Thesis for the degree of MSc in Applied Mathematics in the specialisation of Computational Science and Engineering

Quantum Annealing for Seismic Imaging

Exploring Quantum Annealing Possibilities For Residual Statics Estimation using the D-Wave Advantage System and Hybrid Solver

Master Thesis

To obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on November 11, 2021 at 12:00.

by

S.G. van der Linde

Thesis advisor
Dr. M. Möller

Company supervisors
Dr. M. Dukalski
N.M.P. Neumann MSc
Prof. dr. F. Phillipson
Dr. D. Rovetta

Thesis exam committee
Dr. M. Möller
Prof. dr. D. Gijswijt
Prof. dr. F. Phillipson
Dr. M. Dukalski

Author Details
Student number: 4299779

An electronic version of this dissertation is available at `http://repository.tudelft.nl/`.

Faculty of Electrical Engineering, Mathematics & Computer Science (EEMCS) - Delft University of Technology

Abstract

Recent developments in quantum annealing have shown promising results in logistics, life sciences, machine learning and more. However, in the field of geophysical sciences the applications have been limited. A quantum annealing application was developed for residual statics estimation. Residual statics estimation is a highly non-linear problem in geophysical subsurface imaging. The problem is run on a quantum annealer as well as on a hybrid solver. Quantum annealers solve binary quadratic models, which are a specific type of NP-Hard problem. The hybrid solver uses quantum annealers together with classical computers to solve optimisation problems. Two binary quadratic models were developed for the purpose of residual statics estimation, one based on the one-hot encoding and the other on standard binary encoding. These two models were theoretically analysed and subsequently implemented and tested. Tests with synthetic data were done using the quantum annealer. The solutions found with quantum annealing were often only a few bit-flips away from the global optimum. Therefore, the results were further improved by post-processing of the data with the steepest descent method. It was found that the one-hot encoding with steepest descent performed superior to the other methods. As a final test the hybrid solver was used to solve a business-sized and realistically difficult problem. The hybrid solver outperformed a current industry standard cross-correlation solver.

Contents

1 Introduction

Quantum computing is a fast developing field with increasing relevance for different disciplines. The idea was first coined in 1980 by Paul Benioff, when he described a quantum machine model of the Turing machine [1]. The proposed machine would use key concepts from quantum physics to aid in the computation and make tasks feasible that could be computed on a classical computer. Quantum computers have a wide range of implementations, of which the gate based quantum computers [2] and quantum annealers [3] have gained the most attention. Quantum annealers use properties from quantum physics to find approximate solutions of NP-Hard optimisation problems [4]. NP-Hard problems frequently occur in real life applications. Therefore, it is not surprising that quantum annealers have already been shown to have promising results in a wide variety of fields such as logistics [5–7], life sciences [8, 9] and machine learning [10, 11]. Despite these promising applications, only a few proof-of-principle studies have been done in the field of geophysical imaging of the subsurface [12–14]

Subsurface imaging is a process which produces an image that captures the different layers of the Earth and has a wide variety of applications. Applications include geothermal energy explorations, CO_2 storage, oil and gas exploration, localising mineral deposits and aquifers, and determining foundations for large buildings in hard to build environments such as deserts or marsh. For all these applications, high resolution images are a key factor in reducing operational risk and cost. The process of making high resolution subsurface images consists of many steps, most of which are well understood. However, in the presence of large near surface anomalies, the step of residual statics estimation remains challenging.

Residual statics estimation corrects for errors caused by near-surface anomalies. Frequently used solvers, like the cross-correlation solver, try to solve the residual statics estimation problem by modelling the problem as a linear inversion problem. As the inversion problem is highly non-linear, these methods fail in the presence of large near-surface anomalies [15]. A different approach is the more robust stack power optimisation method introduced by Ronen and Claerbout [16]. However, the objective function of the stack power optimisation method is highly multimodal. Therefore, the true global maximum is usually not found in reasonable time, even with state of the art global solvers [17]. Residual statics estimation by stack power maximisation can be formulated as a combinatorial optimisation problem. This, taken together with the multimodal nature of the objective function, makes quantum annealing a promising candidate to solve this computationally complex problem.

The quantum computing company D-Wave currently has the most advanced quantum annealers [10]. Recently D-Wave released the Advantage system, a quantum annealer which was used for purposes of solving the problem presented in this thesis. Furthermore, before the end of 2024 the Advantage 2 system is expected to be launched [18], which promises solutions to even larger problems. Many of the pioneering applications that are developed now, will therefore be able to solve more complex problems in the future. As the current quantum annealers were not able to solve an industry sized problem, another approach needed to be taken For this purpose the D-Wave Hybrid solver was used. The Hybrid Solver combines quantum annealing with classical computing to allow *quantum aided computations* for problems with up to 20,000 or 1,000,000 vari-

ables, depending on the structure of the problem [19].

In this research, the application of quantum annealers for residual statics estimations was studied. To use the quantum annealers and Hybrid solver, the residual statics estimation problem had to be converted to a binary quadratic model, which is a specific type of NP-Hard problem. To this end, two different binary quadratic models were developed. These two models were both theoretically analysed and tested with synthetic data on the quantum annealers and the Hybrid solver. Finally, the steepest descent method was used as a post-processing step to improve some of the quantum annealer solutions.

This thesis is structured as follows: First, we give an introduction into subsurface imaging and the residual statics estimation problem in Chapter 2, followed by a brief discussion of some classical approaches and their shortcomings in Chapter 3. Thereafter, we give an introduction of the quantum annealing paradigm in Chapter 4. Next, we derive two binary quadratic models in Chapter 5, which are subsequently theoretically analysed in Chapter 6. In Chapter 7, we present the results of both quantum annealing directly on the Advantage system as well as the Hybrid solver. Finally, we conclude the thesis with a conclusion and outlook.

2 Introduction to Seismic Data Acquisition and Processing

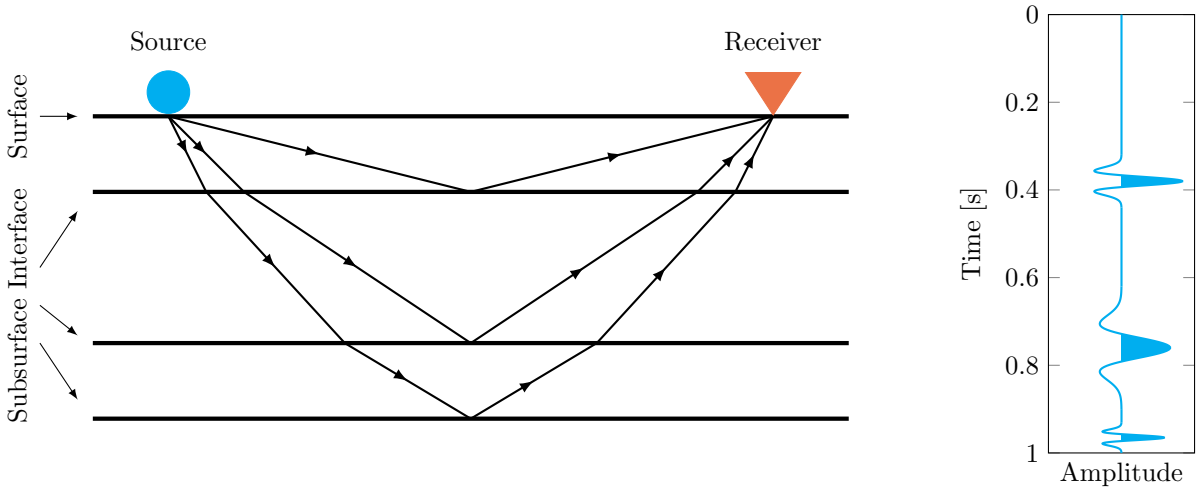
Beneath the surface, the Earth consists out of many layers. Each layer has its own properties, like thickness, strength, fluid content and more. Knowledge of these layers allows for detailed planning of subsurface work, decreasing operational costs and risks. Subsurface imaging is a process where knowledge of these layers is acquired by sending waves through the subsurface and processing the resulting signals. However, the most shallow layers, the *near-surface* layers, are often very irregular. These near-surface anomalies cause timing delays in the recording of waves used to produce a subsurface image. Consequently, the timing delays cause misalignment of recordings, which either misposition the structure or lower the resolution of the final image. Important timing delays are *statics*, which are constant shifts of the whole signal. By field measurement and other methods these statics can be (partially) corrected. However, not all statics are corrected by these methods, the statics that remain after corrections of other methods are called *residual statics*. Finding a correction for these residual statics is the focus of this thesis.

To discuss residual statics and their estimation methods, first seismic data acquisition will be discussed when no near-surface anomalies are present. Next, the effect of the near-surface will be discussed together with strategies to counteract the problems that arise because of near-surface anomalies. Thereafter, a mathematical model describing the residual statics will be presented. More specifically, the residual statics estimation problem is formulated as an optimisation problem.

2.1 Seismic Data Acquisition and Pre-processing

A seismic image starts with a *seismic experiment*, where waves are sent into the subsurface and are subsequently recorded. In Figure 2.1a the layout of such a seismic experiment is shown. In a seismic experiment, *sources* and *receivers* are placed on the surface of the Earth. The sources send vibrations through the Earth and waves start propagating through the subsurface. When the waves reach a point where two subsurface layers meet, called a *subsurface interface*, the waves are partially reflected and partially transmitted according to Snell's law. The reflected waves are collected by the receivers. Figure 2.1a shows the paths of the dominant waves and Figure 2.1b shows an example recording of these waves, which is called a *trace*. The vertical axis of the trace is the time axis, which increases when going downwards. This reversed vertical axis was chosen, because it gives an immediate visual aid for examining the subsurface. Note that for each subsurface interface in Figure 2.1a, there is a wavelet in Figure 2.1b.

The depth of the Earth is in the distance domain, whereas the recorded data, the traces, are in the time domain. Therefore the traces must be transformed from the time domain to the distance domain. To achieve this, the velocity is estimated at different times, such that the conversion from the time domain to the distance domain can be performed. Just like the structure of the subsurface, the velocity is also unknown and a velocity model must therefore be made. The building of velocity models is a well-established but



(a) Simplified cross section of the Earth.

(b) Trace

Figure 2.1: (a) Shows a seismic experiment. A source excites the Earth and waves start to propagate through the subsurface. Upon a subsurface interface, the waves either reflect or are transmitted and go deeper. The waves are collected by the receiver. (b) An example trace of the seismic experiment of (a).

complicated procedure. It is outside the scope of this thesis to discuss velocity models in detail, but the interested reader is referred to [20]. It is important to point out that the velocity models have low resolution. The low resolution of the velocity models is problematic for modelling the most shallow layers, which are called the *near-surface layers*. The near-surface layers behave differently from the other subsurface layers, because of their exposure to the atmosphere. In these layers, the velocity changes very rapidly. These rapid velocity changes are caused by the the fact that there are substantial velocity differences for each of the layers. Moreover, also in a single layer there can be large differences. These differences can be caused through near-surface anomalies such as caves and underground rivers. In this thesis we wish to estimate these effects of the near-surface. The modelled waves in a seismic experiment with the near-surface taken into account is shown in Figure 2.2.

