Mean Square Error* (NRMSE) between the analytic solution and that produced by our algorithm. Formally,

$$\text{NRMSE} \equiv \frac{\sqrt{\frac{1}{D}\sum_{i=1}^{D}\left(v_i^{\text{analytic}} - v_i^{\text{memetic}}\right)^2}}{\left(v_{max}^{\text{analytic}} - v_{min}^{\text{analytic}}\right)} \tag{2.18}$$

where D is the number of the free parameters, v_i^{analytic} is the i-th free parameter of the analytic solution, v_i^{memetic} is the i-th free parameter of the memetic solution, and, $v_{max}^{\text{analytic}}$ and $v_{min}^{\text{analytic}}$ are the maximum and the minimum of the free parameters of the analytic solution, respectively. In detail, in the case $N = 3$, the median fitness solution (reported in Table 2.5) yields NRMSE = 0.0098, while for $N = 4$, the median fitness solution (reported in Table 2.6) yields NRMSE = 0.0114. For sake of completeness, Table 2.7 reports the statistics such as minimum (min), maximum (max), mean and standard deviation (std) related to the NRMSE values over all runs in both considered cases. As shown, the proposed MA produces solutions with a low NRMSE value.

2.4.3 Comparison with other meta-heuristics

After discussing the feasibility of the proposed approach, this section is devoted to show the improvement provided by MA with respect to other meta-heuristics. In order to achieve this aim, this second experimental session involves the comparison between our MA and the Genetic Algorithm (GA) and the Hill Climbing (HC)

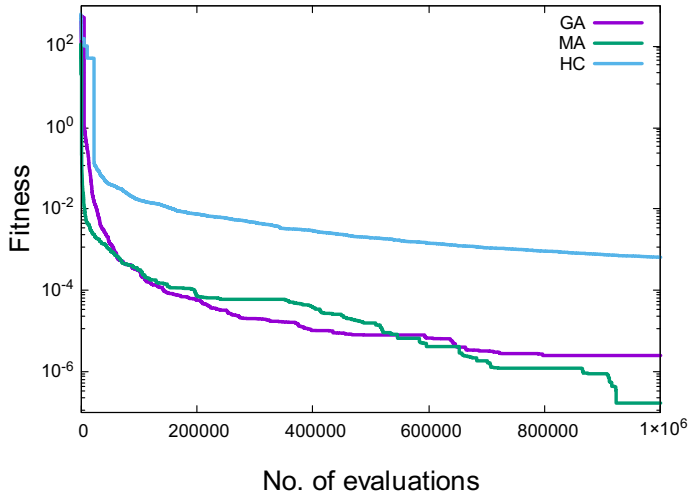
Table 2.7: The statistics for the NRMSE values computed between the analytical solution and the memetic solutions obtained in all executed runs for both considered cases.

Case $N = 3$ and $M = 4$			
min	max	mean	std
0.0038	0.0277	0.0127	0.0044
Case $N = 4$ and $M = 6$			
min	max	mean	std
0.0077	0.0223	0.0134	0.0029

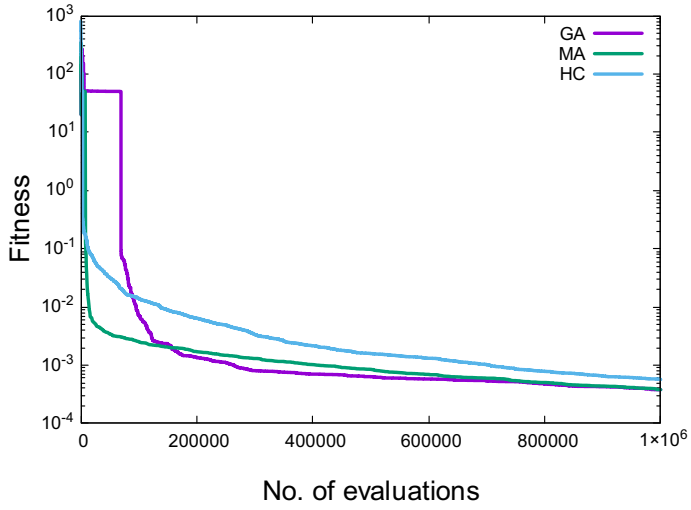
characterized by the same symbolic hyper-parameters (i.e., selection mechanism, crossover, mutation and local mutation) of our approach. This comparison will permit to show that the integration between GA and HC executed by our MA leads to a benefit. As for the values for the numerical hyper-parameters (i.e., mutation and crossover probabilities, local mutation probability) of the compared algorithms, they are the best ones, set up experimentally. All compared algorithms end after evaluating 1,000,000 solutions. The performance of the compared algorithms is assessed by using the fitness value. Therefore, the lower is the fitness value, the higher is the performance.

Fig. 2.7 shows the behaviour of the compared algorithms in a test run for the two considered problem instances. Precisely, the fitness values against the number of evaluations for all compared algorithms are reported. Starting from these figures, it is possible to see how MA achieves a better fitness value with respect to the other meta-heuristics in the same or smaller number of fitness evaluations.

However, as all compared algorithms are characterized by a stochastic nature, in



(a)



(b)

Figure 2.7: Fitness values against the number of evaluations for GA, MA and HC in the cases (a) $N = 3$ and $M = 4$ and (b) $N = 4$ and $M = 6$. The values for each algorithm are obtained from a test run.

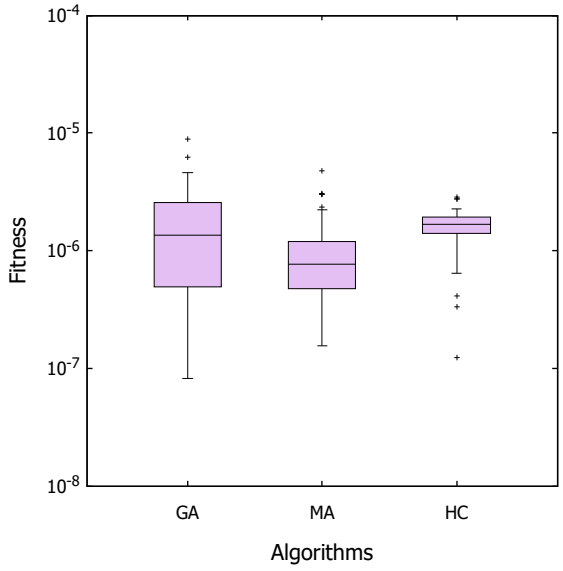
order to better study their behavior we perform 100 runs for each one of them. Fig. 2.8 shows the fitness values obtained on the 100 runs by all compared algorithms for both problem instances. In detail, for the case $N = 3$, the median fitness values for GA, MA and HC are 1.3473×10^{-6}, 7.6298×10^{-7} and 1.6875×10^{-6}, respectively. For the case $N = 4$, the median fitness values for GA, MA and HC are 3.4285×10^{-4}, 1.172×10^{-4} and 5.3624×10^{-4}, respectively. Starting from these results, it is possible to say that MA relatively improves of about 55% on average on GA and of 67% on average on HC. However, in order to give further significance to these results, we perform a pairwise statistical comparison between MA and the compared algorithms independently. In particular, we apply the Wilcoxon's signed rank test (Wilcoxon 1992; García et al. 2009), a non-parametric pairwise test used to answer this question: do two samples represent two different populations? Hence, it can be employed to detect significant differences between the behavior of two algorithms and so it is suitable for our experimentation (Giovanni Acampora, Avella, et al. 2011). In our context, the samples are composed of the fitness values obtained on the 100 runs. The results in terms of p-values computed by the Wilcoxon's signed rank test are reported in Table 2.8. By analysing them, it is possible to say that our memetic approach statistically outperforms the compared meta-heuristics at 99% confidence level (the significance level α has been set to 0.01) for both considered problem instances.

Table 2.8: The p-values computed by the Wilcoxon's signed rank test

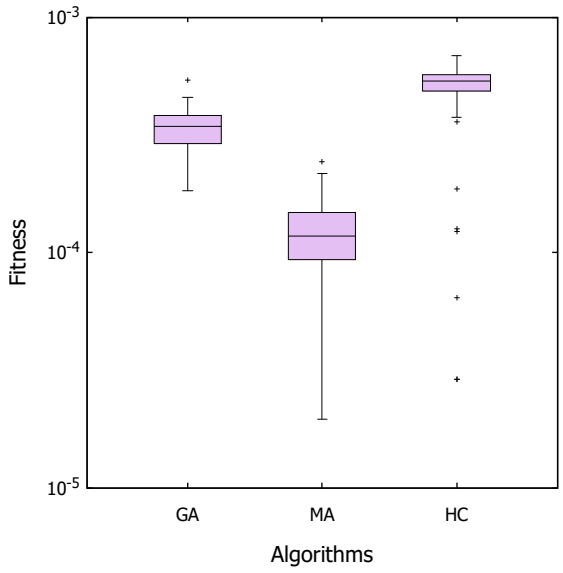
Case $N = 3$ and $M = 4$		Case $N = 4$ and $M = 6$	
MA vs GA	MA vs HC	MA vs GA	MA vs HC
1.0869×10^{-4}	1.5831×10^{-10}	1.9780×10^{-18}	4.5904×10^{-18}

2.5 Conclusions

Using a genetic algorithm, we have mapped the ferromagnetic p-spin Hamiltonian into a Hamiltonian with only 2-body interactions. We have shown, in two analytically solvable cases, that our strategy can successfully be used for this task. We have also discussed and compared the adiabatic quantum computation of the original and the mapped Hamiltonian systems. Further, the mapping is improved by using memetic algorithms, where the performance is enhanced by the local search procedure. For a given number of fitness evaluations, we have made the comparisons of the performances of heuristic techniques of genetic algorithms, memetic algorithms and hill climbing and shown that memetic algorithms outperform the other heuristics considered in terms of the final fitness values. The computational time of evaluating 1,000,000 fitness values in a serial manner for N = 3 is 777.44 s. While for N=4, it is 2864.83 s.



(a)



(b)

Figure 2.8: Fitness values of 100 runs for the compared algorithms (a) in the case $N = 3$ and $M = 4$ (b) in the case $N = 4$ and $M = 6$.

.

3. Genetic optimization of quantum annealing[1]

3.1 Introduction

Small spectral gaps is a bottleneck of adiabatic quantum computation and quantum annealing (Kadowaki and Nishimori 1998; G. E. Santoro et al. 2002; Jörg et al. 2010; Knysh 2016). In these paradigms of quantum computation, the goal is to read the ground state of a target Hamiltonian H_z, encoding an NP-complete or NP-hard problem (Lucas 2014). Starting from the (easy to prepare) ground state $|\psi(0)\rangle$ of a transverse field Hamiltonian $H_x = -\Gamma \sum_{i=1}^{N} \sigma_i^x$, where N is the number of qubits and Γ is the strength of the transverse field, the system is evolved with the time-dependent Hamiltonian $H_0(t) = A(t)H_x + B(t)H_z$. The annealing schedule is given by the pair $\{A(t), B(t)\}$, satisfying $A(0) \gg B(0)$ and $0 = A(T) \ll B(T)$,

[1]The sections of this chapter are published in Pratibha Raghupati Hegde et al. 2022. Reprinted Chapter with permission from [(Pratibha Raghupati Hegde et al. 2022) as follows: Pratibha Raghupati Hegde, Gianluca Passarelli, Annarita Scocco, Procolo Lucignano, Physical Review A, Vol 105, 012612, 2022] License Number:RNP/22/JAN/049647, License Date: 26th Jan 2022 by the American Physical Society.

where T is the annealing time. At $t = T$, the system is found in the target ground state with a high probability, provided T is longer than the inverse square of the smallest gap between the ground state and the first excited state (M. S. Sarandy, L.-A. Wu, and D. A. Lidar 2004). During the dynamics the system may cross a quantum phase transition (Sachdev 2011), correspondingly the gap takes its minimum value $\Delta = \min_t [E_1(t) - E_0(t)]$ which results in long annealing times to satisfy the adiabatic condition, thus making the algorithm ineffective.

If the annealing time T is shorter than that predicted by the adiabatic theorem, the fidelity of the final solution is compromised; if T is longer, the system suffers decoherence. Therefore, the goal here is to modify the annealing dynamics in order to achieve high fidelities even breaking the adiabatic criterion, before decoherence sets in.

This can be achieved by taking advantage of different improved schemes. We mention optimal control theory (Glaser et al. 2015), which is limited, in principle, only by the quantum speed limit (Caneva et al. 2009; Hegerfeldt 2013); shortcuts to adiabaticity (STA) (Torrontegui et al. 2013; Del Campo 2013; Campbell, De Chiara, et al. 2015; Mukherjee, Montangero, and Rosario Fazio 2016; Campbell and Deffner 2017; Abah and Lutz 2018; Funo et al. 2017; Y.-H. Chen et al. 2016; Alan C. Santos and Marcelo S. Sarandy 2015; Coulamy et al. 2016; Alan C Santos and Marcelo S Sarandy 2017; Hu et al. 2018; Alan C. Santos, Nicotina, et al. 2020) or modulating in a controlled way the annealing schedules (Susa and Nishimori 2021; Matsuura, Buck, et al. 2021; Bölte and Thonemann 1996; Roland and Cerf 2002).

A possible STA consists in adopting counterdiabatic (CD) driving (Torrontegui et al. 2013; Del Campo 2013; Campbell, De Chiara, et al. 2015; Mukherjee, Montangero, and Rosario Fazio 2016; Campbell and Deffner 2017; Abah and Lutz 2018; Funo et al. 2017; Y.-H. Chen et al. 2016). In transitionless or CD driving, a time-dependent potential $H_{\mathrm{cd}}(t)$ is added to the unperturbed Hamiltonian $H_0(t)$ so that diabatic Landau-Zener transitions are completely suppressed at all times and for all choices of the annealing time T. The total Hamiltonian reads $H(t) = H_0(t) + H_{\mathrm{cd}}(t)$. The CD operator satisfies the constraint $H_{\mathrm{cd}}(0) = H_{\mathrm{cd}}(T) = 0$ and does not modify the starting and target Hamiltonians. Computing the exact CD potential requires knowledge of the (generally unknown) instantaneous spectrum of the Hamiltonian $H_0(t)$. Moreover, the resulting operator is highly nonlocal, hardly implementable on actual quantum machines, and generally unbounded in the thermodynamic limit (Berry 2009).

Recently, much effort has been put forth to build approximate CD potentials. In some very simple cases, such as the Ising model with longitudinal and transverse fields, linear combination of local operators provide good approximations of the CD potential, e. g., $H_{\mathrm{cd}}(t) \approx \sum_k \alpha_k(t) O_k$. The operators O_k are generally Hermitian products of a small number of Pauli operators. The coefficients $\alpha_k(t)$ can be determined by variational optimization (Sels and Polkovnikov 2017; Hartmann and Lechner 2019). For more complicated many-body Hamiltonians, other choices for operators O_k involve nested commutators between $H_0(t)$ and its time derivative (Claeys et al. 2019). However in the former case, we do not know in advance which and how many local operators are needed to build a good "enough" CD

operator. In the latter case, nested commutators can be highly non-local, as much as the exact CD potential. Moreover, the number of nested commutators is expected to diverge in the thermodynamic limit when the system undergoes a quantum phase transition (G. Passarelli, V. Cataudella, R. Fazio, et al. 2020).

In this chapter, we derive an alternative route and we focus on the study of optimal annealing schedules $A(t)$ and $B(t)$ as well as on an optimal driving (OD) potential $H_{\rm od}(t)$ that are variationally improved so to achieve the maximum fidelity at the final time T. The search for variational minima is approached using computational intelligence tools (Bölte and Thonemann 1996), in particular we adopt a genetic algorithm, i. e., an evolutionary strategy inspired by the Darwinian theory of the survival of the fittest (Yao 1993). We consider time schedules that are polynomial functions of time and we consider local operators for the OD. In our approach, the coefficients of the polynomials and the OD operator are represented as a real-valued chromosome. Each chromosome is characterized by a fitness value. At each generation, chromosomes will mate and randomly mutate. Only the fittest individuals will survive to the next generation. We show that a simple choice of the fitness function can lead to optimized annealing schedules as well as to OD potentials that largely improve the fidelity of the target quantum ground state of H_z, compared to the bare case. We discuss the adiabatic quantum computation of a prototypical system, the ferromagnetic p-spin model, an exactly solvable model with a nontrivial phase diagram, which encodes a Grover-like search (Lov K Grover 1996; Roland and Cerf 2002) for large and odd p.

This chapter is organized as follows. In Sec. 3.2, we describe the ferromagnetic

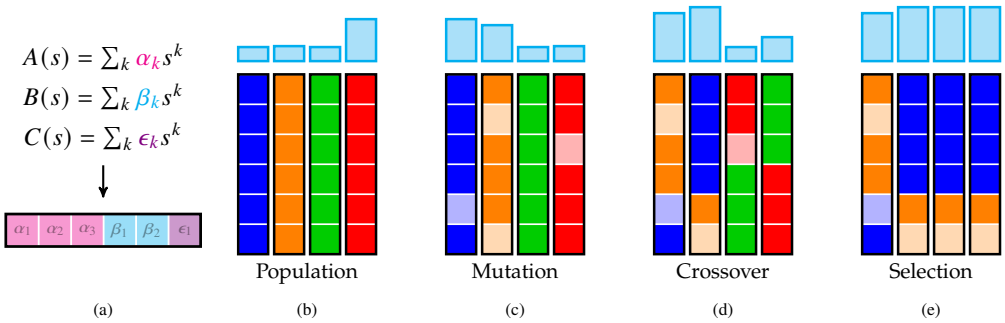


Figure 3.1: Cartoon of our genetic algorithm. a) The free parameters of the annealing schedules are stored into a chromosome. b) We first randomly generate N_{pop} individuals. c) Then random gene mutation occurs in each individual. d) Then we apply two individual crossover. e) We select the fittest individuals and start again from c) until convergence. The azure bars identify the fitness values: the larger the better.

p-spin model. In Sec. 3.3, we introduce the genetic algorithms and the construction of chromosomes for the problems of optimization of annealing schedules and OD potentials. We also define fitness functions for single objective genetic algorithm (SOGAs) and multi-objective genetic algorithms (MOGAs). In Sec. 3.4, we present the results obtained by optimizing the annealing schedules, OD potentials individually and together using genetic algorithms. In Sec. 3.5 we discuss the possibility of extending our techniques to the quantum annealing of random Ising models. We finally derive our conclusions in Sec. 3.6.

