

## Quantum computer-assisted global optimization in geophysics illustrated with stack-power maximization for refraction residual statics estimation

Dukalski, Marcin; Rovetta, Diego ; van de Linde, Stan; Möller, Matthias; Neumann, Niels; Phillipson, Frank

**DOI**

[10.1190/geo2022-0253.1](https://doi.org/10.1190/geo2022-0253.1)

**Publication date**

2023

**Document Version**

Final published version

**Published in**

Geophysics

**Citation (APA)**

Dukalski, M., Rovetta, D., van de Linde, S., Möller, M., Neumann, N., & Phillipson, F. (2023). Quantum computer-assisted global optimization in geophysics illustrated with stack-power maximization for refraction residual statics estimation. *Geophysics*, 88(2), V75-V91. <https://doi.org/10.1190/geo2022-0253.1>

**Important note**

To cite this publication, please use the final published version (if applicable).  
Please check the document version above.

**Copyright**

Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

**Takedown policy**

Please contact us and provide details if you believe this document breaches copyrights.  
We will remove access to the work immediately and investigate your claim.

***Green Open Access added to TU Delft Institutional Repository***

***'You share, we take care!' - Taverne project***

**<https://www.openaccess.nl/en/you-share-we-take-care>**

Otherwise as indicated in the copyright section: the publisher is the copyright holder of this work and the author uses the Dutch legislation to make this work public.

# Quantum computer-assisted global optimization in geophysics illustrated with stack-power maximization for refraction residual statics estimation

Marcin Dukalski<sup>1</sup>, Diego Rovetta<sup>1</sup>, Stan van der Linde<sup>2</sup>, Matthias Möller<sup>3</sup>, Niels Neumann<sup>4</sup>, and Frank Phillipson<sup>5</sup>

## ABSTRACT

Much of recent progress in geophysics can be attributed to the adaptation of heterogeneous high-performance computing architectures. It is projected that the next major leap in many areas of science, and hence hopefully in geophysics too, will be due to the emergence of quantum computers. Finding a right combination of hardware, algorithms, and a use case, however, proves to be a very challenging task — especially when looking for a relevant application that scales efficiently on a quantum computer and is difficult to solve using classical means. We find that maximizing stack power for residual statics correction, an NP-hard combinatorial optimization problem, appears to naturally fit a particular type of quantum computing known as quantum annealing. We express the underlying objective function as a quadratic unconstrained binary optimization, which is a

quantum-native formulation of the problem. We choose some solution space and define a proper encoding to translate the problem variables into qubit states. We find that these choices can have a significant impact on the maximum problem size that can fit on the quantum annealer and on the fidelity of the final result. To improve the latter, we embed the quantum optimization step in a hybrid classical-quantum workflow, which aims to increase the frequency of finding the global, rather than some local, optimum of the objective function. Finally, we find that a generic, black-box, hybrid classical-quantum solver also could be used to solve stack-power maximization problems proximal to industrial relevance and capable of surpassing deterministic solvers prone to cycle skipping. A custom-built workflow capable of solving larger problems with an even higher robustness and greater control of the user appears to be within reach in the very near future.

## INTRODUCTION

Geophysics features some of the most challenging problems that can be found in the computational sciences. Those that can be reliably solved require sufficient data from measurements, efficient algorithms, and enough computing power. Improvements in either of the three aspects facilitate much of the progress in geophysics and help bolster our confidence in the information we have about the subsurface.

Adaptation of heterogeneous high-performance computing (HPC), e.g., the use of central and graphical processing units (CPUs

and GPUs), was a major driver behind the expansion of the set of problems that we can address in geophysics today. It is expected, and perhaps with excessive hype (Grumblin and Horowitz, 2018; Ruane et al., 2022), that the quantum processing units (QPUs) would address some specific computational tasks, which are beyond the reach of CPUs and GPUs (Möller and Vuk, 2017). However, the question is, what are these tasks? Are any of them of any practical relevance? And are there any geophysical problems that can benefit from them? Here, we show that answering these questions requires sufficient understanding of quantum hardware, quantum

Manuscript received by the Editor 25 April 2022; revised manuscript received 2 August 2022; published ahead of production 22 November 2022; published online 14 February 2023.

<sup>1</sup>Aramco Global Research Center Delft, The Netherlands. E-mail: marcin.dukalski@aramcooverseas.com (corresponding author); diego.rovetta.1@aramcooverseas.com.

<sup>2</sup>Delft University of Technology, Institute of Applied Mathematics, The Netherlands and TNO, Department of Applied Crypto and Quantum Algorithms, Den Haag, The Netherlands. E-mail: stan.vanderlinde@tno.nl.

<sup>3</sup>Delft University of Technology, Delft Institute of Applied Mathematics, Delft, The Netherlands. E-mail: m.moller@tudelft.nl.

<sup>4</sup>TNO, Department of Applied Crypto and Quantum Algorithms, Den Haag, The Netherlands. E-mail: niels.neumann@tno.nl.

<sup>5</sup>TNO, Department of Applied Crypto and Quantum Algorithms, Den Haag, The Netherlands and Maastricht University, School of Business and Economics, Maastricht, The Netherlands. E-mail: frank.phillipson@tno.nl.

© 2023 Society of Exploration Geophysicists. All rights reserved.

software, and the use case. One needs to consider the capabilities and limitations of the hardware, the available hardware-dependent software, and its ability to handle classical information. Finally, the use case has to be one that is relatively hard enough to solve, yet not be constrained by the hardware and software limitations.

The field of quantum computing is very broad in scope, with the underlying principle being using quantum effects to perform computations (Feynman, 1985; Mermin, 2007; Nielsen and Chuang, 2010). The field features several physical quantum realizations, e.g., cold atoms/ions (Bruzewicz et al., 2019; Henriot et al., 2020), various integrated circuits (Humble et al., 2019; Gonzalez-Zalba et al., 2021), and photons (Flamini et al., 2018). On top of that, there also are two different quantum computing philosophies: (1) gate-based universal or programmable (Rieffel and Polak, 2011) and (2) annealing/adiabatic quantum computing (AQC) (McGeoch, 2014; Albash and Lidar, 2018). Each combination of physical realization and philosophy comes with its own strengths and limitations. This can have a very big impact on the likelihood of success at superseding a classical computer at one's use case (Leymann and Barzen, 2020), which is especially true because many of these systems are still in their relative infancy (Preskill, 2018; Córcoles et al., 2019).

Contrary to some expectations, gate-based QPUs are very limited in what they can do at the moment or could do in the future. In addition, it can be difficult to assess what relevant task they can do faster, cheaper, or better than CPUs or GPUs (Leymann and Barzen, 2020). As a result, the range of potential use cases, at least for now, is relatively narrow. Using quantum systems to simulate other quantum systems (e.g., in the field of quantum chemistry; Cao et al., 2019; Bauer et al., 2020; Outeiral et al., 2021) was one of the earliest and most obvious use cases. The relatively small pool of gate-based quantum computing algorithms (Montanaro, 2016) still gave rise to a range of future potential applications in, for example, finance (Orus et al., 2019; Woerner and Egger, 2019), telecom (Phillipson et al., 2021a), and energy system optimization (Koretsky et al., 2021). A growing recent popularity of quantum machine learning (QML) (Biamonte et al., 2017) could give rise to applications in many other fields.

The concept of quantum annealing (QA) QPU goes back to a QA algorithm (Kadowaki and Nishimori, 1998), which also found its applications in geosciences (Liu et al., 2018). This algorithm extends the concepts of simulated (thermal) annealing and appends them with a classical simulation of quantum effects, such as quantum tunneling and superposition, aiding transitions between local minima. This promises higher certainty that the optimization finds a global, rather than a local, optimum of some multimodal objective function. A QA QPU is a physical realization of the QA algorithm — a device consisting of superconducting qubits and couplers, which can be adjusted to emulate the quantum dynamics of a custom Ising model (IM) Hamiltonian. Recently, Abel et al. (2021) show that this device is very likely to be more successful in finding a global optimum of an objective function than many classical methods. Because QA QPUs specialize in solving optimization problems, these machines have so far shown a much wider range of use cases (Laumann et al., 2015; Neukart et al., 2017; Crosson and Lidar, 2021; Mato et al., 2021; Phillipson and Chiscop, 2021c; Phillipson et al., 2021b). Quantum annealers also are seen as possible enablers in training any ML/artificial intelligence applications (Delilbas et al., 2021; Phillipson et al., 2021a).

For these reasons, using QA QPUs, henceforth just called QA, is very interesting in geophysical applications, in which getting stuck in

a local optimum is a not-uncommon problem (Bosisio et al. 2014). In many cases, finding a second best optimum, one which is far away in some parameter space from the global one, is of little interest and could offer little improvement to, or worse, provide a false sense of certainty about the subsurface. O'Malley (2018), Greer and O'Malley (2020), and Souza et al. (2022) show first examples of the types of problems that can be encountered in geophysics and solved using a quantum annealer, albeit with a lesser focus on addressing a realistic and business relevant problem with a multimodal objective function.

The main result of this work is to present a use case of quantum computing that is both scalable and practically relevant. We demonstrate that using QA to solve a stack-power maximization (SPM) problem satisfies the conditions of finding the right combination of hardware, software, and relevance. SPM can be used, for example, in refraction residual statics estimation (RRSE). This problem often suffers from cycle skipping and finding the optimal solution may require clever sampling of the objective function (Wiggins et al., 1976). This can be done, for example, through some meta-heuristic approaches (Rothmann, 1985; Pierini et al., 2019). We demonstrate the QA implementation of the problem on a 5000-qubit machine (McGeoch et al., 2020) and propose a classical-quantum workflow, which improves the probability of finding the global optimum. We also use qbsolv (Booth et al., 2017), a standard hybrid classical-quantum solver, to demonstrate that this very first implementation is at least on par with the industry standard classical solvers on problems of industrially relevant size.

This work is structured as follows. We dedicate the first three sections to explain and justify the choices that we made when selecting the quantum computing philosophy, the hardware, and the use case. In the “Are all problems equally difficult?” section, we explain the computer scientist perception of what makes a problem hard and contrast it with that of a geophysicist in the “Difficult problems in geophysics” section. Upon establishing this, it should become clear why there might be little point of trying to solve certain geophysical problems using quantum computers. In the “Quantum computing in a nutshell” section, we review the relevant basics of quantum computing, in which we briefly describe and compare the gate and the adiabatic models of quantum computing. We use this analysis to justify the choice of the AQC, and QA specifically because of the current most likely platform for geophysical applications. This narrows down the class of geophysically relevant optimization problems, and in the “Stackpower maximization (SPM) for refraction residual statics estimation (RRSE)” section, we explain and show that SPM makes for a good candidate. We also discuss the application of simulated annealing (SA) to solve this problem. Furthermore, in the “Stack-power maximization as a quadratic unconstrained binary optimization” section, we show how to translate the problem into one that is native to a quantum computer and consider different variable encodings and constraints. We also explain why this quantum computation requires classical preprocessing and may benefit from classical postprocessing in practice. In the “Discussion” section, we present the process of identifying another quantum-assisted optimization candidate and offer some outlook. The acronyms used throughout this paper are listed in the “Nomenclature” section. This work is an extension of the work previously presented by van der Linde (2021) and van der Linde et al. (2022a, 2022b).

## ARE ALL PROBLEMS EQUALLY DIFFICULT?

Combinatorial optimization problems often are considered to be harder to solve than continuous ones (Schrijver, 2003). Admittedly,

one may argue whether, for example, in an inverse scattering problem, finding the set of parameters that characterize a given medium and explain the measured data should be considered easy, but, at least conceptually, the continuous nature of the problem enables a variety of systematic and well-established computational approaches to approximate the optimal solution step by step. It often is difficult to tune all these parameters in such a way that the global optimum is attained exactly — especially in multiobjective optimization problems — but in many practical cases a good-enough solution that is close to the optimum can be obtained efficiently. It is the continuous nature of this type of optimization problems that ensures that a small change in the input implicates a not-too-large change in the output.

This is no longer the case in combinatorial optimization problems due to their discrete nature. Since a small discrete change in problem parameters can lead to a very large change in the output, one may need to exhaustively check among all the outcomes before guaranteeing that a better solution cannot be found. The difficulty level of such problems is commonly characterized by the classes P and NP. Problems for which a deterministic algorithm with polynomial runtime exists belong to the class P and are considered easy to solve. Polynomial runtime of course covers a wide range, from linear or quadratic in the number of inputs to very large powers that make them prohibitively expensive and far from what one would in practice call easy. An example is the solution of a linear system of equations of the form  $\mathbf{Ax} = \mathbf{b}$  using the Gaussian elimination algorithm, which scales cubically in the number of vector entries  $n$ , i.e.,  $O(n^3)$ . Although this is still polynomial complexity, and therefore classified as easy in the P-NP landscape, this algorithm is considered prohibitively expensive for practical large-scale computations. As a remedy, special-purpose iterative solution algorithms have been developed for specific instances of the system matrix  $\mathbf{A}$ , which, at best, lead to linearly scaling runtime, i.e.,  $O(n)$ .

On the other hand, the class NP comprises problems for which a nondeterministic algorithm exists to solve the problem at hand in polynomial time. This concept might need some further explanation. For example, we consider an  $n \times n$  Sudoku puzzle. Finding the solution often is difficult (and can take hours), but checking that a given answer is indeed a solution is easy, that is, doable in polynomial time. More formally, NP problems often assume a so-called oracle — the nondeterministic part — that guesses the solution somewhat magically, whereas the verification step can be performed deterministically and in polynomial time.

The reason why many combinatorial optimization problems belong to the class NP is because of the fact that a small number of input variables can lead to exponentially many combinations of outcomes that need to be considered. Think of packing different items, each having a different weight and value, into a knapsack with fixed-size capacity. The goal is to maximize the overall value of the items selected for being put into the knapsack while at the same time not exceeding the capacity limit.

NP-hard problems are even worse than their NP counterparts because they are considered to be at least as hard to solve as the hardest problem in NP. Moreover, for NP-hard problems, it is as difficult to find or verify that something is a solution to a problem. The intersection of NP and NP-hard problems is called the class of NP-complete problems and is commonly referred to as the most difficult to solve problems in NP. It can be shown that a solution to any problem in NP-complete implies that one is able to solve any problem in NP.