The more detailed effects of the near-surface will be discussed using the simplified drawing shown in Figure 2.2. As can be seen in Figure 2.2, the dominant waves are modelled to propagate vertically through the near-surface. The wave propagation is modelled vertically, due to the effects of refraction caused by the low velocity at which the waves propagate through the near-surface [16]. Below the near-surface, the waves are modelled to propagate the same way as they did when no near-surface was present. As demonstrated in Figure 2.2, the effect of the near-surface is twofold. The near-surface affects the waves when they propagate downwards, directly beneath the source, and when they propagate upwards again, directly beneath the receivers. The effect the near-surface has on the traces is modelled to be a shift in time over the whole trace. Because the effect is the constant over the whole time domain of the traces, this effect is called a *static*. Some of these statics can be estimated by measurements of the surface. Such statics are called field statics. After removing the field statics, some effects of the near-surface anomalies still remain. These effects are called the *residual statics*, which we wish to estimate in this thesis. At first glance, the effects of residual statics might seem like small errors. However, in the presence of large near surface anomalies, the errors caused by the residual statics can be substantial.

For the purpose of residual statics estimation, common midpoint gathers will be introduced. In general, a seismic experiment will have many sources and receivers. This causes a lot of redundancy in the data,

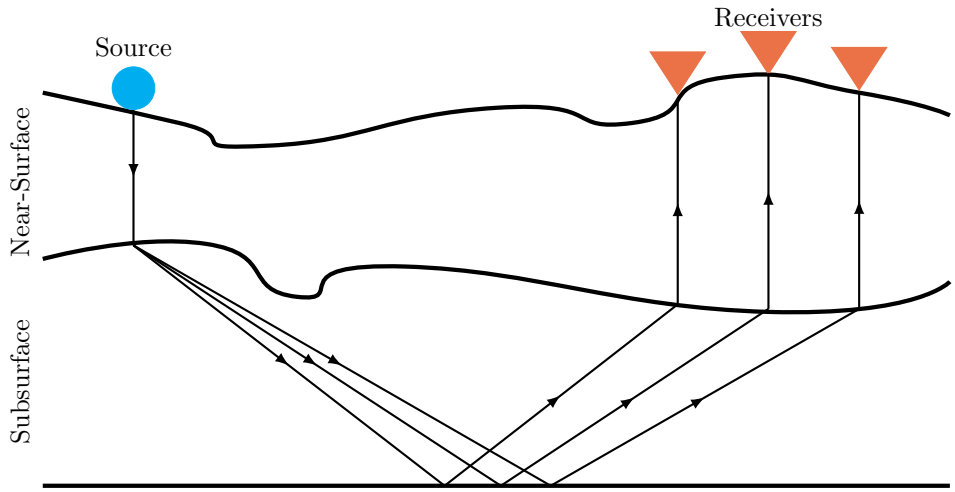
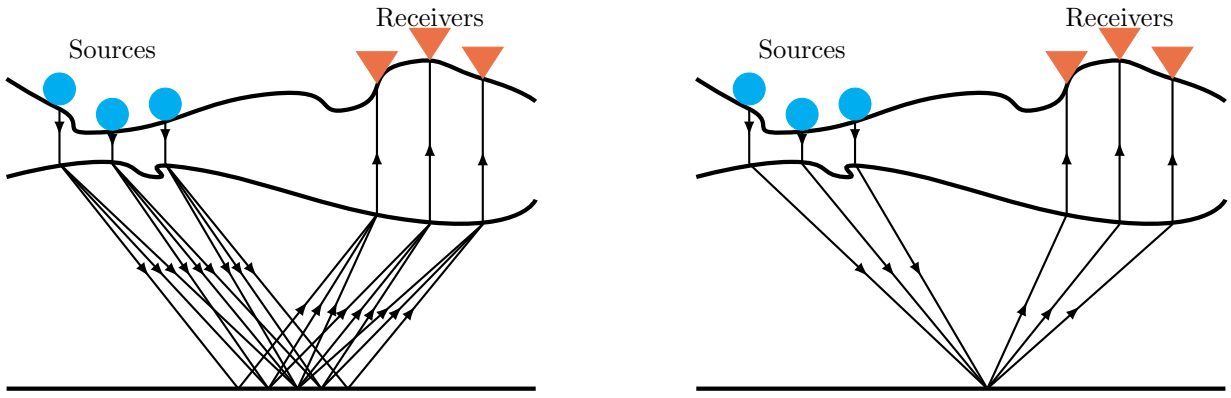


Figure 2.2: Seismic experiment with 1 source and 3 receivers. In this simplified cross section of the Earth, the effects of an exaggerated near-surface are drawn, which causes the dominant waves to propagate vertically.

which can be used to improve the quality of the final subsurface image. This redundancy becomes evident in the *common midpoint gather*. Figure 2.3 illustrates the idea behind such a common midpoint gather.



(a) All transmission paths of the seismic experiment

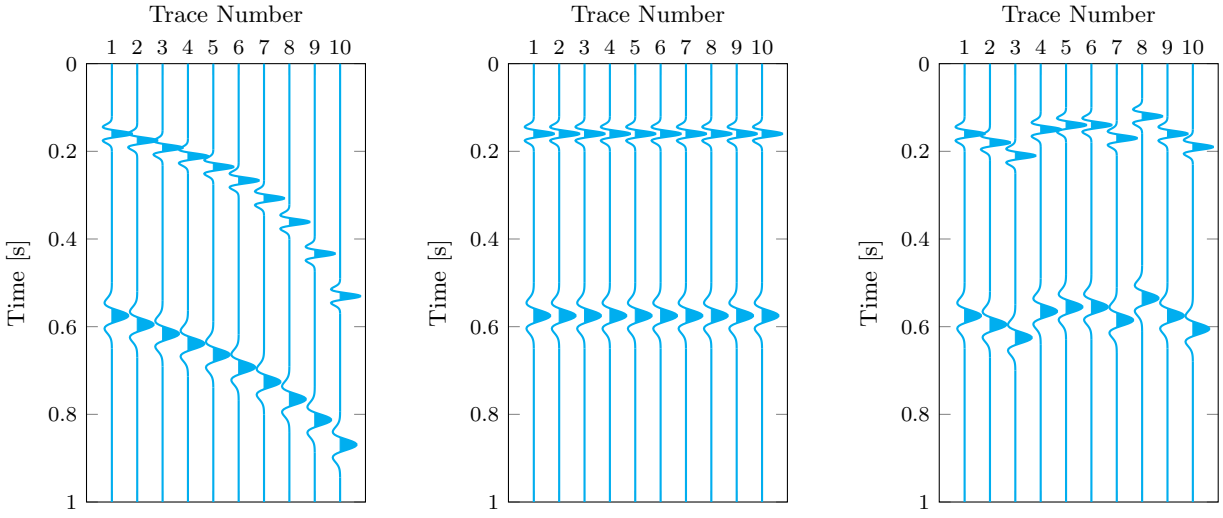
(b) Common midpoint gather

Figure 2.3: Both figures show seismic experiments with 3 sources and 3 receivers. In (a) the transmission paths of all traces are shown. (b) shows the subset of the traces in (a) which have a common midpoint.

Figure 2.3a shows a seismic experiment with 3 sources and 3 receivers. Each source sends vibrations through the Earth, waves propagate through the subsurface, reflect and are subsequently recorded by each receiver. Figure 2.3b shows three traces of this seismic experiment which have the same midpoint. That is, the source and receiver of these traces have the same horizontal midpoint. As depicted in Figure 2.3b, these three traces also have the same reflection point beneath the ground. This is true under the assumption that the Earth is horizontally layered, but it is also a good approximation for layers at a slight angle [15, 21, 22]. This idea of the same reflection point will be the main arguments for the method used in this thesis. Because each signal in the common midpoint gather has the same reflection point, this reflection point should be visible in each trace in the common midpoint gather. Hence, every trace in the common midpoint gather should be identical, except for the effects of the statics and the different travelling times in the sub-surface due to the different

paths. The different travelling times in the subsurface can be accounted for by normal moveout correction [23].

The general idea of normal moveout correction is shown in Figure 2.4. Figure 2.4 shows 10 example traces from a central midpoint gather in the simplified case that no near surface anomalies are present. Because of the different paths the waves take, the distance between the wavelets is different for each trace. Normal moveout correction is performed to correct for these different paths. An example result is shown in Figure 2.4b. All traces are perfectly aligned and the relative distance between wavelets is constant. This is what we would expect to see, because in the central midpoint gather, the traces reflect from the same point.



(a) No normal moveout correction

(b) With normal moveout correction

(c) With normal moveout correction and near-surface anomalies

Figure 2.4

When large near-surface anomalies are present, this behaviour changes. Examples of the effects of near-surface anomalies are shown in Figure 2.4c. The relative distances within each trace remain constant, but the traces are shifted relative to each other. Note that the distances between the wavelets remain constant between all traces. These relative shifts are the residual statics. Therefore, if the shifts can be determined such that the traces are aligned as in Figure 2.4b, we have an estimation of the residual statics. This is the basis of the residual statics estimation method used in this thesis.

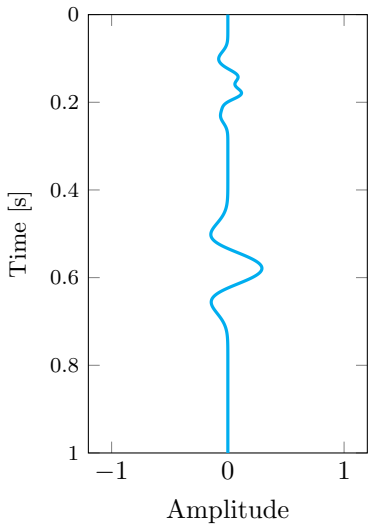
2.2 Mathematical Model for Residual Statics Estimation

To perform residual statics estimation, it will be necessary to define a measure for the quality of the residual statics estimations. Therefore, we present a mathematical model to describe the traces and the quality of the estimated statics. Let us assume we have a stack of M traces, which have been normal moveout corrected. Let $\mathbf{d}_i \in \mathbb{R}^N$ be a vector representing the amplitudes of the traces $i = 1, \ldots, M$ (the data), each containing N datapoints. Let $\tau_i \in \mathbb{R}$ be a possible shift, then $\mathbf{d}_i(\tau_i)$ denotes trace i, which is shifted by a static of τ_i. A common midpoint *stack* gives important insight of the quality of the statics. A midpoint stack is the sum

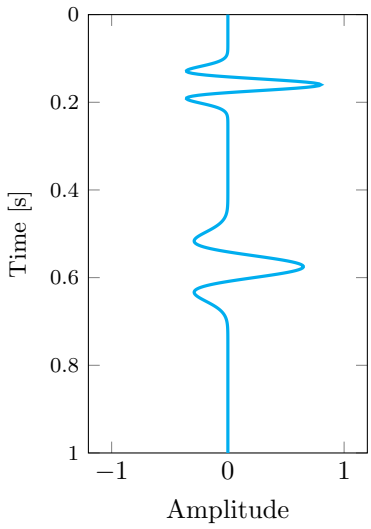
over all the traces, thus the stack is

$$\sum_{i=1}^{M} \mathbf{d}_i(\tau_i). \tag{2.1}$$

Figure 2.5 shows example stacks of the traces shown in Figure 2.4. Figure 2.5a shows the stack when near-surface anomalies are present, but no statics are applied. Figure 2.5b shows the same traces, but corrected with statics such that the traces were aligned as in Figure 2.4b. Note how the amplitudes of the wavelets in Figure 2.5a are much smaller and the wavelets are more smeared out as compared to the wavelets in Figure 2.5b. These higher amplitudes are caused by the high amount of constructive interference that happens when the traces are perfectly aligned. Hence, constructive interference is what we want to maximise.



(a) Stacked traces of Figure 2.4c.



(b) Stack of traces of Figure 2.4c after statics correction.

Figure 2.5: Common midpoint stacks of the traces shown in Figure 2.4c before (a) and after (b) statics correction.