3.2 Problem definition

In this chapter, we consider the fully-connected ferromagnetic p-spin model (Derrida 1981; Gross and Mezard 1984) as case study. The Hamiltonian of this model

is

$$H_z = -JN\left(\frac{1}{N}\sum_{i=1}^{N}\sigma_i^z\right)^p, \tag{3.1}$$

with $J > 0$ and $p \geq 2$. For odd p, its ground state is ferromagnetic with all qubits in the state $|0\rangle$. For even p, the ground state manifold is 2-dimensional ($|00\cdots0\rangle$ and $|11\cdots1\rangle$) due to the Z_2 symmetry. If we study the quantum annealing with time-dependent Hamiltonian $H_0(t)$ using as a target Hamiltonian H_z defined in Eq. (3.1), we observe a dynamical quantum phase transition separating a paramagnetic phase (at short times) from a ferromagnetic phase (at long times). For $p = 2$, the quantum phase transition is of second order, while for $p \geq 3$ it is of first order. The latter is the hardest case for quantum annealing, as the minimal gap Δ closes exponentially as a function of N (Bapst and Semerjian 2012). This feature motivates the broad interest in this system as a toy model of NP-hard problems (Seoane and Nishimori 2012; Seki and Nishimori 2012; Ohkuwa, Nishimori, and Daniel A. Lidar 2018a; Matsuura, Nishimori, Vinci, Albash, et al. 2017; Susa, Yamashiro, et al. 2018; Matsuura, Nishimori, Vinci, and Daniel A. Lidar 2019; Yamashiro et al. 2019; G. Passarelli, De Filippis, et al. 2018; G. Passarelli, V. Cataudella, and Lucignano 2019; Giovanni Acampora, Vittorio Cataudella, Pratibha R Hegde, et al. 2019; G. Passarelli, V. Cataudella, R. Fazio, et al. 2020; Gianluca Passarelli, Yip, et al. 2020).

The model Hamiltonian is permutationally invariant and commutes with the total spin operator S^2 at all times. The starting and the target state belong to the subspace with maximum spin $S = N/2$ and the dynamics will occur within the

same (maximum spin) subspace. Therefore, we can work in this $(n = N+1)$-dimensional sector. In the following, we will consider J as unit of energy. Times are expressed in units of J^{-1} ($\hbar = 1$ here and in the following).

We perform adiabatic evolutions of the system described by the p-spin model assisted by genetic algorithms. We aim at improving the final state fidelity of the system by following three strategies: a) optimizing annealing schedules b) optimizing local OD with the traditional linear annealing schedules and c) optimizing both annealing schedules and local OD operator. These strategies are explained in detail later in the chapter, see Sec. 3.3. Further, we choose an annealing time sufficiently shorter than the timescale T_{AD} predicted by the adiabatic theorem, i. e.,

$$T_{\mathrm{AD}} = \max_{\lambda\in[0,1]} \frac{|\langle E_0(\lambda)|\partial_\lambda H(\lambda)|E_1(\lambda)\rangle|}{|E_1(\lambda)-E_0(\lambda)|^2}, \qquad \lambda = t/T. \tag{3.2}$$

3.3 Methods: Genetic algorithms

We use a class of evolutionary algorithms known as genetic algorithms to find optimized annealing schedules for adiabatic evolutions. In addition, we also manage to demonstrate the efficiency of genetic algorithms in the paradigm of shortcuts to adiabaticity by finding optimized, local OD operators.

Genetic algorithms are inspired by Darwin's theory of evolution. These algorithms offer solutions to optimization problems conditioned by a single objective or multiple objectives (Giovanni Acampora, Vittorio Cataudella, Pratibha R Hegde, et al. 2019; Giovanni Acampora, Vittorio Cataudella, Pratibha Raghupati Hegde,

et al. 2021; Deb et al. 2002; Fortin et al. 2012). In both cases, the possible solutions to the problem are encoded as a string of real numbers called chromosomes. The construction of a chromosome depends on the optimization problem. In this article, broadly speaking, we address three optimization problems, all of which aid the performance of adiabatic evolution, i. e., finding the system in a ground state of the problem Hamiltonian H_z with maximum probability at the end of the evolution. The three problems are as follows.

3.3.1 Optimization of annealing schedules

Here we try to optimize the performance of quantum annealing by optimizing its annealing schedules $A(t)$ and $B(t)$ (Matsuura, Buck, et al. 2021; Bölte and Thonemann 1996; Susa and Nishimori 2021). Firstly, we express the annealing schedules as dimensionless time functions of $s = t/T$ throughout this chapter. We consider polynomial expansions of $A(s)$ and $B(s)$ as candidates for the possible annealing time schedules, i. e., $A(s,\alpha) = \sum_{i=1}^{k_a+1} \alpha_i s^i$, $B(s,\beta) = \sum_{j=1}^{k_b+1} \beta_j s^j$. Moreover, these time-dependent functions have to satisfy the boundary conditions, $A(0) = 1, A(1) = 0$ and $B(0) = 0, B(1) = 1$, and therefore can expressed as

$$\begin{aligned} A(s,\alpha) &= 1 + \sum_{i=1}^{k_a} \alpha_i s^i + \left(-1 - \sum_{i=1}^{k_a} \alpha_i \right) s^{k_a+1}, \\ B(s,\beta) &= \sum_{j=1}^{k_b} \beta_j s^j + \left(1 - \sum_{j=1}^{k_b} \beta_j \right) s^{k_b+1}. \end{aligned} \tag{3.3}$$

We optimize the coefficients of these polynomial expansions as chromosomes of

the genetic algorithm and the structure of the chromosome for this problem is

$$D_1 = \left[\alpha_1, \alpha_2, \cdots \alpha_{k_a}, \beta_1, \beta_2, \cdots \beta_{k_b}\right]. \tag{3.4}$$

The length of the chromosome is $k_a + k_b$.

3.3.2 Optimization of the local OD operator

In this section, we adopt the strategy of shortcuts to adiabaticity to optimize the performance of quantum annealing (Berry 2009; Sels and Polkovnikov 2017; Claeys et al. 2019; G. Passarelli, V. Cataudella, R. Fazio, et al. 2020; Hartmann and Lechner 2019). Keeping the annealing schedules to be fixed and as linear functions, i. e., $A(s) = 1 - s$ and $B(s) = s$, we optimize an OD operator which successfully avoids Landau-Zener transitions resulting in a better fidelity of the state of the system with the exact ground state at $t = T$. We assume that the OD operator $H_{\mathrm{od}}(s)$ can be expanded as the sum of local spin operators,

$$H_{\mathrm{od}}(s, \gamma) = C(s) \sum_{i=1}^{d} \gamma_i O_i, \tag{3.5}$$

where O_i are the total spins along the x, y and z directions, i. e., S_x, S_y and S_z, and their products. Especially, we consider only single local operators and cumulatively add the set of all possible 2-spin operators and the set of all 3-spin operators. These

local operators can be explicitly written as

$$\begin{aligned} H_{\mathrm{od}}(s,\gamma)_{d=3} &= C(s)\sum_{i=1}^{3}\gamma_i S_i, \\ H_{\mathrm{od}}(s,\gamma)_{d=9} &= H_{\mathrm{od}}(s)_{d=3} + C(s)\sum_{i,j=1}^{3}\gamma_{i,j} S_i S_j, \\ H_{\mathrm{od}}(s,\gamma)_{d=21} &= H_{\mathrm{od}}(s)_{d=9} + C(s)\sum_{i,j,k=1}^{3}\gamma_{i,j,k} S_i S_j S_k. \end{aligned} \tag{3.6}$$

The chromosome of the genetic genetic algorithm for this problem is the set of coefficients of the local operators,

$$D_2 = [\gamma_1, \cdots, \gamma_3, \gamma_{11}, \gamma_{12}, \cdots, \gamma_{33}, \gamma_{111}, \gamma_{112}, \cdots, \gamma_{333}], \tag{3.7}$$

whose length is equal to the number of local operators, d. In this work, we are able to achieve remarkable results by optimizing the local OD operator with only single spin operators, i. e., $H_{\mathrm{od}}(s,\gamma)_{d=3} = C(s)\sum_{i=1}^{3}\gamma_i S_i$, and therefore we discuss and demonstrate our results for the case with $d = 3$. The higher terms of 2-spin and 3-spin operators are omitted since they do not produce any significant improvements. The time schedule $C(s)$ is fixed in this approach and is given by $C(s) = A(s)B(s) = (1-s)s$. The function $C(s)$ controls the pace of evolution dictated by the OD operator $H_{\mathrm{od}}(s)$.

3.3.3 Optimization of the time schedules and the local OD

Finally, here we optimize the annealing schedules $A(s)$, $B(s)$ and the local OD operator altogether (Susa and Nishimori 2021; Matsuura, Buck, et al. 2021). The time schedule $C(s)$ is optimized by absorbing it as the coefficients of the local OD operators, i. e., $H_{\rm od}(s) = \sum_{i=1}^{d} C_i(s) O_i$. We consider each $C_i(s)$ to be a polynomial of order $k_c + 1$, which satisfies the boundary condition $C_i(0) = 0$ and $C_i(1) = 0$. Therefore, the OD operator can be explicitly written as

$$H_{\rm od}(s,\varepsilon) = \sum_{i=1}^{d} \left(\sum_{j=1}^{k_c} \varepsilon_{ji} s^j + \left(- \sum_{j=1}^{k_c} \varepsilon_{ji} \right) s^{k_c+1} \right) O_i. \tag{3.8}$$

We optimize the free parameters ε_{ji}, in addition to the free parameters α_i and β_i in Eq. (3.4). The time-dependent Hamiltonian of the system is given by

$$H(s) = A(s,\alpha) H_x + B(s,\beta) H_z + H_{\rm od}(s,\varepsilon), \tag{3.9}$$

where

$$\alpha = \{\alpha_1, ..., \alpha_{k_a}\}$$

$$\beta = \{\beta_1, ..., \beta_{k_b}\}$$

$$\varepsilon = \{\varepsilon_{11}, ..., \varepsilon_{k_c d}\}.$$

Therefore, the chromosome for this optimization problem can be expressed as

$$D_3 = \left[\alpha_1, \cdots \alpha_{k_a}, \beta_1, \cdots, \beta_{k_b}, \varepsilon_{11}, \cdots \varepsilon_{k_c 1}, \cdots \varepsilon_{k_c d}\right]. \tag{3.10}$$

The length of the chromosome in this case is $k_a + k_b + (d \times k_c)$. Again here, we are able to obtain high fidelity of the state of the system by considering only single spin operators in the expansion of the local OD operator. Therefore we stick to the case of $d = 3$.

The key aspect of genetic algorithms is the definition of fitness function. It is a function which takes each chromosome as a variable and gives it a fitness value according to the quality of the solution generated by the given chromosome. In the course of a genetic algorithm, we intend to either maximize or minimize this fitness function. Depending on the number of conditions the chromosomes have to satisfy, the genetic algorithms are characterized by fitness functions which are single objective or multi-objective. In the remaining of this section, we describe the fitness function and the workflow of Single Objective Genetic Algorithms (SOGAs) and Multi-Objective Genetic Algorithms (MOGAs).

3.3.4 Single objective genetic algorithms

SOGAs follow the workflow of a standard genetic algorithm. We define the fitness of each chromosome as the fidelity, which is the ground state probability at $t = T$, i. e.,

$$f_{\text{so}} \equiv P_{\text{gs}}(T) = |\langle E_0(T)|U(T)|\psi(0)\rangle|^2, \tag{3.11}$$

where $U(t,0) = \mathscr{T}_+ \exp\{-i\int_0^t [H(t')]dt'\}$ is the time evolution operator and $\mathscr{T}_+$ is the time ordering. The time evolution is computed with the QuTiP toolbox (Johansson, Nation, and Nori 2012; Johansson, Nation, and Nori 2013). An alternative

fitness function would be to use the negative of the mean energy at the final time T, i. e., $-\langle\psi(0)|U^{\dagger}(T,0)H_z U(T,0)|\psi(0)\rangle$. This choice does not require the knowledge of any spectral property of the Hamiltonian. The fittest individuals, maximizing $f_{\rm so}$, are those with higher fidelities and are likely to survive along generations. At the end of the genetic optimization, we will obtain a chromosome defined according to the problem. However, all three problems considered here aim at giving a higher fidelity.

We initialize a starting population of $N_{\rm pop}$ individuals, whose genes are randomly extracted in the interval $[g_{\rm min}, g_{\rm max}]$. Then, we repeatedly apply the three genetic operators (mutation, crossover and selection (Herrera, Lozano, and Sánchez 2003)) until a convergent solution is achieved. The genetic algorithm is implemented by using the DEAP package (Fortin et al. 2012). Here, we briefly describe the genetic operators adopted, also sketched in Fig. 3.1.

i) Gaussian mutation—Among the population of individuals, random individuals are selected with a probability p_m for mutation. Each gene is independently mutated with a probability $p_{\rm ind}$, by adding a normal variable, extracted from a Gaussian with mean value $\mu = 0$ and variance $\sigma^2 = 1$ [see Fig. 3.1(c)]. The mutation probability of each gene, i. e., the product $p_m p_{\rm ind}$, should be neither too large nor too small (a quantitative description is given in Appendix D.1). In the former case, the genetic algorithm will turn into a random search. In the latter case, the algorithm would be nonergodic. These random mutations increase variability in the population and reduce the probability of being trapped in local minima.

ii) Two-point crossover—After mutation process, we randomly select two parents

from the chromosome population. Two random integers are randomly extracted in the interval $[0, L-1]$, where L is the length of the chromosome, which is the number of free parameters to be optimized using a genetic algorithm and is problem-specific. Two children are produced by mixing alternating parts of the two parents, obtained by cutting the chromosomes at the two extracted indices [see Fig. 3.1(d)]. Note that the exchange of the fragments is only symbolic in Fig. 3.1(d) and represents a one-point crossover for the sake of visual clarity. In our experiments, we resort to a two-point crossover operator which yields the fastest convergence in this case. The whole process occurs with a probability p_c. Small p_c ensures slow but accurate convergence to the optimal solution. On the other hand, large p_c ensures quick convergence but can lead to sub optimal solutions. Hence, p_c has to be carefully tuned to find a compromise between speed of convergence and accuracy of the solution.

iii) Selection by tournament—After mutation and crossover, a new population is produced. N_{T} competitors are selected from the population and their fitness is compared [see Fig. 3.1(e)]. Only the fittest individual survives to the next generation. This tournament is repeated until we obtain a new set of N_{pop} individuals.

3.3.5 Multi-objective genetic algorithms

While SOGAs aim at maximizing the ground state probability at the final time T, they sometime lead to practically not feasible solutions during the time of evolution. For example, some of the solutions returned by the algorithm can have energy level crossings between the ground state and the first excited state. In an

attempt to avoid these solutions produced by SOGAs, we add another objective to the fitness function. Other than maximizing the fidelity at $t = T$, we choose to maximize it together with the area under the curve of the instantaneous ground state probabilities of the system computed at N_t time intervals. The latter assures that the ground state occupation is maximum at all the intermediate times, in the spirit of counterdiabatic dynamics. The ground state probability at time t is given by $P_{\rm gs}(t) = |\langle E_0(t)|U(t)|\psi(0)\rangle|^2$. The fitness of a chromosome in MOGA are defined as

$$f_{\rm mo} \equiv \left\{ \frac{1}{T}\int_0^T P_{\rm gs}(t)\,dt, P_{\rm gs}(T) \right\}. \tag{3.12}$$

We stress here the fact that this is not the same as imposing local adiabaticity as by Roland and Cerf (2002). MOGAs deviate from the standard genetic algorithms. In particular, they work using the strategy of Non-dominated Sorting Genetic Algorithm II (NSGA-II) (Deb et al. 2002; Chivilikhin et al. 2020). NSGA-II uses an elitist method of evolutionary algorithms. The parent and offspring generations are clubbed together and are ranked into fronts based on non-dominated sorting. The population of the following generation is filled with the best fronts until $N_{\rm pop}$ is reached. In case that only some chromosomes have to be selected from a front in the process, the most diverse solutions are chosen based on the crowding distance. Given the new population, by the above non-dominated sorting process, the chromosomes undergo selection (a binary tournament selected based on both rank and crowding distance), mutation and crossover processes. In the end of $N_{\rm g}$ generations, the Pareto optimal front with the best ranking is obtained. The

details of selecting the chromosome from the Pareto optimal front is given in Appendix D.2.

3.4 Results

In this section we present the results obtained by performing adiabatic quantum computation of the ferromagnetic p-spin model assisted by genetic algorithms. In particular, we concentrate on a system with 15 spins and $p = 3$ to demonstrate our results. The adiabatic time scale of Eq. (3.2) for this system is $T_{\mathrm{AD}} \approx 30$. We choose the annealing time $T = T_{\mathrm{AD}}/10 \approx 3$ in order to be far from adiabaticity. Throughout the time evolution, we store the data of energy gaps between the ground state and the first excited state, time schedule function values and ground state probabilities. We initiate the genetic algorithm with a population of $N_{\mathrm{pop}} = 20$ individuals, and run it for a large enough number of generations until the algorithm gives convergent values. When implementing genetic algorithms, it is advisable to perform an initial experimentation to optimize the hyperparameters involved in mutation, crossover and selection processes. The details of this procedure are given in Appendix D.1. We consider the optimal hyperparameters to repeatedly perform genetic algorithms and to analyse the results obtained from their solutions. With the optimized annealing schedules and an optimal driving (OD) operator, the Schrödinger equation is solved in the time domain $[0, T]$ and sampled at 100 evenly spaced points in this interval. The system is initialized in the ground state of H_x. When we optimize the annealing schedules, we evolve the Schrödinger equation

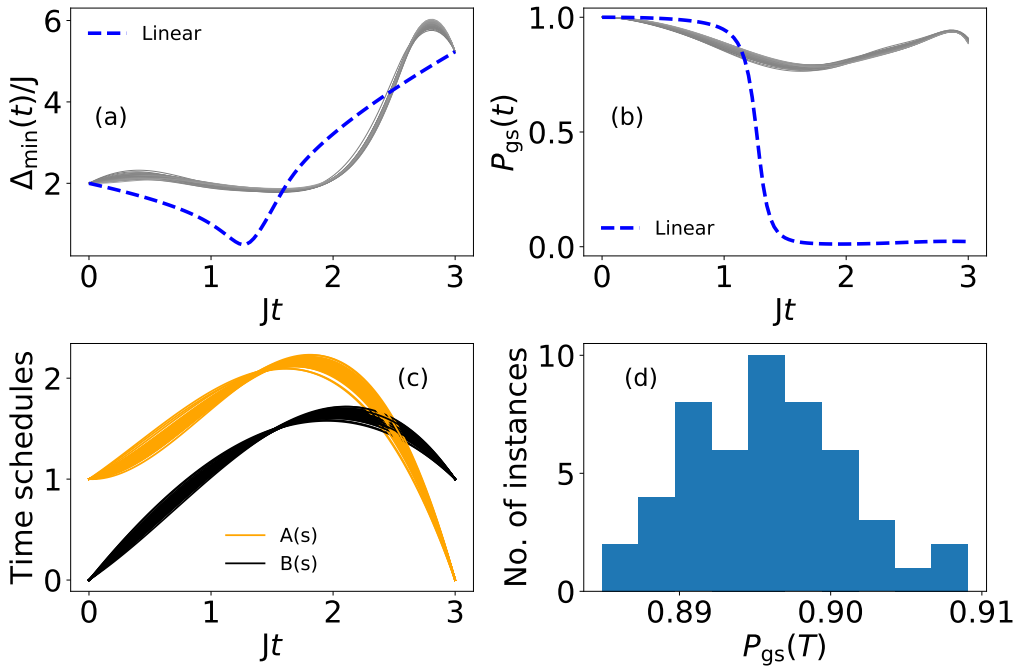


Figure 3.2: Summary of the results obtained by using the optimized annealing schedules for solving the p-spin model using SOGA. We investigate the system with 15 spins with both the annealing schedules $A(s)$ and $B(s)$ expanded up to a degree of 3. In other words, $k_a = k_b = 2$. In panel (a), we show the instantaneous energy gaps during the dynamics of the adiabatic evolution, in panel (b) we show the instantaneous ground state probability using the optimised schedules and by using a simple linear schedule, in panel (c) we show the annealing schedules $A(s)$ and $B(s)$, and in panel (d) we show the histogram of the fidelities for 50 runs of the algorithm.