The question whether the two classes P and NP fall together is one of the outstanding challenges in modern theoretical computer science. Luckily, we do not have to wait for the ultimate solution of the P-versus-NP problem to exploit the internal structure of the different complexity classes. It often is possible to convert one difficult problem A into another difficult problem B in polynomial time. This does not change the overall complexity of the original problem, but it might well be that finding an efficient algorithm for problem B is easier. This is exactly the process that we will use in the section “SPM for RRSE”, in which we will attempt to rewrite the problem in terms of the so-called IM, a prominent example of an NP-hard problem. This forms the basis for using QA to solve combinatorial problems.

## DIFFICULT PROBLEMS IN GEOPHYSICS

The problems (just) outside of the intersection of algorithms, data, and HPC are what geophysicists consider to be hard; however, that does not make them hard in the context of computational complexity classes discussed previously. For example, 3D seismic multiple attenuations could be deemed difficult (Dragoset et al., 2010) because it requires a very large amount of data (with costly input/output [I/O]) and adaptive subtraction, which has to be assessed by the end user. This, however, is a problem that is not hard in our context because its computational complexity grows polynomially (i.e.,  $O(N^k)$ , where  $N$  is the number of traces and  $k$  is some problem-dependent constant, which happens to be large). In addition, for simple enough cases with only one or two relevant multiples that need to be removed, the correct outcome is either known a priori or can be easily assessed to be correct on output, e.g., for first-order deepwater surface-related multiples. This becomes hard in shallow water, where consecutive orders of multiples overlap. The only known way to address that is to formulate this as underconstrained inverse problem, whereby for each possible source wavelet there is a possible surface primary, both of which are a priori unknown (see, e.g., Lin and Herrmann [2014]). Full-waveform inversion (FWI) is a great example of another hard problem from the perspective of a geophysicist, in which its multimodal objective function can only be explored by repeatedly modeling large volumes of (elastic) data and by using clever tricks aimed at avoiding local minima or barren plateaus (Yang et al., 2018). From the point of view of a (quantum) computational scientist, however, it is mainly the question of certainty of whether the minimum found is global and how to get there efficiently that makes the problem hard. The modeling, even though it might consume most of computational time, falls into the easy or deterministic class.

The question remains, are these problems suitable for a quantum computer? And if so, which quantum-computing philosophy — gate-based universal quantum computing or QA — is the most suitable one? To better appreciate the answer to this, we have to look closely at what these computers can and cannot do and the size of the current intersection of the three sets: field measurement, quantum algorithms, and quantum compute power. In particular, here we will find an additional quandary of how to move the measurements to a quantum computer, as well as physical limitations on the range of calculations that a quantum algorithm can carry out. We will discuss two forms of quantum computing: so-called gate or circuit model (requiring algorithms) and AQC or QA (algorithm free and optimization focused). We will show how the range of applications in geophysics on the former is quite limited, whereas the latter can already boast a few use cases (delegating the optimization step of FWI on a



toy model example being one of them; Greer and O'Malley, 2020). However, we wish to show that the SPM problem, such as for residual statics estimation, is an interesting candidate for being solved efficiently by a quantum computer. This is because this problem is not only hard but also easily implementable and scalable for today's quantum computers. We also show that (much like in the Greer and O'Malley [2020] example) classical computing will remain indispensable in theory and also in practice, in which all the imperfections from the interface of the classical and quantum worlds enter our solutions.

## QUANTUM COMPUTING IN A NUTSHELL

### The basic elements

In this section, we wish to give only a very small glimpse at the aspects of quantum computing that are relevant for potential geophysical applications. A wealth of details on what we cover here and beyond can be found in Nielsen and Chuang (2010).

A basic element of any quantum computer is a so-called quantum bit (qubit). Unlike a classical bit, which can be either zero or one, a qubit can be in a superposition of two states at once — much like a superposition of two differently polarized waves — which we denote by  $a_0|0\rangle + a_1|1\rangle$ , using the Dirac's ket  $|\dots\rangle$  notation, with complex-valued amplitudes  $a_k$ . To perform any practically meaningful computation, one needs a wavefunction  $\psi$  of a register of  $N$  qubits

$$|\psi\rangle = a_{00\dots0}|00\dots0\rangle + a_{00\dots1}|00\dots1\rangle + \dots + a_{11\dots1}|11\dots1\rangle, \quad (1)$$

which form a superposition of up to  $2^N$   $N$ -bit long classical registers. This superposition state also could be used to efficiently store and encode information. For instance, a vector of length  $M$  can be mapped onto a quantum state of  $\log_2 M$  qubits, e.g., a vector  $[b_1, b_2, \dots, b_{16}]$  can be encoded onto four qubits with  $a_{0000} = b_1$ ,  $a_{0001} = b_2, \dots, a_{1111} = b_{16}$ . A vector of length  $M = 10^9$  can be encoded with only 30 qubits. Computation amounts to changes in the amplitudes  $a_k$ . The extraction of the information encoded in and processed by the quantum states requires an irreversible measurement, which collapses the qubit register to any of the possible binary registers, which takes place with probability given by the square of the amplitude in front of the given register, e.g.,  $|a_{0010011}|^2$ . However, the latter can only be estimated with increasing precision by repeating the quantum computation and the measurement for a given number of times. This is similar to determining the probabilities of a biased coin. In spite of this additional computational overhead, there are some calculations that, in theory, using perfect quantum computers, can still be done much faster (in terms of number of elementary computational steps) using a quantum computer than its classical counterpart.

### The gate/circuit model

The gate-based quantum computing model uses a sequence of single- and two-qubit reversible operations and the irreversible measurement to perform computations. There are only very few algorithms known that perform some meaningful operation with a very clear application with sights of scaling it up to an industry-relevant problem and at the same time offer some kind of quantum

advantage. These include scenario testing, or unstructured search algorithm (Grover, 1996), factorization of semiprime numbers (Shor, 1994), or solving (some) linear inverse problems through the Harrow Hassidim Lloyd (HHL) algorithm (Harrow et al., 2009). The development of new quantum algorithms can be a huge challenge. First, algorithm design often relies on human intuition, which can be at odds with the laws of quantum mechanics (Venegas-An-draca et al., 2018). Second, even a correctly functioning quantum algorithm might be of little value if it does not provide a significant quantum advantage; that is, it outperforms its best known classical counterpart that solves the same problem. In addition, with the state-of-the-art circuit model machines of today, quantum algorithms that require a large number of gates are unusable in practice because gate application is imperfect and introduces a significant amount of noise that quickly ruins the results.

For example, the HHL algorithm, offering to solve least-squares minimization problems of the kind  $\|\mathbf{Ax} - \mathbf{b}\|_2^2$  with a theoretically exponential speed up over a classical algorithm, could be appealing as a subroutine in several applications in geophysics. However, Aaronson (2015) points out several caveats, which were discussed by Dukalski (2019) for geophysical applications. First, the algorithm only offers a speed up for inverse problems in which only a single scalar piece of information about  $\mathbf{x}$  is needed, and it does so with a probability proportional to the inverse square of the condition number of  $\mathbf{A}$ . If field data measurements are part of  $\mathbf{A}$  or  $\mathbf{b}$ , then feeding them to a quantum computer can be a complex and very computationally expensive process — more so than solving the initial problem classically (Clader et al., 2013). Current hardware has so far only produced examples of very small, often custom built, examples of solving such an inverse problem (see, e.g., Barz et al., 2014; Pan et al., 2014; Zheng et al., 2017). Dukalski (2021) discusses a potential geophysical application example (solving the Gelfand-Levitan-Marchenko equation) that appears to fit most of the HHL algorithm caveats reasonably well, but only once the classical problem is reformulated to better fit the requirements imposed by quantum mechanics. Moreover, it appears that directly solving the wave equation in an arbitrarily complicated heterogeneous medium might, even in theory, be currently not possible using the HHL algorithm. More research is needed to see if this problem can be solved by other means, e.g., using some physics-informed and quantum-assisted ML (Kyriienko et al., 2021).

For these reasons, gate-model quantum computing does not appear to offer many advantages for solving geophysical problems, which is why we now turn to a lesser-known form of quantum computing.

### AQC and QA

AQC solves problems by mapping the objective function of a problem of interest onto the energy landscape of some tunable quantum system (Albash and Lidar, 2018). This way, starting in a simple system with a known global optimum, we can slowly tune the parameters of the quantum system (and have it such that its energy landscape is more complex). Performing this process slowly allows for the initial system configuration to adjust as well, such that it remains in the global optimum. This process amounts to “computation” as the final system configuration encodes the global optimum of the complex objective function of the problem we were trying to solve.

$$H(t) = f(t)H_{\text{problem}} + (1 - f(t))H_{\text{simple}}, \quad (2)$$

where  $H_{\text{problem}}$  is the Hamiltonian that represents our problem of interest and  $H_{\text{simple}}$  encodes the simple problem whose global optimum is known a priori. Moreover,  $f(t)$  is some smooth function of time  $t$ , with  $f(0) = 0$  and  $\lim_{t \rightarrow T} f(t) = 1$ . Since everything is governed by laws of quantum mechanics, then one can additionally benefit from quantum tunneling between local minima — a process mathematically similar to seismic wave tunneling as an evanescent mode through thin high-velocity layers in the subsurface. AQC could therefore be a great tool for solving problems described by multimodal objective functions in which only a global optimum is of interest — not only found quicker than other meta-heuristic approaches but also offering additional certainty that no other better solution can be found. AQC can be shown to be (in theory) equivalent to the gate-model QC, in which the translation between the two can be carried out in polynomial time and without any constraints on the physical device (Mizel et al., 2007). In practice, however, with limitations imposed by the devices, this equivalence is no longer that clear.

QA is very similar to AQC. There we have the simple system — a so-called driving or tunneling Hamiltonian, which introduces magnetic field fluctuations that have a simple and a priori known ground state. The so-called IM is particularly useful:

$$H_{\text{problem}} = H_{\text{IM}} = \sum_{i \in V} h_i s_i + \sum_{(i,j) \in E} J_{ij} s_i s_j, \quad (3)$$

where  $s_i$  represents the variables taking values  $\pm 1$ ,  $h_i$  represents the energies (biases) of individual spins (the magnetic models), and  $J_{ij}$  represents the coupling/interaction strengths. The objective function of some relevant problem is mapped onto a Hamiltonian in equation 3, by means of physically fixing the biases and the coupling strengths. The latter form a so-called graph  $G$ , which will be discussed later. The subtle difference between AQC and QA is that the terms in front of  $H_{\text{simple}}$  and  $H_{\text{problem}}$  need not add to one. Finding the minimum energy of the IM for the majority of graphs  $G$  is NP-hard, which makes it particularly suitable for solving a wide range of (NP-hard) problems. Hence, it is a great design choice for a quantum annealer. A notable example of the latter is comprised of an array of interconnected tunable superconducting flux qubits (Venegas-Andraca et al., 2018). From an engineering standpoint, when constructing such a device, one has to balance the ability to control and tune the energies of the individual qubits and the coupling between them, as well as reading out their state at the end of the calculation. This means that one has to decide on the so-called hardware graph, a quantum computer architecture that specifies which set of qubits (on the vertices of the graph) are connected (by edges of the graph) to other qubits. Construction of the next generation of quantum annealers amounts to balancing the increase in the qubit-qubit connectivity, while not making the device overall noisier and harder to control.

A good analogy of this system would be a set of compasses (in which the needle plays the role of the qubit with  $|0\rangle$  and  $|1\rangle$  corresponding to the north–south and east–west directions, respectively), with varying field strengths (which dictate their resistance to change their orientation) and varying coupling to each other (e.g., controlled by their relative distance) in the absence of any external magnetic field. By appropriately magnetizing the magnets beforehand and choosing their relative position to control their coupling, we can make sure that the energy landscape (the energy as a function of the

magnetic needle orientation of each of the compasses) is the same as some objective function that we wish to optimize. This system will settle in a local minimum and some externally applied stimulus could displace its configuration toward another local (and perhaps asymptotically global) minimum. However, this system is not capable of tunneling. The driving or tunneling Hamiltonian could be linked to an external oscillating magnetic field, which sets all the compasses in some spinning motion, allowing them to explore all possible combinations of solutions. As the external field strength is reduced, their mutual interactions start dominating and increasing the chances of finding the ground state, albeit here without tunneling.

## Solving problems using quantum annealers

Solving a (combinatorial optimization) problem with a quantum annealer in practice requires first converting its original formulation into an IM formulation (Venegas-Andraca et al., 2018). This new problem will most likely be described by a so-called problem graph. Subsequently, to solve the problem of choice, one needs to translate this intermediate problem graph into a hardware graph — a process called minor embedding. Finding an optimal minor embedding is itself an NP-hard problem, but without much sacrifice to the original problem, one can use heuristics to obtain acceptable solutions (Cai et al., 2014). This enables quantum annealers to solve a very large class of (combinatorial) optimization problems in which most of the work is dedicated outside of the QA step, namely, to finding an efficient way to carry out the “inexpensive” (problem-conversion) steps and interpret the solutions in the presence of noise intrinsic to the quantum annealer.

To illustrate the concept of minor embedding, we consider the IM Hamiltonian:

$$H = 6s_1 + 5s_2 + 3s_3 + 9s_4 + s_5 + 20s_1s_2 + 8s_2s_3 + 14s_2s_4 + 5s_3s_4 + 7s_4s_5, \quad (4)$$

with its problem graph example shown in Figure 1a. The so-called Chimera hardware graph, a particular quantum annealer architecture, is shown in Figure 1b. It can be rigorously proven to be impossible to map the problem to the hardware graph because the connectivity of nodes in the latter is not sufficient (the problem graph has a threefold loop, whereas the hardware graph has fourfold loops and no threefold ones). Minor embedding in this case duplicates the problem graph’s node five, which represents the variable  $s_5$ , and insert a so-called chain between the two copies to tie them together so that they do not behave as two individual variables (see Figure 1c). The strength of this tying together is controlled by the chain-strength parameter that is addressed later in this paper.