Traditionally, maximising the constructive interference was done by creating a reference trace and calculating the cross-correlation with the reference trace for each trace. However, unless a perfect reference trace is created (which is as difficult as solving the problem), the true solution is often not found. A more robust method, stack power maximisation, was first proposed by Ronen and Claerbout [16]. They noticed that the Euclidean norm emphasises large amplitudes and thus promotes large constructive interference. Hence, they proposed to maximise the Euclidean norm of the stack, which they called the *stack power*. To simplify the computations, the Euclidean norm is squared. Thus, the residual statics can be estimated by solving the following maximisation problem:

$$\max_{\boldsymbol{\tau} \in S \subset \mathbb{R}^n} \left\| \sum_{i=1}^{M} \mathbf{d}_i(\tau_i) \right\|_2^2. \tag{2.2}$$

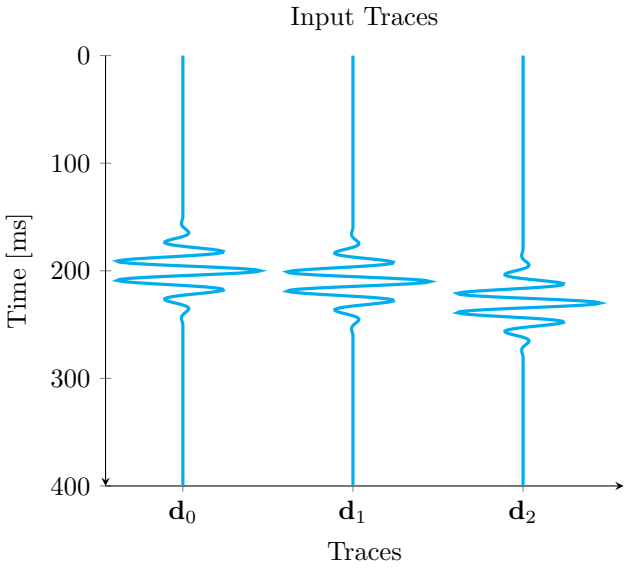
Note that there is no reference trace present and all the data is collected in this one objective function. In this representation the whole gather can be shifted by a constant to produce the same optimal value. Therefore, there are an infinite amount of statics that solve the problem. This can be solved by keeping the static of one trace fixed, or by reducing the search space.

2.3 Difficulties of Stack Power Maximisation

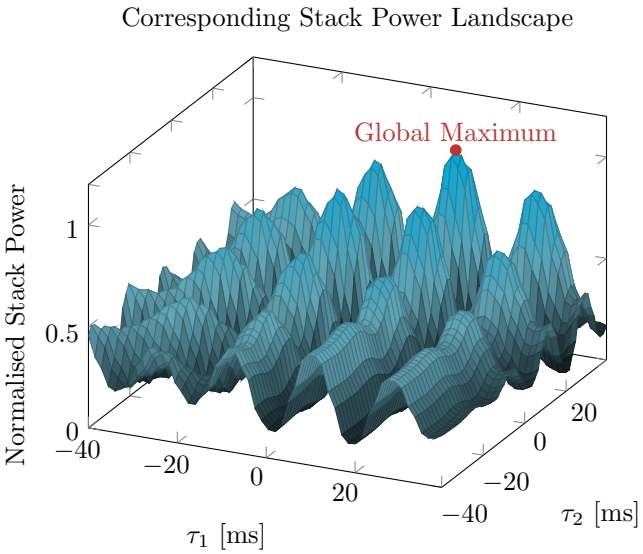
At first glance, stack power maximisation might seem like a simple problem. However, maximisation of stack power is a highly multimodal problem. To illustrate this, let us consider a small example of stack power maximisation. Figure 2.6a shows a fixed trace $\mathbf{d}_0$ together with two traces $\mathbf{d}_1$ and $\mathbf{d}_2$, each containing 401 datapoints. The statics of $\mathbf{d}_1$ and $\mathbf{d}_2$ relative to $\mathbf{d}_0$ are 10 and 30 respectively, these are the statics we wish to estimate by maximising the stack power. Let τ_1 be a shift of $\mathbf{d}_1$ and τ_2 be a shift of $\mathbf{d}_2$. Maximising the stack power for this problem is the same as

$$\max_{\tau_1,\tau_2} \|\mathbf{d}_0 + \mathbf{d}_1(\tau_1) + \mathbf{d}_2(\tau_2)\|_2^2. \tag{2.3}$$

The objective function of Equation (2.3) is visualised in Figure 2.6b. Note that even though the global maximum is obtained at $\tau_1 = 10$ and $\tau_2 = 30$, there are many local maxima near it with almost equal stack power. It is a challenging task to maximise the stack power, because of the steepness of the local maxima and the highly multimodal nature of the stack power objective function. Any local solver will almost certainly get trapped in a local maximum, whereas global optimisers run the risk of stopping prematurely when encountering a steep maximum.



(a) Reference trace $\mathbf{d}_0$ and 2 traces $\mathbf{d}_1$ and $\mathbf{d}_2$ with statics of 10 and 30 respectively. Each trace contains 401 datapoints.



(b) Corresponding stack power landscape of the traces of (a) for varying values of τ_1 and τ_2.

Figure 2.6: Example of 3 traces and their corresponding stack power landscape.

3 | Classical Solver

Residual statics estimation is a difficult aspect of subsurface imaging. Nonetheless, residual statics are crucial for making high resolution subsurface images in the presence of large near-surface anomalies. Therefore, multiple approaches have been tried to solve this problem. The most relevant classical approaches will be discussed in this chapter, together with some of the drawbacks of their implementation. First, a deterministic solver will be discussed. This solver does not work with stack power maximisation, but maximises the overlap with a reference trace. Although this gives less accurate results compared to stack power maximisation, its low computational complexity and ease of implementation have made it an industry standard. Next, simulated annealing will be discussed, which was first used in [15] to optimise the stack power objective function. Lastly, two state of the art global solvers will be discussed in the context of stack power maximisation: the Firefly Algorithm and Whale Swarm Optimisation.

3.1 Deterministic Cross-Correlation Solver

The deterministic cross-correlation solver estimates the residual statics by calculating the cross-correlation between each trace and a chosen reference trace and subsequently maximising it. The cross-correlation between two traces $\mathbf{d}_1$ and $\mathbf{d}_2$, denoted $\mathbf{d}_1 \star \mathbf{d}_2$, is a function describing the overlap of two traces for different shifts. More precisely, the cross-correlation between two traces is a function $(\mathbf{d}_1 \star \mathbf{d}_2) : \mathbb{R} \to \mathbb{R}$ defined by

$$(\mathbf{d}_1 \star \mathbf{d}_2)(\tau) = \langle \mathbf{d}_1, \mathbf{d}_2(\tau) \rangle. \tag{3.1}$$

When there is a high amount of overlap for a certain shift, the resulting cross-correlation is a positive number and relatively high. On the other hand, when there is only little overlap for a certain shift, the resulting cross-correlation is small or even negative. The idea is to make a reference trace, which is often the average of the stack, and maximise the cross-correlation between each trace and the reference trace. Hence, for every trace the following maximisation problem is solved:

$$\max_{\tau \in \mathbb{R}} \quad (\mathbf{d}_r \star \mathbf{d}_i)(\tau). \tag{3.2}$$

For discrete input, like the traces used in this thesis, the cross-correlation can be efficiently computed with the Fast Fourier Transform. Therefore, computing the cross-correlation has a computational complexity of $\mathcal{O}(N \log_2 N)$ for each trace.

For each iteration, the solver exactly maximises the cross-correlation trace by trace with the use of the Fast Fourier Transform. Next the reference trace is updated accordingly and a new iteration starts. The iterations are repeated until some stopping criterion is met, which is usually a maximum number of iterations. The basic outline of the algorithm is shown in Algorithm 1.

In practice, the Cross-Correlation solver is often used with only one iteration, as it is an extraordinarily fast and easy-to-implement algorithm. Therefore, it is still an industry standard today. However, as it solves

Algorithm 1: Deterministic Cross-Correlation Solver

input : Normal-moveout corrected traces $\mathbf{d}_1, \ldots, \mathbf{d}_M$
output: Estimation of the residual statics $\tau_1, \ldots, \tau_M$

Set reference trace $\mathbf{d}_r \leftarrow \frac{1}{M} \sum_{i=1}^{M} \mathbf{d}_i$;
Set initial statics $\tau_1, \ldots, \tau_M \leftarrow 0$;
repeat
  for $m \leftarrow 1$ **to** M **do**
    $\tau_i \leftarrow \arg\max(\mathbf{d}_r \star \mathbf{d}_i)$;
  $\mathbf{d}_r \leftarrow \sum_{i=1}^{M} \mathbf{d}_i(\tau_i)$;
until *stopping criterion*;

Equation (3.2) trace by trace and not the whole problem of Equation (2.2), the optimum found often differs from the true global optimum, which can have substantial consequences for the final subsurface image.

3.2 Simulated Annealing

Simulated annealing is named after a process in metallurgy called annealing. Annealing is a process where a material is cooled down slowly to change its structural properties. Slowly cooling the material results in a material where the average energy across the molecules in the material is lower as compared to the energy of the same material cooled quickly [24]. The key observation Kirkpatrick, et al. [24] made, was that the search for lower energy in the annealing process is similar to searching for lower output of the objective function of an minimisation problem. Therefore, the mathematical description of the process could be used to solve general optimisation problems.

The simulated annealing algorithm is outlined in its most basic form in Algorithm 2. It works by starting with an initial solution x_0 and an initial temperature T_0. Then, the current optimal solution x^* and temperature T are set to x_0 and T_0 respectively. Next, a process starts in which the following steps are repeated iteratively. In the first step, a neighbouring solution x from a set of neighbours of the current optimal solution $N(x^*)$ is generated. In the second step, the difference in objective functions of both solutions is calculated $E = f(x) - f(x^*)$. If $E \leq 0$, the new solution is better or equal, and it is accepted. If $E > 0$, the new solution is worse and is accepted with a probability $p = \exp\left(\frac{-E}{T}\right)$, depending on the current temperature. At the end of each iteration, the temperature is lowered according to a cooling schedule $T(k, T_0)$ and the new temperature and current optimal solution are used for the next iteration.

Although Algorithm 2 takes only the initial guess x_0 and initial temperature T_0 as input, the user must make many choices in the implementations. This includes a function to generate neighbours, a cooling schedule, a stopping criterion and a repetition counter for each temperature. All these choices influence the performance of the algorithm significantly [27]. However, there are no criteria for 'good' choices which is considered a downside of the algorithm [26].

The computational complexity of simulated annealing is relatively low. As can be seen from Algorithm 2, the computational complexity is $\mathcal{O}(kmMN)$, where k is the number of outer cycles and m the number of inner cycles per outer cycle. Furthermore, Geman and Geman [28] proved that simulated annealing converges to the global optimum when a logarithmic (very slow) annealing schedule is used. However, in practice, logarithmic annealing schedules cool too slow to find any meaningful results of larger system in an acceptable amount of time.

Algorithm 2: Pseudo-code for simulated annealing. Adapted from [25] and [26]

input : Initial state $x_0 \in X$
Initial temperature $T_0 > 0$

output: Approximate global optimal value x^*.

Set optimal value $x^* \leftarrow x_0$;
Set temperature $T \leftarrow T_0$;
repeat
  for $m \leftarrow 0$ **to** M_k **do**
    Generate a neighbour $x \in N(x^*)$;
    Calculate $E = f(x) - f(x^*)$;
    if $E \leq 0$ **then**
      $x^* \leftarrow x$;
    else if $E > 0$ **then**
      $x^* \leftarrow x$ with probability $p = \exp\left(\frac{-E}{T}\right)$;
  $k \leftarrow k + 1$;
  Cool $T \leftarrow T(k, T_0)$;
until *stopping criterion*;

3.3 State of the Art Solvers

In a recent study [17], new developments in global optimisers were put to the test to solve the residual statics estimation by maximising stack power. Of these global optimisers, the nature inspired meta-heuristic Firefly Algorithm and Whale Swarm Optimisation techniques seem to be the most promising. They find the global maximum more consistently as compared to other techniques and find the highest stack power on average.