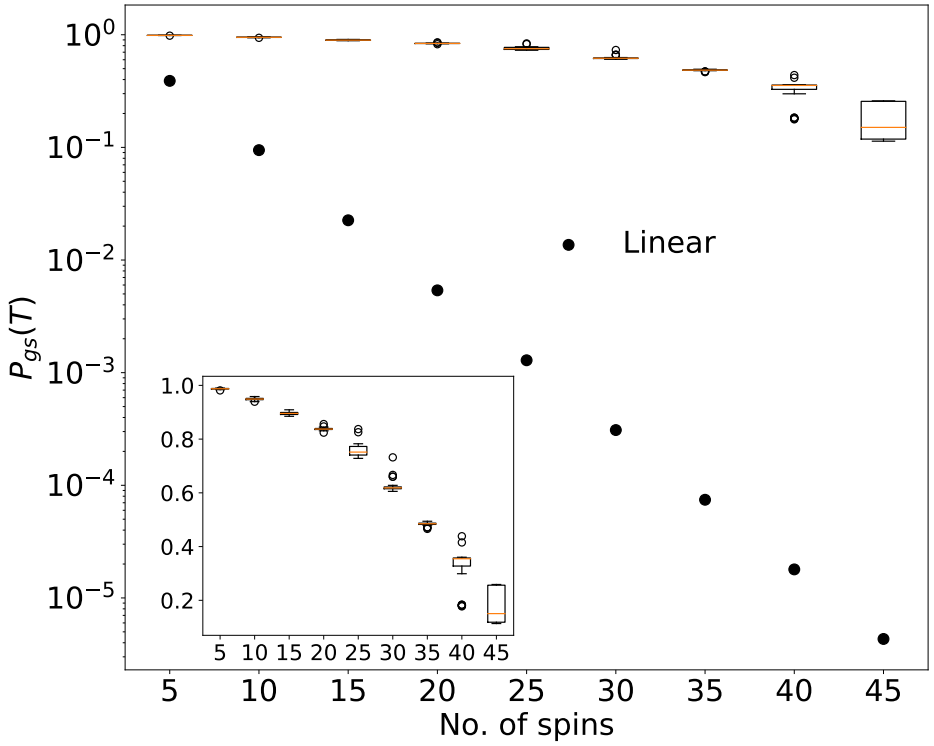


Figure 3.3: Boxplot of fidelities of the states of systems with different sizes. Each box represents the first quartile and the third quartile and the red line represents the median of the data for 50 runs of SOGA which optimizes the annealing schedules $A(s)$ and $B(s)$, each of which is expanded up to a third-degree polynomial.

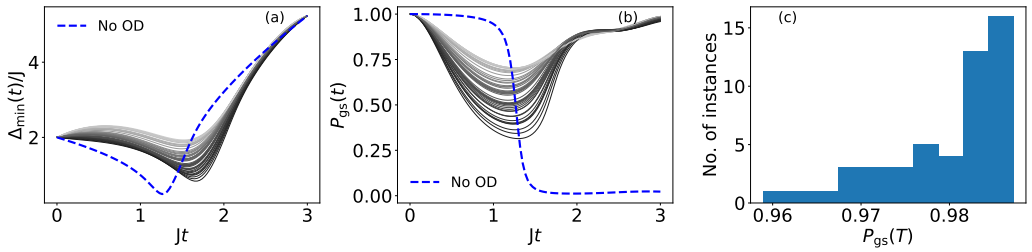


Figure 3.4: Results obtained by optimizing the time independent part of the local OD operator for the ferromagnetic p-spin model with 15 spins and $p = 3$. The OD operator chosen is $H_{\text{od}}(s,\gamma)_{d=3}$. The plots depict the data for 50 runs of the SOGA and the corresponding results obtained by adiabatic quantum computation. a) Instantaneous energy gaps between the ground state and the first excited state. b) Instantaneous ground state probabilities. c) Histogram of the fidelities for 50 instances.

with the Hamiltonian in Eq. (3.9), but without the optimal driving $H_{\text{od}}(s,\varepsilon)$ term. In the case of optimal driving optimization, we evolve the Schrodinger equation with the Hamiltonian in Eq. (3.9). The ground state probability of the system is computed along the genetically optimized path of quantum annealing. The Schrödinger equation evolution is simulated using the QuTiP library (Johansson, Nation, and Nori 2012; Johansson, Nation, and Nori 2013). Further, we repeat the genetic algorithms 50 times and compute the corresponding results pertaining the dynamics of the system. Hereafter, we present the results obtained by using the three strategies assisted by SOGA. We discuss the cases where MOGAs can be opted over SOGAs in order to obtain meaningful results. Further, we test our methods with systems of varying number of spins.

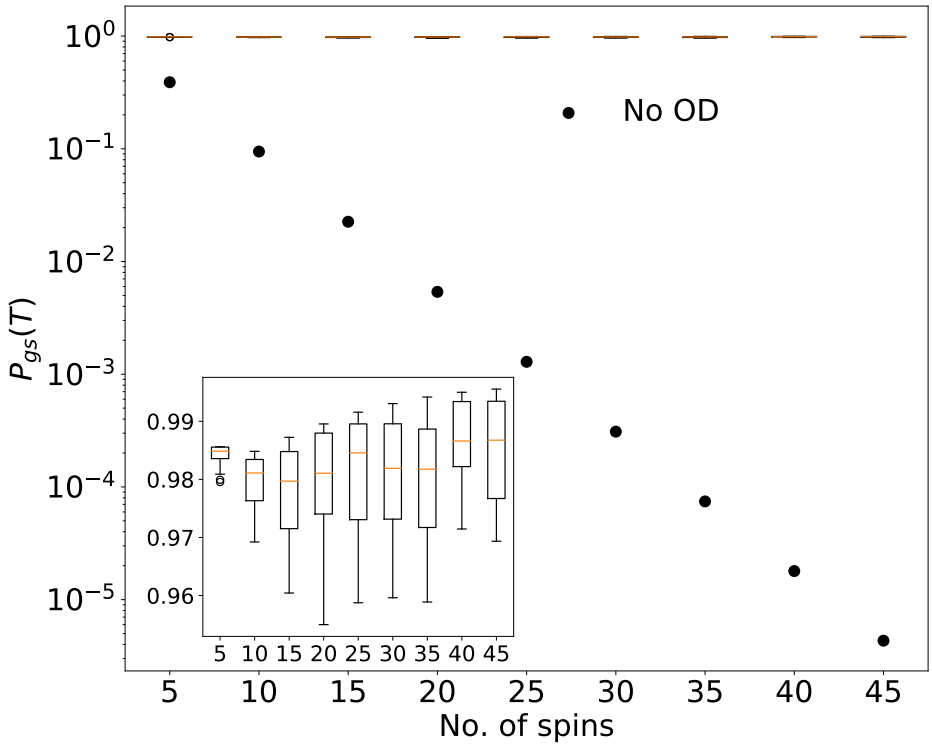


Figure 3.5: Boxplot of fidelities of the states of systems with different sizes. Each box represents the data for 50 runs of SOGA iterated for 1000 generations, which optimizes the time independent part of the OD operator with $d = 3$.

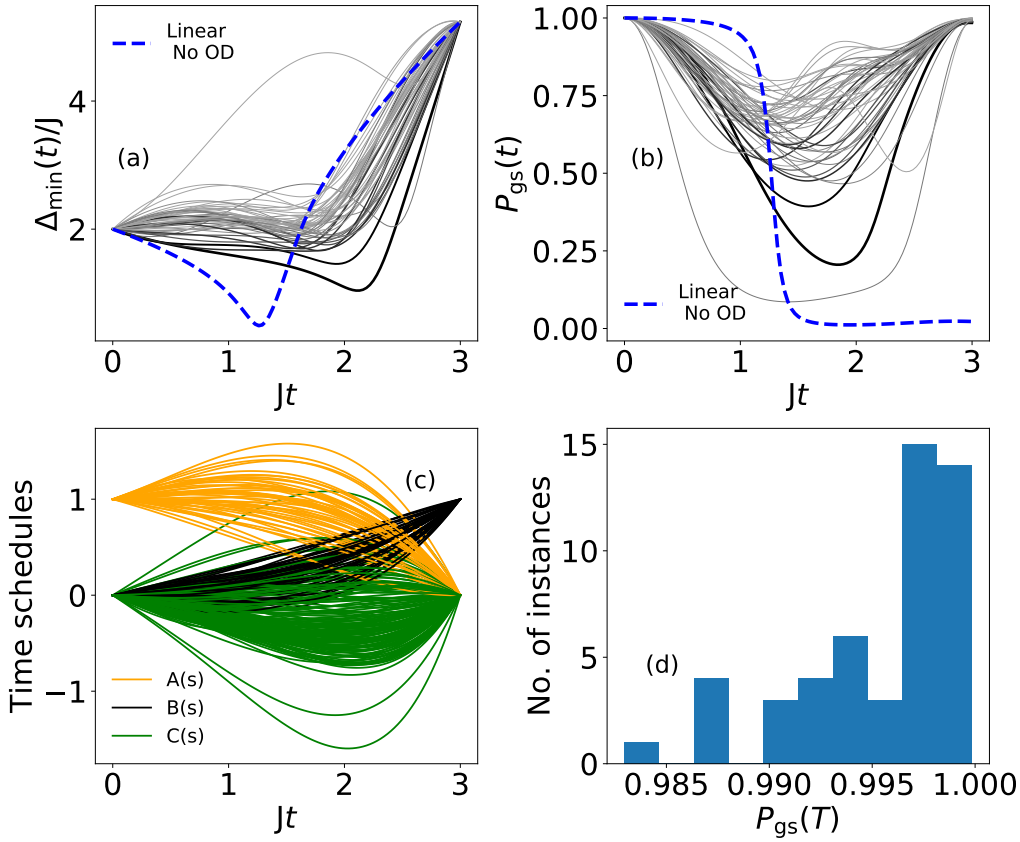


Figure 3.6: Results obtained by optimizing the annealing schedules $A(s)$, $B(s)$ and time dependent local OD operator for the ferromagnetic p-spin model with 15 spins and $p = 3$. The OD operator chosen is $H_{\text{od}}(s,\gamma)_{d=3}$, and $k_a = k_b = 2$ and $k_c = 3$. The plots depict the data for 50 runs of the SOGA and the corresponding results obtained by adiabatic evolution. a) Instantaneous energy gaps between the ground state and the first excited state. b) Instantaneous ground state probabilities. c) Time schedule functions, $A(s)$, $B(s)$ and $C(s)$ d) Histogram of the fidelities for 50 instances.

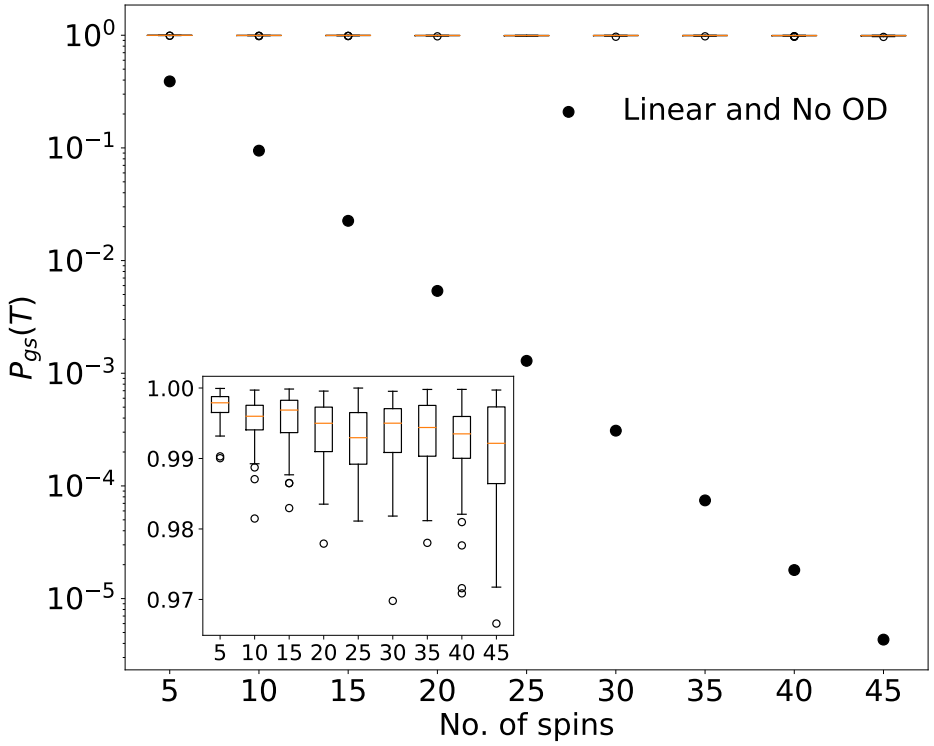


Figure 3.7: Boxplot of fidelities of the states of systems with different sizes. Each box represents the data for 50 runs of SOGA iterated for 1000 generations which optimizes the annealing schedules $A(s)$ and $B(s)$, and the scheduling of local OD operator $[H_{\text{od}}(s,\gamma)_{d=3}]$, with $k_a = k_b = 2$ and $k_c = 3$.

3.4.1 Optimization of annealing schedules $A(s)$ and $B(s)$

As described in Sec. 3.3, we optimize the annealing schedules by encoding the coefficients of the polynomials in Eq. (3.3) as chromosomes D_1 in Eq. (3.4). With the optimal annealing schedules given by the genetic algorithms, we simulate the adiabatic quantum computation. We focus on the cases when $k_a = 2$ and $k_b = 2$ and hence the length of the chromosome is 4. We have run the algorithm for 5000 generations.

The summary of the results by optimizing the annealing schedules using SOGA is provided in Fig. 3.2. In Fig. 3.2(a), we show that the optimized path increases the minimum gap between the ground state and the first excited state only slightly: energy scales remain within practical limits. Meanwhile, in Fig. 3.2(b), the ground state probabilities remain higher throughout the evolution and, around the final time, there is a slight drop in the fidelity. This could possibly be overcome by using MOGA by imposing a condition in the fitness function that the derivative of the ground state probability evolution curve remains smaller. In Fig. 3.2(c), we show the annealing schedules optimized by the SOGA. We see that both the schedules $A(s)$ and $B(s)$ increase to a value larger than one and gradually decrease to their respective boundary values, as opposed to the traditionally used monotonic functions (M. S. Sarandy, L.-A. Wu, and D. A. Lidar 2004; Giovanni Acampora, Vittorio Cataudella, Pratibha R Hegde, et al. 2019). We point out that the nature of our optimized time schedules is different from the exact solution of annealing schedule function derived for example in Ref. (Roland and Cerf 2002). This is due to the fact that we do not impose the local adiabaticity at all points of time,

but only at the final time. Fig. 3.2(d) shows the histogram of fidelities for 50 runs of the algorithm. Fidelities are distributed in a small window with median value of the distribution being ≈ 0.895, which is about two orders of magnitude higher with respect to the linear schedule.

To conclude this section, we study the genetic optimization of annealing schedules for varying system sizes. In Fig. 3.3, we fix the chromosome length to be 4, and we run the genetic algorithms for 5000 generations for system sizes up to 45 spins. We plot the fidelities of the adiabatic evolution as a box plot where each box represents the interval between the first and third quartiles and the red line is the median fidelity over 50 repetitions. The solutions by the genetic algorithms decrease for larger system sizes. However, the performance is strikingly better than the corresponding results using linear annealing schedules, by several orders of magnitude.

3.4.2 Optimization of OD

Here we optimize the local OD operators alone fixing linear annealing schedules as described in Sec. 3.3. The chromosome is D_2 in Eq. (3.7). We focus on optimizing the local operators with only single spin operators $H_{\mathrm{od}}(s,\gamma)_{d=3}$ from Eq. (3.6) and show that optimizing only 3 parameters, we obtain good fidelities. The higher number of local terms lead to many trivial solutions of simply increasing the energy scale of the system beyond practical capabilities due to the large solution space, at the same time being computationally expensive.

The summary of the results obtained by genetic optimization of $H_{\mathrm{od}}(s,\gamma)_{d=3}$ are

shown in Fig. 3.4. In Fig. 3.4(a) we show the energy gaps between the ground state and the first excited state. The minimum energy gap is slightly higher than the original system driven with no OD potentials. In panel (b), we show the corresponding results of the evolution of ground state probabilities. Even though the ground state probabilities are comparatively lower during the evolution, the fidelities are high at the final time. The probabilities can be controlled to be higher also during the evolution using MOGA. The results are not very diverse due to the small chromosome size and yet these set of solutions are feasible. Finally in Fig. 3.4, we show the distribution of fidelities for 50 solutions of SOGA. All the solutions show very high fidelity with the median value of ≈ 0.98. We analyzed the data of optimized chromosomes to understand the contribution of each of the local operator term in the expansion of optimized local OD operator. The contribution of S_y is larger for all the cases considered, which is expected since the S_y term is the starting point for many known expansions of the OD operator (Claeys et al. 2019; Sels and Polkovnikov 2017; G. Passarelli, V. Cataudella, R. Fazio, et al. 2020; Hartmann and Lechner 2019).