The IM formulation and the interacting magnetic moments analogy is a convenient way to understand the workings of the quantum annealer. However, it is sometimes preferable to formulate the optimization problem as a quadratic unconstrained binary optimization (QUBO) problem of the form

$$\min_{\mathbf{x} \in \{0,1\}^n} \mathbf{x}^T Q \mathbf{x}, \quad (5)$$

where  $Q$  is an upper triangular real matrix and  $\mathbf{x}$  and its transpose  $\mathbf{x}^T$  are the vectors of binary variables  $x_i$  with the property  $x_i^2 = x_i$ , i.e., taking values zero or one. The entries of  $Q$  can be mapped

to those of  $h_i$  and  $J_{ij}$  from equation 3 through the transformation  $x_i = (s_i + 1)/2$ .

Very few problems in geophysics are naturally discrete and formulated in terms of binary variables. To use the QUBO formulation, the underlying variables not only need to be discretized, but a set of allowed discrete values also needs to be specified a priori. Furthermore, these discrete values have to be encoded in terms of the binary variables  $x_i$ . Depending on the choice of encoding, additional constraints or penalty terms (with their corresponding Lagrange multiplier) might need to be appended to the original QUBO to obtain meaningful solutions. This may introduce a host of new complications, which not only have an impact on the maximum problem size that a quantum computer can handle as well as tuning of the QA parameters, but these also could require additional classical pre- and postprocessing.

In the “SPM as a QUBO” section, we illustrate and discuss the transformation process for a geophysical application — the problem of RRSE. This problem is particularly suited because it is of great industrial importance for land seismic exploration and near-surface characterization for civil engineering, for example, and at the same time fits very well with the requirements imposed by the quantum annealer. Before we come to the RRSE application, we would like to compare QA and SA and point out some important differences between the two approaches.

### Quantum versus simulated (thermal) annealing

SA imitates the technique of annealing in metallurgy in which the physical properties of a material are altered by heating, followed by controlled cooling. Slow cooling produces a material with average energy across the molecules lower when compared with the energy of the same material cooled quickly. Kirkpatrick et al. (1983) realize that the behavior of a thermodynamic system reaching the low-energy state at the equilibrium during the annealing process is similar to the one of an optimization algorithm searching for the global optimum of an objective function, and therefore, the mathematical description of such a process could be used to solve general optimization problems. The algorithm is an adaptation of the Metropolis-Hastings algorithm, a Monte Carlo method to generate sample states of a thermodynamic system, and works by starting with an initial solution and an initial temperature. Then, the temperature progressively decreases, and at each decreasing step the algorithm

randomly selects a solution close to the current one following a criterion based on an adaptation of the Metropolis-Hastings algorithm, in which the acceptance probability is a function of the temperature. Two nested cycles related to the random moves and to the adjustment of the steps are repeated, respectively,  $N_s$  and  $N_t$  times, for each temperature reduction  $N_r$  and for each search dimension  $B$  (Corana et al., 1987). The computational complexity of SA is therefore equal to  $O(N_s N_t N_r B)$ . Geman and Geman (1984) prove that SA converges to the global optimum when a logarithmic (very slow) annealing schedule is used. However, in practice, logarithmic annealing schedules are too slow to find meaningful results for high-dimensional problems in an acceptable amount of time.

In SA, the objective function of the problem is the same as the energy function, whereas in QA, the initial objective function needs to be translated to an IM, which then gets embedded into the hardware. SA starts with some initial guess, whereas QA starts with an eigenstate of the drive Hamiltonian, which is a superposition state of all possible solutions. QA and SA are similar in that movement between different local minima is facilitated by modulation of some kind of fluctuations — the driving Hamiltonian (in QA) and the “temperature” of the system (in SA). In addition, these approaches are probabilistic in nature, which is why the processes need to be repeated multiple times, and the result in these cases could strongly depend on the annealing time  $T$ . The difference there however is that SA and QA might still be stuck in some local minimum (though this is expected to be less likely so for QA), but more importantly, at the end of QA, the system could still be in a superposition state of ground and excited (higher energy) state, and a quantum measurement collapses the system to a configuration that could still be in an excited state.

Let us now turn to our demonstrator application — the problem of RRSE. In the next section, we will start with a brief description of the problem, followed by two different encodings. We will then present the overall classical-quantum hybrid workflow composed of classical preprocessing, QA, and classical postprocessing step, which are needed to obtain the best solution.

### SPM FOR RRSE

Subsurface seismic imaging in land environments often is difficult due to the presence of complex geologic features in the near surface that affect the propagation of seismic waves distorting their amplitude and phase (Taner et al., 1974). As a result, the acquired seismic data (traces) corresponding to the waves traveling through subsurface anomalies will show seismic events that are amplified or reduced and anticipated or delayed when compared with the ones associated with the waves not traveling through such anomalies. The relative delays of the seismic events in the traces due to this phenomenon also are known as statics.

If the anomalies are resolvable, statics can be estimated and corrected by building a subsurface compressional velocity model through techniques, such as reflection/refraction tomography or FWI (Virieux and Operto, 2009). Amplitude distortions also can be calculated and corrected with similar approaches, such as amplitude tomography (Quan and Harris, 1997). However, if the anomalies are

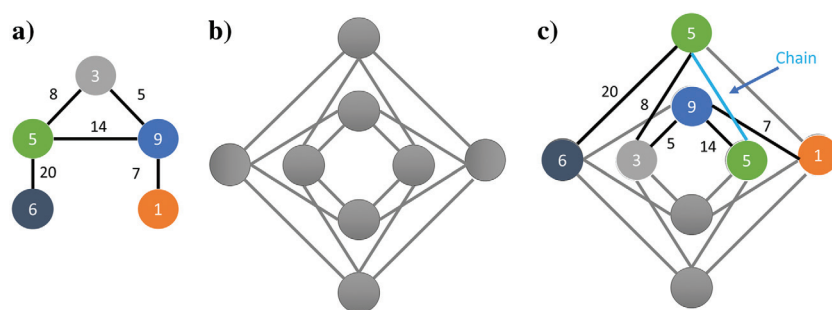


Figure 1. (a) The problem graph corresponding to the IM from equation 4. The colored circles correspond to the set of vertices  $V$ , with the numbers inside corresponding to the biases  $h_i$ , and the black lines are the edges  $E$  with the numbers next to them corresponding to the coupling strengths  $J_{ij}$ . (b) The Chimera hardware graph, in which the strength of each node and vertex is tunable. (c) The embedding of (a) onto (b) with vertex and edge strengths denoted, and the blue line denoting a chain.



not resolvable with tomographic techniques, then there will be amplitude and phase residuals in the seismic traces.

A standard way in the industry of tackling the problem of compensating for such residuals is to use surface-consistent approaches (Wiggins et al., 1976; Taner and Koehler, 1981), trying to decouple the wave propagation effects occurring at the surface (or at the shallow subsurface) to the ones occurring at the (deeper) subsurface. After removing the deeper subsurface distortions, the analysis focuses on a statistical evaluation and inversion of the residual components of the wave propagation that are characteristic of surface positions occupied by either seismic sources or receivers. Even though surface-consistent approaches have been initially developed for reflection seismology, they also can be applied to the transmitted part of the seismic wavefield, which is the one mostly traveling in the near surface and mostly sensing its complex heterogeneities (Colombo et al., 2021). In addition, surface-consistent statics correction methods can address most of the statics and therefore are routinely applied in seismic data processing; it is worth mentioning that nonsurface-consistent statics correction approaches also are possible, sometimes providing more accurate near-surface corrections (Alfaraj et al., 2021).

In this work, we are focusing on the problem of phase distortion of the seismic waves in terms of RRSE and correction. We follow the surface-consistent framework for the transmitted portion of the wavefield as detailed by Colombo et al. (2021). Starting from the convolutional model of the wave propagation (Cambois and Stoffa, 1992), the authors isolate the contribution of the subsurface from the source and receiver impulse responses that are mapped as a function of surface positions. The decomposition has been adapted to the transmitted wavefield involving sorting in common-midpoint and offset (CMP)-offset (the so-called XYO domain), in which the traces are collected in small bins. A hypercube of X-Y-offset dimensions and the corresponding XYO gathers are generated, and the surface-consistent analysis is performed for each of these bins and gathers. More details can be found in Colombo et al. (2016, 2020).

In the frequency domain, the convolutional model of the wave propagation for each XYO gather can be written as

$$\mathbf{d}_m(f) = \mathbf{w}(f)\mathbf{s}_i(f)\mathbf{r}_j(f)\mathbf{q}(f), \quad (6)$$

where  $f$  denotes the frequency;  $\mathbf{d}_m(f)$  is the seismic trace of index  $m$  in that specific gather of  $M$  traces, related to the source and receiver index couple  $(i, j)$ ;  $\mathbf{s}_i(f)$  and  $\mathbf{r}_j(f)$  are the source and receiver surface-consistent residual terms, respectively;  $\mathbf{q}(f)$  is a subsurface-related residual term; and  $\mathbf{w}(f)$  represents the XYO gather statistical response (pilot trace) that can be approximated by the stack of the gather traces.

The decomposition for the residuals is then obtained by solving

$$\frac{\mathbf{d}_m(f)}{\mathbf{w}(f)} = \mathbf{s}_i(f)\mathbf{r}_j(f)\mathbf{q}(f). \quad (7)$$

By specifying equation 7 for phase differences and assuming linear distortion in the frequency domain (Taner and Koehler, 1981), the correction is reduced to a constant time shift for each surface position

$$\Delta\tau_m = \tau_{D_m} - \tau_W = \tau_{s_i} + \tau_{r_j} + \tau_q, \quad (8)$$

where  $\Delta\tau_m$  is the time shift of the trace  $\mathbf{d}_m(t)$  with respect to the pilot trace  $\mathbf{w}(t)$ , and  $\tau_{s_i}$ ,  $\tau_{r_j}$ , and  $\tau_q$  are the time shifts added by source, receiver, and XYO bin operators, respectively. Equation 8 defines a system of equations involving all the source and receiver combinations  $(i, j)$  for which a residual is calculated, and it can be solved through a preconditioned conjugate gradient algorithm (Barrett et al., 1994).

The crosscorrelation operation, which is computing the time shifts  $\Delta\tau_m$ , is therefore solving

$$\max_{\Delta\tau_m} \int_{-T}^T \mathbf{d}_m(t)\mathbf{w}(t + \tau)dt, \quad (9)$$

where  $2T$  corresponds to the time length of the crosscorrelation window and  $\mathbf{w}(t)$  is the time-domain pilot trace introduced previously.

The previous operations to calculate  $\Delta\tau_m$  suffer from cycle skips, which is when one seismic arrival on the trace  $\mathbf{d}_m(t)$  is aligned with the wrong arrival on the pilot trace  $\mathbf{w}(t)$  because there are multiple wiggles within the crosscorrelation window. This can happen for many reasons, including limited signal bandwidth, noise in the data, and a short crosscorrelation window.

For each XYO gather, one crosscorrelation and the related maximization operations need to be repeated  $M$  times, one for each trace of the gather. The crosscorrelation operation can be efficiently computed with the fast Fourier transform and its computational complexity is therefore  $O(N \log 2N)$  for each trace and for each bin, with  $N$  denoting the number of samples per trace.

Trying to counteract the effect of the cycle skips, Ronen and Claerbout (1985) suggest reformulating the problem as an equivalent SPM of the form

$$\max_{\Delta\tau_m} \left\| \sum_{m=1}^M \mathbf{d}_m(t - \tau_m)_L \right\|_2^2, \quad (10)$$

where  $\Delta\tau_m$  is the vector of the optimal time shifts of the traces in the XYO gather. In this approach, instead of solving  $M$  maximizations of  $M$  mono-dimensional crosscorrelation functions, an  $M$ -dimensional function, corresponding to the power of the stack of the traces, has to be maximized.

Equation 10 corresponds to a global optimization problem that can be tackled with different global optimization techniques, the most popular and successful of which, in the field of SPM, include SA (Rothmann, 1985) and others inspired by natural processes, such as swarm intelligence (Beni and Wang, 1993). Unfortunately, none of these methods guarantee to find the global optimum in an affordable amount of time (Pierini et al., 2019).

Using SA for the SPM problem has been studied by Rovetta (2006) for a simple three-trace problem (see Figure 2) with the complicated multimodal objective function shown in Figure 3a. Here, variables  $\tau_1$  and  $\tau_2$  are the relative time shifts of signals  $\mathbf{s}_1$  and  $\mathbf{s}_2$  with respect to the reference signal  $\mathbf{s}_0$ , which also could be the stack of the two traces. After few iterations, the algorithm is moving from the initial solution to the optimal solution. Some evaluations of the stack power at different annealing temperatures are shown in Figure 3b–3d. At high temperatures, the algorithm works as a common uniform sampler, whereas, at low temperatures, the sampling interval is located in the region corresponding to the highest probability of finding the global maximum. SA is typically computationally expensive, but its performances can be improved when it

is combined with linear optimization techniques (i.e., steepest descent). In this case, a hybrid algorithm can be designed to use the global optimization capabilities of the SA during the first part of the process, which identifies the region around the global maximum, and subsequently the local optimizer is used in the second part of the process to speed up the convergence (Bosisio et al., 2014).

Current hardware limitations restrict the maximum dimension of the function to be optimized. In addition, for this reason, in this work we have applied QA on the RRSE problem in the specific formulation of XYO bins and gathers, in which the number of traces is relatively small. Finally, as for the SA hybrid algorithm, in which local optimizers are used in the proximity of the global optimum, for the QA we also make use of hybrid solvers, combining the best of the quantum and classical approaches, to circumvent current limitations on the number of variables to be estimated.

## SPM AS A QUBO

Finding the QUBO form of the stack maximization problem can be done in three steps:

- 1) Determining discretization and solution space. This amounts to deciding what values the shift variables  $\tau_i$  will be allowed to take.
- 2) Expressing the objective function in terms of the time-shift operators and establishing the encoding, in which such discrete shifts are expressed in terms of the binary variables  $x_i$ . We will consider two choices for such encoding.
- 3) Checking if, upon the chosen encoding, the problem is indeed a QUBO and/or if it is still an SPM objective function with physically meaningful solutions. If not, we need to undertake additional steps to make it so (if possible).