Both of these algorithms are based on the so-called 'particle swarm technique' and resemble some aspects of the behaviour of a specific animal. In the case of the Firefly Algorithm the behaviour is based on the mating patterns of fireflies, whereas for the Whale Swarm Optimisation the rules are based on echo-locating behaviour of Sperm Whales. Stripped down to their essence, the algorithms can be explained in a few simple steps. First, some initial guesses are made, where each initial guess represents an animal. Then the animals move in each iteration based on a set of bioinspired rules. These rules use both the objective function and the distance to the other animals. Finally, after the set convergence criteria are met, the best found solution is returned.

The computational complexity of both the nature inspired global solver is again relatively low. Let n denote the number of animals in the swarm and let k denote the number of iterations (movements), then the computational complexity of the nature inspired solver is $\mathcal{O}(kMNn \log_2 n)$ [29, 30]. For both solvers there is no proof of convergence to the global optimum for arbitrary objective functions. Thus, while nature inspired solvers are fast and find relatively good local optima quickly, they have no way to guarantee that the global optimum will be found.

4 | Quantum Annealing

In the 80s of the last century Richard Feynman proposed a new type of computing, better equipped to simulate the complexities of nature [31]. He challenged the computer scientists of his time to develop a computer based on the principles of quantum mechanics, crucially providing them with the idea of the so-called CNOT gate. The first physical quantum computer came into existence when Y. Yamamoto and K. Igeta realised the first physical implementation of a gate-based Quantum Computer using Feynman's CNOT gate [32]. Now, almost forty years later, multiple different types of quantum computing principles have withstood the test of time and gained significant popularity: the gate based quantum computers and the annealing based quantum computers. In this chapter both types of quantum computing will be explored, but focus will be laid on the latter as all simulations for this thesis have been conducted with a quantum annealer.

4.1 Qubits

At the heart of all quantum computing methods lies the qubit. The analogy to bits as used in classical computers are often made due to their apparent similarities. Classical bits can be manipulated to perform computations by changing their state from 0 to 1 or vice versa; they can exclusively be 1 or 0 referring to the presence or absence of a current. Qubits on the other hand are a linear combination of two quantum states, often denoted as $|0\rangle$ and $|1\rangle$, which might refer to the quantum physical spin up or spin down. This means that although the state can only be measured to be in one of two states, the qubits are in a so-called *superposition* before they are measured. This superposition is a linear combination of the form

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle, \tag{4.1}$$

where $\alpha, \beta \in \mathbb{C}$ and $|\alpha|^2 + |\beta|^2 = 1$.

The probability that the outcome of a measurement on a qubit will be $|0\rangle$ or $|1\rangle$ is $|\alpha|^2$ or $|\beta|^2$ respectively. Therefore, contrary to the deterministic classical computer, quantum computers behave probabilistically. Although this gives rise to more complexities, this probabilistic nature can also be exploited to speed up specific types of computations. Another property of qubits that distinguish them from the classical bits is a property called *entanglement*. This means that if two qubits are entangled, the measurement of one them influences the state of the other.

These two properties, superposition and entanglement, shed some light on the differences between classical and quantum computing. It also sheds some light on how it is possible that quantum computing can speed up certain tasks significantly [4]. However, it must be noted that quantum computing is not a replacement for classical computing. Instead, the two different types of computing are good at solving different problems. As the field quantum computing applications is still a relatively young and unexplored field, much research remains to be done to find out which specific types of problems benefit most from this new computational approach.

4.2 Gate Based Quantum Computing

In the field of gate based quantum computing, unitary transformations are performed on qubits that are prepared in a state that is known beforehand. A unitary transformation means that the inner product of a qubit is preserved when performing the operation. These unitary transformations are called gates. The gates are used to design circuits in such a way that the resulting qubits at the end of the circuit refer to a meaningful answer at the end of the computation. However, due to the probabilistic nature of such a device, some algorithms require multiple runs need to deduce what is the most likely answer to the question posed.

Gate-based quantum computing has continued developing up until this day and age. Some promising algorithms have been proposed for this type of quantum computer already. Nonetheless, all of the experiments described in this thesis were done with another type of quantum computing; quantum annealing, which will be described in more detail in the next section. The interested reader is referred to [2], [33] or [34] for more information about gate-based quantum computing.

4.3 Quantum Annealing

The quantum computing technique used for this research study is quantum annealing. Quantum annealing algorithms are well-fit for solving optimisation problems and are therefore of interest for this study. The strategy used in quantum annealing is based upon the workings of the metaheuristic global optimisation method of simulated annealing described in Section 3.2. Crucially, Kadowaki and Nishimori [35] recognised that there is a similarity between thermal fluctuations and quantum fluctuations. Similar to the thermal fluctuations that are used in simulated annealing, quantum fluctuations cause transitions of states. It was this shared property that served as the basis for the development of quantum annealing. In both methods a solution is sought by traversing a solution landscape to find the global minimum. Analogous to the thermal fluctuations that are modulated in simulated annealing, quantum fluctuations can be tuned as well. However, instead of lowering the temperature gradually like in simulated annealing, a transverse field coefficient is modulated that is slowly lowered in its strength over the course of the simulation. It is by modulating this coefficient that the traversibility of the solution landscape is altered and an extensive exploration of the solution landscape can be done to find the optimal solution.

The operator that describes the transverse field is often called the *tunnelling Hamiltonian* [36]. The term tunnelling refers to a quantum mechanical phenomenon where a particle goes through an energy barrier, while not having enough energy to go over the barrier. This process is called tunnelling, because such an energy barrier is often depicted as a mountain. A schematic drawing of this concept is depicted in Figure 4.1. The quantum annealing path is more efficient as compared to the thermal jump, which results in a more efficient way to search the solution space.

There are other differences between the two optimisation methods, that become apparent when comparing the two methods side by side.

1. In simulated annealing the energy function directly represents the objective function of the problem. In quantum annealing, the objective function must first be translated to a pseudo-Boolean objective function, such that a problem specific magnetic field can be prepared. This pseudo-Boolean objective function will be described in more detail in the next section.

2. In simulated annealing the algorithm starts with an initial guess for the optimal point, quantum annealing starts with a superposition state that is realised by applying a transverse field.

3. In simulated annealing the initial ‘temperature’ of the system is high and slowly cooled down. Similarly, the strength of the transverse field, that causes the quantum fluctuations and processes like tunnelling

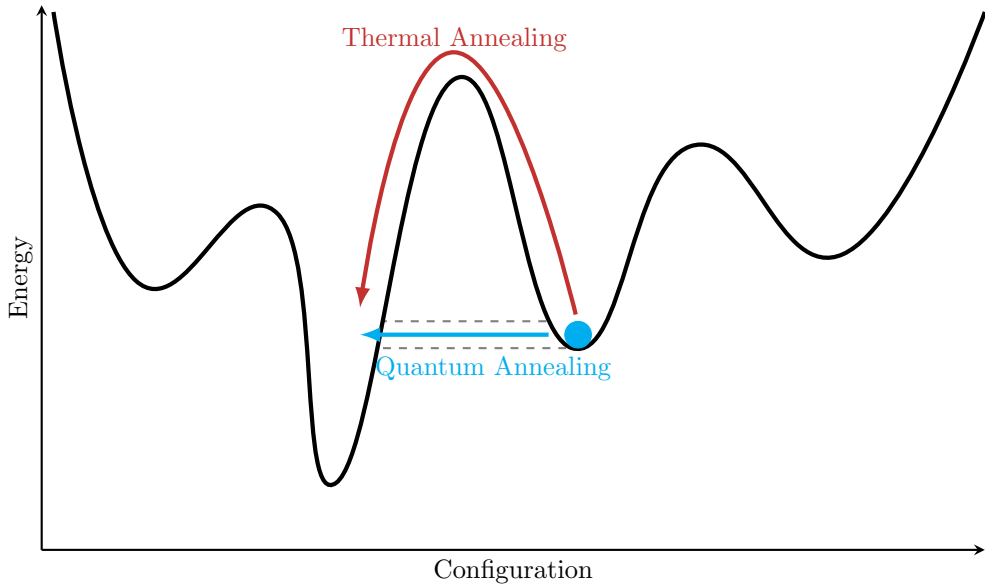


Figure 4.1: A cost function is shown in black and the cyan coloured dot is a possible configuration. The system moves towards the lowest cost (lowest energy). In red the path of a thermal annealing jump toward the lowest possible energy is shown, whereas the cyan coloured path shows the quantum annealing path.

to happen, is lowered. The strength of the magnetic field that describes the objective function is increased simultaneously.

Like simulated annealing, quantum annealing is probabilistic in nature. There is always a non-zero and sometimes high probability that the optimal solution is not found [4]. In the current systems, the initialisation times of the problem specific field are high relative to the time it takes to perform the annealing and measurement procedure [4]. Therefore, it is advised to repeat the annealing process multiple times.

4.4 The Binary Quadratic Model

During quantum annealing, a problem specific magnetic field is turned on, which slowly increases in strength during the annealing process. Hence, some methodologies must be used to encode the problem into such a magnetic field. To this end, the problem must be written into the form of a very specific type: *the binary quadratic model*. The binary quadratic model describes both the ferromagnetic Ising model and its binary equivalent QUBO model, which both belong to the class of NP-Hard problems [10].

The binary quadratic model describes the energy of a system in the form of a polynomial where the variables are either 0 or 1 (or equivalently -1 or 1) and the degree of the polynomial is at most 2. However, we will only concern ourselves with the binary quadratic model with variables in the set $\mathbb{B} = \{0, 1\}$. Furthermore, the binary quadratic model does not allow for any constraints imposed on the variables (other than that they are binary). This gives the following form:

$$E(\mathbf{x}) = \sum_{i=1}^{N} a_i x_i + \sum_{i=1}^{N} \sum_{j=i+1}^{N} b_{ij} x_i x_j + c, \tag{4.2}$$

where $x_i \in \mathbb{B}$ for all $i = 1, \ldots, N$ and a_i, b_{ij} and c are real valued constants.

For many problems in the class of NP-Hard problems, equivalent binary quadratic models are readily available [37, 38]. For other problems, like the stack power maximisation problem, there exist well-established techniques to transform them to the binary quadratic model [37]. In this thesis, two of these techniques will be of importance: (1) including linear equality constraints into the objective function and (2) degree reduction of higher order binary polynomials.

4.4.1 Including Linear Equality Constraints

The binary quadratic model does not allow for constraints, therefore they must be included into the binary quadratic model itself. This is done by adding a penalty term (also known as a Lagrange multiplier) to the objective function. This penalty term is constructed in such a way that the objective function will favour solutions where no constraints are violated. This can be done for both equality constraints and inequality constraints. However, for the topics discussed in this thesis, only linear equality constraints are of interest. Let x_i be binary variables, a linear equality constraint has the following form:

$$\sum_{i=1}^{N} a_i x_i = b, \tag{4.3}$$

where a_i and b are real constants. A penalty term for this constraint is then:

$$\left(\sum_{i=1}^{N} a_i x_i - b \right)^2. \tag{4.4}$$

Equation (4.4) is zero if and only if the constraint of Equation (4.3) is met. When this constraint is violated it is always larger than zero. The term of Equation (4.4) is scaled with a penalty parameter p and added to the objective function. The exact value of p is important for the performance of the quantum annealing algorithm. However, other than the general rule of thumb that p should be roughly the magnitude of the largest constant of the binary quadratic model, there are no hard rules for the optimal values [6].