We verify the robustness of the genetic optimization approach in OD driving for larger system sizes. In Fig. 3.5 we compare the fidelities of states of the systems up to 45 spins. The fidelities are very high despite increasing the number of spins by optimizing only single spin operators (i. e., $d = 3$). Nevertheless, when we increase the size of the system, some of the solutions given by the genetic algorithms lead to energy level crossings between the ground state and the first excited state. The corresponding ground state probabilities fall to very low values in these points and

regain better values towards the end of the evolution. However, this is an unphysical scenario. We resort to MOGA in this case, which makes sure the ground state probabilities are higher throughout the evolution by avoiding the situations of energy crossings. An example of improvement of the results using MOGA for a system with 40 spins is demonstrated in Appendix D.2.

3.4.3 Optimization of time schedules and the local OD

Here we optimize the free parameters of the time schedules $A(s,\alpha)$, $B(s,\beta)$, and $C(s,\varepsilon)$ all together as a chromosome D_3 in Eq. (3.10). We choose $k_a = 2$, $k_b = 2$ and the number of local operators $d = 3$, each accompanied by a time schedule $C_i(s,\varepsilon)$ as described in Eq. (3.8) with $k_c = 3$. It is sufficient to run the algorithm up to 1000 generations in this case in order to obtain convergent results.

Fig. 3.6 shows the summary of the results obtained by optimizing all the time schedules in the realm of shortcuts to adiabaticity. In Fig. 3.6(a), we show the minimum energy gaps. In this case, the solutions are quite diverse because of the larger search space. The same is reflected in the evolution of ground state probabilities in Fig. 3.6(b). In Fig. 3.6(c), we show the optimized annealing schedules. While some of the solutions show the same increase and decrease patterns seen in the previous case, some others are monotonic between the boundary values. The schedules $C(s)$ plotted in green color are composed of the three time functions $\{C_1(s), C_2(s), C_3(s)\}$ of each of the local operators in the expansion of the OD potential. We show the distribution of fidelities in the solutions given by the genetic algorithm in Fig. 3.6(d). The fidelities are exceptionally higher with a

median value of ≈ 0.997.

In Fig. 3.7, we compare the fidelities of adiabatic quantum computation assisted by genetic algorithms for varying system sizes. Here we have fixed the chromosome length to be 13 and we run the algorithm for 1000 generations for all the cases. The performance of genetic optimization is consistently higher even for larger system sizes.

3.5 Generalization to random Ising models

In order to test the feasibility of our method in a more general framework, we additionally studied the performance of the genetic optimization for a random Ising model. We considered a system of $n = 5$ qubits arranged in the graph shown in Fig. 3.8, described by the following Hamiltonian,

$$H_z = H_{\mathrm{I}} = \frac{1}{2}\sum_{\langle ij\rangle}\left(\mathbb{1} - J_{ij}\sigma_i^z\sigma_j^z\right), \tag{3.13}$$

where the sum acts on qubits connected by the graph bonds and the couplings J_{ij} are random uniform variables in $[-1,1]$. The idea here is to apply the genetic routine to a family of randomized models so as to see if some general features of optimized annealing schedules/OD operators can be inferred. This would allow us to significantly speed up computation since it would remove the need to repeat the genetic optimization on an instance-by-instance basis. We generated $N_{\mathrm{inst}} = 50$ random instances and repeated the (stochastic) genetic optimization $N_{\mathrm{rep}} = 30$

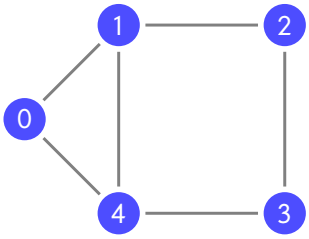


Figure 3.8: Graph of the Ising model discussed in Sec. 3.5.

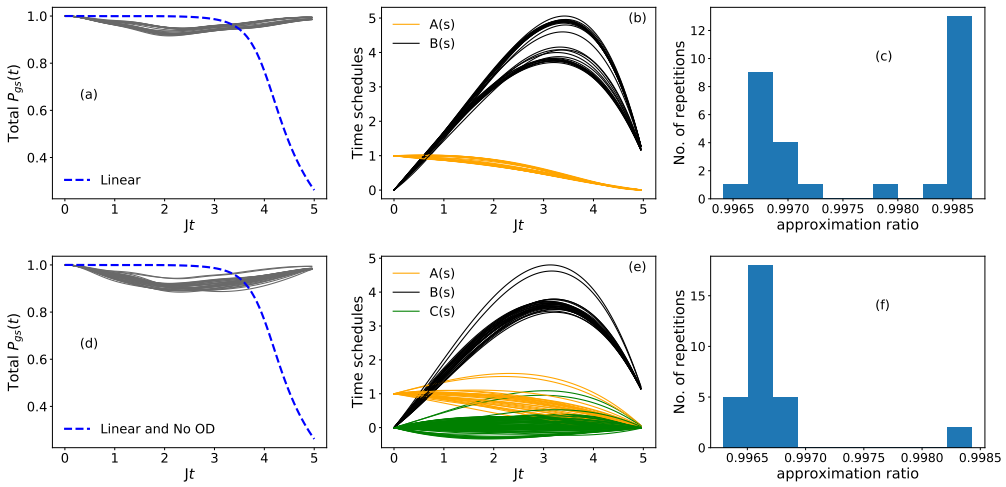


Figure 3.9: Results of genetic optimization of quantum annealing of a random Ising model (Eq. (3.13)), for a typical random instance. The plot shows the data of 30 genetic optimizations of the considered random instance. Top panel depicts the optimization of annealing schedules alone, with the parameters, $k_a = k_b = 3$. Here, (a) Instantaneous total probability of obtaining degenerate ground states using optimized polynomial schedules vs using linear schedules (b) Optimized annealing schedules $A(s)$ and $B(s)$ (c) Approximation ratios of 30 genetic optimizations of the given random instance. In the bottom panel, (d), (e) and (f) are the corresponding results obtained by optimizing annealing schedules and OD operator($H_{\text{od}}(s,\gamma)_{d=3}$) together. The parameters considered in this case are $k_a = k_b = 2$ and $k_c = 3$.

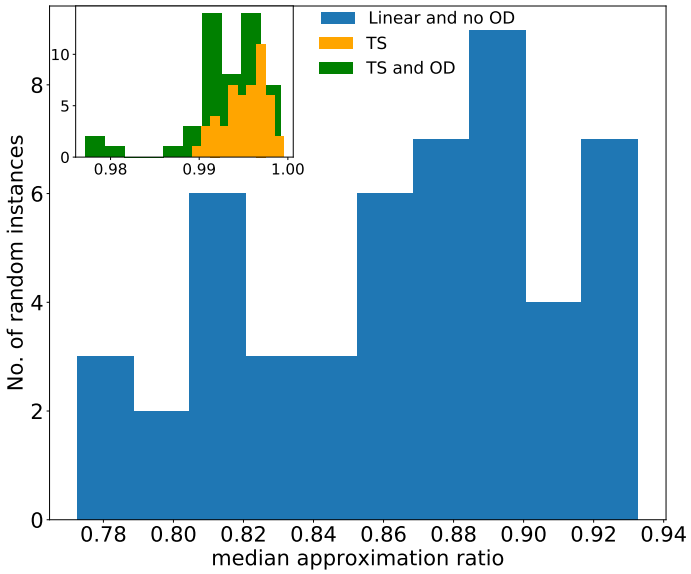


Figure 3.10: Median approximation ratio distribution for $N_{\text{inst}} = 50$ random instances over $N_{\text{rep}} = 30$ repetitions of SOGA for each instance. The histogram shows the approximation ratios of traditional quantum annealing protocol. In the inset plot, we show the corresponding approximation ratios of quantum annealing with optimized time schedules, and quantum annealing with optimized annealing schedules and OD operator.

times for each instance and for each choice of the parameters of the simulation. In particular, we considered two different annealing times ($T = 5$ and 10). For the optimization of annealing schedules alone, we considered the parameters of the polynomial ansatz, $k_a = k_b = 3$, while for the optimization of the OD operator we considered $k_a = k_b = 2$ and $k_c = 3$. We focused our attention on SOGAs and, since the target Hamiltonian is Z_2 symmetric and the ground state is doubly degenerate, we resorted to the average final energy as fitness function: $f_{\rm so} = \langle H_{\rm I} \rangle$. We quickly note that the Hamiltonian in Eq. (3.13) is commonly used to encode MaxCut and MinCut problems (Crooks 2018). This is why, in the following, we will show data concerning the so-called approximation ratio, i. e. the ratio between the final fitness value and the true ground-state energy, which is a commonly used figure of merit in approximate optimization of this kind of problems (Farhi, Goldstone, and Gutmann 2014; Crooks 2018).

We show the results for a typical random instance in Fig. 3.9 by optimizing annealing schedules alone and by optimizing both annealing schedules and OD operator. First, we focus on the optimization of the annealing schedules (see Fig. 3.9 (a)(b)(c)). In all cases analyzed, the annealing schedules are nonmonotonic like for the p-spin model of Sec. 3.2. In addition, since the final typical energy scale is smaller than the starting one, we note that the schedule $B(s)$ is always larger than $A(s)$. The energy scales remain comparable with the ones of linear annealing schedules, but the approximation ratio is substantially improved compared with the linear schedules. The results are similar when annealing schedules and OD operators are optimized together (see Fig. 3.9 (d)(e)(f)). Especially, the annealing

schedules continue to show nonmonotonic features, and we note that the schedules of the OD operator $C(s)$ is bounded within a smaller range of values. Also, the approximation ratios are significantly higher than the bare case. In Fig. 3.10, we compare the median approximation ratios (median of $N_{rep} = 30$ SOGA repetitions) of 50 random instances of the Ising model. It is evident that the genetically optimized quantum annealing protocols show consistently higher approximation ratios than the traditional quantum annealing with linear annealing schedules and without OD.

Even though the preliminary analysis of this problem shows considerably promising results, the question of whether one can find a general optimal schedules or OD operator which optimizes any random instance of Ising model remains open. We considered the average time schedule obtained from the data of 50 random instances, and investigated if this averaged schedule optimizes new random instances. In most cases it shows slight improvement when compared to the bare case. However, our analysis is far from being comprehensive in this test case and we reserve the possibility of expanding on this aspect in future works together with machine learning techniques.

3.6 Conclusions

In this chapter, we used genetic algorithms to optimize the performance of quantum annealing. We demonstrated the efficiency of our method for the ferromagnetic p-spin model with $p = 3$. In the beginning, we optimized the annealing schedules

of the standard adiabatic quantum computation protocol using genetic algorithms. We considered the time schedules to be polynomial expansions, whose coefficients were optimized as chromosomes of genetic algorithms. For a system with 15 spins, we were able to achieve a median fidelity of ≈ 0.895, by optimizing 4 free parameters of the polynomials.

We used the genetic algorithms in the paradigm of shortcuts to adiabaticity as well. Here, we optimized a practically implementable local Hamiltonian composed of only single spin operators which when added to the system Hamiltonian can improve the fidelity of the state of the system. In the first step, we fixed the annealing schedules to be linear functions of time and the time schedule of the optimal driving operator to be a quadratic function. By optimizing only the coefficients of single spin operators, i. e., by optimizing only 3 free parameters, we were able to achieve a median fidelity of ≈ 0.98, for a system with 15 spins. As a next step, we optimized the annealing schedules, and the time-dependent coefficients of the local operators together. In this case, the time schedule of each of the optimal driving operator were absorbed as their coefficients and were assumed to be polynomial functions of time. By optimizing 13 free parameters of polynomials, we were able to obtain median fidelity ≈ 0.997.

Further, we tested our methodology for varying system sizes. While optimizing annealing schedules alone showed a decrease in the fidelities, optimization of optimal driving showed consistent performance even for larger systems with up to 45 spins by optimizing only local single spin operators. We also discussed the cases when the single objective genetic algorithms give unphysical solutions of

energy crossings and the possibility of using multi-objective genetic algorithms to tackle this problem.

We tested the technique of SOGAs for a generic case of random Ising models. We generated 50 random instances of Ising models. We separately analyzed the results when only annealing schedules are optimized with chromosome size 6 and as well as in the picture of optimal driving with chromosome size 13. We compared the approximation ratios (the ratio between the energy of the final state and the energy of the true ground state) of the traditional quantum annealing with those of genetically optimized quantum annealing and demonstrated that genetic algorithms are promising tools also in optimizing quantum annealing of random Ising models. Optimized annealing schedules and local optimal driving operators can enhance the efficiency of quantum annealers in solving optimization problems. To give a practical example, D-Wave quantum annealers allow the experimentalists to control the dynamics globally by submitting a piecewise linear approximation of the annealing schedules. The shape of this approximation can then be tweaked using our genetic algorithm. Our method is flexible in terms of choice of the ansatz, definition of fitness function, simulation of time dynamics, and can be fine-tuned accordingly to match the experimental platforms and their limitations.

4. Conclusions

In this thesis we have approached the problem of mapping a p-body Hamiltonian into a Hamiltonian with only 2 body interactions. We have used a genetic algorithm (GA) where a fitness function minimizes the energy difference between the lower spectrum of the original and the final Hamiltonian by optimizing the free parameters of the fully connected 2-body Hamiltonian. The process of mapping highly interacting models into 2 local interactions requires the introduction of ancillae qubits, which are desired to be minimum in number to avoid huge resources. We have considered two analytically solvable cases of a ferromagnetic p-spin model; the number spins N=3 and N=4. For N=3, we have used one ancillary degree of freedom leading to the final Hamiltonian with the number of qubits M=4. For N=4, we introduce two ancillae leading to M=6. The optimized free parameters obtained from genetic algorithms is in good agreement with the analytical values for the cases considered. We further implement memetic algorithms (MA) which are advantaged by the local searches around the region of the solution obtained from a genetic algorithm. We consider the same cases of the ferromagnetic p-spin model and show that memetic algorithm outperforms other heuristics in terms of

the median fitness value (taken over 100 runs of the algorithm) for a given number of fitness evaluations. Specifically, for the case of $N = 3$, the median fitness values for GA, MA and hillclimbing (HC) are 1.3473×10^{-6}, 7.6298×10^{-7} and 1.6875×10^{-6}, respectively. For the case $N = 4$, the median fitness values for GA, MA and HC are 3.4285×10^{-4}, 1.172×10^{-4} and 5.3624×10^{-4}, respectively. Further, we have proposed the optimization of quantum annealing using genetic algorithms. Specifically, we engineered the annealing schedules which helps the system to avoid diabatic transitions. We chose the polynomial ansatz whose coefficients are constructed as a string of real numbers called chromosome. The chromosome is optimized by genetic algorithm with the goal to maximize the final fidelity of the adiabatic evolution. We have used ferromagnetic p-spin model to benchmark our technique. For the ferromagnetic p-spin model with 15 spins, this technique is able to achieve the median fidelity approximately equal to 0.895 by optimizing only 4 parameters. When the number of spins were increased to a large number, the fidelity was shown to be decreasing by engineering the annealing schedules alone. However by adding an optimal driving operator to the time dependent driving Hamiltonian which remains zero in the initial and the final time we were able to achieve very high fidelity even for a large number of spins. We constructed a practically implementable local optimal driving Hamiltonian from single spin operators. By optimizing only 3 parameters using genetic algorithms, the adiabatic evolution showed the final time median fidelity of approximately 0.98 for a system of 15 spins. Further by optimizing the annealing schedules and the optimal driving operator together, the median fidelity was further increased to

approximately 0.997. In the optimal driving scenario, The fidelity remained high even for the large number of spins (upto 45 spins).

The results of this thesis have shown that the evolutionary algorithms are very effective techniques in aiding quantum computation. They are effective in performing efficient embedding of optimization problems into quantum hardware and they are also capable of modelling and engineering the optimal interactions deviating from the standard theoretical quantum algorithms making it possible to implement them on the real quantum hardware. Likewise, in the near future we would like to use these techniques for other problems within the subject of quantum computation. The quantum approximate optimization algorithms (QAOAs) have been recently shown to have advantaged by shortcuts to adiabaticity by implementing an optimal quantum circuit (Hegade, X. Chen, and Solano 2022; Wurtz and Love 2021). It will be definitely a worthwhile study to investigate the use of genetic algorithms in this problem. The preliminary study of finding optimal driving of a random Ising model in our work is a starting point to investigate the use of machine learning techniques to find general patterns in optimal schedules, perhaps also using different ansatz functions. It would be also interesting to study the effectiveness of evolutionary strategies in finding an optimal driving in open quantum systems.

Bibliography

Abah, Obinna and Eric Lutz (Sept. 2018). "Performance of shortcut-to-adiabaticity quantum engines". In: *Phys. Rev. E* 98 (3), p. 032121. DOI: 10.1103/PhysRevE.98.032121. URL: https://link.aps.org/doi/10.1103/PhysRevE.98.032121.

Acampora, G. and A. Vitiello (June 2012). "Improving agent interoperability through a memetic ontology alignment: A comparative study". In: *2012 IEEE International Conference on Fuzzy Systems*, pp. 1–8.

Acampora, Giovanni, Pasquale Avella, et al. (2011). "Improving ontology alignment through memetic algorithms". In: *2011 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE 2011)*. IEEE, pp. 1783–1790.

Acampora, Giovanni, Jose Manuel Cadenas, et al. (2011). "Achieving memetic adaptability by means of agent-based machine learning". In: *IEEE Transactions on Industrial Informatics* 7.4, pp. 557–569.

Acampora, Giovanni, Vittorio Cataudella, Pratibha R Hegde, et al. (2019). "An evolutionary strategy for finding effective quantum 2-body Hamiltonians of p-body interacting systems". In: *Quantum Machine Intelligence* 1.3, pp. 113–122. DOI: https://doi.org/10.1007/s42484-019-00011-8.

Acampora, Giovanni, Vittorio Cataudella, Pratibha Raghupati Hegde, et al. (2021). "Memetic algorithms for mapping p-body interacting systems in effective quantum 2-body Hamiltonians". In: *Applied Soft Computing* 110, p. 107634. ISSN: 1568-4946. DOI: https://doi.org/10.1016/j.asoc.2021.107634. URL: https://www.sciencedirect.com/science/article/pii/S156849462100555X.

Adachi, Steven H. and Maxwell P. Henderson (Oct. 2015). "Application of Quantum Annealing to Training of Deep Neural Networks". In: *arXiv:1510.06356 [quant-ph, stat]*. arXiv: 1510.06356. URL: http://arxiv.org/abs/1510.06356.