## Determining the solution space

In the first step, we denote a solution space (all the allowable values) of the shift variable  $\tau_i$  with  $S_{i,K} = S_K = \{t_1, \dots, t_K\}$ . We assume the same set of possible shifts for all traces, for example, for all of the shift variable  $\tau_i$ , which is why we drop the index  $i$  on  $S_K$  in what follows. This choice will have an impact on the problem size and the energy landscape of the quantum annealer. For example, one could consider a simple set ( $K = 3$ ), where  $t_1 = 0$ , and  $t_2 = -t_3 = 2$  ms. As a result, any relative shift larger than 2 ms will not be a part of the solution. A good choice of  $K$  or the exact values of  $t_a$  should be based on the data used. Note that adding a constant to all elements of  $S_K$  will have no impact on the solution because this operation would introduce an overall shift to a gather and hence not influence the trace alignment.

By looking at the data, one also could decide to fix one, e.g., the  $n$ th, trace and hence set  $S_{n,K} = \emptyset$ , which would not only slightly reduce the number of variables but also improve the choices of maximum and minimum shifts applied to all the other traces. This is of course not limited to the QA implementation, but to solving the SPM in general. However, the problem is that we do not know a priori which trace to fix. For sufficiently large problem sizes, currently those involving tens of traces, we have observed that fixing the “wrong” trace leads the quantum computer to very rarely return a result corresponding to a global minimum, and oftentimes not even a local one.

## Reexpressing the objective function

We can use the sampling property of the Kronecker delta function  $\delta_{ij}$  and denote the process of shifting a particular trace by a particular amount as a selection from a set of traces shifted by all possible values of  $t_a$ . In particular, the shift of  $t_{x_i}$  of the  $i$ th trace reads

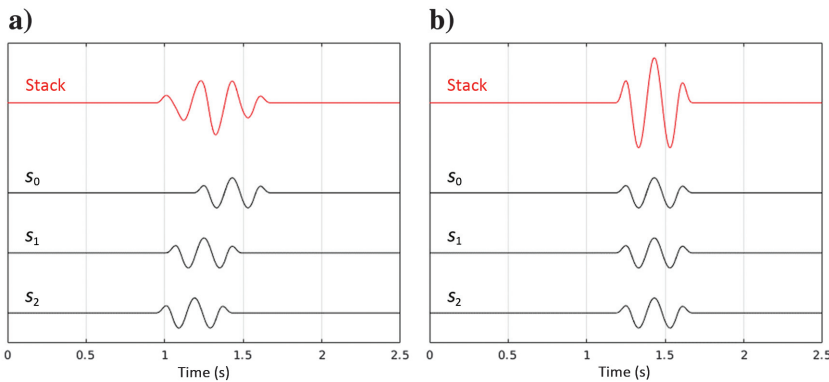


Figure 2. Seismic gather of three traces to be aligned (a) before and (b) after SPM.

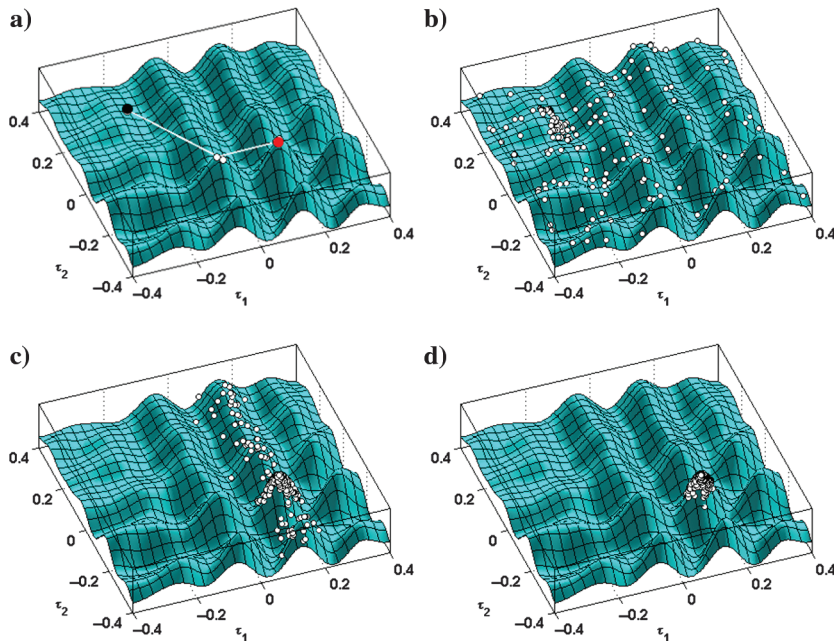


Figure 3. Two-dimensional objective function to be maximized to solve the problem of Figure 2: (a) path of the SA algorithm from the starting solution (marked with a black dot) to the final one (marked with a red dot) and (b–d) evaluations of the objective function, respectively, at temperatures of  $T_0 = 300[\text{K}]$ ,  $T = 23.33[\text{K}]$ , and  $T = 14[\text{K}]$ .

$$\mathbf{d}_i(t_{x_i}) = \sum_{a=1}^K \delta_{x_i a} \mathbf{d}_i(t_a). \quad (11)$$

This should give us a first hint of the classical preprocessing because it suggests that we will need to calculate traces with all the possible shifts applied to them. Substitution in equation 10 gives the objective function

$$\operatorname{argmax}_{\mathbf{x} \in \{1, \dots, K\}^M} \left\| \sum_{i=1}^M \sum_{a=1}^K \delta_{x_i a} \mathbf{d}_i(t_a) \right\|_2^2. \quad (12)$$

Next, we want to reexpress the preceding result in terms of products of Kronecker delta functions, which later will be expressed in terms of the binary variables  $x_i$ . Using basic linear algebra gives

$$\begin{aligned} \operatorname{argmax}_{\mathbf{x} \in \{1, \dots, K\}^M} & \sum_{i=1}^M \sum_{a=1}^K (\delta_{x_i a})^2 \langle \mathbf{d}_i(t_a), \mathbf{d}_i(t_a) \rangle \\ & + 2 \sum_{i=1}^M \sum_{j=i+1}^M \sum_{a=1}^K \sum_{b=1}^K \delta_{x_i a} \delta_{x_j b} \langle \mathbf{d}_i(t_a), \mathbf{d}_j(t_b) \rangle, \end{aligned} \quad (13)$$

where  $\langle \mathbf{d}_i(t_a), \mathbf{d}_j(t_b) \rangle$  is the inner product (i.e., the crosscorrelation evaluated at  $t = 0$ ) between the  $i$ th and the  $j$ th traces shifted by  $t_a$  and  $t_b$ , respectively. The term  $\langle \mathbf{d}_i(t_a), \mathbf{d}_i(t_a) \rangle = \|\mathbf{d}_i(t_a)\|_2^2 = \|\mathbf{d}_i\|_2^2$  is independent of  $t_a$ , and hence we will drop it because it does not affect the solution.

## Encoding

The next step is to encode the variables in the combinatorial optimization problem of equation 13 using binary variables. For this purpose, we use two different encodings: an indicator (or one-hot) encoding (IE) and the standard binary encoding (BE).

### Indicator encoding

With the IE, we introduce  $MK$  binary variables  $x_{ia}$ ,  $1 \leq i < M$ ,  $1 \leq a \leq K$ , where  $x_{ia}$  is one when the shift  $t_a$  is applied to trace  $i$  and zero otherwise. This means that we can replace  $\delta_{x_i a}$  by  $x_{ia}$  with the additional constraint  $\sum_{a=1}^K x_{ia} = 1$  because each shifted trace can be present only once (as dictated by the Kronecker delta). However, this encoding allows for more solutions with more than one  $x_{ia} = 1$ , e.g.,  $x_{i1} = 1$  and  $x_{iK} = 1$ , which means that the same trace will contribute to the stack-power twice — once shifted by  $t_1$  and another time shifted by  $t_K$ . The only way to avoid this happening is by introducing a constraint into the QUBO formulation with the aid of a penalty term. Here, we choose  $p \sum_{i=1}^M (\sum_{a=1}^K x_{ia} - 1)^2$ , where  $p$  is a Lagrange multiplier. The outer sum over  $i$  enforces the constraint over each of the  $M$  traces. This penalty term is nonzero if more than one  $x_{ia}$  is equal to one. Substituting the binary variables and adding the penalty produces the IEQUBO

$$\begin{aligned} \max_{\mathbf{x} \in \{0,1\}^{MK}} & \sum_{i=1}^M \sum_{j=i+1}^M \sum_{a=1}^K \sum_{b=1}^K x_{ia} x_{jb} \langle \mathbf{d}_i(t_a), \mathbf{d}_j(t_b) \rangle \\ & - p \sum_{i=1}^M \left( \sum_{a=1}^K x_{ia} - 1 \right)^2. \end{aligned} \quad (14)$$

In the process, we also have dropped the factor two in front of the second term in equation 13. There are two apparent disadvantages of this encoding: (1) a relatively noncompact representation of the Kronecker deltas and (2) as a result the need for the penalty term. The first disadvantage will result in requiring probably more than theoretically necessary qubits with as consequence the restriction to smaller problems fitting on the QPU, whereas the second one introduces a free parameter with some additional disadvantages, which we will discuss later.

### Binary encoding

The second strategy is to encode the variables in equation 13 using the standard BE, i.e., three is represented by variables  $x_1 = x_2 = 1$  and  $x_{k>2} = 0$  (i.e.,  $11_2 = 3_{10}$ ) and 16 is represented by  $x_5 = 1$  and all others are zero (i.e.,  $10000_2 = 16_{10}$ ). The hope is that with this “more natural” encoding choice, additional constraints become obsolete and that fewer binary variables are required to encode the problem and its solution. As a consequence, this would allow for a larger problem (with either a much greater number of shifts or a larger number of traces) to fit into the QPU and be simultaneously optimized. At first glance, one would need  $Mk$  variables, where  $k = \lceil \log_2 K \rceil$ ; however, this is not the case, as we will show next.

We introduce the BE by defining a Boolean polynomial of degree  $k$ ,  $c_a: \{0, 1\}^k \rightarrow \{0, 1\}$ ,

$$c_a(\mathbf{x}) = \sum_{A \in P(V)} (-1)^{w(\mathbf{a}) + |A|} \prod_{i \in A^c} \bar{a}_i \prod_{i \in A} x_i. \quad (15)$$

Previously, we have used  $\mathbf{a} = [a_1, a_2, \dots, a_k]$  as a vector of zeros and ones representing  $a$ , (e.g.,  $[1, 0, 1]$  describes five),  $\bar{a}_i = \text{mod}_2(a_i + 1)$  means not  $a_i$ , and  $w(\mathbf{a})$  is the Hamming weight — the  $\ell_0$  “norm” of  $\mathbf{a}$ . Furthermore,  $P(V)$  denotes the powerset of  $V = \{1, \dots, k\}$ , e.g., for  $V = \{1, 2, 3\}$ , we have  $P(V) = \{\emptyset, \{1\}, \{2\}, \{3\}, \{1, 2\}, \{1, 3\}, \{2, 3\}, \{1, 2, 3\}\}$ . For instance, for  $k = 3$ , we obtain

$$\begin{aligned} c_a(\mathbf{x}) = & \pm \bar{a}_1 \bar{a}_2 \bar{a}_3 \pm \bar{a}_1 \bar{a}_2 x_3 \pm \bar{a}_1 \bar{a}_3 x_2 \pm \bar{a}_2 \bar{a}_3 x_1 \\ & \pm \bar{a}_1 x_2 x_3 \pm \bar{a}_2 x_1 x_3 \pm \bar{a}_3 x_1 x_2 \pm x_1 x_2 x_3, \end{aligned} \quad (16)$$

where we sum, in equation 15, over all these in subsets of the powerset of  $V$  (these are the  $A$  and their complements  $A^c$ , e.g., with the first one being  $A = \emptyset$  and  $A^c = V/A = \{1, 2, 3\}$ ). Each  $\pm$  choice is independent of the other ones and comes from the  $(-1)^{w(\mathbf{a}) + |A|}$  term. Thus, to encode a shift corresponding to  $t_{a=5}$ , which has  $\mathbf{a} = [1, 0, 1]$ , we would use a polynomial of the form

$$c_5(\mathbf{x}) = x_1 x_3 - x_1 x_2 x_3. \quad (17)$$

By definition,  $c_a(\mathbf{x})$  is one if and only if the binary representation of  $a$  corresponds to that of  $\mathbf{x}$ . In this example, it is easy to check that  $c_5([1 \ 1 \ 0]) = 0$  and  $c_5([0 \ 0 \ 0]) = 0$ , but  $c_5([1 \ 0 \ 1]) = 1$ .

When we replace the Kronecker deltas in equation 13 with the encoding from equation 15, and because  $c_a(\mathbf{x}_i)$  is polynomials, we obtain a higher-order unconstrained binary optimization (HUBO) problem:

$$\max_{\mathbf{x} \in \{0,1\}^{MK}} \sum_{i=1}^M \sum_{j=i+1}^M \sum_{a=0}^{2^k-1} \sum_{b=0}^{2^k-1} c_a(\mathbf{x}_i) c_b(\mathbf{x}_j) \langle \mathbf{d}_i(t_a), \mathbf{d}_j(t_b) \rangle. \quad (18)$$



To obtain a standard BE, QUBO polynomials  $c_a(\mathbf{x}_i)$  need to be replaced with a new (much larger) set of binary (auxiliary) variables  $\tilde{\mathbf{x}}_i$ . Moreover, this has to be done in such a way that the inclusion of the auxiliary binary variables does not shift the optimal point of the original higher-order model. There is a variety of different methods to achieve this task (Venegas-Andraca et al., 2018). We wish to select a method that does not add a very large number of auxiliary variables (which unnecessarily use up space on the QPU). Methods due to Freedman and Drineas (2005) or Ishikawa (2010), handle terms like  $x_1 x_2 x_5$ . The nonmonomials in individual variables  $x_i$ , for example, terms of the form  $x_1^2 x_2 x_5$ , are handled by the pairwise substitution method (Dattani and Bryans, 2014; Tanburn et al., 2015). This replaces frequently used variable pairs in higher-order terms with the same and a new set of auxiliary variables. Moreover, a penalty term is added to ensure that the optimum point does not move. The quadratization process is carried out using the “make\_quadratic” functionality (McGeoch et al., 2020), which uses a blend of the aforementioned three methods. In the end, with this more promising encoding, we are still left with a constraint and several variables that are larger than initially expected. In the next section, we will try to determine the maximum problem size that could fit on the QPU for these two encodings.