4.4.2 Degree Reduction

The binary quadratic model only allows for polynomials of at most degree two. When the degree is higher, the higher order model must be reduced to a binary quadratic model. This is done by adding auxiliary variables, such that the optimal point of the reduction, without any auxiliary variables, is the same as the optimal point of the higher order model. In other words, if $h(x)$ is the objective function of the higher order model and $f(x, y)$ is the objective function of the reduction, then

$$\max_x h(x) = \max_{x,y} f(x, y). \tag{4.5}$$

A variety of different degree reduction methods exists [4]. However, because of the limited number of qubits, methods that require the least amount of auxiliary variables are of interest. These methods are

- The Freedman method [39]
- The Ishikawa method [40]
- Pairwise substitution [41, 42]

The Freedman and Ishikawa methods only work on mononomials and not the whole structure of the problem. On the other hand, pairwise degree reduction reduces the degree of pairs of variables that are used the most in the whole polynomial, not just mononomials.

The Freedman method works exclusively for binary mononomials with a negative coefficient and it only requires one auxiliary variable. The Freedman method exploits the fact that negative binary mononomials are almost always zero, except when all variables in the mononomial are 1, then the outcome is -1. Hence, for any binary mononomial of degree d we have the following identity:

$$-\prod_{i=1}^{d} x_i = \min_{y\in\mathbb{B}} y\left(d-1-\sum_{i=1}^{d} x_i\right). \tag{4.6}$$

The identity shown in Equation (4.6) is then substituted into the original objective function.

The Ishikawa method works only for positive binary mononomials of degree d. However, the Ishikawa method adds $k = \lfloor \frac{d-1}{2} \rfloor$ auxiliary variables. For any binary mononomial of degree d we have the following identity:

$$\prod_{i=1}^{d} x_i = \begin{cases} \displaystyle\sum_{i=1}^{d}\sum_{j=i+1}^{d} x_i x_j + \min_{\mathbf{y}\in\mathbb{B}^k}\left[\sum_{i=1}^{k}(4i-1)y_i - 2\sum_{i=1}^{k}\sum_{j=1}^{d} y_i x_j\right] & \text{if } d \text{ is even,} \\ \displaystyle\sum_{i=1}^{d}\sum_{j=i+1}^{d} x_i x_j + \min_{\mathbf{y}\in\mathbb{B}^k}\left[\sum_{i=1}^{k}(4i-1)y_i - 2\sum_{i=1}^{d}\sum_{i=1}^{k} x_i y_j + y_k\left(1-d+\sum_{i=1}^{d} x_i\right)\right] & \text{if } d \text{ is odd.} \end{cases} \tag{4.7}$$

This is then substituted in the original objective function.

The pairwise substitution method searches for the pairs of variables that are used in the higher order terms most often. When these are found, they are substituted throughout the whole objective function by a new auxiliary variable and a penalty term is added to ensure that the auxiliary variable gives the same result. Just like the penalty term in the previous section, this penalty term greatly influences the performance of the quantum annealing algorithm. Moreover, just like for the other penalty term, no hard rules are yet found for determining the optimal penalty term p, except for the rule of thumb that p should be in the order of magnitude of the largest coefficient of the binary quadratic model.

4.5 Minor-Embedding

The D-Wave Quantum annealers are designed to solve binary quadratic models. This is done by representing both the binary quadratic model and the topology of the QPU as a graph and then finding a mapping between the two graphs. First, the binary quadratic model will be represented as a graph. Let $G_P = (V_P, E_P)$ denote a graph of the binary quadratic model, the problem, where V_P is the set of vertices and E_P is the set of edges. Let a_i be weights on the vertices and b_{ij} the weights on the edges. Next, the QPU topology will be described in terms of a graph. Let $G_T = (V_T, E_T)$ denote a graph describing the topology of the QPU. The set of vertices V_T represents the qubits and the set of edges E_T represents the couplers between the qubits.

The goal is to find a mapping from G_P with weights a_i and b_{ij} to G_T that is both injective and structure preserving. Such a mapping is called an *embedding*. The QPU topology is not fully connected, thus it is often the case that the degree of the vertices in the binary quadratic model graph are larger compared to those available in the topology graph of the QPU. Therefore, weights on edges on the problem graph are mapped to weights on edges on the topology graph, but weights on vertices are mapped to weights on vertices of trees in the topology graph. They must act as one variable. This is done by mapping a new parameter *chain strength* to the edges of the trees.

The chain strength parameter must be sufficiently large, to ensure that the tree acts as one variable. Edges and vertices in the topology graph that are not used are assigned a weight of 0. The resulting weighted topology graph has the binary quadratic graph as a minor graph , which can be produced by deleting all

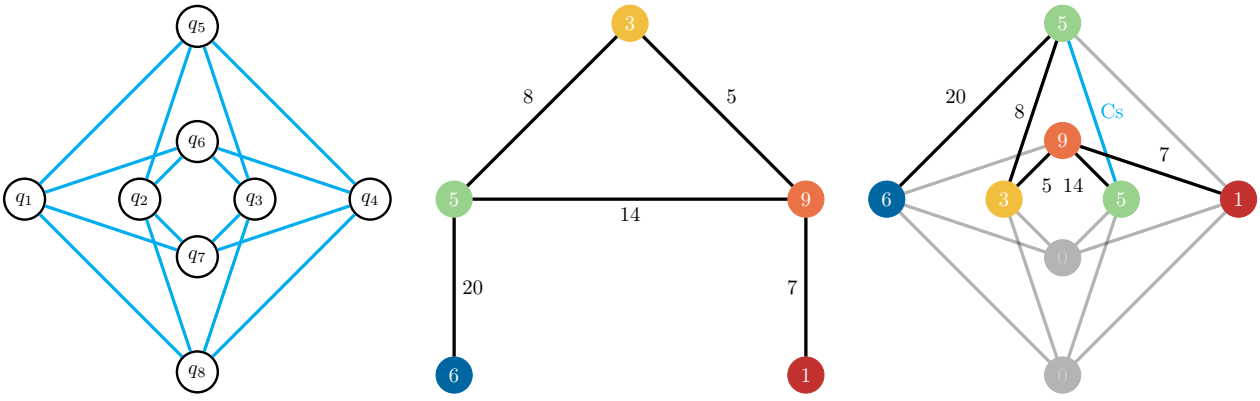
edges and nodes with weight 0 and by contracting edges of the trees. The aforementioned embedding is therefore called a *minor-embedding*. The quality of a minor-embedding is described in terms of the *chain length*, which is the size of the largest chain.

Another important parameter that has a great influence on the performance of the quantum annealing algorithm, is the chain strength of the minor-embedding. It must be large enough such that the nodes of a tree act as one variable. However, it cannot be too large, because then it will interfere with the other qubits [43]. In this thesis sufficiently large chain strengths were calculated using the *uniform_torque_compensation* functionality of D-Wave.

4.5.1 Example Embedding

Consider the embedding of a small problem onto the unit cell of the Chimera topology, which is a type of QPU topology. Even though the Chimera topology is no longer the state of the art topology for quantum annealers, as it is replaced by the next generation Pegasus topology, it is still good for an example. The Chimera unit cell, depicted in Figure 4.2a, is a complete bipartite graph with two vertex sets containing 4 vertices, denoted $K_{4,4}$. Even though the degree of each node is 4, there are no odd cycles possible. This is because the graph is bipartite and bipartite graphs cannot contain odd cycles. Even a simple 3-cycle (a triangle) does not exist on the Chimera unit cell and must therefore be embedded.

Suppose we want to embed the problem shown in Figure 4.2b, where the nodes and edges correspond to the a_i and b_{ij} coefficients of equation Equation (4.2) respectively. Because the problem graph is not bipartite, it is not possible to map every node of the problem graph to a node on the Chimera unit cell. To correct this, nodes from the binary quadratic model are mapped to trees in the Chimera unit cell which are connected with a chain with a certain chain strength. An example embedding is shown in Figure 4.2c. The green node with weight 5 appears twice in the embedding and is connected with the cyan chain with a chain strength of Cs. Note that by deleting the unused edges and nodes and by contracting the cyan edge, the binary quadratic model graph is retrieved.



(a) A unit cell of the chimera type topology.

(b) Input binary quadratic graph

(c) Embedding on the chimera unit cell topology

Figure 4.2: Embedding of an binary quadratic model problem onto the unit cell of the chimera topology.

4.6 The D-Wave Hybrid Solver

The amount of qubits available in the current quantum annealers is still limited. Moreover, the number of couplers is also rather limited, giving rise to large minor-embeddings and consequently even more usage of the limited number of qubits. Hence, solving large problems directly on the quantum annealers is still out of reach for the current quantum annealers. For these larger problems the D-Wave Hyrbid Solver was developed. The D-Wave hybrid solver tries to combine the best both classical computing and quantum annealing.

When using the hybrid solver a binary quadratic model together with a maximum runtime is sent to the D-Wave machines. There, multiple subproblems are made of the original problem which are sent to the quantum annealers. The results of the quantum annealers are combined with a classical algorithm to search for the global minimum of the problem.

5 Two Binary Quadratic Models for Residual Statics Estimation

In this chapter the stack power optimisation problem will be written as a binary quadratic model, such that it can be solved by the D-Wave Advantage quantum annealer and the Hybrid solver. To this end, the stack power problem will be rewritten such that the shift operator and Euclidean norm are removed from the variables. Next, this new formulation will be encoded in two ways: (1) a 'one-hot' encoding model and (2) a 'binary' encoding model. Lastly, a workflow for using these models is presented.

5.1 Removing the Shift Operator and Euclidean Norm from the Variables

To solve the stack power maximisation problem with the D-Wave quantum annealers, Equation (2.2) must be written into the binary quadratic model format. Therefore, the objective function must be transformed to a quadratic polynomial and the real search space must be changed to a binary search space. For any given quantum annealer, there is fixed number of qubits and couplers, which directly influence the maximum size of the binary search space. Therefore, the binary search space must be kept sufficiently small, i.e. there is a maximum number of shifts that can be implemented for a given set of traces.

The search space S will therefore be defined more precisely as a discrete solution space, which will be enforced by two different encodings in the following sections. As mentioned at the end of Section 2.2, there can be an infinite amount of solutions for Equation (2.2). This problem can be resolved by either keeping the static of one trace fixed, or by reducing the search space. Keeping one static fixed can have drastic consequences for the size of the search space needed to find the optimal solution. This is best illustrated with an example.

Some example traces are shown in Figure 5.1. The traces in Figure 5.1 are the same traces as shown in Section 2.3. Thus, the misalignment with respect to $\mathbf{d}_0$ are 10 ms and 30 ms for $\mathbf{d}_1$ and $\mathbf{d}_1$ respectively. If $\mathbf{d}_1$ is fixed, then the largest shift to reach perfect alignment is 20 ms. On the other hand, when either $\mathbf{d}_0$ or $\mathbf{d}_2$ is fixed, then the largest shift needed is 30 ms. So, to find the optimal solution when either $\mathbf{d}_0$ or $\mathbf{d}_2$ are fixed, the search space must be 50 % larger compared to the search space then when $\mathbf{d}_1$ is fixed. As the size of the search space is limited by the number of qubits and couplers of the quantum annealer, it is important to find the best trace to fix. Otherwise, it can quickly happen that the optimal solution will not be present in the search space. This would lead to a result that is not the global optimum of the complete problem. In general it is difficult to find the best trace to fix, as it requires preexisting knowledge of the optimal solution. To keep the search space small, it should therefore be preferred to abstain from fixing a trace when using quantum annealing methods.