Aharonov, Dorit et al. (2008). "Adiabatic Quantum Computation Is Equivalent to Standard Quantum Computation". In: *SIAM Review* 50.4, pp. 755–787. ISSN: 00361445. URL: http://www.jstor.org/stable/20454175.

Albash, Tameem and Daniel A. Lidar (Jan. 2018). "Adiabatic quantum computation". In: *Rev. Mod. Phys.* 90 (1), p. 015002. DOI: 10.1103/RevModPhys.90.015002. URL: https://link.aps.org/doi/10.1103/RevModPhys.90.015002.

Arute, Frank et al. (2019). "Quantum supremacy using a programmable superconducting processor". In: *Nature* 574.7779, pp. 505–510. DOI: 10.1038/s41586-019-1666-5. arXiv: 1910.11333 [quant-ph].

Babbush, Ryan et al. (Apr. 2014). "Construction of Energy Functions for Lattice Heteropolymer Models: A Case Study in Constraint Satisfaction Programming and Adiabatic Quantum Optimization". In: *arXiv:1211.3422 [cond-mat, physics:math-ph, physics:physics, physics:quant-ph]*. arXiv: 1211.3422, pp. 201–244. DOI: 10.1002/9781118755815.ch05. URL: http://arxiv.org/abs/1211.3422.

Bäck, Thomas and Hans-Paul Schwefel (1993). "An Overview of Evolutionary Algorithms for Parameter Optimization". In: *Evolutionary Computation* 1.1, pp. 1–23. DOI: 10.1162/evco.1993.1.1.1.

Bapst, Victor and Guilhem Semerjian (June 2012). "On quantum mean-field models and their quantum annealing". In: *Journal of Statistical Mechanics: Theory and Experiment* 2012.06, P06007. URL: http://stacks.iop.org/1742-5468/2012/i=06/a=P06007.

Berry, M V (Aug. 2009). "Transitionless quantum driving". In: *Journal of Physics A: Mathematical and Theoretical* 42.36, p. 365303. DOI: 10.1088/1751-8113/42/36/365303. URL: https://doi.org/10.1088%2F1751-8113%2F42%2F36%2F365303.

Biamonte, J. D. (May 2008). "Nonperturbative k-body to two-body commuting conversion Hamiltonians and embedding problem instances into Ising spins". In: *Phys. Rev. A* 77 (5), p. 052331. DOI: 10.1103/PhysRevA.77.052331. URL: https://link.aps.org/doi/10.1103/PhysRevA.77.052331.

Biamonte, Jacob D. and Peter J. Love (July 2008). "Realizable Hamiltonians for universal adiabatic quantum computers". In: *Phys. Rev. A* 78 (1), p. 012352. DOI: 10.1103/PhysRevA.78.012352. URL: https://link.aps.org/doi/10.1103/PhysRevA.78.012352.

Bian, Zhengbing et al. (2020). "Solving SAT (and MaxSAT) with a quantum annealer: Foundations, encodings, and preliminary results". In: *Information and Computation* 275, p. 104609. ISSN: 0890-5401. DOI: https://doi.org/

10.1016/j.ic.2020.104609. URL: https://www.sciencedirect.com/science/article/pii/S0890540120300973.

Boev, A. S. et al. (Dec. 2021). "Genome assembly using quantum and quantum-inspired annealing". en. In: *Scientific Reports* 11.1, p. 13183. ISSN: 2045-2322. DOI: 10.1038/s41598-021-88321-5. URL: http://www.nature.com/articles/s41598-021-88321-5.

Bölte, Andreas and Ulrich Wilhelm Thonemann (1996). "Optimizing simulated annealing schedules with genetic programming". In: *European Journal of Operational Research* 92.2, pp. 402–416. ISSN: 0377-2217. DOI: https://doi.org/10.1016/0377-2217(94)00350-5. URL: https://www.sciencedirect.com/science/article/pii/0377221794003505.

Born, M. and V. Fock (Mar. 1928). "Beweis des Adiabatensatzes". de. In: *Zeitschrift für Physik* 51.3-4, pp. 165–180. ISSN: 1434-6001, 1434-601X. DOI: 10.1007/BF01343193. URL: http://link.springer.com/10.1007/BF01343193.

Campbell, Steve, Gabriele De Chiara, et al. (May 2015). "Shortcut to Adiabaticity in the Lipkin-Meshkov-Glick Model". In: *Phys. Rev. Lett.* 114 (17), p. 177206. DOI: 10.1103/PhysRevLett.114.177206. URL: https://link.aps.org/doi/10.1103/PhysRevLett.114.177206.

Campbell, Steve and Sebastian Deffner (Mar. 2017). "Trade-Off Between Speed and Cost in Shortcuts to Adiabaticity". In: *Phys. Rev. Lett.* 118 (10), p. 100601. DOI: 10.1103/PhysRevLett.118.100601. URL: https://link.aps.org/doi/10.1103/PhysRevLett.118.100601.

Caneva, T. et al. (Dec. 2009). "Optimal Control at the Quantum Speed Limit". In: *Phys. Rev. Lett.* 103 (24), p. 240501. DOI: 10.1103/PhysRevLett.103.240501. URL: https://link.aps.org/doi/10.1103/PhysRevLett.103.240501.

Chen, Ye-Hong et al. (May 2016). "Method for constructing shortcuts to adiabaticity by a substitute of counterdiabatic driving terms". In: *Phys. Rev. A* 93 (5), p. 052109. DOI: 10.1103/PhysRevA.93.052109. URL: https://link.aps.org/doi/10.1103/PhysRevA.93.052109.

Chen, Hongwei et al. (Mar. 2011). "Experimental demonstration of a quantum annealing algorithm for the traveling salesman problem in a nuclear-magnetic-resonance quantum simulator". In: *Phys. Rev. A* 83 (3), p. 032314. DOI: 10.1103/PhysRevA.83.032314. URL: https://link.aps.org/doi/10.1103/PhysRevA.83.032314.

Chivilikhin, D. et al. (July 2020). "MoG-VQE: Multiobjective genetic variational quantum eigensolver". In: *arXiv:2007.04424 [cond-mat, physics:quant-ph]*. arXiv: 2007.04424. URL: http://arxiv.org/abs/2007.04424.

Choi, Vicky (Oct. 2008). "Minor-embedding in adiabatic quantum computation: I. The parameter setting problem". In: *Quantum Information Processing* 7.5, pp. 193–209. ISSN: 1573-1332. DOI: 10.1007/s11128-008-0082-9. URL: https://doi.org/10.1007/s11128-008-0082-9.

— (June 2011). "Minor-embedding in adiabatic quantum computation: II. Minor-universal graph design". In: *Quantum Information Processing* 10.3, pp. 343–353. ISSN: 1573-1332. DOI: 10.1007/s11128-010-0200-3. URL: https://doi.org/10.1007/s11128-010-0200-3.

Claeys, Pieter W. et al. (Aug. 2019). "Floquet-Engineering Counterdiabatic Protocols in Quantum Many-Body Systems". In: *Phys. Rev. Lett.* 123 (9), p. 090602. DOI: 10.1103/PhysRevLett.123.090602. URL: https://link.aps.org/doi/10.1103/PhysRevLett.123.090602.

Comparat, Daniel (July 2009). "General conditions for quantum adiabatic evolution". In: *Phys. Rev. A* 80 (1), p. 012106. DOI: 10.1103/PhysRevA.80.012106. URL: https://link.aps.org/doi/10.1103/PhysRevA.80.012106.

Cook, Stephen A. (1971). "The Complexity of Theorem-proving Procedures". In: *Proceedings of the Third Annual ACM Symposium on Theory of Computing*. STOC '71. Shaker Heights, Ohio, USA: ACM, pp. 151–158. DOI: 10.1145/800157.805047. URL: http://doi.acm.org/10.1145/800157.805047.

Coulamy, Ivan B. et al. (2016). "Energetic Cost of Superadiabatic Quantum Computation". In: *Frontiers in ICT* 3, p. 19. ISSN: 2297-198X. DOI: 10.3389/fict.2016.00019. URL: https://www.frontiersin.org/article/10.3389/fict.2016.00019.

Crooks, Gavin E. (Nov. 2018). "Performance of the Quantum Approximate Optimization Algorithm on the Maximum Cut Problem". In: *arXiv:1811.08419 [quant-ph]*. arXiv: 1811.08419. URL: http://arxiv.org/abs/1811.08419.

Dam, W. van, M. Mosca, and U. Vazirani (2001). "How powerful is adiabatic quantum computation?" In: *Proceedings 42nd IEEE Symposium on Foundations of Computer Science*, pp. 279–287. DOI: 10.1109/SFCS.2001.959902.

Date, Prasanna et al. (Apr. 2019). "Efficiently Embedding QUBO Problems on Adiabatic Quantum Computers". In: *Quantum Information Processing* 18.4, pp. 1–31. ISSN: 1570-0755. DOI: 10.1007/s11128-019-2236-3. URL: https://doi.org/10.1007/s11128-019-2236-3.

Deb, K. et al. (2002). "A fast and elitist multiobjective genetic algorithm: NSGA-II". In: *IEEE Transactions on Evolutionary Computation* 6.2, pp. 182–197. DOI: 10.1109/4235.996017.

Del Campo, Adolfo (Sept. 2013). "Shortcuts to Adiabaticity by Counterdiabatic Driving". In: *Phys. Rev. Lett.* 111 (10), p. 100502. DOI: 10.1103/PhysRevLett.111.100502. URL: https://link.aps.org/doi/10.1103/PhysRevLett.111.100502.

Derrida, Bernard (Sept. 1981). "Random-energy model: An exactly solvable model of disordered systems". In: *Phys. Rev. B* 24 (5), pp. 2613–2626. DOI: 10.1103/PhysRevB.24.2613. URL: https://link.aps.org/doi/10.1103/PhysRevB.24.2613.

Deutsch, David and Richard Jozsa (Dec. 1992). "Rapid Solution of Problems by Quantum Computation". In: *Proceedings of the Royal Society of London Series A* 439.1907, pp. 553–558. DOI: 10.1098/rspa.1992.0167.

Dowling, Jonathan P. and Gerard J. Milburn (Aug. 2003). "Quantum technology: the second quantum revolution". en. In: *Philosophical Transactions of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences* 361.1809. Ed. by A. G. J. MacFarlane, pp. 1655–1674. ISSN: 1364-503X, 1471-2962. DOI: 10.1098/rsta.2003.1227. URL: https://royalsocietypublishing.org/doi/10.1098/rsta.2003.1227.

Farhi, Edward, Jeffrey Goldstone, and Sam Gutmann (Jan. 2002). "Quantum Adiabatic Evolution Algorithms versus Simulated Annealing". In: *arXiv:quant-ph/0201031*. arXiv: quant-ph/0201031. URL: http://arxiv.org/abs/quant-ph/0201031.

— (Nov. 2014). "A Quantum Approximate Optimization Algorithm". In: *arXiv:1411.4028 [quant-ph]*. arXiv: 1411.4028. URL: http://arxiv.org/abs/1411.4028.

Farhi, Edward, Jeffrey Goldstone, Sam Gutmann, Joshua Lapan, et al. (2001). "A Quantum Adiabatic Evolution Algorithm Applied to Random Instances of an NP-Complete Problem". In: *Science* 292.5516, pp. 472–475. DOI: 10.1126/science.1057726. eprint: https://www.science.org/doi/pdf/10.1126/science.1057726. URL: https://www.science.org/doi/abs/10.1126/science.1057726.

Farhi, Edward, Jeffrey Goldstone, Sam Gutmann, and Michael Sipser (Jan. 2000). "Quantum Computation by Adiabatic Evolution". In: *arXiv e-prints*, quant-ph/0001106, quant–ph/0001106. arXiv: quant-ph/0001106 [quant-ph].

Feynman, Richard P. (1982). "Simulating physics with computers". In: *Int. J. Theor. Phys.* 21. Ed. by L. M. Brown, pp. 467–488. DOI: 10.1007/BF02650179.

Finnila, A.B. et al. (1994). "Quantum annealing: A new method for minimizing multidimensional functions". In: *Chemical Physics Letters* 219.5, pp. 343–348. ISSN: 0009-2614. DOI: https://doi.org/10.1016/0009-2614(94)00117-

0. URL: https://www.sciencedirect.com/science/article/pii/0009261494001170.

Flynn, Brian et al. (Dec. 2021). "Quantum Model Learning Agent: characterisation of quantum systems through machine learning". In: *arXiv:2112.08409 [quant-ph]*. arXiv: 2112.08409. URL: http://arxiv.org/abs/2112.08409.

Fonseca, Carlos M. and Peter J. Fleming (1995). "An Overview of Evolutionary Algorithms in Multiobjective Optimization". In: *Evolutionary Computation* 3.1, pp. 1–16. DOI: 10.1162/evco.1995.3.1.1.

Fortin, Félix-Antoine et al. (July 2012). "DEAP: Evolutionary Algorithms Made Easy". In: *Journal of Machine Learning Research* 13, pp. 2171–2175. URL: http://jmlr.org/papers/v13/fortin12a.html.

Funo, Ken et al. (Mar. 2017). "Universal Work Fluctuations During Shortcuts to Adiabaticity by Counterdiabatic Driving". In: *Phys. Rev. Lett.* 118 (10), p. 100602. DOI: 10.1103/PhysRevLett.118.100602. URL: https://link.aps.org/doi/10.1103/PhysRevLett.118.100602.

García, Salvador et al. (Dec. 2009). "A study on the use of non-parametric tests for analyzing the evolutionary algorithms' behaviour: a case study on the CEC'2005 Special Session on Real Parameter Optimization". en. In: *Journal of Heuristics* 15.6, pp. 617–644. ISSN: 1381-1231, 1572-9397. DOI: 10.1007/s10732-008-9080-4. URL: http://link.springer.com/10.1007/s10732-008-9080-4.

Giraldi, Gilson A., Renato Portugal, and Ricardo N. Thess (Mar. 2004). "Genetic Algorithms and Quantum Computation". In: *arXiv:cs/0403003*. arXiv: cs/0403003. URL: http://arxiv.org/abs/cs/0403003.

Glaser, Steffen J. et al. (2015). "Training Schrödinger's cat: quantum optimal control". In: *The European Physical Journal D* 69.12, p. 279. DOI: 10.1140/epjd/e2015-60464-1. URL: https://doi.org/10.1140/epjd/e2015-60464-1.

Goldberg, David E. and John H. Holland (Oct. 1988). "Genetic Algorithms and Machine Learning". In: *Machine Learning* 3.2, pp. 95–99. ISSN: 1573-0565. DOI: 10.1023/A:1022602019183. URL: https://doi.org/10.1023/A:1022602019183.

Gross, D.J. and M. Mezard (1984). "The simplest spin glass". In: *Nuclear Physics B* 240.4, pp. 431–452. ISSN: 0550-3213. DOI: https://doi.org/10.1016/0550-3213(84)90237-2. URL: http://www.sciencedirect.com/science/article/pii/0550321384902372.

Grover, L. K. (1996). "A Fast Quantum Mechanical Algorithm for Database Search". In: *Proceedings of the Twenty-eighth Annual ACM Symposium on*

Theory of Computing. STOC '96. ACM, pp. 212–219. ISBN: 0-89791-785-5. URL: http://doi.acm.org/10.1145/237814.237866.

Grover, Lov K (1996). "A fast quantum mechanical algorithm for database search". In: *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*, pp. 212–219.

Guéry-Odelin, D. et al. (Oct. 2019). "Shortcuts to adiabaticity: Concepts, methods, and applications". In: *Rev. Mod. Phys.* 91 (4), p. 045001. DOI: 10.1103/RevModPhys.91.045001. URL: https://link.aps.org/doi/10.1103/RevModPhys.91.045001.

Hardy, Yorick and W-H Steeb (2010). "Genetic algorithms and optimization problems in quantum computing". In: *International Journal of Modern Physics C* 21.11, pp. 1359–1375.

Harris, R. et al. (2011). "Quantum annealing with manufactured spins". In: *Nature* 473 (7346), pp. 194–198. URL: http://dx.doi.org/10.1038/nature10012.

Hartmann, Andreas and Wolfgang Lechner (Apr. 2019). "Rapid counter-diabatic sweeps in lattice gauge adiabatic quantum computing". In: *New Journal of Physics* 21.4, p. 043025. DOI: 10.1088/1367-2630/ab14a0. URL: https://doi.org/10.1088/1367-2630/ab14a0.

Hauke, P. et al. (2019). "Perspectives of quantum annealing: Methods and Perspectives of quantum annealing: Methods and implementations". arXiv:1903.06559.

Hauke, Philipp et al. (Mar. 2019). "Perspectives of quantum annealing: Methods and implementations". In: *arXiv e-prints*, arXiv:1903.06559, arXiv:1903.06559. arXiv: 1903.06559 [quant-ph].

Hegade, Narendra N., Xi Chen, and Enrique Solano (Jan. 2022). "Digitized-Counterdiabatic Quantum Optimization". In: *arXiv:2201.00790 [cond-mat, physics:quant-ph]*. arXiv: 2201.00790. URL: http://arxiv.org/abs/2201.00790.

Hegde, Pratibha Raghupati et al. (Jan. 2022). "Genetic optimization of quantum annealing". In: *Phys. Rev. A* 105 (1), p. 012612. DOI: 10.1103/PhysRevA.105.012612. URL: https://link.aps.org/doi/10.1103/PhysRevA.105.012612.

Hegerfeldt, Gerhard C. (Dec. 2013). "Driving at the Quantum Speed Limit: Optimal Control of a Two-Level System". In: *Phys. Rev. Lett.* 111 (26), p. 260501. DOI: 10.1103/PhysRevLett.111.260501. URL: https://link.aps.org/doi/10.1103/PhysRevLett.111.260501.

Herrera, F., M. Lozano, and A. M. Sánchez (2003). "A taxonomy for the crossover operator for real-coded genetic algorithms: An experimental study". In: *Inter-*

national Journal of Intelligent Systems 18, pp. 309–338. DOI: https://doi.org/10.1002/int.10091.

Hogg, Tad (Feb. 2003). "Adiabatic quantum computing for random satisfiability problems". In: *Phys. Rev. A* 67 (2), p. 022314. DOI: 10.1103/PhysRevA.67.022314. URL: https://link.aps.org/doi/10.1103/PhysRevA.67.022314.

Hu, Chang-Kang et al. (July 2018). "Experimental implementation of generalized transitionless quantum driving". In: *Opt. Lett.* 43.13, pp. 3136–3139. DOI: 10.1364/OL.43.003136. URL: http://ol.osa.org/abstract.cfm?URI=ol-43-13-3136.