### Maximum problem size for purely quantum optimization

The machine’s hardware graph and the minor embedding strategy determine the maximum problem size that can be fit on the QPU at once, and hence, the advantage one could expect from using a quantum computer. Minor embedding determines how many next-neighbor coupled qubits need to be “tied together” to match the problem graph, and hence, need to be excluded as an independent variable. The more connected the machine’s hardware graph, the less computationally intensive it is to embed any QUBO and the fewer qubits become “sacrificed.” The worst-case scenario is where the QUBO is “fully connected,” in that all variables are coupled to each other (no zero entries in  $\mathbf{Q}$ ), and that *appears* to be the case for SPM because the static shift of one trace relative to any other trace contributes to the overall energy.

In fact, one can rigorously show that the indicator encoded problem is fully connected, but it is less straightforward to show this result for the easier BE. We instead studied this issue experimentally. We try to determine the maximum value of  $K_{\max}(M)$  for some fixed value of  $M$  by trying to find if a minor embedding is possible. The results are shown in Figure 4a. With  $M$  traces in the CMP gather, and the number of possible shifts  $K$ , the IEQUBO has  $MK$  variables, and  $(1/2)MK(MK - 1)$  quadratic interactions — a fully connected graph. This means that in theory, when embedding the problem graph on a quantum annealer,  $MK$  will be approximately constant (as  $M$  and  $K$  vary). In practice, however, this is not necessarily the case due to fabrication imperfections that render certain couplings and individual qubits inaccessible. This is why the orange line shown in Figure 4a is not straight. For the BE, the theoretical bounds analysis is much more difficult. The binary HUBO consists of  $M \log_2 K$  variables; however, after quadratization, the exact number of variables is not immediately known. We have found empirically this number to be roughly  $(1 - (1/K))MK$ . The number of quadratic terms was roughly  $(1/2)MK^2 \log K$  for  $K \leq 2^3$ , and asymptotically ( $K \gg 2^3$ ) tends to  $(1/2)MK(MK - 1)$  — the number of quadratic terms found in IEQUBO. Thus, for a smaller number of shifts, the BE model contains fewer variables and quadratic terms than IEQUBO and is *not* a fully connected problem, perhaps offering some advantages, such as handling a much larger number of traces at once. This could be of interest when extending the hybrid workflow to include chunking of large CMP gathers and multiple attempts at realignment with different sets of allowable shifts  $S_K$ . Finally, we also see that the total number of variables  $MK \sim 2^7$  is similar to that found for fully connected problems on the 5000 qubits quantum annealer (McGeoch et al., 2020). This is of course no longer true for a BE model, where for  $K \leq 2^3$  the problem is visibly “less than fully connected.”

However, it should be stressed that the minor-embedding computation by the quantum computer software suite took only a few seconds for IEQUBO in contrast to minutes for maximum size BE-QUBOs. This could be due to two reasons. First, a deterministic algorithm for an a priori known to be fully connected graph is more efficient, rather than for one which is just short to meet this qualification (such as the BEQUBO). Second, the quadratization step could become a dominant part in the computational cost for large  $K$  problems. Thus, when trying to optimize alignments of many CMP gathers in a processing flow, perhaps a predetermined  $M$  and  $K$  quadratization algorithm could be provided by the user once, such that the structure of the BEQUBO problem graph is fixed and only the  $\langle \mathbf{d}_i(t_a), \mathbf{d}_j(t_b) \rangle$  terms need to be recomputed every time.

### Classical preprocessing

The preprocessing is relatively straightforward and requires finding the inner products  $\langle \mathbf{d}_i(t_a), \mathbf{d}_j(t_b) \rangle$  for all possible pairs of time shifts and traces. This may seem like a near-brute-force sampling approach that we are trying to avoid with QA; however, it is important to note that here we only need to calculate the pairwise overlaps, and the ones involving a larger number of traces need not be found. These inner products can be calculated efficiently as

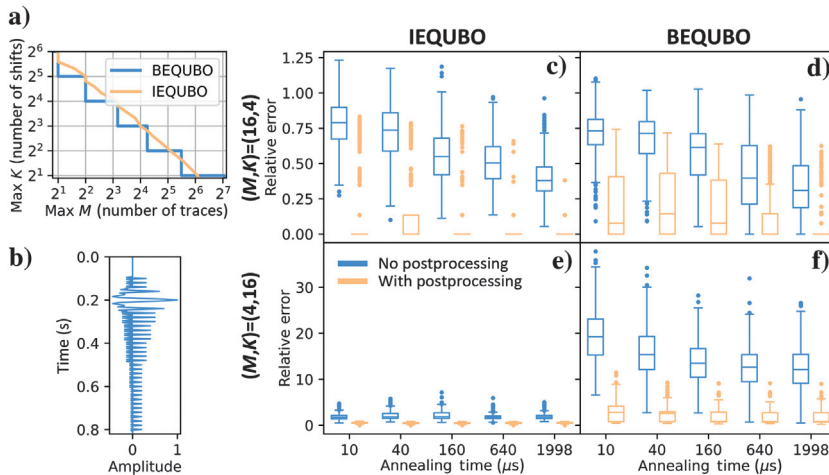


Figure 4. (a) A log-log plot showing the maximum size of a problem that can fit on a QPU at once. (b) The trace used to generate gathers for the controlled experiments. (c–f) The box-and-whisker diagrams of the QUBO-relative errors, with dots showing outliers 1.5 times outside the interquartile range.



$$\langle \mathbf{d}_i(t_a), \mathbf{d}_j(t_b) \rangle = \sum_{\omega} \mathcal{F}[\mathbf{d}_i(0)]^* \mathcal{F}[\mathbf{d}_j(0)] e^{i\omega(t_a - t_b)}, \quad (19)$$

where  $\mathcal{F}$  denotes the Fourier transform. Here, the sum is over all frequencies  $\omega$ , and  $*$  denotes the complex conjugation. The Fourier transforms need to be calculated only once per trace, and maximally only  $K(K-1)$  possible combinations of  $e^{i\omega(t_a - t_b)}$  are required. Alternatively, if the allowable shifts in the set  $S_K$  correspond to the integer multiples of the signal sampling interval, then we can find all possible inner products at once from  $\mathcal{F}^{-1}[\mathcal{F}[\mathbf{d}_i(0)]^* \mathcal{F}[\mathbf{d}_j(0)]]$  and collecting the time sample corresponding to time  $t = t_a - t_b$ .

It is important to note a major advantage of this formulation relative to the I/O limitations discussed in the “Quantum computing in a nutshell” section. Here, the classical information contained in the original problem — the crosscorrelations between pairs of shifted traces — remains classical because it is mapped onto the biases and the coupling strengths of the Ising Hamiltonian, which is a tuning parameter inside a quantum annealer. The relevant information is not only abstracted away from a classical problem in a very compact fashion — just one scalar for each trace pair and relative shift — but also at no point do we have to create a quantum register, in which the information about the seismic signal is encoded in and manipulated within quantum amplitudes  $a_k$  from equation 1.

## Quantum annealing

The quantum computer software toolkit (McGeoch et al., 2020) makes this part of the programming rather simple. The developer has to specify only three parameters: the annealing time, the chain strength, and the Lagrange multiplier, all of which have an impact on the solution (Coffrin, 2019). In what follows, we will briefly describe the meaning of the three parameters and report on our experience with selecting particular value ranges.

The annealing time can be chosen in the 1–2000 range and represents the actual physical time in microseconds spent on the annealing process on the quantum computer. Based on the controlled experiments discussed later, we have found that setting this parameter in the 100–500  $\mu$ s range is sufficient because longer annealing times were not the main driver leading to improved solutions.

The chain strength plays a major role in minor embedding because it specifies the strength of the physical coupling between neighboring (on the machine’s hardware graph) qubits, whose values have to be “tied-together.” Tying together qubits means that they should not act as independent variables. The chain-strength parameter has to be set high enough, such that the physical quantum system ensures that the qubits return the same value; however, there is no guarantee of that happening, which results in “chain-breaks” — an often-found problem with the previous generation of quantum computers (Rieffel et al., 2015). Setting this value to infinity would of course seem like an obvious choice; however, (1) this is an actual physical system, thus it cannot have an infinite amount of energy, and (2) this would prohibit finding the solution to the problem. The latter is caused by the fact that the system would be tuned toward favoring the tied-up qubits having the same value, rather than anything else. Finding the solution to the actual problem (the qubits relevant for the problem and not the machine’s hardware graph) would be of secondary importance. The last problem has to do with resolution, which we will discuss later. The solution to the problem is found using the “uniform\_torque\_compensation” functionality,

which is a proprietary algorithm that attempts to strike the right balance between optimally unbroken chains and maximum resolution.

Finally, the Lagrange multiplier  $p$  has to be determined, which is done by experimentation. The penalty term adds energy on top of the first term, which means that a large value of  $p$  that ensures that the encoding-resultant constraint is preserved will suppress the energy of the first term. This is the same problem as with setting the chain strength, except that the penalty term pertains to the problem rather than the machine’s hardware graph. Quantum annealers have a limited energy resolution that the machine Hamiltonian can handle. This has the consequence that adding a large amount of energy in the constraints (also from chains) pushes the energies of all the other solutions together. This is a problem that has two origins, quantum and classical, which dictate that the actual energy of a quantum system (and hence the encoding of the solution to the optimization problem we seek) is broadened. Quantum mechanics, specifically the Heisenberg’s uncertainty principle, dictates that the exact energy of something can only be found if it is measured for an infinitely long time (much like a DC component of any signal). Classically, we should remember that the qubits are subject to thermal fluctuations (noise) present inside and outside of the QPU, which can shift the energy levels of each qubit and hence result in the same solution corresponding to slightly different energies every time a qubit register is read out. Therefore, a good balance between resolution and the enforcement of the constraint (magnitude of  $p$ ) has to be found. A general rule could be to set  $p$  an order of magnitude larger than the largest coefficient in the first term. However, more advanced approaches might be required to better fine tune the value of  $p$ , e.g., by experimentation on problems with a known solution. In our experiments, we have noticed that when applied to the large BEQUBO problems, this could be a time-consuming process. This is because most of the computational effort is spent on HUBO quadratization. In the next section, we show that the problem of finding the right constraint or chain strength could perhaps be lessened by (relatively) inexpensive classical postprocessing.

## Classical postprocessing

With the list of challenges mentioned previously, it should not come as a surprise that QA in some cases failed to output a global optimum. As we will show later, the output often was not even a local minimum. The cause of this could be multifold: incomplete tunneling to a global optimum, suboptimal parameter setting, and quantum and environmental noise. To obtain a more robust outcome and reduce the requirements and sensitivity to careful parameter tuning, we propose to use a classical postprocessing procedure to correct the QA output. For cases in which the outcome was at most a couple of bit flips away from a global optimum, we can use a steepest descent algorithm, which greedily evaluates if flipping a given bit increases the QUBO energy. The postprocessing step can be run within the quantum computer provider software suite.

## NUMERICAL EXAMPLES

In the spirit of geophysical methods, we first verify our approach on several simple synthetic examples with known solutions and later test the approach on a realistic case. We will use the former to verify that our classical-quantum workflow works, in which we will be able to study every part from input to output separately. We then switch to testing the generic and proprietary black-box classical-

quantum hybrid solver, in which a larger problem, in a QUBO formulation, is optimized by a combination of problem segmentation, QA of a smaller subproblem, and merging, as well as classical optimization.

### Controlled experiments

The purpose of the controlled experiments is to illustrate the method described previously in detail and to test the efficacy of the workflow and potential problems encountered at each step. The approximately 5000 qubit machine that was used for our experiment can handle a maximum problem size of 180 variables, which needs to be distributed between  $M$  and  $K$ . In the following experiments, we have opted for  $MK = 64$  because, for  $MK > 100$ , we have observed that the chances of finding the global optimum without preprocessing have practically shrunk to zero, the reasons for which require further research. We have studied two extreme cases: a many-trace problem with 16 traces scrambled with four shifts  $(M, K) = (16, 4)$ , and a many-shift problem with four traces scrambled over 16 shifts  $(M, K) = (4, 16)$ . We have generated these 4- and 16-trace gathers using multiple copies of the same trace (see Figure 4b), which was chosen to generate a large number of possible local optima. The traces were shifted by an amount from  $S_K$  that was later used in encoding. In this way, the problem has an exact and known solution (global optimum). A range of annealing times was used because we expect that the longer that parameter the more likely it is to reach the global optimum. To generate output for further statistical analysis of solution distribution, we repeated the QA process 500 times. The results are shown in Figure 4c–4f, in which we calculate the relative error  $\epsilon(\mathbf{x}')$  of the quantum annealer's output  $\mathbf{x}'$  relative to the actual global optimum configuration  $\mathbf{x}_0$  given by

$$\epsilon(\mathbf{x}') = \left| \frac{\mathbf{x}_0^T \mathbf{Q} \mathbf{x}_0 - \mathbf{x}'^T \mathbf{Q} \mathbf{x}'}{\mathbf{x}_0^T \mathbf{Q} \mathbf{x}_0} \right|, \quad (20)$$

before and after postprocessing. Note that this relative error might differ from that for the original SPM in equation 10 as  $\epsilon(\mathbf{x}')$  is evaluated at the QUBO level in equation 5, which, in particular for the BE, features auxiliary variables with a constraint function.

For many traces, unlike for BEQUBO, QA IEQUBO did not yield the global optimum. Applying postprocessing changed the solution energy distribution significantly such that the optimal solution was found by IEQUBO more often (see Figure 4c and 4d).