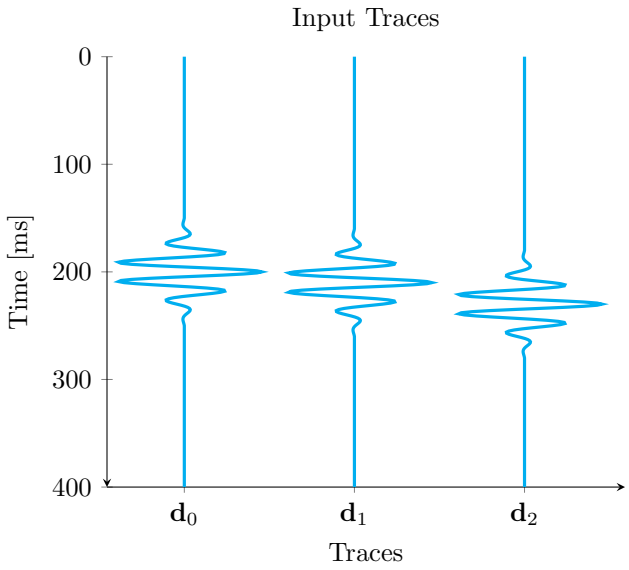


Figure 5.1: Three example traces. The misalignment with respect to $\mathbf{d}_0$ are 10 ms and 30 ms for $\mathbf{d}_1$ and $\mathbf{d}_2$ respectively.

To define the search space more precisely, let S_K consist of K equidistant discrete timesteps, which are exactly the same for each trace, i.e.

$$S_K = \{t_1, t_2, \ldots, t_K\}^M, \tag{5.1}$$

such that $|t_i - t_{i+1}| = c$ for all $i = 1, \ldots, K-1$. By using the Kronecker delta, denoted δ_{xa}, the shift operators can be made independent of the variables. Note that for any trace $\mathbf{d}$ and any integer $x \in \{1, 2, \ldots, K\}$ the following identity holds true

$$\mathbf{d}(t_x) = \sum_{a=1}^{K} \delta_{xa} \mathbf{d}(t_a). \tag{5.2}$$

Substituting the identity presented in Equation (5.2) into Equation (2.2) together with the identity $\|\mathbf{x}\|_2^2 = \langle \mathbf{x}, \mathbf{x} \rangle$, where $\langle \cdot, \cdot \rangle$ denotes the standard inner product, results in the following equivalent optimisation problem:

$$\max_{\mathbf{x} \in \{1,2,\ldots,K\}^M} \left\langle \sum_{i=1}^{M} \sum_{a=1}^{K} \delta_{x_i a} \mathbf{d}_i(t_a), \sum_{j=1}^{M} \sum_{b=1}^{K} \delta_{x_j b} \mathbf{d}_j(t_b) \right\rangle. \tag{5.3}$$

Using the bilinearity of the inner product and the property that $\delta_{x_i a} \delta_{x_i b} = 0$ whenever $a \neq b$ then gives

$$\max_{\mathbf{x} \in \{1,2,\ldots,K\}^M} \sum_{i=1}^{M} \sum_{a=1}^{K} (\delta_{x_i a})^2 \langle \mathbf{d}_i(t_a), \mathbf{d}_i(t_a) \rangle + 2 \sum_{i=1}^{M} \sum_{j=i+1}^{M} \sum_{a=1}^{K} \sum_{b=1}^{K} \delta_{x_i a} \delta_{x_j b} \langle \mathbf{d}_i(t_a), \mathbf{d}_j(t_b) \rangle. \tag{5.4}$$

Note that $\mathbf{d}_i(t_a)$ is just a shift of $\mathbf{d}_i$ and therefore does not alter the Euclidean norm. Therefore, we have that $\|\mathbf{d}_i\|_2^2 = \langle \mathbf{d}_i(t_a), \mathbf{d}_i(t_a) \rangle$ for all $t_a \in S_K$. Using this together with the property of the Kronecker delta that $\sum_i \delta_{ij} c_i = c_j$ we get:

$$\max_{\mathbf{x} \in \{1,2,\ldots,K\}^M} \sum_{i=1}^{M} \|\mathbf{d}_i\|_2^2 + 2 \sum_{i=1}^{M} \sum_{j=i+1}^{M} \sum_{a=1}^{K} \sum_{b=1}^{K} \delta_{x_i a} \delta_{x_j b} \langle \mathbf{d}_i(t_a), \mathbf{d}_j(t_b) \rangle. \tag{5.5}$$

Note that the first term in Equation (5.5) is constant. Furthermore, the inner products have become coefficients, which must be calculated before the optimisation procedure. Moreover, the variables in Equation (5.5) only alter the Kronecker delta, which will be encoded by binary variables in the next two sections.

5.2 The 'One-Hot' Encoding

The easiest way to encode the Kronecker deltas, is by replacing them with binary variables. In other words, introduce binary variables x_{ia} such that

$$x_{ia} = \begin{cases} 1 & \text{, if trace } i \text{ and shift } a \text{ are selected,} \\ 0 & \text{, otherwise.} \end{cases} \tag{5.6}$$

and demanding that every trace can only be selected exactly once. The latter will be enforced with a constraint. Mathematically this constraint can be formulated as $\sum_{a=1}^{K} x_{ia} = 1$, for all $i = 1, \ldots, M$. Therefore, the maximisation problem shown in Equation (5.5) becomes

$$\begin{aligned} \max_{\mathbf{x} \in \mathbb{B}^{MK}} \quad & \sum_{i=1}^{M} \|\mathbf{d}_i\|_2^2 + 2 \sum_{i=1}^{M} \sum_{j=i+1}^{M} \sum_{a=1}^{K} \sum_{b=1}^{K} x_{ia} x_{jb} \langle \mathbf{d}_i(t_a), \mathbf{d}_j(t_b) \rangle, \\ \text{such that} \quad & \sum_{a=1}^{K} x_{ia} = 1 \qquad \text{for all } i = 1, \ldots, M. \end{aligned} \tag{5.7}$$

The objective function of Equation (5.7) is already in the binary quadratic model format. However, the M constraints are not allowed. Therefore, the technique shown in Equation (4.3) to add a penalty term to the objective function, such that the objective function will favour solutions where these constraints are met, will be used. Using the property that $x^2 = x$ for any $x \in \mathbb{B}$, we have

$$\left(\sum_{a=1}^{K} x_{ia} - 1 \right)^2 = 2 \sum_{a=1}^{K} \sum_{b=a+1}^{K} x_{ia} x_{ib} - \sum_{a=1}^{K} x_{ia} + 1. \tag{5.8}$$

The penalty term given by Equation (5.8) must be added for each of the M traces. The resulting binary quadratic model is then

$$\begin{aligned} \max_{\mathbf{x} \in \mathbb{B}^{MK}} \quad & \sum_{i=1}^{M} \|\mathbf{d}_i\|_2^2 + 2 \sum_{i=1}^{M} \sum_{j=i+1}^{M} \sum_{a=1}^{K} \sum_{b=1}^{K} x_{ia} x_{jb} \langle \mathbf{d}_i(t_a), \mathbf{d}_j(t_b) \rangle \\ & - p \left(2 \sum_{i=1}^{M} \sum_{a=1}^{K} \sum_{b=a+1}^{K} x_{ia} x_{ib} - \sum_{i=1}^{M} \sum_{a=1}^{K} x_{ia} + M \right) \end{aligned} \tag{5.9}$$

The use of binary variables as described in this section, where there is one bit used for every possible outcome, is called the 'one-hot' encoding. Therefore, the binary quadratic model described in Equation (5.9) will be called the *one-hot encoding model*.

5.3 The 'Binary' Encoding

In classical computing, binary representations are used to encode 2^k combinations with just k bits. To reduce the amount of used variables, the idea is to use a similar kind of binary encoding to encode the possibilities of the Kronecker delta. Hence, we want to find some sort of function $c_a : \mathbb{B}^k \to \mathbb{B}$ which encodes the behaviour of the Kronecker delta. For this purpose, the function bin : $V = \{1, \ldots, 2^k\} \to \mathbb{B}^k$ such that bin(a) is a k

dimensional binary vector representing the binary number of $a-1$. Using $\text{bin}(a)$ we can define c_a as the function which behaves according to

$$c_a(\mathbf{x}) = \begin{cases} 1 & \text{, if } \text{bin}(a) = \mathbf{x}, \\ 0 & \text{, otherwise.} \end{cases} \tag{5.10}$$

In other words, c_a is a binary function with $2^k - 1$ roots. Hence, Equation (5.5) becomes

$$\max_{\mathbf{x}\in\mathbb{B}^{MK}} \sum_{i=1}^{M} \|\mathbf{d}_i\|_2^2 + 2\sum_{i=1}^{M}\sum_{j=i+1}^{M}\sum_{a=1}^{2^k}\sum_{b=1}^{2^k} c_a(\mathbf{x}_i)c_b(\mathbf{x}_j)\langle \mathbf{d}_i(t_a), \mathbf{d}_j(t_b)\rangle. \tag{5.11}$$

To compute the binary quadratic model of Equation (5.11), an explicit polynomial form of Equation (5.11) must be found. Therefore, the functions c_a must be represented as a polynomial. To shorten the notation, let $\text{bin}(a) = \begin{bmatrix} a_1 & a_2 & \cdots & a_k \end{bmatrix}^\intercal = \mathfrak{a}$. Furthermore, it will be necessary to introduce a concept from information theory: the *Hamming Weight* [44]. The Hamming Weight of a vector counts the number of nonzero entries and is denoted $w(\mathbf{x})$, where $\mathbf{x} \in \mathbb{B}^k$. For example, for $\mathbf{x} = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 \end{bmatrix}^\intercal$, the Hamming Weight is $w(\mathbf{x}) = 2$. Let $\mathcal{P}(V)$ denote the powerset of V, then $c_a(\mathbf{x})$ can be explicitly written as

$$c_a(\mathbf{x}) = \sum_{A\in\mathcal{P}(V)} (-1)^{w(\mathfrak{a})+|A|} \prod_{i\in A^c} \bar{a}_i \prod_{i\in A} x_i, \tag{5.12}$$

if we set $\prod_{i\in\emptyset} a_i = 1$. Therefore the product $c_a(\mathbf{x})c_b(\mathbf{y})$ becomes

$$c_a(\mathbf{x})c_b(\mathbf{y}) = \sum_{A\in\mathcal{P}(V)}\sum_{A\in\mathcal{P}(V)} (-1)^{w(\mathfrak{a})+w(\mathfrak{b})+|A|+|B|} \prod_{i\in A^c}\prod_{j\in B^c} \bar{a}_i\bar{b}_j \prod_{i\in A}\prod_{j\in B} x_i y_j. \tag{5.13}$$

A useful and easily verified property of the Hamming Weight is $w(\mathfrak{a}) + w(\mathfrak{b}) = w(\mathfrak{a}\oplus\mathfrak{b}) + 2\langle\mathfrak{a},\mathfrak{b}\rangle$, where $\oplus$ denotes the XOR operation. Using this property together with the identity $|A| + |B| = |A\Delta B| + 2|A\cap B|$, where Δ denotes the symmetric difference of two sets, simplifies Equation (5.13) to

$$c_a(\mathbf{x})c_b(\mathbf{y}) = \sum_{A\in\mathcal{P}(V)}\sum_{B\in\mathcal{P}(V)} (-1)^{w(\mathfrak{a}\oplus\mathfrak{b})+|A\Delta B|} \prod_{i\in A^c}\prod_{j\in B^c} \bar{a}_i\bar{b}_j \prod_{i\in A}\prod_{j\in B} x_i y_j. \tag{5.14}$$

5.3.1 Final 'Binary' Encoding Model

Substitution of the results of Equation (5.13) into Equation (5.11) gives the following explicit form:

$$\begin{aligned} &\max_{\mathbf{x}\in\mathbb{B}^{Mk}} \sum_{i=1}^{M} \|\mathbf{d}_i\|_2^2 \\ &+ 2\sum_{i=1}^{M}\sum_{j=i+1}^{M}\sum_{a=1}^{K}\sum_{b=1}^{K}\sum_{A\in\mathcal{P}(V)}\sum_{B\in\mathcal{P}(V)} (-1)^{w(\mathfrak{a}\oplus\mathfrak{b})+|A\Delta B|} \prod_{n\in A^c}\prod_{m\in B^c} \bar{a}_n\bar{b}_m \prod_{n\in A}\prod_{m\in B} x_i^{(n)} x_j^{(m)} \langle \mathbf{d}_i(t_a), \mathbf{d}_j(t_b)\rangle. \end{aligned} \tag{5.15}$$

Note that this is not yet a binary quadratic model, but a higher order binary model. However, this explicit form of the objective function can be quadratised using built-in D-Wave functionalities, like make_quadratic. Because the Kronecker delta was encoded with a standard binary encoding, the model presented in this section is called the *binary encoding model*.