Ikeda, Kazuki, Yuma Nakamura, and Travis S. Humble (Dec. 2019). "Application of Quantum Annealing to Nurse Scheduling Problem". en. In: *Scientific Reports* 9.1, p. 12837. ISSN: 2045-2322. DOI: 10.1038/s41598-019-49172-3. URL: http://www.nature.com/articles/s41598-019-49172-3.

Inack, E. M. and S. Pilati (Nov. 2015). "Simulated quantum annealing of double-well and multiwell potentials". In: *Phys. Rev. E* 92 (5), p. 053304. DOI: 10.1103/PhysRevE.92.053304. URL: https://link.aps.org/doi/10.1103/PhysRevE.92.053304.

James, Daniel F. V. et al. (Oct. 2001). "Measurement of qubits". en. In: *Physical Review A* 64.5, p. 052312. ISSN: 1050-2947, 1094-1622. DOI: 10.1103/PhysRevA.64.052312. URL: https://link.aps.org/doi/10.1103/PhysRevA.64.052312.

Johansson, J.R., P.D. Nation, and F. Nori (Aug. 2012). "QuTiP: An open-source Python framework for the dynamics of open quantum systems". In: *Computer Physics Communications* 183.8, pp. 1760–1772. DOI: 10.1016/j.cpc.2012.02.021. URL: https://doi.org/10.1016/j.cpc.2012.02.021.

— (Apr. 2013). "QuTiP 2: A Python framework for the dynamics of open quantum systems". In: *Computer Physics Communications* 184.4, pp. 1234–1240. DOI: 10.1016/j.cpc.2012.11.019. URL: https://doi.org/10.1016/j.cpc.2012.11.019.

Jörg, T. et al. (Feb. 2010). "Energy gaps in quantum first-order mean-field–like transitions: The problems that quantum annealing cannot solve". In: *EPL (Europhysics Letters)* 89.4, p. 40004. DOI: 10.1209/0295-5075/89/40004. URL: https://doi.org/10.1209%2F0295-5075%2F89%2F40004.

Kadowaki, Tadashi and Hidetoshi Nishimori (Nov. 1998). "Quantum annealing in the transverse Ising model". In: *Phys. Rev. E* 58 (5), pp. 5355–5363. DOI: 10.1103/PhysRevE.58.5355. URL: https://link.aps.org/doi/10.1103/PhysRevE.58.5355.

Kato, Tosio (1950). "On the Adiabatic Theorem of Quantum Mechanics". In: *Journal of the Physical Society of Japan* 5.6, pp. 435–439. DOI: 10.1143/JPSJ.5.435. eprint: https://doi.org/10.1143/JPSJ.5.435. URL: https://doi.org/10.1143/JPSJ.5.435.

Knysh, Sergey (2016). "Zero-temperature quantum annealing bottlenecks in the spin-glass phase". In: *Nature Communications* 7.1, p. 12370. DOI: 10.1038/ncomms12370. URL: https://doi.org/10.1038/ncomms12370.

Krasnogor, Natalio and James Smith (2005). "A tutorial for competent memetic algorithms: model, taxonomy, and design issues". In: *IEEE Transactions on Evolutionary Computation* 9.5, pp. 474–488.

Laarhoven, Peter J. M. van and Emile H. L. Aarts (1987). "Simulated annealing". In: *Simulated Annealing: Theory and Applications*. Dordrecht: Springer Netherlands, pp. 7–15. ISBN: 978-94-015-7744-1. DOI: 10.1007/978-94-015-7744-1_2. URL: https://doi.org/10.1007/978-94-015-7744-1_2.

Las Heras, U. et al. (June 2016). "Genetic Algorithms for Digital Quantum Simulations". In: *Phys. Rev. Lett.* 116 (23), p. 230504. DOI: 10.1103/PhysRevLett.116.230504. URL: https://link.aps.org/doi/10.1103/PhysRevLett.116.230504.

Leib, Martin, Peter Zoller, and Wolfgang Lechner (Dec. 2016). "A transmon quantum annealer: decomposing many-body Ising constraints into pair interactions". In: *Quantum Science and Technology* 1.1, p. 015008. DOI: 10.1088/2058-9565/1/1/015008. URL: https://doi.org/10.1088%5C%2F2058-9565%5C%2F1%5C%2F1%5C%2F015008.

Li, Rui et al. (2017). "Approximate Quantum Adders with Genetic Algorithms: An IBM Quantum Experience". In: *Quantum Measurements and Quantum Metrology* 4.1, pp. 1–7. DOI: doi:10.1515/qmetro-2017-0001. URL: https://doi.org/10.1515/qmetro-2017-0001.

Loss, Daniel and David P. DiVincenzo (Jan. 1998). "Quantum computation with quantum dots". In: *Phys. Rev. A* 57 (1), pp. 120–126. DOI: 10.1103/PhysRevA.57.120. URL: https://link.aps.org/doi/10.1103/PhysRevA.57.120.

Lucas, Andrew (2014). "Ising formulations of many NP problems". In: *Frontiers in Physics* 2, p. 5. ISSN: 2296-424X. DOI: 10.3389/fphy.2014.00005. URL: https://www.frontiersin.org/article/10.3389/fphy.2014.00005.

Matsuda, Yoshiki, Hidetoshi Nishimori, and Helmut G Katzgraber (Jan. 2009). "Quantum annealing for problems with ground-state degeneracy". In: *Journal of Physics: Conference Series* 143, p. 012003. DOI: 10.1088/1742-6596/143/1/012003. URL: https://doi.org/10.1088/1742-6596/143/1/012003.

Matsuura, Shunji, Samantha Buck, et al. (May 2021). "Variationally scheduled quantum simulation". In: *Phys. Rev. A* 103 (5), p. 052435. DOI: 10.1103/PhysRevA.103.052435. URL: https://link.aps.org/doi/10.1103/PhysRevA.103.052435.

Matsuura, Shunji, Hidetoshi Nishimori, Walter Vinci, Tameem Albash, et al. (Feb. 2017). "Quantum-annealing correction at finite temperature: Ferromagnetic p-spin models". In: *Phys. Rev. A* 95 (2), p. 022308. DOI: 10.1103/PhysRevA.95.022308. URL: https://link.aps.org/doi/10.1103/PhysRevA.95.022308.

Matsuura, Shunji, Hidetoshi Nishimori, Walter Vinci, and Daniel A. Lidar (June 2019). "Nested quantum annealing correction at finite temperature: p-spin models". In: *Phys. Rev. A* 99 (6), p. 062307. DOI: 10.1103/PhysRevA.99.062307. URL: https://link.aps.org/doi/10.1103/PhysRevA.99.062307.

McGeoch, Catherine C. (2014). *Adiabatic quantum computation and quantum annealing: theory and practice*. eng. Synthesis lectures on quantum computing 8. San Rafael, Calif.: Morgan & Claypool. ISBN: 9781627053358.

Messiah, Quantum Mechanics (1962). *Vol. II, Chap. XXI. 13*.

Mukherjee, Victor, Simone Montangero, and Rosario Fazio (June 2016). "Local shortcut to adiabaticity for quantum many-body systems". In: *Phys. Rev. A* 93 (6), p. 062108. DOI: 10.1103/PhysRevA.93.062108. URL: https://link.aps.org/doi/10.1103/PhysRevA.93.062108.

Nenciu, G (Feb. 1980). "On the adiabatic theorem of quantum mechanics". In: *Journal of Physics A: Mathematical and General* 13.2, pp. L15–L18. DOI: 10.1088/0305-4470/13/2/002. URL: https://doi.org/10.1088/0305-4470/13/2/002.

Nielsen, Michael A and Isaac Chuang (2002). *Quantum computation and quantum information*.

Nishimori, Hidetoshi and Kabuki Takada (2017). "Exponential Enhancement of the Efficiency of Quantum Annealing by Non-Stoquastic Hamiltonians". In: *Frontiers in ICT* 4, p. 2. ISSN: 2297-198X. DOI: 10.3389/fict.2017.00002. URL: https://www.frontiersin.org/article/10.3389/fict.2017.00002.

O'Driscoll, L., R. Nichols, and P. A. Knott (Mar. 2019). "A hybrid machine learning algorithm for designing quantum experiments". In: *Quantum Machine Intelligence*. ISSN: 2524-4914. DOI: 10.1007/s42484-019-00003-8. URL: https://doi.org/10.1007/s42484-019-00003-8.

O'Malley, Daniel et al. (Dec. 2018). "Nonnegative/Binary matrix factorization with a D-Wave quantum annealer". en. In: *PLOS ONE* 13.12. Ed. by Ivan Olier, e0206653. ISSN: 1932-6203. DOI: 10.1371/journal.pone.0206653. URL: https://dx.plos.org/10.1371/journal.pone.0206653 (visited on 03/07/2022).

Ohkuwa, Masaki, Hidetoshi Nishimori, and Daniel A. Lidar (Aug. 2018a). "Reverse annealing for the fully connected p-spin model". In: *Phys. Rev. A* 98 (2), p. 022314. DOI: 10.1103/PhysRevA.98.022314. URL: https://link.aps.org/doi/10.1103/PhysRevA.98.022314.

— (Aug. 2018b). "Reverse annealing for the fully connected p-spin model". In: *Phys. Rev. A* 98 (2), p. 022314. DOI: 10.1103/PhysRevA.98.022314. URL: https://link.aps.org/doi/10.1103/PhysRevA.98.022314.

Okada, Shuntaro et al. (2019). "Improving solutions by embedding larger subproblems in a D-Wave quantum annealer". In: *Scientific reports* 9.1, p. 2098.

Orlando, T. P. et al. (Dec. 1999). "Superconducting persistent-current qubit". In: *Phys. Rev. B* 60 (22), pp. 15398–15413. DOI: 10.1103/PhysRevB.60.15398. URL: https://link.aps.org/doi/10.1103/PhysRevB.60.15398.

Passarelli, G., V. Cataudella, R. Fazio, et al. (Mar. 2020). "Counterdiabatic driving in the quantum annealing of the p-spin model: A variational approach". In: *Phys. Rev. Research* 2 (1), p. 013283. DOI: 10.1103/PhysRevResearch.2.013283. URL: https://link.aps.org/doi/10.1103/PhysRevResearch.2.013283.

Passarelli, G., V. Cataudella, and P. Lucignano (July 2019). "Improving quantum annealing of the ferromagnetic p-spin model through pausing". In: *Phys. Rev. B* 100 (2), p. 024302. DOI: 10.1103/PhysRevB.100.024302. URL: https://link.aps.org/doi/10.1103/PhysRevB.100.024302.

Passarelli, G., G. De Filippis, et al. (Feb. 2018). "Dissipative environment may improve the quantum annealing performances of the ferromagnetic p-spin model". In: *Phys. Rev. A* 97 (2), p. 022319. DOI: 10.1103/PhysRevA.97.022319. URL: https://link.aps.org/doi/10.1103/PhysRevA.97.022319.

Passarelli, Gianluca, Giulio De Filippis, et al. (Jan. 2019). "May a dissipative environment be beneficial for quantum annealing?" In: *arXiv e-prints*, arXiv:1901.07787, arXiv:1901.07787. arXiv: 1901.07787 [quant-ph].

Passarelli, Gianluca, Ka-Wa Yip, et al. (Feb. 2020). "Reverse quantum annealing of the p-spin model with relaxation". In: *Phys. Rev. A* 101 (2), p. 022331. DOI: 10.1103/PhysRevA.101.022331. URL: https://link.aps.org/doi/10.1103/PhysRevA.101.022331.

Preskill, John (Aug. 2018). "Quantum Computing in the NISQ era and beyond". In: *Quantum* 2, p. 79. ISSN: 2521-327X. DOI: 10.22331/q-2018-08-06-79. URL: https://doi.org/10.22331/q-2018-08-06-79.

Qiu, Xingze, Peter Zoller, and Xiaopeng Li (Nov. 2020). "Programmable Quantum Annealing Architectures with Ising Quantum Wires". In: *PRX Quantum* 1 (2), p. 020311. DOI: 10.1103/PRXQuantum.1.020311. URL: https://link.aps.org/doi/10.1103/PRXQuantum.1.020311.

Renders, J-M and Hugues Bersini (1994). "Hybridizing genetic algorithms with hill-climbing methods for global optimization: two possible ways". In: *Proceedings of the First IEEE Conference on Evolutionary Computation. IEEE World Congress on Computational Intelligence*. IEEE, pp. 312–317.

Roland, Jérémie and Nicolas J. Cerf (Mar. 2002). "Quantum search by local adiabatic evolution". In: *Phys. Rev. A* 65 (4), p. 042308. DOI: 10.1103/PhysRevA.65.042308. URL: https://link.aps.org/doi/10.1103/PhysRevA.65.042308.

Rønnow, Troels F. et al. (July 2014). "Defining and detecting quantum speedup". en. In: *Science* 345.6195, pp. 420–424. ISSN: 0036-8075, 1095-9203. DOI: 10.1126/science.1252319. URL: https://www.science.org/doi/10.1126/science.1252319.

Rothlauf, Franz (2006). "Representations for Genetic and Evolutionary Algorithms". In: *Representations for Genetic and Evolutionary Algorithms*. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 9–32. ISBN: 978-3-540-32444-7. DOI: 10.1007/3-540-32444-5_2. URL: https://doi.org/10.1007/3-540-32444-5_2.

Sachdev, Subir (2011). *Quantum Phase Transitions*. 2nd ed. Cambridge University Press. DOI: 10.1017/CBO9780511973765.

Santoro, G.E. and E. Tosatti (2006). "Optimization using quantum mechanics: quantum annealing through adiabatic evolution". In: *Journal of Physics A: Mathematical and General* 39, R393.

Santoro, Giuseppe E. et al. (2002). "Theory of Quantum Annealing of an Ising Spin Glass". In: *Science* 295.5564, pp. 2427–2430. ISSN: 0036-8075. DOI: 10.1126/science.1068774. URL: https://science.sciencemag.org/content/295/5564/2427.

Santos, Alan C and Marcelo S Sarandy (Dec. 2017). "Generalized shortcuts to adiabaticity and enhanced robustness against decoherence". In: *Journal of Physics A: Mathematical and Theoretical* 51.2, p. 025301. DOI: 10.1088/1751-8121/aa96f1. URL: https://doi.org/10.1088/1751-8121/aa96f1.

Santos, Alan C., Amanda Nicotina, et al. (Mar. 2020). "Optimizing NMR quantum information processing via generalized transitionless quantum driving". In: *EPL (Europhysics Letters)* 129.3, p. 30008. DOI: 10.1209/0295-5075/129/30008. URL: https://doi.org/10.1209/0295-5075/129/30008.

Santos, Alan C. and Marcelo S. Sarandy (Dec. 2015). "Superadiabatic Controlled Evolutions and Universal Quantum Computation". In: *Scientific Reports* 5.1, p. 15775. ISSN: 2045-2322. DOI: 10.1038/srep15775.

Sarandy, M. S., L.-A. Wu, and D. A. Lidar (Dec. 2004). "Consistency of the Adiabatic Theorem". In: *Quantum Information Processing* 3.6, pp. 331–349. ISSN: 1573-1332. DOI: 10.1007/s11128-004-7712-7. URL: https://doi.org/10.1007/s11128-004-7712-7.

Sastry, Kumara, David Goldberg, and Graham Kendall (2005). "Genetic Algorithms". In: *Search Methodologies: Introductory Tutorials in Optimization and Decision Support Techniques*. Ed. by Edmund K. Burke and Graham Kendall. Boston, MA: Springer US, pp. 97–125. ISBN: 978-0-387-28356-2. DOI: 10.1007/0-387-28356-0_4. URL: https://doi.org/10.1007/0-387-28356-0_4.

Seki, Yuya and Hidetoshi Nishimori (May 2012). "Quantum annealing with antiferromagnetic fluctuations". In: *Phys. Rev. E* 85 (5), p. 051112. DOI: 10.1103/PhysRevE.85.051112. URL: https://link.aps.org/doi/10.1103/PhysRevE.85.051112.

Sels, Dries and Anatoli Polkovnikov (2017). "Minimizing irreversible losses in quantum systems by local counterdiabatic driving". In: *Proceedings of the National Academy of Sciences* 114.20, E3909–E3916. ISSN: 0027-8424. DOI: 10.1073/pnas.1619826114. eprint: https://www.pnas.org/content/114/20/E3909.full.pdf. URL: https://www.pnas.org/content/114/20/E3909.

Seoane, Beatriz and Hidetoshi Nishimori (Oct. 2012). "Many-body transverse interactions in the quantum annealing of the p-spin ferromagnet". In: *Journal of Physics A: Mathematical and Theoretical* 45.43, p. 435301. DOI: 10.1088/1751-8113/45/43/435301. URL: https://doi.org/10.1088%2F1751-8113%2F45%2F43%2F435301.

Shahamatnia, Ehsan et al. (2011). "Adaptive imitation scheme for memetic algorithms". In: *Doctoral Conference on Computing, Electrical and Industrial Systems*. Springer, pp. 109–116.

Shnirman, Alexander, Gerd Schön, and Ziv Hermon (Sept. 1997). "Quantum Manipulations of Small Josephson Junctions". In: *Phys. Rev. Lett.* 79 (12),

pp. 2371–2374. DOI: 10.1103/PhysRevLett.79.2371. URL: https://link.aps.org/doi/10.1103/PhysRevLett.79.2371.

Shor, Peter W. (Oct. 1997). "Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer". en. In: *SIAM Journal on Computing* 26.5, pp. 1484–1509. ISSN: 0097-5397, 1095-7111. DOI: 10.1137/S0097539795293172. URL: http://epubs.siam.org/doi/10.1137/S0097539795293172.

Smith, Adam et al. (2019). "Simulating quantum many-body dynamics on a current digital quantum computer". In: *npj Quantum Inf.* 5, p. 106. DOI: 10.1038/s41534-019-0217-0. arXiv: 1906.06343 [quant-ph].

Steane, Andrew (Feb. 1998). "Quantum computing". In: *Reports on Progress in Physics* 61.2, pp. 117–173. DOI: 10.1088/0034-4885/61/2/002. URL: https://doi.org/10.1088/0034-4885/61/2/002.

Susa, Yuki and Hidetoshi Nishimori (Feb. 2021). "Variational optimization of the quantum annealing schedule for the Lechner-Hauke-Zoller scheme". In: *Phys. Rev. A* 103 (2), p. 022619. DOI: 10.1103/PhysRevA.103.022619.

Susa, Yuki, Yu Yamashiro, et al. (Oct. 2018). "Quantum annealing of the p-spin model under inhomogeneous transverse field driving". In: *Phys. Rev. A* 98 (4), p. 042326. DOI: 10.1103/PhysRevA.98.042326. URL: https://link.aps.org/doi/10.1103/PhysRevA.98.042326.