Interesting differences between the two encodings can be observed for the many-shift problem (see Figure 4e and 4f). There, at first BEQUBO found solutions with a greater stack power compared with the IEQUBO; however, neither resulted in finding the global optimum. The distribution of solutions obtained by BEQUBO produces many of them with a far larger relative error compared with the IEQUBO. Postprocessing helped the IEQUBO to find the optimal solution, which required approximately three iterations of the greedy algorithm. This postprocessing choice, however, did not help the BEQUBO. This step was, at most, as computationally expensive as building the QUBOs and had very minor overall effect on the computational cost. We anticipate that the postprocessing choice is particularly helpful and native to the IEQUBO, whereas this is less so for the BEQUBO. This could be due to bit flips on the auxiliary (constraint) variables introduced by quadratization. The discussion on the impact of these variables on the initial problem and the encoding-spe-

cific postprocessing choice are beyond the scope of this work and require further in-depth investigation.

The differences between the many-shift or many-trace problem for the two encodings sheds some light on the optimal choices and the solution quality within a hybrid workflow. This is particularly evident for larger problems. For example, one could benefit from solving a few shifts and many traces, maximum size problem, and simultaneously loop over different choices of possible shifts in  $S_K$ . We also see that there is a relationship between the encoding choice and how successful that, or another type of postprocessing, could be in obtaining the global optimum. The observations require further investigation and are likely to be important when problem-specific classical problem segmentation is used, or when the problem is run on a more powerful quantum annealer, such as the one slated for release in the coming years.

### Larger uncontrolled experiments and hybrid solvers

We also have tested the generic classical-quantum hybrid solver that is provided by the quantum annealer (McGeoch et al., 2020) to estimate optimal residual statics for a much larger data set. We have decided to use the Society of Exploration Geophysicists Advanced Modeling (SEAM) Arid Model synthetic data set, representing a realistic desert environment characterized by complex geologic features, such as karsts, wadis, sand dunes and other unconsolidated sediments, outcropping bedrock, and highly variable topography (Oristaglio, 2015). One of the seismic data sets produced by SEAM for this model, called “classic data set 1,” has been decimated to reproduce a standard seismic survey with sources and receiver crosslines spaced 100 m and 150 m, respectively, and inline spatial sampling of 50 m for the sources and 25 m for the receivers. Such a decimated data set has been processed with a transmission surface-consistent scheme (Colombo et al., 2021): the traces have been sorted in a CMP and common-offset domain (XYO) before applying residual statics correction. In this example, we show the behavior of the quantum annealer for one XYO bin consisting of 108 traces in which we have used  $S_K$  with 16 equidistant shifts every 6 ms, giving  $16^{108} \approx 10^{126}$  possible solutions. Approximately 1700 variable QUBO was constructed in the same fashion as outlined previously and sent to the quantum annealer. For this problem, unlike the previous examples, it is now known a priori what the solution looks like.

The difference between the generic hybrid and the dedicated quantum workflow discussed previously is many fold. First, the generic hybrid solver optimizes the QUBO by means of classical algorithms as well as by segmenting the problem and optimizing parts of it on a quantum annealer. This way a larger than  $MK \approx 2^7$  problem can be processed by the hybrid solver. No control or ability to significantly influence the process is granted to the user. The optimization also does not pay attention to the physical nature of the initial problem and only focuses on the QUBO, which by then is stripped of any physical meaning known only to the user. At the end, the solver outputs only a single optimal outcome and provides the total amount of time that was spent on the quantum annealer. Whether the final outcome, or parts of it, are found using the quantum computer or just by means of classical algorithms is not communicated to the user, which makes it difficult to interpret the degree of “quantumness” of outcome of the hybrid solver with scientific rigor.

In Figure 5, we show the data used on input, the outcome of the simple crosscorrelational solver (sensitive to cycle skipping), and

the result of the hybrid solver applied to the IEQUBO. The hybrid solver output has the energy 10.7% higher than that of the simple crosscorrelational approach. Applying the hybrid solver to the BE-QUBO was unsuccessful, resulting in solutions with the stack power smaller than the input, hence, with a much worse alignment. This could be due to greater focus on satisfying the constraint (auxiliary) variables and perhaps additional difficulties when automatically partitioning the larger problem. We believe that instead of further researching the reasons for that, it might be more worthwhile to construct a purpose-built problem partitioning classical-hybrid workflow using the findings from the previous section.

## DISCUSSION

Quantum computing is entering a higher technological readiness level, and it is now important to ask what applications it will be most suitable for. In the process, it is becoming clear that perhaps the initially considered gate-model applications, such as factorization or unstructured database search, are not going to be the most relevant applications everyone is hoping for. It is more likely, however, that the future could feature a significant amount of quantum-assisted optimization. This is particularly true for the noisy intermediate scale machines of the 2020s. In all this, the adiabatic or analog quantum computing paradigm appears to be gaining popularity and QA appears to be a very interesting candidate for solving real-world applications. The main reason for that is that the use cases are relatively clear, and they require relatively straight-forward classical algorithms to support the quantum optimization step. These machines also are much more robust to noise — a topic that we were only able to cover briefly. For these reasons, understanding and developing the QUBO or IM formulation on any optimization problem in geophysics might be worthwhile exploring. Because several recipes for NP-hard problems to QUBO or IM conversion exist (Lucas, 2014), it might be very convenient to identify the computational complexity classes and types of optimization problems commonly found in geophysics. In the end, it will be the work of geophysicists to find the QUBO for their geophysical problem and incorporate it in a meaningful workflow. This process can and should be started already today, while waiting for the purely quantum engineering problems to be solved, which will enable building quantum annealers that feature larger and more interconnected QPUs. Quantum computing in geoscience could benefit from a structured framework in which software practitioners, researchers, academic educators, and professional associations work together and learn from each other to accelerate developments in this exciting field (see, e.g., Piattini et al, 2020).

Based on what we have seen for the SPM problem, there are two relevant questions that need to be answered for a successful porting of an application onto a quantum computer: (1) what are we trying to accomplish when we attempt to solve a problem on a quantum computer and (2) what are the characteristics of a suitable candidate for a QA application? Quantum computing often is promoted as something

that would solve a very specific problem with some computational complexity scaling speed up. For geophysical applications *today* that speed up is most likely to be found when trying to find strictly global optimum of some multimodal optimization problem solved using a quantum annealer. This is because the gate-model QC is still in its relative infancy, and appropriate algorithms are lacking. As far as QA is concerned, we have to consider the pre- and postprocessing, as well as the partitioning overhead and embedding as additional computational cost. Therefore, it is only the search for the global optimum in a problem with a second-best local optimum far away in the parameter space that could bring substantial advantages. In other words, we are only interested in the global optimum, and we would like the added certainty that we have found it rather than just a “better” solution. In addition, a suitable geophysical problem has to be combinatorial in nature, where continuum optimization problem tools do not yield good results (like sudden shallow anomalies induced discontinuities) or a nonconvex optimization problem, which currently is being replaced/approximated with a second-best alternative. Finally, a problem which (1) is susceptible to easy partitioning, (2) has possible encoding which results in a sparsely connected problem graph, and (3) has an intuitive and easy to implement quality control and classical postprocessing measures is more likely to be solved with quantum annealers with a limited number of qubits.

A problem frequently found in geophysics, yet not strictly combinatorial in its nature, which appears to meet these specifications is least-squares optimization with sparsity requirements on the solution. This technique often is used in seismic data interpolation (Cao et al., 2011), simultaneous source separation (deblending) (Kumar et al., 2015), and denoising (Pilikos and Faul, 2017). One of the most prominent ways to enforce sparsity is to use the least absolute shrinkage and selection operator (LASSO) objective function (Tibshirani, 1996). In this unconstrained formulation,

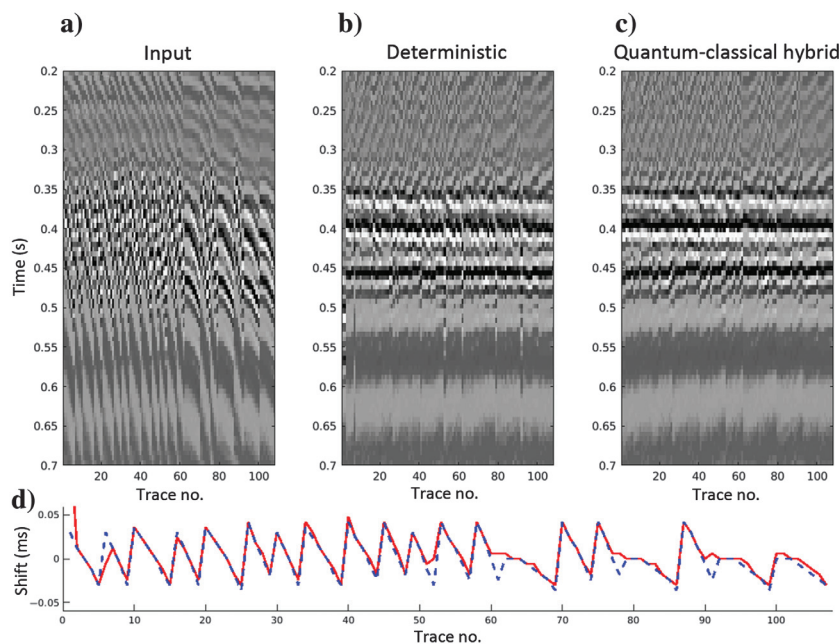


Figure 5. Results of 108 traces of the SEAM arid model data set in the XYO domain. (a) Unaligned traces used on input. (b) The result of a crosscorrelational (deterministic) solver. (c) The result of a generic classical-quantum hybrid solver. (d) Shifts applied to (a) to produce (b) in solid red and (c) in dashed blue.



the objective function consists of two terms: a data-misfit quadratic term and a sparsity-promoting regularization term — typically, the  $\ell_1$ -norm of the solution vector. The LASSO formulation is of particular interest to this discussion because it has been demonstrated in Ayanzadeh et al. (2020) that a version of it with binary variables can be converted into a QUBO problem. However, geophysical applications usually require real-valued solution vectors. A way to address this issue is an extension, described in Ide and Ohzeki (2022), that allows for approximating vectors of real numbers by a product of a matrix of predefined weights and a binary vector — a form of encoding not too dissimilar as the sampling trick used in equation 12. This, of course, comes at the cost of a reduced number of “real-valued” variables available for optimizing (similar to the set  $S_K$  in the RRSE problem). A partial remedy is to work with small patches of data, which is common practice in denoising and deblending workflows. An intriguing aspect of the quantum version of LASSO is that the  $\ell_0$ -norm is used in the regularization term. Classical computer implementations tend to use the  $\ell_1$ -norm for sparsity regularization because the  $\ell_0$  version of the problem is NP hard. The fact that the quantum LASSO solves the  $\ell_0$ -regularized objective function could lead to sparser solutions than the classical version, possibly less affected by signal noise leakage.

A problem frequently found in geophysics and *perceived* as costly is sorting a list of numbers. Other than applications during standard seismic data processing, sorting also is used as a subroutine in many other algorithms, for example, calculation of truncated ( $\alpha$ -trimmed) mean values, in which a controlled percentage of outliers are discarded in the process (Bednar and Watt, 1984). This is particularly useful as a robust estimator of the mean value in the presence of noise, which finds applications in tomography (Gersztenkorn and Scales, 1988), stack gather denoising (Watt and Bednar, 1983; Haldorsen and Farmer, 1989), seismic interpretation (Al-Dossary and Marfurt, 2007), and robust inner product calculation (Moore et al., 2016), which sometimes replaces the usual inner product in regression algorithms. A QUBO formulation for this problem has been proposed by Bauckhage and Welke (2021), and hence it would seem that this would be an interesting candidate for QA. However, it is important to consider the full picture here because sorting only becomes prohibitively costly for large data sets. The limitation stems mainly from I/O in which the trace headers are being sorted, but the entire trace needs to be accessed with it. The sorting process has efficient algorithms, with the fastest implementation scaling as  $O(n \log n)$ , and the solution is relatively easy to verify to be correct, putting this problem in the P computational complexity scaling class and hence be an unlikely candidate for QA. Moreover, it is unlikely that a QA would be able to be a stand-alone solution, especially when its QUBO is fully connected and consumes many qubits to minor embed. It is much more likely that again a classical-quantum hybrid sorting algorithm would be required, but the benefit of using a quantum computer for this task is not that obvious.

A last plausible example pertains to a field that has seen growing popularity in the recent years — ML. The training of an ML algorithm is a large optimization problem, which can suffer from many local minima and barren plateaus, hence it could benefit from QA or other quantum computing algorithms (Phillipson and Chiscop, 2021c).

Finally, regarding the SPM problem, there are several elements worth considering in the follow-up work. First, we have noticed that

the encoding choice could play a major role for QA, and recently a more machine-native, so-called domain-wall, encoding of discrete variables has been proposed by Chancellor (2019). Second, this encoding can be thought of as a running sum of the IE, and hence transitioning from one discrete variable to the next requires only a single bit flip instead of two for IE or potentially many more for BE, which is one of the reasons why it is thought to be more efficient (Chen et al., 2021). It also offers some improvement when it comes to the number of possible variables it can encode, which could be advantageous provided that the constraints that would need to be introduced are not too complex to implement. Third, one can attempt to exercise greater control over the annealing schedule and implement some of the techniques presented by Pelofske et al. (2020). Fourth, there are several classical tricks known from “classical” RRSE or other optimization algorithms, such as FWI, that can be incorporated in the hybrid workflow as pre- or postprocessing. Fifth, most likely the real power of QA for SPM would be unlocked with a hybrid solver in which a too-large-for-the-QPU problem is split into subproblems individually optimized and then merged again. SPM is definitely a problem that can be subject to easy partitioning and using a purpose built, rather than the all-purpose classical-quantum hybrid workflow used in this work (Booth et al., 2017), would be the natural next step. Finally, SPM is a step for other purposes in seismic data processing, (for example, Etgen, 2020), and may require a different hybrid workflow than the one proposed here.