5.4 Workflow for the Two Models

In this section the workflow for using the binary quadratic models presented in Section 5.2 and Section 5.3 will be given. The general outline for both models is depicted in Figure 5.2, together with the places in the workflow where user input must be given. Note that only the fourth level of the flowchart (QA) actually uses the quantum annealer, the rest is either pre-processing or post-processing.

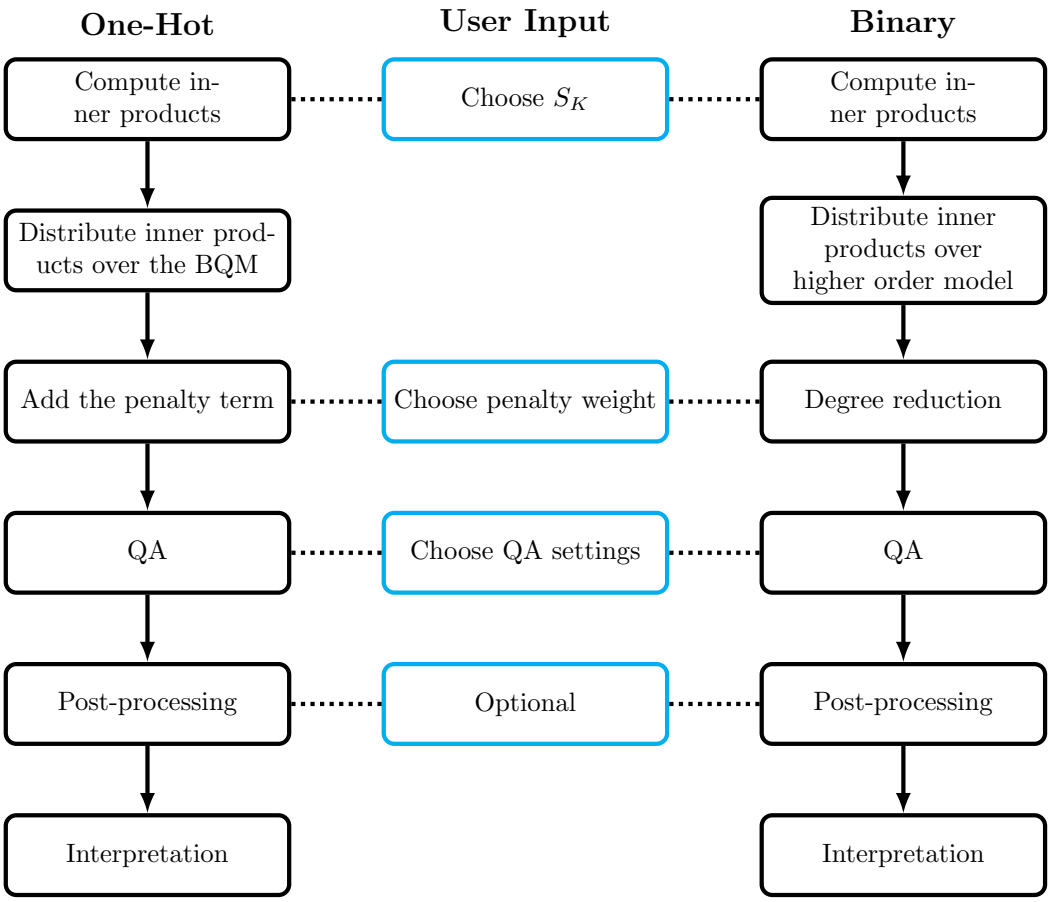


Figure 5.2: Flowchart of the workflow of the one-hot encoding model (left) and the binary encoding model (right). The cyan outlined squares in the middle represent the user input at the different levels.

5.4.1 One-Hot Encoding Workflow

First, the workflow of the one-hot encoding model will be discussed. To solve the residual statics estimation problem, the coefficients of the binary quadratic model shown in Equation (5.9) must be computed. This computation is done on a classical computer. To this end, $\|\mathbf{d}_i\|_2^2$ must be calculated for $i = 1, \ldots, M$ and $\langle d_i(t_a), d_j(t_b)\rangle$ must be calculated for all $i, j = 1, \cdots, M$ and all $t_a, t_b \in S_K$. Hence, the first user input that must be considered is what to choose for S_K. In the case of a synthetic set of which the solution is known, an appropriate size is easily chosen. In the case that the solution is not known, visual inspection can aid in choosing S_K. Once S_K is chosen and the inner products are computed, the inner products can be distributed over the variables as shown in Equation (5.9).

After the distribution, the penalty term can be added and therefore the weight of the penalty p must be determined. As a general rule of thumb, the weight should be in the same order of magnitude as the largest absolute coefficient, which is $\max 2\,|\langle d_i(t_a), d_j(t_b)\rangle|$. The size of the penalty term can have great effect on the performance of the quantum annealing process [6]. Therefore, fine-tuning it beyond just the order of magnitude can greatly enhance the quality of the solutions returned by the quantum annealer. As simulated annealing often gives similar types of results, the weight of the penalty term was fine-tuned using fast runs of a simulated annealing algorithm provided by D-Wave.

After adding the penalty term, the binary quadratic model is complete and can be solved by a quantum annealer or Hybrid solver. Both also require some extra parameters. During this master thesis research, it was often found that results returned by the quantum annealer were not in local minima (and thus certainly not in the global minimum). To counteract this, the local search algorithm of steepest descent can be added to improve the results. In this binary case, steepest descent means that we greedily search for the bit-flip that improves the energy the most, repeating until no improvements can be made. After the optional post-processing step the results can be interpreted.

5.4.2 Binary Encoding Workflow

The workflow for the binary encoding model is very similar. First, the coefficients of the higher model must be calculated, for which we again need the inner products $\langle d_i(t_a), d_j(t_b)\rangle$ and Euclidean norms $\|\mathbf{d}_i\|_2^2$. Hence, we need to choose S_K for which we assume $K = 2^k$. Note that this limits the size of S_K to powers of 2. Thereafter, the coefficients must be distributed over the higher order model according to Equation (5.15). Next, degree reduction provided by D-Wave is used to reduce the degree to be quadratic. The degree reduction uses a combination of the three methods discussed in subsection 4.4.2. Therefore, a penalty term is applied for which the weight must be determined. This weight was determined similarly to the one-hot encoding model with some fast runs of simulated annealing. It must be stressed that this penalty term is different as it ensures that the auxiliary variables have the right values, wheres for the one-hot encoding there are no auxiliary variables. After quadratisation the binary encoding model can be solved by the quantum annealer and Hybrid solver. Optionally, steepest descent can be added in post-processing to further improve the results after which the results can be interpreted.

6 Theoretical Analysis of the Binary Quadratic Models

In this chapter the binary quadratic models will be theoretically analysed. First, the pre-processing steps will be analysed, which are computed on a classical computer. Thereafter, the 'quantum costs' are discussed to give insight into the scalability (in terms of number of traces and number of shifts) of the one-hot and binary encoding models. Furthermore, a note is made on how we interpret the two different models.

6.1 Complexity Analysis of the Pre-processing Steps

The pre-processing steps for both models consist out of three steps. First, the inner products must be calculated for both of the models . Next, the inner products must be distributed over the quadratic and higher order model for the one-hot model and binary model respectively. Lastly, for the one-hot model the penalty term must be added, while for the binary model degree reduction must be applied. In this section, the computational cost of each of these steps will be analysed using asymptotic analysis.

The fist step in the pre-processing for both models is the computation of the inner product. The inner product can be calculated in two ways: (1) directly or (2) using the Fast Fourier Transform. Calculating the inner product of two N dimensional traces directly is of computational complexity of $\mathcal{O}(N)$. It is assumed that all shifts defined in Equation (5.1) are equidistant. Therefore, there $M(M-1)(2K-1)$ inner product must be calculated in total. Hence, the computational complexity of this pre-processing step using the direct method is $\mathcal{O}(M^2NK)$. Another way to calculate the inner products is by calculating the cross-correlation between all traces, since the cross-correlation consists of all the inner products and shift combinations between two traces. Cross-correlation methods frequently utilise the Fast Fourier Transform, therefore calculating the cross-correlation between two traces is of complexity $\mathcal{O}(N \log_2 N)$. Hence, computing all the needed inner products using the cross-correlation method has a computational cost of $\mathcal{O}(M^2 N \log_2 N)$. Because K is much smaller relative to N and because of the simplicity of its implementation, in this thesis the aforementioned direct method will be used.

Next, the cross-correlation must be distributed to calculate the binary quadratic model and the higher order model of the one-hot model and binary model respectively. For the binary quadratic model of the one-hot encoding, as presented in Equation (5.9), there is an inner product present in the coefficient of the quadratic terms between two different traces for each possible shift. Therefore, the complexity of distributing the inner products over the coefficients has a computational complexity of $\mathcal{O}(M^2K^2)$. On the other hand, the penalty term is distributed over every shift combination for every trace separately and therefore does not add any additional computational complexity. Therefore, calculating the binary quadratic model of the one-hot encoding using pre-calculated inner products is of complexity $\mathcal{O}(M^2K^2)$. For the higher order model of the binary encoding model presented in Equation (5.15), there is an inner product present in the six sums. The first two sums are over all the M traces , the next two over all the K shifts and the last two over all

the subsets of V. Since $|V| = k$, we have $\mathcal{P}(V) = 2^k = K$. Lastly, calculating the products is of complexity $\mathcal{O}(k)$. Therefore, calculating the higher order model of the binary encoding model is of complexity $\mathcal{O}(M^2K^4k)$

Calculating the quadratisation of the binary encoding is theoretically estimated to be bounded by a complexity $\mathcal{O}(M^2K^2k)$ for each of the methods discussed in subsection 4.4.2. The D-Wave method used in this thesis uses a combination of the three methods to find even smaller quadratisations. However, this comes at the cost of a significant increase in computation time with an unknown scaling.

The results of this section are summarised in Table 6.1. From Table 6.1 it becomes clear that computational complexity of the pre-processing of the binary encoding model are much larger compared to those of the one-hot encoding model.