Suzuki, Sei and Masato Okada (2005). "Residual Energies after Slow Quantum Annealing". In: *Journal of the Physical Society of Japan* 74.6, pp. 1649–1652. DOI: 10.1143/JPSJ.74.1649. eprint: https://doi.org/10.1143/JPSJ.74.1649. URL: https://doi.org/10.1143/JPSJ.74.1649.

Torrontegui, Erik et al. (2013). "Chapter 2 - Shortcuts to Adiabaticity". In: *Advances in Atomic, Molecular, and Optical Physics*. Ed. by Ennio Arimondo, Paul R. Berman, and Chun C. Lin. Vol. 62. Advances In Atomic, Molecular, and Optical Physics. Academic Press, pp. 117–169. DOI: https://doi.org/10.1016/B978-0-12-408090-4.00002-5. URL: http://www.sciencedirect.com/science/article/pii/B9780124080904000025.

Trauzettel, Björn et al. (Mar. 2007). "Spin qubits in graphene quantum dots". en. In: *Nature Physics* 3.3, pp. 192–196. ISSN: 1745-2473, 1745-2481. DOI: 10.1038/nphys544. URL: http://www.nature.com/articles/nphys544.

Vedral, V. and M. B. Plenio (Mar. 1998). "Entanglement measures and purification procedures". In: *Phys. Rev. A* 57 (3), pp. 1619–1633. DOI: 10.1103/PhysRevA.57.1619. URL: https://link.aps.org/doi/10.1103/PhysRevA.57.1619.

Venturelli, Davide and Alexei Kondratyev (May 2019). "Reverse quantum annealing approach to portfolio optimization problems". en. In: *Quantum Machine Intelligence* 1.1-2, pp. 17–30. ISSN: 2524-4906, 2524-4914. DOI: 10.1007/s42484-019-00001-w. URL: http://link.springer.com/10.1007/s42484-019-00001-w.

Wang, Hongfeng, Dingwei Wang, and Shengxiang Yang (2009). "A memetic algorithm with adaptive hill climbing strategy for dynamic optimization problems". In: *Soft Computing* 13.8, pp. 763–780.

Wang, Hongfeng, Shengxiang Yang, et al. (2010). "A particle swarm optimization based memetic algorithm for dynamic optimization problems". In: *Natural Computing* 9.3, pp. 703–725.

Wilcoxon, Frank (1992). "Individual comparisons by ranking methods". In: *Breakthroughs in statistics*. Springer, pp. 196–202.

Wu, Jian-da et al. (June 2008). "Adiabatic condition and quantum geometric potential". In: *Phys. Rev. A* 77 (6), p. 062114. DOI: 10.1103/PhysRevA.77.062114. URL: https://link.aps.org/doi/10.1103/PhysRevA.77.062114.

Wu, Yulin et al. (Oct. 2021). "Strong Quantum Computational Advantage Using a Superconducting Quantum Processor". In: *Phys. Rev. Lett.* 127 (18), p. 180501. DOI: 10.1103/PhysRevLett.127.180501. URL: https://link.aps.org/doi/10.1103/PhysRevLett.127.180501.

Wurtz, Jonathan and Peter J. Love (July 2021). "Counterdiabaticity and the quantum approximate optimization algorithm". In: *arXiv:2106.15645 [quant-ph]*. arXiv: 2106.15645. URL: http://arxiv.org/abs/2106.15645.

Yamashiro, Yu et al. (Nov. 2019). "Dynamics of reverse annealing for the fully connected *p*-spin model". In: *Phys. Rev. A* 100 (5), p. 052321. DOI: 10.1103/PhysRevA.100.052321.

Yao, Xin (1993). "An empirical study of genetic operators in genetic algorithms". In: *Microprocessing and Microprogramming* 38.1. Proceedings Euromicro 93 Open System Design: Hardware, Software and Applications, pp. 707–714. ISSN: 0165-6074. DOI: https://doi.org/10.1016/0165-6074(93)90215-7. URL: http://www.sciencedirect.com/science/article/pii/0165607493902157.

A. Adiabatic theorem of quantum mechanics

Adiabatic theorem states that if a system is prepared in the nth eigenstate of a time dependent non-degenerate Hamiltonian in the initial time, it continues to be in the same eigenstate at every instant of time, up to a multiplicative phase factor if the Hamiltonian is driven slowly enough (Born and Fock 1928).
The time independent Schrödinger equation of system described by the Hamiltonian H at time t is given by,

$$H(t)|\psi_n(t)\rangle = E_n(t)|\psi_n(t)\rangle. \tag{A.1}$$

When the system is driven by the Hamiltonian $H(t)$, the state of the system at time t can be expressed in terms of the instantaneous eigen bases (obtained by solving Eq. A.1) at time t of the driving Hamiltonian $H(t)$. i. e.,

$$\Psi(t) = \sum_n c'_n(t)\psi_n(t) \tag{A.2}$$

The coefficients $c'_n(t)$ can be rewritten as

$$\Psi(t) = \sum_n c_n(t)\psi_n e^{i\theta_n(t)}, \tag{A.3}$$

where $\theta_n(t) = -\int_0^t E_n(t)dt$. The evolution of this state with time is described by the time-dependent Schrödinger equation as follows,

$$i\hbar\frac{\Psi(t)}{dt} = H(t)\Psi(t), \tag{A.4}$$

Using the expression A.3 and further solving, we get

$$i\hbar\sum_n\left[\dot{c}_n(t)\psi_n(t)e^{i\theta_n(t)}+c_n(t)\dot{\psi}_n(t)e^{i\theta_n(t)}+c_n(t)\psi_n(t)\dot{\theta}_n(t)e^{i\theta_n(t)}\right] = \sum_n c_n(t)H(t)\psi_n(t)e^{i\theta_n(t)}. \tag{A.5}$$

Further simplifying, we get the expression,

$$\sum_n[\dot{c}_n(t)\psi_n(t)+c_n(t)\dot{\psi}_n(t)]=0. \tag{A.6}$$

Now taking inner product with another instantaneous eigen state $\psi_m(t)$,

$$\sum_n\dot{c}_n\langle\psi_m|\psi_n\rangle e^{i\theta_n(t)}=-\sum_n c_n\langle\psi_m|\dot{\psi}_n\rangle e^{i\theta_n(t)}. \tag{A.7}$$

In the above equation all the terms are time dependent and are not explicity denoted in the rest of the derivation. Because of the orthogonality condition, the inner products of the mth state survive on the left hand side, leading to,

$$\dot{c}_m=-\sum_n c_n\langle\psi_m|\dot{\psi}_n\rangle e^{i(\theta_n-\theta_m)}. \tag{A.8}$$

To find an equivalent expression for the term $\langle\psi_m|\dot{\psi}_n\rangle$, differentiate the time independent Schrödinger equation and take innerproduct with an arbitrary eigen state $|m\rangle$. This gives us,

$$\dot{H}\psi_n+H\dot{\psi}_n=\dot{E}_n\psi_n+E_n\dot{\psi}_n, \tag{A.9}$$

$$\langle\psi_m|\dot{H}|\psi_n\rangle+E_m\langle\psi_m|\dot{\psi}_n\rangle=\dot{E}_n\delta_{mn}+E_n\langle\psi_m|\dot{\psi}_n\rangle. \tag{A.10}$$

For $n\neq m$ we get,

$$\langle\psi_m|\dot{\psi}_n\rangle=\frac{\langle\psi_m|\dot{H}|\psi_n\rangle}{E_n-E_m}. \tag{A.11}$$

Since we are considering adiabatic regime where the rate of change of Hamiltonian is very small, for the case when $n\neq m$, the term $\langle\psi_m|\psi_n\rangle$ can be set to zero. This leads to to the equation of change of the coefficients to be

$$\dot{c}_m=-c_m\langle\psi_m|\dot{\psi}_m\rangle. \tag{A.12}$$

Solving this equation by integration gives,

$$c_m(t) = c_m(0)e^{i\gamma(t)}, \text{ where } \gamma(t) = i\int_0^t \langle \psi_m | \dot{\psi}_m \rangle \, dt. \tag{A.13}$$

Here $\gamma(t)$ is called the geometric phase. It is evident from this equation that if the system is initially in the nth eigen state which means $c_n(0) = 1$, then the probability of finding the system in the nth eigenstate at a time t is also equal to 1. This is the gist of the adiabatic theorem.

A.0.1 Adiabatic criterion

The question of how slow the evolution should be to be in adiabatic limit can be addressed by deriving a quantitative adiabatic criterion. Consider the intermediate step in the previous derivation in Eq. A.8.

$$\begin{aligned} \dot{c}_n &= -\sum_k c_m \langle \psi_n | \dot{\psi}_m \rangle \, e^{i(\theta_m - \theta_n)} \\ &= \sum_m c_m \frac{\langle \psi_n | \dot{H} | \psi_m \rangle}{E_n(t) - E_m(t)} e^{-\frac{i}{\hbar}\int_0^t E_m(t') - E_n(t') dt'} \end{aligned} \tag{A.14}$$

For the adiabatic theorem to be true, we know that $c_m(t) = 1$ whenever $c_{n\neq m}(0) = 1$. Applying the same in the above equation,

$$\dot{c}_{n\neq m} = \frac{\langle \psi_n | \dot{H} | \psi_m \rangle}{E_n(t) - E_m(t)} e^{-\frac{i}{\hbar}\int_0^t E_m(t') - E_n(t') dt'}. \tag{A.15}$$

Integrating the above equation in the limit 0 and T(the final time),

$$c_{n\neq m}(T) = \int_0^T \frac{\langle \psi_n | \dot{H} | \psi_m \rangle}{E_n(t) - E_m(t)} e^{\frac{i}{\hbar}\int_0^t E_n(t') - E_m(t') dt'} dt. \tag{A.16}$$

We know that $c_{n\neq m}(T)$ should be close to zero to satisfy the adiabatic criterion. In the integral, the largest contribution comes from the largest matrix element of $\langle \psi_n(t) | \dot{H} | \psi_m(t) \rangle \approx \overline{\langle \psi |_n \dot{H} | \psi_n \rangle}$, and the smallest energy difference $E_n(t) - E_m(t) \approx \overline{E_n - E_m}$ during the time evolution. The integral in the above equation can

be approximately written as,

$$\begin{aligned} c_{n\neq m}(T) &\approx \int_0^T \frac{\overline{\langle\psi_n|\dot{H}|\psi_m\rangle}}{\overline{E_n(t)-E_m(t)}} e^{\frac{i}{\hbar}\int_0^t \overline{E_n(t')-E_m(t')}dt'} dt \\ &= \frac{\overline{\langle\psi_n|\dot{H}|\psi_m\rangle}}{\overline{E_n(t)-E_m(t)}} \frac{i\hbar}{\overline{E_n-E_m}} \left(e^{-i\overline{E_n-E_m}T/\hbar}-1\right) \\ &\approx \frac{\overline{\langle\psi_n|\dot{H}|\psi_m\rangle}}{\overline{E_n(t)-E_m(t)}} \frac{i\hbar}{\overline{E_n(t)-E_m(t)}}. \end{aligned} \tag{A.17}$$

Since the last term is oscillating the upper bound is considered to obtain the condition for adiabaticity. Now this expression of $c_{n\neq m}(T) \ll 1$. Therefore the condition for adiabaticity is,

$$\frac{\hbar\overline{\langle\psi_n|\dot{H}|\psi_m\rangle}}{\overline{E_n(t)-E_m(t)}^2} \ll 1, \tag{A.18}$$

or more explicitly ($\hbar = 1$),

$$\max_{t\in[0,T]} \frac{|\langle\psi_n(t)|\dot{H}(t)|\psi_m(t)\rangle|}{|E_n(t)-E_m(t)|^2} \ll 1. \tag{A.19}$$

The above equation indicates how slow the Hamiltonian is allowed to be varied with time such that the system is in ground state at a given time. This is one of the simplest and the earliest adiabatic criterion derived. There are other adiabatic conditions in the literature which have proven to be more rigorous (J.-d. Wu et al. 2008; Comparat 2009; Albash and Daniel A. Lidar 2018).

B. Representing ferromagnetic p-spin model in maximum spin subspace

Since the Hamiltonian of the ferromagnetic p-spin model commutes with the total spin operator S^2, the model can be represented in the maximum spin subspace spanned by total spin operators. Here we derive the equivalent Pauli spin operators σ_x, σ_y, σ_z in this subspace.
Spin operators and Pauli matrices are related by,

$$S_{x,y,z} = \frac{\hbar}{2}\sigma_{x,y,z}. \tag{B.1}$$

From here, we set $\hbar = 1$.
The total spin operator S_z operator acts on a system of N spin-$\frac{1}{2}$ particles as,

$$S_z |J,m\rangle = m |J,m\rangle , \tag{B.2}$$

Where J is the maximum spin equal to $N/2$ and $m \in \{-J...J\}$ being $N+1$ in number. The eigen value equation of σ_z in the new subspace is straight forward i. e.,

$$\sigma_z \left|\frac{N}{2},m\right\rangle = 2m \left|\frac{N}{2},m\right\rangle . \tag{B.3}$$

σ_z is diagonal in this subspace with the ith eigen value being $N-2i$. Therefore the matrix elements of σ_z are

$$\sigma_z[i,i] = N-2i, \tag{B.4}$$

where $i = 0...N$.

To find σ_x and σ_y operators, consider the ladder operators S_+ and S_-, defined by,

$$S_\pm |J,m\rangle = S_x \pm iS_y \implies S_x = \frac{S_+ + S_-}{2},\ S_y = \frac{S_+ - S_-}{2i} \tag{B.5}$$

The action of these ladder operators on a state $|J,m\rangle$ is

$$S_\pm |J,m\rangle = \sqrt{J(J+1) \mp m(m+1)}\, |J,m\pm 1\rangle . \tag{B.6}$$

With this definition, the matrix elements of S_x, $\langle i| S_x |i'\rangle$ can be computed. From Eq. B.5,

$$\begin{aligned}\left\langle \frac{N}{2},i \middle| S_x \middle| \frac{N}{2},i' \right\rangle &= \left\langle \frac{N}{2},i \middle| \frac{S_+ + S_-}{2} \middle| \frac{N}{2},i' \right\rangle \\ &= \frac{1}{2}\sqrt{\frac{N}{2}\left(\frac{N}{2}+1\right) - i'(i'+1)}\,\delta_{i,i'+1} \\ &+ \frac{1}{2}\sqrt{\frac{N}{2}\left(\frac{N}{2}+1\right) + i'(i'+1)}\,\delta_{i,i'-1}\end{aligned} \tag{B.7}$$

This implies that only the elements $\langle i| S_x |i-1\rangle$ and $\langle i| S_x |i+1\rangle$ have non zero values,

$$\left\langle \frac{N}{2},i \middle| S_x \middle| \frac{N}{2},i+1 \right\rangle = \frac{1}{2}\sqrt{\frac{N}{2}\left(\frac{N}{2}+1\right) - \left(\frac{N}{2}-i-1\right)\left(\frac{N}{2}-i\right)} \tag{B.8}$$

$$= \frac{1}{2}\sqrt{(i+1)(N-i)}. \tag{B.9}$$

Similarly,

$$\begin{aligned}\left\langle \frac{N}{2},i+1 \middle| S_x \middle| \frac{N}{2},i \right\rangle &= \frac{1}{2}\sqrt{(i+1)(N-i)}, \\ \left\langle \frac{N}{2},i \middle| S_y \middle| \frac{N}{2},i+1 \right\rangle &= \frac{1}{2i}\sqrt{(i+1)(N-i)}, \\ \left\langle \frac{N}{2},i+1 \middle| S_y \middle| \frac{N}{2},i \right\rangle &= -\frac{1}{2i}\sqrt{(i+1)(N-i)}.\end{aligned} \tag{B.10}$$

The corresponding equivalent Pauli matrices $\sigma_{x,y,z}$ can be obtained from the relation in Eq. B.1. The ferromagnetic p-spin model can be defined using these new

operators with the dimension of the Hamiltonian being $N+1$.

C. Numerical details of memetic algorithms

C.1 Experimental setup

In order to evaluate the proposed approach of mapping p-body Hamiltonian into 2-body Hamiltonian using memetic algorithms, we consider two instances of the problem at hand. In particular, we consider the mapping of the ferromagnetic p-spin model with (i) $N = 3$ and $p = 3$, needing one ancilla which implies $M = 4$; (ii) with $N = 4$ and $p = 3$, needing two ancillae which makes $M = 6$. The evaluation involves these two instances because it is possible to find the analytical solution for them and, as a consequence, this allows us to study the suitability of our proposal in finding the most opportune free parameters. Formally, the correct free parameters for these two configurations are, respectively,

$$\vec{v}_{M=4} \equiv (-3, -3\delta, \frac{26}{9}, \frac{26}{9}, \frac{26}{9}, 2\delta, 2\delta, \frac{16}{3}, -\frac{8}{3} - \delta, -\frac{8}{3}, -\frac{8}{3}) \tag{C.1}$$

$$\begin{aligned}\vec{v}_{M=6} \equiv (&-4, -3\delta, -3\delta, \frac{7}{2}, \frac{7}{2}, \frac{7}{2}, \frac{7}{2}, 0, 2\delta, 2\delta, \\ &3, 3, 3, 3, 2\delta, 2\delta, -3-\delta, -3, -3, -3, -3, -3-\delta).\end{aligned} \tag{C.2}$$

In our experimentation, we will fix the constant δ used to compute the penality cost E^{vec}_{pen} reported in Eq. 3.11 to 50 because this value provides a large separation between the largest eigenvalue of the target subspace and the smallest eigenvalue of the nonphysical one, in both considered problem instances, as described in (**Acampora2019**).

To visualize the results of the experimentation, a graphical methodology, named *box plots*, is used. A box plot is a standardized way of displaying data based on a

five-number summary: the minimum, the maximum, the sample median, and the first (Q1) and third (Q3) quartiles. Graphically, a boxplot is constructed of two parts, a box and a set of whiskers. The lowest point is the minimum of the data set and the highest point is the maximum of the data set. The box is drawn from Q1 to Q3 with a horizontal line drawn in the middle to denote the median. Moreover, outliers are plotted as individual points or crosses. Box plots have been chosen because they are a non-parametric method: they display variation in samples of a statistical population without making any assumptions of the underlying statistical distribution. Indeed, the used samples will be fitness values obtained by different runs of the algorithms whose the statistical distribution is unknown.