## CONCLUSION

Quantum computing is an umbrella term for several emerging hardware architectures, which by controlling the quantum state of some physical system try to perform calculations. The problem is that it is not obvious what practical application and at what scale could be implemented on what device. In this work, we have shown that QA could be a great tool for solving NP-hard problems in geophysics, characterized by rapidly varying multimodal objective functions. This could be a great tool in particular, in which only the global optimum is relevant, and all other local minima (especially those far in the parameter space) should be avoided. In particular, we have shown how to cast SPM as a QUBO — a quantum-native form of the problem. We have discussed two discrete variable encodings, their advantages and disadvantages, which are likely to be present for other geophysical use cases. Most significantly, we have shown the importance of embedding the quantum computer within a classical-quantum hybrid workflow, in which not just the best quantum result, but all outputs, are used and postprocessed to identify the most plausible global optimum. This approach to analyzing the quantum computer output transcends this particular application. We have identified several extensions aimed at potentially further improving the probability of success as well as optimize larger problems on a quantum computer as we scale up to take steps toward attempting to benchmark these results against the best classical algorithms. Finally, with this work we have offered not only a tutorial into how to carry out SPM on a quantum annealer but also presented an approach to discover and consider other applications than RRSE or completely other optimization problems (QUBO) objective functions that could be run on an even larger quantum annealer in the near future.



## DATA AND MATERIALS AVAILABILITY

No data have been required for this paper.

## NOMENCLATURE

AI	=	Artificial intelligence
AQC	=	Adiabatic quantum computing
BE	=	Binary encoding
BEQUBO	=	Binary encoded quadratic unconstrained binary optimization
CPU	=	Central processing unit
GPU	=	Graphical processing unit
HHL	=	Harrow Hassidim Lloyd (algorithm)
HPC	=	High-performance computing
HUBO	=	Higher-order unconstrained binary optimization
IE	=	Indicator encoding
IEQUBO	=	Indicator encoded quadratic unconstrained binary optimization
IM	=	Ising model
LASSO	=	Least absolute shrinkage and selection operator
ML	=	Machine learning
QA	=	Quantum annealing
QML	=	Quantum machine learning
QPU	=	Quantum processing unit
QUBO	=	Quadratic unconstrained binary optimization
RRSE	=	Refraction residual statics estimation
SA	=	Simulated annealing
SEAM	=	SEG advanced modeling
SPM	=	Stack-power maximization
XYO	=	Horizontal (X, Y) and offset (O) sorted data

## REFERENCES

- Aaronson, S., 2015, Read the fine print: *Nature Physics*, **11**, 291–293, doi: [10.1038/nphys3272](https://doi.org/10.1038/nphys3272).
- Abel, S., N. Chancellor, and M. Spannowsky, 2021, Quantum computing for quantum tunneling: *Physical Review D*, **103**, 016008, doi: [10.1103/PhysRevD.103.016008](https://doi.org/10.1103/PhysRevD.103.016008).
- Albash, T., and D. A. Lidar, 2018, Adiabatic quantum computation: *Reviews of Modern Physics*, **90**, 015002, doi: [10.1103/RevModPhys.90.015002](https://doi.org/10.1103/RevModPhys.90.015002).
- Al-Dossary, S., and K. J. Marfurt, 2007, Lineament-preserving filtering: *Geophysics*, **72**, no. 1, P1–P8, doi: [10.1190/1.2387138](https://doi.org/10.1190/1.2387138).
- Alfaraj, A. M., E. Verschuur, and F. J. Herrmann, 2021, Residual statics correction without NMO — A rank-based approach: *First International Meeting for Applied Geoscience & Energy*, Expanded Abstracts, 2565–2569, doi: [10.1190/segam2021-3583455.1](https://doi.org/10.1190/segam2021-3583455.1).
- Ayanzadeh, R., M. Halem, and T. Finin, 2020, An ensemble approach for compressive sensing with quantum annealers: *IEEE International Geoscience and Remote Sensing Symposium*, 3517–3520.
- Barrett, R., M. Berry, T. F. Chan, J. Demmel, J. M. Donato, J. Dongarra, V. Eijkhout, R. Pozo, C. Romine, and H. van der Vorst, 1994, *Templates for the solution of linear systems: Building blocks for iterative methods*: SIAM.
- Barz, S., I. Kassal, M. Ringbauer, Y. O. Lipp, B. Dakic, A. Aspuru-Guzik, and P. Walther, 2014, A two-qubit photonic quantum processor and its application to solving systems of linear equations: *Scientific Reports*, **4**, 6115, doi: [10.1038/srep06115](https://doi.org/10.1038/srep06115).
- Bauchhage, C., and P. Welke, 2021, ML2R coding nuggets: Sorting as QUBO: Technical Report, MLAI, University of Bonn.
- Bauer, B., S. Bravyi, M. Motta, and G. K. L. Chan, 2020, Quantum algorithms for quantum chemistry and quantum materials science: *Chemical Reviews*, **120**, 12685–12717, doi: [10.1021/acs.chemrev.9b00829](https://doi.org/10.1021/acs.chemrev.9b00829).
- Bednar, J., and T. Watt, 1984, Alpha-trimmed means and their relationship to median filters: *IEEE Transactions on Acoustics, Speech, and Signal Processing*, **32**, 145–153, doi: [10.1109/TASSP.1984.1164279](https://doi.org/10.1109/TASSP.1984.1164279).
- Beni, G., and J. Wang, 1993, Swarm intelligence in cellular robotic systems, in P. Dario, G. Sandini, and P. Aebischer, eds., *Robots and biological systems: Towards a new bionics*: Springer, 703–712, doi: [10.1007/978-3-642-58069-7\\_38](https://doi.org/10.1007/978-3-642-58069-7_38).
- Biamonte, J., P. Wittek, N. Pancotti, P. Rebentrost, N. Wiebe, and S. Lloyd, 2017, Quantum machine learning: *Nature*, **549**, 195–202, doi: [10.1038/nature23474](https://doi.org/10.1038/nature23474).
- Booth, M., S. Reinhardt, and A. Roy, 2017, Partitioning optimization problems for hybrid classical/quantum execution: D-Wave Technical Report Series, 14-1006A-A.
- Bosisio, A. V., G. Drufuca, and D. Rovetta, 2014, The concept of distance in global optimization applied to non-linear inverse problems: *Inverse Problems in Science and Engineering*, **22**, 683–706, doi: [10.1080/17415977.2013.823410](https://doi.org/10.1080/17415977.2013.823410).
- Bruzewicz, C. D., J. Chilverini, R. McConnell, and J. M. Sage, 2019, Trapped-ion quantum computing: Progress and challenges: *Applied Physics Reviews*, **6**, 021314, doi: [10.1063/1.5088164](https://doi.org/10.1063/1.5088164).
- Cai, J., W. G. Macready, and A. Roy, 2014, A practical heuristic for finding graph minors: *arXiv preprint*, arXiv:1406.2741.
- Cambois, G., and P. L. Stoffa, 1992, Surface-consistent deconvolution in the log/Fourier domain: *Geophysics*, **57**, 823–840, doi: [10.1190/1.1443296](https://doi.org/10.1190/1.1443296).
- Cao, J., Y. Wang, J. Zhao, and C. Yang, 2011, A review on restoration of seismic wavefields based on regularization and compressive sensing: *Inverse Problems in Science and Engineering*, **19**, 679–704, doi: [10.1080/17415977.2011.576342](https://doi.org/10.1080/17415977.2011.576342).
- Cao, Y., J. Romero, J. P. Olson, M. Degroote, P. D. Johnson, M. Kieferová, I. D. Kivlichan, T. Menke, B. Peropadre, N. P. D. Sawaya, S. Sim, L. Veis, and A. Aspuru-Guzik, 2019, Quantum chemistry in the age of quantum computing: *Chemical Reviews*, **119**, 10856–10915, doi: [10.1021/acs.chemrev.8b00803](https://doi.org/10.1021/acs.chemrev.8b00803).
- Chancellor, N., 2019, Domain wall encoding of discrete variables for quantum annealing and QAOA: *Quantum Science and Technology*, **4**, 045004, doi: [10.1088/2058-9565/ab33c2](https://doi.org/10.1088/2058-9565/ab33c2).
- Chen, J., T. Stollenwerk, and N. Chancellor, 2021, Performance of domain-wall encoding for quantum annealing: *IEEE Transactions on Quantum Engineering*, **2**, 1–14.
- Clader, B. D., B. C. Jacobs, and C. R. Sprouse, 2013, Preconditioned quantum linear system algorithm: *Physical Review Letters*, **110**, 250504, doi: [10.1103/PhysRevLett.110.250504](https://doi.org/10.1103/PhysRevLett.110.250504).
- Coffrin, C. J., 2019, Challenges with chains: Testing the limits of a D-wave quantum annealer for discrete optimization (no. LA-UR-19-21739): Los Alamos National Lab. (LANL), Los Alamos, NM (United States).
- Colombo, D., F. Miorelli, E. Sandoval-Curiel, and K. Erickson, 2016, Fully automated near-surface analysis by surface-consistent refraction method: *Geophysics*, **81**, no. 4, U39–U49, doi: [10.1190/geo2016-0018.1](https://doi.org/10.1190/geo2016-0018.1).
- Colombo, D., D. Rovetta, E. Sandoval-Curiel, and A. Kontakis, 2020, Transmission-based near surface deconvolution: *Geophysics*, **85**, no. 2, V169–V181, doi: [10.1190/geo2019-0443.1](https://doi.org/10.1190/geo2019-0443.1).
- Colombo, D., E. Sandoval-Curiel, D. Rovetta, and A. Kontakis, 2021, Near-surface full-waveform inversion in a transmission surface-consistent scheme: *Geophysics*, **86**, no. 2, U15–U29, doi: [10.1190/geo2020-0474.1](https://doi.org/10.1190/geo2020-0474.1).
- Corana, A., M. Marchesi, C. Martini, and S. Ridella, 1987, Minimizing multimodal functions of continuous variables with the simulated annealing algorithm: *ACM Transactions on Mathematical Software*, **13**, 262–280, doi: [10.1145/29380.29864](https://doi.org/10.1145/29380.29864).
- Córcolas, A. D., A. Kandala, A. Javadi-Abhari, D. T. McClure, A. W. Cross, K. Temme, P. D. Nation, M. Steffen, and J. M. Gambetta, 2019, Challenges and opportunities of near-term quantum computing systems: *arXiv preprint*, arXiv:1910.02894.
- Crosson, E., and D. Lidar, 2021, Prospects for quantum enhancement with diabatic quantum annealing: *Nature Reviews Physics*, **3**, 466–489, doi: [10.1038/s42254-021-00313-6](https://doi.org/10.1038/s42254-021-00313-6).
- Dattani, N. S., and N. Bryans, 2014, Quantum factorization of 56153 with only 4 qubits: *arXiv preprint*, arXiv:1411.6758.
- Delilbasic, A., G. Cavallaro, M. Willsch, F. Melgani, M. Riedel, and K. Michielsen, 2021, Quantum support vector machine algorithms for remote sensing data classification: *IEEE International Geoscience and Remote Sensing Symposium*, 2608–2611.
- Dragoset, B., E. Verschuur, I. Moore, and R. Bisley, 2010, A perspective on 3D surface-related multiple elimination: *Geophysics*, **75**, no. 5, 75A245–75A261, doi: [10.1190/1.3475413](https://doi.org/10.1190/1.3475413).
- Dukalski, M., 2019, Potential applications of quantum computing in upstream: Fourth EAGE Workshop on High Performance Computing for Upstream, doi: [10.3997/2214-4609.201903288](https://doi.org/10.3997/2214-4609.201903288).
- Dukalski, M., 2021, Toward an application of quantum computing in geophysics: Fifth EAGE Workshop on High Performance Computing for Upstream, doi: [10.3997/2214-4609.2021612005](https://doi.org/10.3997/2214-4609.2021612005).
- Etgen, J. T., 2020, Why FWI isn't the only answer in seismic velocity model building: 82nd Annual International Conference and Exhibition, EAGE, Extended Abstracts, doi: [10.3997/2214-4609.202012151](https://doi.org/10.3997/2214-4609.202012151).
- Feynman, R. P., 1985, Quantum mechanical computers: *Optics News*, **11**, 11–20, doi: [10.1364/ON.11.2.000011](https://doi.org/10.1364/ON.11.2.000011).
- Flamini, F., N. Spagnolo, and F. Sciariolo, 2018, Photonic quantum information processing: A review: *Reports on Progress in Physics*, **82**, 016001, doi: [10.1088/1361-6633/aa5b2](https://doi.org/10.1088/1361-6633/aa5b2).