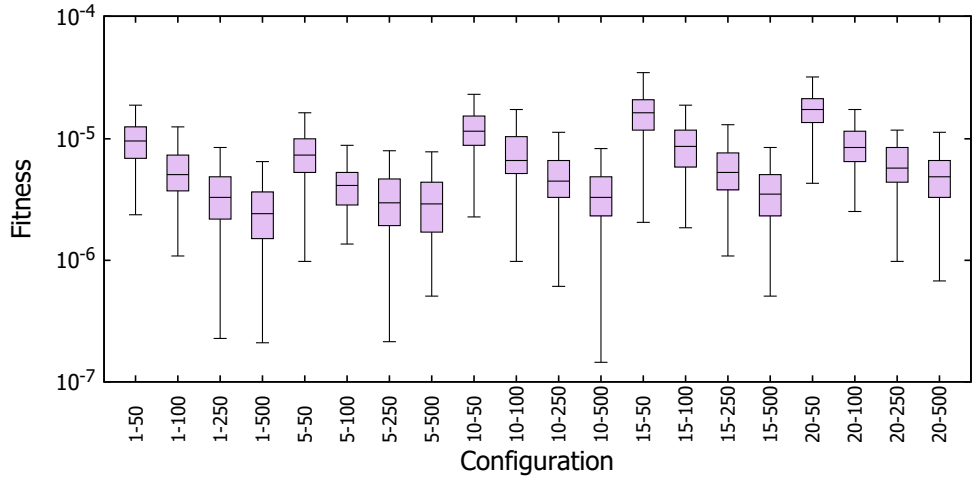
C.2 Tuning of hyper-parameters of the memetic algorithm

The configuration of the memetic algorithm used in the experiments is the result of a design study involving the genetic operators and the other hyper-parameters as described in this section. In particular, we perform a design study involving the tuning of the following hyper-parameters: crossover probability, mutation probability, tournament size, frequency and intensity. The tested values for each one of them together with the other hyper-parameter values are reported in Table C.1. By using 3 different values for the crossover probability and mutation probability, 2 for tournament size, 4 for intensity and 5 for frequency, our design study would have involved the assessment of 360 different combinations. Hence, to reduce the computational effort related to the hyper-parameter tuning, we have decided to perform a tuning study in two steps. The first step is devoted to analyse the trend of the frequency and intensity values. In this first step, the values for crossover probability, mutation probability and tournament size are fixed. The second step, instead, has involved the study of all hyper-parameters by reducing the number of different values for the intensity and frequency to the best ones obtained in the first step. Being MAs stochastic procedures, we perform 100 runs for each configuration in both steps of the design study.
Fig. C.1 shows the results of the first step for both the considered problem instances by using box plots. As shown by the median fitness values represented in the box plots, the configurations with a low frequency are better than the others for both instances (in particular, frequency value 1 characterizes the best configurations for both instances). Instead, the intensity hyper-parameter is not characterized by a

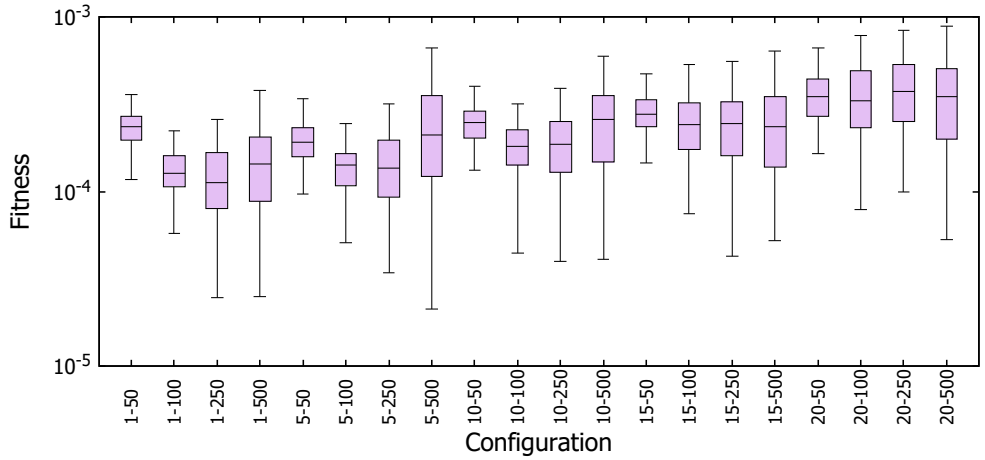
Table C.1: Parameter settings for the tuning of the memetic hyper-parameters

Population size	20
Stopping criteria	1,000,000 fitness evaluations
Crossover probability	0.2, 0.5 and 0.9
Mutation probability	0.02, 0.05 and 0.09
Tournament size	3 and 5
Intensity	50, 100, 250 and 500 fitness evaluations
Frequency	1, 5, 10, 15 and 20 individuals

clear trend. Indeed, for the case $N = 3$ and $M = 4$, the best values are the highest ones, whereas, for the case $N = 4$ and $M = 6$, the best values are the lowest ones. For this reason, we select the following values to perform the second step: 1 for frequency and the extreme values (50 and 500) for the intensity.
Fig. C.2 shows the results of the second step for both the considered problem instances by using box plots graphical method. The configuration for which the box has the lowest median is considered to be the best configuration of the memetic algorithm. As shown in the box plots, for the case $N = 3$, the best configuration is the 11th one that is characterized by $p_m = 0.09$, $p_c = 0.2$, $N_T = 5$, $l_i = 50$ and $l_f = 1$, whereas, the best configuration for the case $N = 4$ is the 12th one characterized by $p_m = 0.09$, $p_c = 0.2$, $N_T = 5$, $l_i = 500$ and $l_f = 1$. These configurations for MA are used in the comparison studies reported in the next sections.

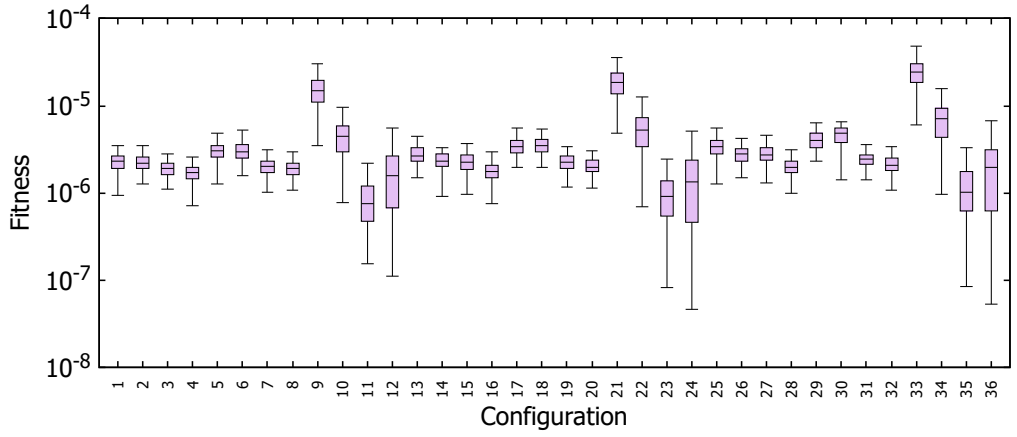


(a)

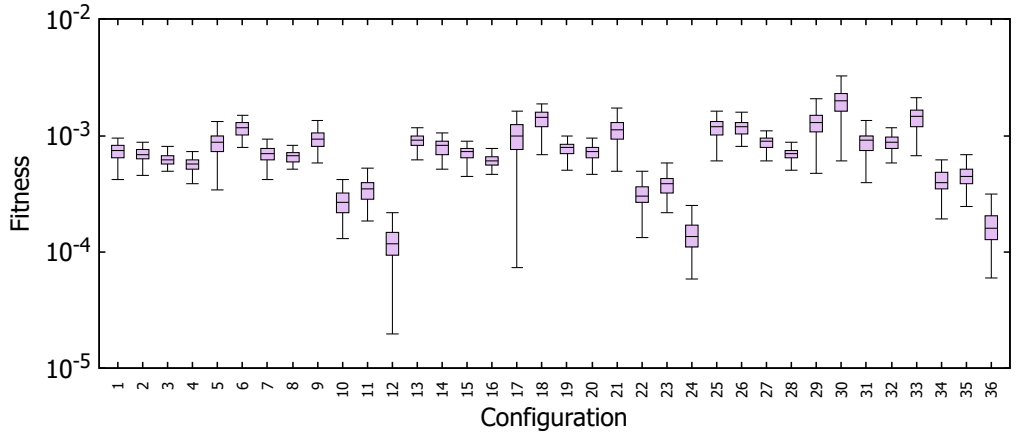


(b)

Figure C.1: Results of the first step of the tuning process for (a) the case $N = 3$ and $M = 4$ and (b) the case $N = 4$ and $M = 6$. Each configuration is denoted as $x - y$ where x is the frequency value and y is the intensity value. For sake of readibility, outliers are not displayed.



(a)



(b)

Figure C.2: Results of the second step of the tuning process for (a) the case $N = 3$ and $M = 4$ and (b) the case $N = 4$ and $M = 6$. Configurations are described in Table C.2. For sake of readibility, outliers are not displayed.

Table C.2: Configurations of the second step of the tuning study. In the table, values for crossover probability(p_c), mutation probability(p_m), tournament selection size (N_T), intensity (l_i) and frequency (l_f) chosen are tabulated.

#	p_c	p_m	N_{T}	l_i	l_f	#	p_c	p_m	N_{T}	l_i	l_f
1	0.2	0.02	3	50	1	19	0.5	0.05	5	50	1
2	0.2	0.02	3	500	1	20	0.5	0.05	5	500	1
3	0.2	0.02	5	50	1	21	0.5	0.09	3	50	1
4	0.2	0.02	5	500	1	22	0.5	0.09	3	500	1
5	0.2	0.05	3	50	1	23	0.5	0.09	5	50	1
6	0.2	0.05	3	500	1	24	0.5	0.09	5	500	1
7	0.2	0.05	5	50	1	25	0.9	0.09	3	50	1
8	0.2	0.05	5	500	1	26	0.9	0.02	3	500	1
9	0.2	0.09	3	50	1	27	0.9	0.02	3	50	1
10	0.2	0.09	3	500	1	28	0.9	0.02	3	500	1
11	0.2	0.09	5	50	1	29	0.9	0.05	5	50	1
12	0.2	0.09	5	500	1	30	0.9	0.05	5	500	1
13	0.5	0.02	3	50	1	31	0.9	0.05	3	50	1
14	0.5	0.02	3	500	1	32	0.9	0.05	3	500	1
15	0.5	0.02	5	50	1	33	0.9	0.09	5	50	1
16	0.5	0.02	5	500	1	34	0.9	0.09	5	500	1
17	0.5	0.05	3	50	1	35	0.9	0.09	3	50	1
18	0.5	0.05	3	500	1	36	0.9	0.09	3	500	1

D. Numerical details of genetic algorithms used in genetic optimization of quantum annealing

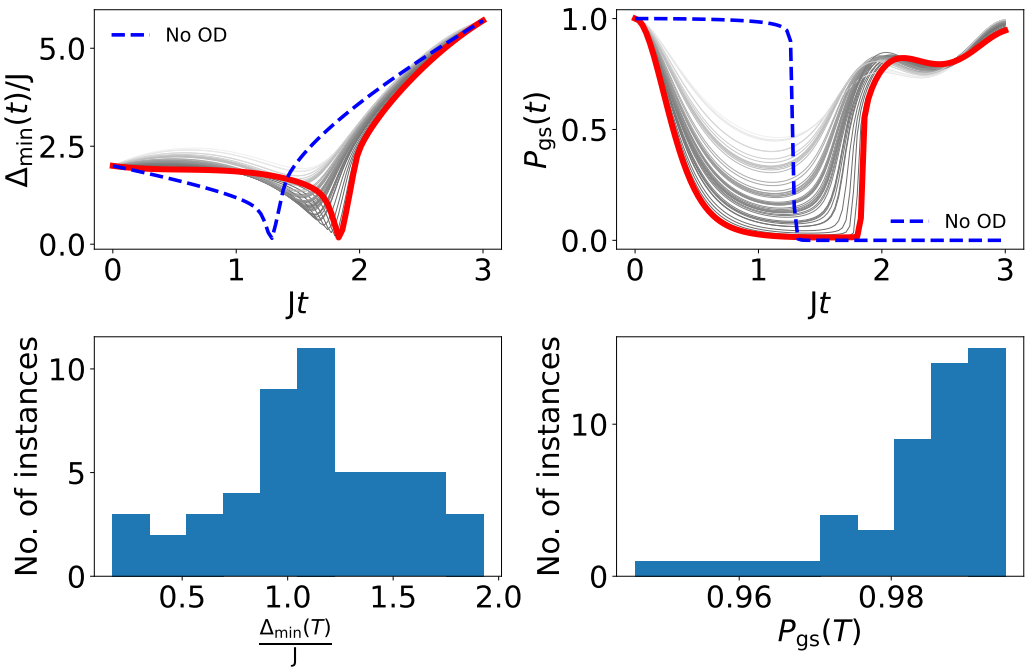
D.1 Optimizing the hyperparameters of genetic algorithms

Genetic algorithms are characterized by hyperparameters pertaining the selection, crossover and mutation processes. To be precise, the individual undergo the process of mutation with a probability of p_m, wherein the real numbers of the chromosome are altered according to a Gaussian distribution with variance σ^2 and mean μ. Further each real number (gene) in the chromosome undergoes mutation with the probability p_{ind}. We perform two-point crossover among the parent chromosomes where a string of values are cut and exchanged between the parents to produce two new solutions and this process occurring with a probability of p_c. We choose the tournament selection process where among every N_{T} individual chromosomes, we choose the best chromosome as parent for producing offspring. This cycle of generation repeats. In general, for each optimization problem it is advisable to perform an initial experimentation to fix these hyperparameters which give the best solution to the problem (Giovanni Acampora, Vittorio Cataudella, Pratibha R Hegde, et al. 2019; Giovanni Acampora, Vittorio Cataudella, Pratibha Raghupati Hegde, et al. 2021; Fortin et al. 2012). In particular, for the problem of annealing schedules optimization, we have tuned and chosen the hyperparameters values, $N_{\text{T}} = 6$, $p_c = 0.75$, $p_m = 0.35$, $p_{\text{ind}} = 0.1$, $\sigma^2 = 0.6$, $\mu = 0$. For the problem of finding the optimal driving, the best combination of hyper parameters is found to be $N_{\text{T}} = 3$, $p_c = 0.3$, $p_m = 0.9$, $p_{\text{ind}} = 0.1$, $\sigma^2 = 1$, $\mu = 0$. However, in this paper, for the optimization problems chosen, varying the hyperparameters have minimal

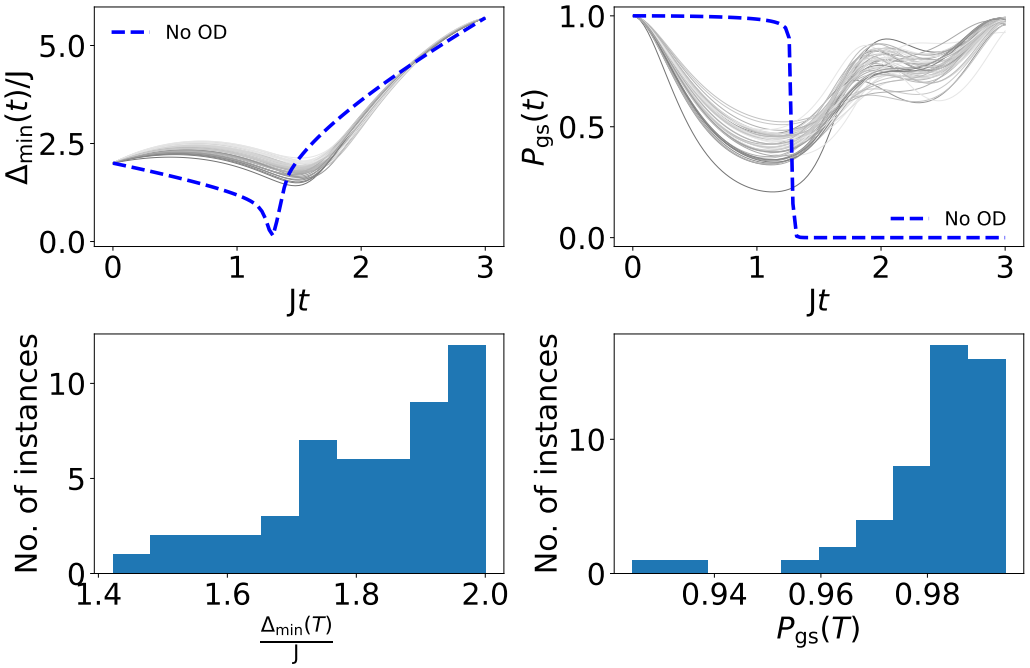
effect on the overall quality of the solutions. For example, $p_m = 0.9$ gives the best fidelity, however decreasing p_m leads to searching in smaller search space which in turn reduces the number of solutions which simply increase the energy scaling of the system. Meanwhile, by doing so, the fidelity is not affected by a great deal.

D.2 Selection of chromosome from the Pareto Optimal Front in MOGA

The output of a MOGA which is implemented using Non-dominated Sorting algorithm II, is a set of chromosomes with the best ranking in terms of their domination over the rest of the chromosomes (Deb et al. 2002; Fortin et al. 2012). This set of chromosomes is called Pareto optimal front. In the end of evolution, we choose one of the chromosomes in the Pareto optimal front, which has a good trade-off between the area under the ground-state probability curve and fidelity. In this work, we choose the chromosome with the maximum value of $0.4 \times \text{area} + 0.6 \times P_{\text{gs}}(T)$ and use this solution to perform adiabatic evolution and compute results. As an example, we consider the ferromagnetic p-spin model with 40 spins and optimize a local optimal driving (OD) operator (with fixed annealing schedules). We show the difference in the solutions obtained from SOGAs and MOGAs in Fig. D.1. In MOGA, with the imposition of large area under the curve of ground state probabilities, the genetic algorithm converges to solutions where there are no energy crossings. The same can be seen in the plots of $\Delta_{\text{min}}(t)$ and the histogram of $\Delta_{\text{min}}(T)$. The median fidelity using the results of SOGA is ≈ 0.983, whereas with MOGA, the median fidelity is ≈ 0.981.



(a)



(b)

Figure D.1: Results of the optimization of local OD operator with 3-local operators for a p-spin model with 40 spins and $p = 3$. We compare the performance of SOGA in (a) and MOGA in (b). The red bold line in the solutions obtained from SOGA indicate the solutions where there are energy level crossings. The corresponding results using MOGA do not show this kind of solutions.