- Freedman, D., and P. Drineas, 2005, Energy minimization via graph cuts: Settling what is possible: IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), IEEE, 939–946.
- Geman, S., and D. Geman, 1984, Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images: IEEE Transactions on Pattern Analysis and Machine Intelligence, **PAMI-6**, 721–741, doi: [10.1109/TPAMI.1984.4767596](#).
- Gersztenkorn, A., and J. A. Scales, 1988, Smoothing seismic tomograms with alpha-trimmed means: Geophysical Journal International, **92**, 67–72, doi: [10.1111/j.1365-246X.1988.tb01121.x](#).
- Gonzalez-Zalba, M. F., S. de Franceschi, E. Charbon, T. Meunier, M. Vinet, and A. S. Dzura, 2021, Scaling silicon-based quantum computing using CMOS technology: Nature Electronics, **4**, 872–884, doi: [10.1038/s41928-021-00681-y](#).
- Greer, S., and D. O'Malley, 2020, An approach to seismic inversion with quantum annealing: 90th Annual International Meeting, SEG, Expanded Abstracts, 2845–2849, doi: [10.1190/segam2020-3424413.1](#).
- Grover, L. K., 1996, A fast quantum mechanical algorithm for database search: Proceedings of the 28th Annual ACM Symposium on Theory of Computing, 212–219.
- Grumbling, E. and M. Horowitz, eds., 2018, Quantum computing: Progress and prospects: National Academies of Sciences, Engineering, and Medicine, The National Academies Press.
- Haldorsen, J. B., and P. A. Farmer, 1989, Suppression of high-energy noise using an alternative stacking procedure: Geophysics, **54**, 181–190, doi: [10.1190/1.1442642](#).
- Harrow, A. W., A. Hassidim, and S. Lloyd, 2009, Quantum algorithm for linear systems of equations: Physical Review Letters, **103**, 150502, doi: [10.1103/PhysRevLett.103.150502](#).
- Henriet, L., L. Beguin, A. Signoles, T. Lahaye, A. Browaeys, G. O. Raymond, and C. Jurczak, 2020, Quantum computing with neutral atoms: Quantum, **4**, 327, doi: [10.22331/q-2020-09-21-327](#).
- Humble, T. S., H. Thapliyal, E. Munoz-Coreas, F. A. Mohiyaddin, and R. S. Bennink, 2019, Quantum computing circuits and devices: IEEE Design & Test, **36**, 69–94, doi: [10.1109/MDAT.2019.2907130](#).
- Ide, N., and M. Ohzeki, 2022, Sparse signal reconstruction with QUBO formulation in l0-regularized linear regression: arXiv preprint, arXiv:2202.00452.
- Ishikawa, H., 2010, Transformation of general binary MRF minimization to the first-order case: IEEE Transactions on Pattern Analysis and Machine Intelligence, **33**, 1234–1249, doi: [10.1109/TPAMI.2010.91](#).
- Kadowaki, T., and H. Nishimori, 1998, Quantum annealing in the transverse Ising model: Physical Review E, **58**, 5355, doi: [10.1103/PhysRevE.58.5355](#).
- Kirkpatrick, S., C. D. Gelatt, and M. P. Vecchi, 1983, Optimization by simulated annealing: Science, **220**, 671–680, doi: [10.1126/science.220.4598.671](#).
- Koretsky, S., P. Gokhale, J. M. Baker, J. Visklai, H. Zheng, N. Gurung, R. Burg, E. A. Paaso, A. Khodaei, R. Eskandarpour, and F. T. Chong, 2021, Adapting quantum approximation optimization algorithm (QAOA) for unit commitment: IEEE International Conference on Quantum Computing and Engineering, 181–187.
- Kumar, R., H. Wason, and F. J. Herrmann, 2015, Source separation for simultaneous towed-streamer marine acquisition — A compressed sensing approach: Geophysics, **80**, no. 6, WD73–WD88, doi: [10.1190/geo2015-0108.1](#).
- Kyriienko, O., A. E. Paine, and V. E. Elfving, 2021, Solving nonlinear differential equations with differentiable quantum circuits: Physical Review A, **103**, 052416, doi: [10.1103/PhysRevA.103.052416](#).
- Laumann, C. R., R. Moessner, A. Scardicchio, and S. L. Sondhi, 2015, Quantum annealing: The fastest route to quantum computation? The European Physical Journal Special Topics, **224**, 75–88, doi: [10.1140/epjst/e2015-02344-2](#).
- Leymann, F., and J. Barzen, 2020, The bitter truth about gate-based quantum algorithms in the NISQ era: Quantum Science and Technology, **5**, 044007, doi: [10.1088/2058-9565/abae7d](#).
- Lin, T. T., and F. J. Herrmann, 2014, Multilevel acceleration strategy for the robust estimation of primaries by sparse inversion: 76th Annual International Conference and Exhibition, EAGE, Extended Abstracts, doi: [10.3997/2214-4609.20140672](#).
- Liu, X., J. Li, X. Chen, K. Guo, C. Li, L. Zhou, and J. Cheng, 2018, Stochastic inversion of facies and reservoir properties based on multi-point geostatistics: Journal of Geophysics and Engineering, **15**, 2455–2468, doi: [10.1088/1742-2140/aac694](#).
- Lucas, A., 2014, Ising formulations of many NP problems: Frontiers in Physics, **2**, 5, doi: [10.3389/fphy.2014.00005](#).
- Mato, K., R. Mengoni, D. Ottaviani, and G. Palermo, 2021, Quantum molecular unfolding: arXiv preprint, arXiv:2107.13607.
- McGeoch, C., 2014, Adiabatic quantum computation and quantum annealing: Theory and practice: Springer, Synthesis Lectures on Quantum Computing 5, 1–93.
- McGeoch, C., P. Farre, and W. Bernoudy, 2020, D-wave hybrid solver service+ advantage: Technology update: Technical Report, 14-1048A-A.
- Mermin, N. D., 2007, Quantum computer science: An introduction: Cambridge University Press.
- Mizel, A., D. A. Lidar, and M. Mitchell, 2007, Simple proof of equivalence between adiabatic quantum computation and the circuit model: Physical Review Letters, **99**, 070502, doi: [10.1103/PhysRevLett.99.070502](#).
- Möller, M., and C. Vuk, 2017, On the impact of quantum computing technology on future developments in high-performance scientific computing: Ethics and Information Technology, **19**, 253–269, doi: [10.1007/s10676-017-9438-0](#).
- Montanaro, A., 2016, Quantum algorithms: An overview: NPJ Quantum Information, **2**, 1–8, doi: [10.1038/npjqi.2015.23](#).
- Moore, I., R. Fletcher, C. Beasley, and C. Castellanos, 2016, Data studies of simultaneous source separation using robust linear algebra: 86th Annual International Meeting, SEG, Expanded Abstracts, 4623–4627, doi: [10.1190/segam2016-13779503.1](#).
- Neukart, F., G. Compostella, C. Seidel, D. von Dollen, S. Yarkoni, and B. Parney, 2017, Traffic flow optimization using a quantum annealer: Frontiers in ICT, **4**, 29, doi: [10.3389/fict.2017.00029](#).
- Nielsen, M. A., and I. L. Chuang, 2010, Quantum computation and quantum information: Cambridge University Press.
- O'Malley, D., 2018, An approach to quantum-computational hydrologic inverse analysis: Scientific Reports, **8**, 6919, doi: [10.1038/s41598-018-25206-0](#).
- Oristaglio, M., 2015, SEAM update: The arid model — Seismic exploration in desert terrains: The Leading Edge, **34**, 466–468, doi: [10.1190/le34040466.1](#).
- Orus, R., S. Mugel, and E. Lizaso, 2019, Quantum computing for finance: Overview and prospects: Reviews in Physics, **4**, 100028, doi: [10.1016/j.revip.2019.100028](#).
- Outeiral, C., M. Strahm, J. Shi, G. M. Morris, S. C. Benjamin, and C. M. Deane, 2021, The prospects of quantum computing in computational molecular biology: Wiley Interdisciplinary Reviews: Computational Molecular Science, **11**, e1481, doi: [10.1002/wcms.1481](#).
- Pan, J., Y. Cao, X. Yao, Z. Li, C. Ju, H. Chen, X. Peng, S. Kais, and J. Du, 2014, Experimental realization of quantum algorithm for solving linear systems of equations: Physical Review A, **89**, 022313, doi: [10.1103/PhysRevA.89.022313](#).
- Pelofske, E., G. Hahn, and H. N. Djidjev, 2020, Advanced anneal paths for improved quantum annealing: IEEE International Conference on Quantum Computing and Engineering, 256–266.
- Phillipson, F., T. Bontekoe, and I. Chiscop, 2021b, Energy storage scheduling: A QUBO formulation for quantum computing: International Conference on Innovations for Community Services, Springer, 251–261.
- Phillipson, F., and I. Chiscop, 2021c, Multimodal container planning: A QUBO formulation and implementation on a quantum annealer: International Conference on Computational Science, Springer, 30–44.
- Phillipson, F., R. S. Wezeman, and I. Chiscop, 2021a, Indoor-outdoor detection in mobile networks using quantum machine learning approaches: Computers, **10**, 71, doi: [10.3390/computers10060071](#).
- Piattini, M., G. Petersen, R. Pérez-Castillo, J. L. Hevia, M. A. Serrano, G. Hernández, I. G. R. de Guzmán, C. A. Paradela, M. Polo, E. Murina, L. Jiménez, J. C. Marquino, R. Gallego, J. Tura, F. Phillipson, J. Murillo, A. Niño, and M. Rodríguez, 2020, The Talavera Manifesto for quantum software engineering and programming: Proceedings of the 1st International Workshop on the Quantum Software Engineering & Programming, 1–5.
- Pierini, S., M. Aleardi, and A. Sajeva, 2019, Comparisons of recent global optimization algorithms: Tests on analytic objective functions and residual statics corrections: 81st Annual International Conference and Exhibition, EAGE, Extended Abstracts, doi: [10.3997/2214-4609.201901383](#).
- Pilikos, G., and A. C. Faul, 2017, Bayesian feature learning for seismic compressive sensing and denoising: Geophysics, **82**, no. 6, O91–O104, doi: [10.1190/geo2016-0373.1](#).
- Preskill, J., 2018, Quantum computing in the NISQ era and beyond: Quantum, **2**, 79, doi: [10.22331/q-2018-08-06-79](#).
- Quan, Y., and J. M. Harris, 1997, Seismic attenuation tomography using the frequency shift method: Geophysics, **62**, 895–905, doi: [10.1190/1.1444197](#).
- Rieffel, E. G., and W. H. Polak, 2011, Quantum computing: A gentle introduction: MIT Press.
- Rieffel, E. G., D. Venturelli, B. O'Gorman, M. B. Do, E. M. Prystay, and V. N. Smelyanskiy, 2015, A case study in programming a quantum annealer for hard operational planning problems: Quantum Information Processing, **14**, 1–36, doi: [10.1007/s11128-014-0892-x](#).
- Ronen, J., and J. F. Claerbout, 1985, Surface-consistent residual statics estimation by stack-power maximization: Geophysics, **50**, 2759–2767, doi: [10.1190/1.1441896](#).
- Rothmann, D. H., 1985, Large near-surface anomalies, seismic reflection data and simulated annealing: Ph.D. dissertation, SEP.
- Rovetta, D., 2006, Inverse problem analysis: Ph.D. dissertation, Politecnico Di Milano.
- Ruane, J., A. McAfee, and W. D. Oliver, 2022, Quantum computing for business leaders: Harvard Business Review.

- Schrijver, A., 2003, Combinatorial optimization: Polyhedra and efficiency: Springer 24, 2.
- Shor, P. W., 1994, Algorithms for quantum computation: Discrete logarithms and factoring: Proceedings of the 35th Annual Symposium on Foundations of Computer Science, 124–134.
- Souza, A. M., E. O. Martins, I. Roditi, N. Sá, R. S. Sarthour, and I. S. Oliveira, 2022, An application of quantum annealing computing to seismic inversion: *Frontiers in Physics*, **9**, 748285, doi: [10.3389/fphy.2021.748285](https://doi.org/10.3389/fphy.2021.748285).
- Tanburn, R., E. Okada, and N. Dattani, 2015, Reducing multi-qubit interactions in adiabatic quantum computation without adding auxiliary qubits. Part I: The “deduc-reduc” method and its application to quantum factorization of numbers: arXiv preprint, arXiv:1508.04816.
- Taner, M. T., and F. Koehler, 1981, Surface-consistent corrections: *Geophysics*, **46**, 17–22, doi: [10.1190/1.1441133](https://doi.org/10.1190/1.1441133).
- Taner, M. T., F. Koehler, and K. A. Alhilali, 1974, Estimation and correction of near-surface time anomalies: *Geophysics*, **39**, 441–463, doi: [10.1190/1.1440441](https://doi.org/10.1190/1.1440441).
- Tibshirani, R., 1996, Regression shrinkage and selection via the lasso: *Journal of the Royal Statistical Society: Series B (Methodological)*, **58**, 267–288, doi: [10.1111/j.2517-6161.1996.tb02080.x](https://doi.org/10.1111/j.2517-6161.1996.tb02080.x).
- van der Linde, S., 2021, Quantum annealing for seismic imaging: Exploring quantum annealing possibilities for residual statics estimation using the D-wave advantage system and hybrid solver: M.S. thesis, Technical University of Delft.
- van der Linde, S., M. Dukalski, M. Möller, N. Neumann, F. Phillipson, and D. Rovetta, 2022a, Residual statics estimation with quantum annealing: 84th Annual International Conference and Exhibition, EAGE, Extended Abstracts, doi: [10.3997/2214-4609.202210180](https://doi.org/10.3997/2214-4609.202210180).
- van der Linde, S., M. Dukalski, M. Möller, N. Neumann, F. Phillipson, and D. Rovetta, 2022b, Using hybrid classical and quantum computing workflow in geophysical inverse problems: The case of quantum annealing for residual statics estimation: Sixth EAGE Workshop on High Performance Computing for Upstream, doi: [10.3997/2214-4609.2022615002](https://doi.org/10.3997/2214-4609.2022615002).
- Venegas-Andraca, S. E., W. Cruz-Santos, C. McGeoch, and M. Lanzagorta, 2018, A cross-disciplinary introduction to quantum annealing-based algorithms: *Contemporary Physics*, **59**, 174–197, doi: [10.1080/00107514.2018.1450720](https://doi.org/10.1080/00107514.2018.1450720).
- Virieux, J., and S. Operto, 2009, An overview of full-waveform inversion in exploration geophysics: *Geophysics*, **74**, no. 6, WCC1–WCC26, doi: [10.1190/1.3238367](https://doi.org/10.1190/1.3238367).
- Watt, T., and J. B. Bednar, 1983, Role of the alpha-trimmed mean in combining and analyzing seismic common-depth-point gathers: 53rd Annual International Meeting, SEG, Expanded Abstracts, 276–277, doi: [10.1190/1.1893840](https://doi.org/10.1190/1.1893840).
- Wiggins, R. A., K. L. Lerner, and R. D. Wisecup, 1976, Residual static analysis as a general linear inverse problem: *Geophysics*, **41**, 922–938, doi: [10.1190/1.1440672](https://doi.org/10.1190/1.1440672).
- Woerner, S., and D. J. Egger, 2019, Quantum risk analysis: *NPJ Quantum Information*, **5**, 15, doi: [10.1038/s41534-019-0130-6](https://doi.org/10.1038/s41534-019-0130-6).
- Yang, Y., B. Engquist, J. Sun, and B. F. Hamfeldt, 2018, Application of optimal transport and the quadratic Wasserstein metric to full-waveform inversion: *Geophysics*, **83**, no. 1, R43–R62, doi: [10.1190/geo2016-0663.1](https://doi.org/10.1190/geo2016-0663.1).
- Zheng, Y., C. Song, M. C. Chen, B. Xia, W. Liu, Q. Guo, L. Zhang, D. Xu, H. Deng, K. Huang, Y. Wu, Z. Yan, D. Zheng, L. Lu, J. W. Pan, H. Wang, C. Y. Lu, and X. Zhu, 2017, Solving systems of linear equations with a superconducting quantum processor: *Physical Review Letters*, **118**, 210504, doi: [10.1103/PhysRevLett.118.210504](https://doi.org/10.1103/PhysRevLett.118.210504).

Biographies and photographs of the authors are not available.