	One-Hot	Binary
Compute inner products	$\mathcal{O}(M^2NK)$	$\mathcal{O}(M^2NK)$
Distributing inner products	$\mathcal{O}(M^2K^2)$	$\mathcal{O}(M^2K^4k)$
Add penalty term	$\mathcal{O}(MK^2)$	-
Degree reduction	-	Larger than $\mathcal{O}(M^2K^4k)$

Table 6.1: Complexity scaling of different parts of pre-processing for the one-hot and binary encoding model.

6.2 Quantum Cost

In this section, the number of variables and number of quadratic terms of both the models will be estimated. The number of variables and quadratic terms of a binary quadratic model give a lower bound on the number of qubits and couplers a quantum annealer would need to be able to handle the binary quadratic model. Together, these will be called the 'quantum cost' of the two models. Keep in mind that even if a binary quadratic model has the same or smaller number of variables and quadratic interactions as the QPU topology, then a minor-embedding might still be necessary. However, it does shed some light on the scaling possibilities of both the one-hot encoding model and binary encoding model.

For the one-hot encoding model, a variable is introduced for each trace and shift combination. Therefore, there are a total of MK variables. Furthermore, there is a quadratic interaction between every variable, which results in $\frac{1}{2}MK(MK-1)$ quadratic interactions.

For the binary encoding model it is more challenging to count the number of variables and quadratic combinations. Before quadratisation, the higher order model consists of Mk variables, where $K = 2^k$ (Equation (5.15)). Quadratisation is done using the D-Wave 'make_quadratic' function, which uses a combination of the methods discussed in subsection 4.4.2. The exact implementation of this quadratisation is not disclosed by D-Wave. Looking at the theoretical bounds, the number of variables is bounded by a complexity of $\mathcal{O}(M^2K^2k)$ as this is the order of mononomials in the higher order model. However, empirical testing seems to indicate a much better scaling in the number of variables of roughly $(1-2^{-k})MK$. Furthermore, based on empirical testing, the number of quadratic terms seems to be roughly $\frac{1}{2}M^2Kk$ for $k = 1, 2, 3$ growing to $\frac{1}{2}MK(MK-1)$ for larger k.

The results of this section are summarised in Table 6.2. From Table 6.2 it can be seen that for a smaller number of shifts, the binary encoding model uses less variables and has a smaller number of quadratic terms. Hence, when the number of shifts is low, the binary encoding model allows for a larger number of traces compared to the one-hot encoding model. However, this difference becomes negligible as the number of shifts increases. Thus, in terms of the number of shifts used, the two models scale the same.

	One-hot	Binary
Number of variables	MK	$(1-2^{-k})MK$
Number of quadratic terms	$\frac{1}{2}MK(MK-1)$	$\frac{1}{2}M^2Kk$ up to $\frac{1}{2}MK(MK-1)$

Table 6.2: Number of variables and number of quadratic terms for the one-hot and binary encoding models. The numbers for the one-hot encoding model are exact, whereas the numbers for the binary encoding model are based on empirical testing.

6.3 Maximum Allowed Problem Size on the Advantage System

The results presented in Section 6.2 are purely theoretical and assume that the number of variables and quadratic terms is directly related to the number of qubits and couplers respectively. However, the QPU topology also plays an important role, as this ultimately determines whether a minor-embedding is needed. The cost of the minor-embedding step is important, but as the topology of the QPUs are constantly improving over time, it is impossible to take it into account.

On the other hand, the current maximum problem size that can be solved on the current state of the art D-Wave Advantage System can be estimated. Two parameters that influence the problems size are the number of traces M and the number of possible shifts K. What we want to know is the maximum value of K given a certain number of traces M. The maximum problem size was determined by trying to find minor-embeddings for the Advantage system. First, one-hot encoding models and binary encoding models were made with a given K and M. Next, the D-Wave minor−miner was used to find an embedding; if one was found M was increased and the process was repeated. This was done for K up 64, as from this point on the number of traces for which a minor-embedding could be found was too small to be relevant. The results are shown in Figure 6.1.

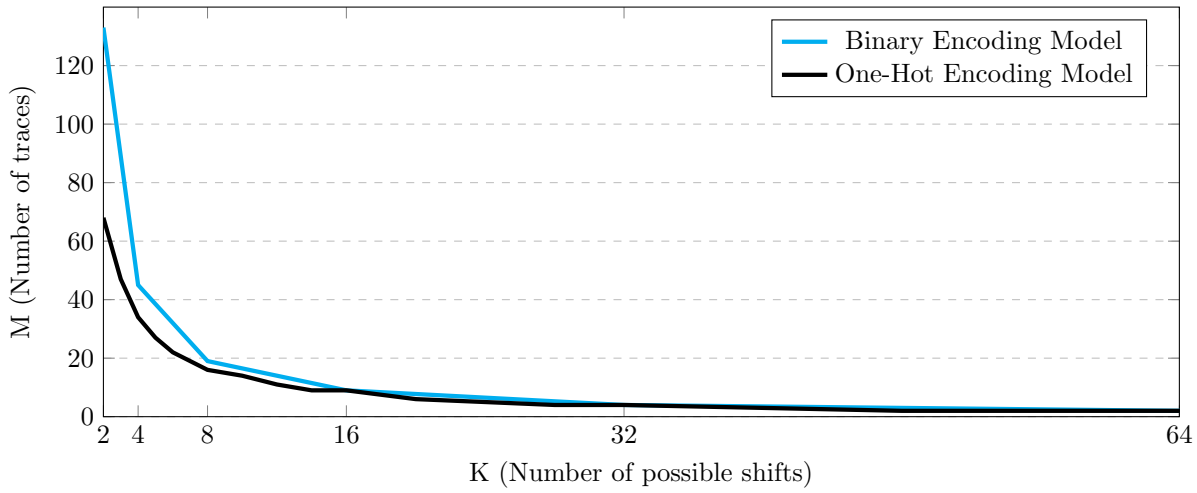


Figure 6.1: Maximum Problem Size on the Advantage System. This was found empirically by searching for a maximum M for which an embedding could be found given a K.

In Figure 6.1 we can see that the binary encoding model allows for a larger number of traces as compared to the one-hot encoding model. However, this is only the case for $K \leq 8$. For a larger number of possible shifts K there is no distinction between the binary encoding model and the one-hot encoding model in terms of maximum problem size. This is in line with the results of Section 6.2, because when K becomes larger, the

number of variables and quadratic terms is approximately equal for both the models.

6.4 Interpretation of the Results

Both the one-hot encoding model and the binary encoding model introduce a penalty term to let the objective function favour solutions where certain constraints are met. When all constraints are met, the interpretation is the same for the two models, as the energy represents the additive inverse of the stack power. When the weight of the penalty term is sufficiently large, a violation of the constraints implies that the solution is not yet optimal. In these cases, the energy has no meaningful interpretation. However, this is where the similarities in the interpretations for the two models end.

In the case of the one-hot encoding model, the constraints ensure that a trace can be selected exactly once. Hence, a violation of a constraints can have two causes. Either, there is a trace which was not selected at all, or a trace was selected with multiple shifts at once. In either case, the shift of such a trace remains unknown and therefore the results are not directly usable. On the other hand, for the binary encoding model, the constraints ensure that the auxiliary variables added in the quadratisation have the right value. Hence, a violation of the constraints means that an auxiliary variable is not yet coinciding with the variables it should represent. However, since it is an auxiliary variable, they can cautiously be ignored, resulting in a possible direct interpretation of the results.

7 Results and Discussion: Quantum Annealing and Hybrid Solver

The theoretical analysis reveals a lot about the interpretation of the results, the scalability and the largest cases of the two models presented in this thesis. However, it provides little information about the quality of the solutions of the actual implementation. Therefore, in this chapter, residual statics of synthetic data will be estimated. This will be done directly on the quantum annealer, as well as with the use of the Hybrid solver.

7.1 Quantum Annealing with the D-Wave Advantage System

To test the performance of the quantum annealer, a set of problems that could run directly on the quantum annealer was made. In this section, the results of these experiments will be shown and the performance of the quantum annealer will be discussed. For all the results that will be shown in this chapter the default quantum annealing settings of the Advantage System of 2021 were used , with the exception of the annealing time. To get an idea of how the annealing time influences the probability of finding the optimal solution different annealing times of 10, 20, 40, 80, 160, 320, 640, 1280 and 1998 μs were used. Finally, for each experiment the number of reads was set to 500.

7.1.1 Proof of Concept: Small Sized Problem

To illustrate the type of results and the way to interpret them, a small proof of concept is shown. This small test problem consists out of 4 traces containing just one wavelet. The gather of the 4 traces is shown in Figure 7.1. The gather of a set of traces shows a 'top view' of the input traces. More specifically, the colour coding of this picture represents the amplitude of parts of the wave. The lighter the colour is, the the more positive the amplitude is. Conversely, the darker the colour the more negative the amplitude is. The maximum positive amplitudes of the wave are thus white, whereas the maximum negative amplitudes are black. This visualisation style makes it easier to detect small shifts such as statics, because you can see at a glance whether the traces are aligned or not by looking at the continuation of the colours.

The traces shown in Figure 7.1 have a temporal discretisation of 0.004 s. With respect to trace 1, the traces 2, 3 and 4 are shifted by 0.004 s, 0.008 s and 0.012 s respectively. First, an appropriate set S_K must be chosen. As this experiment serves as a proof of concept, we want only one optimal solution when choosing the set S_K. The smallest set S_K that has one and only one optimal solution is the set $S_K = \{0,\ 0.004,\ 0.008,\ 0.012\}$. Hence, we have $M = 4$ and $K = 4$. With this S_K the binary quadratic model for both the one-hot encoding and the binary encoding were made, after which they were sent to the quantum annealers.

Figure 7.2 shows a summary of the results. The gather shown in Figure 7.2a represents the gather, where the residual statics of the best solutions are corrected for by shifting the traces correspondingly. These statics,

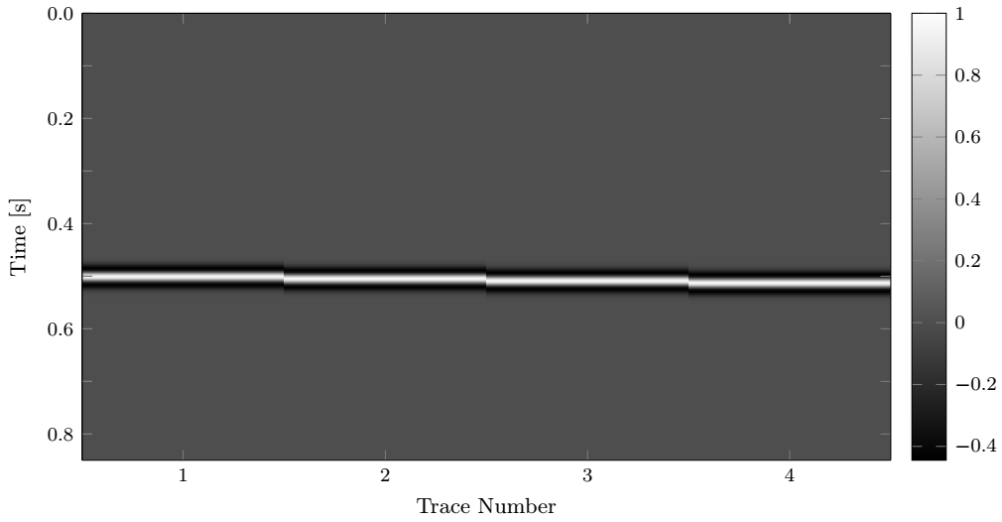


Figure 7.1: Four traces containing one wavelet. The temporal discretisation is 0.004 s and with respect to the first trace the shifts are 0.004 s, 0.008 s and 0.012 s for the second, third and fourth trace respectively.

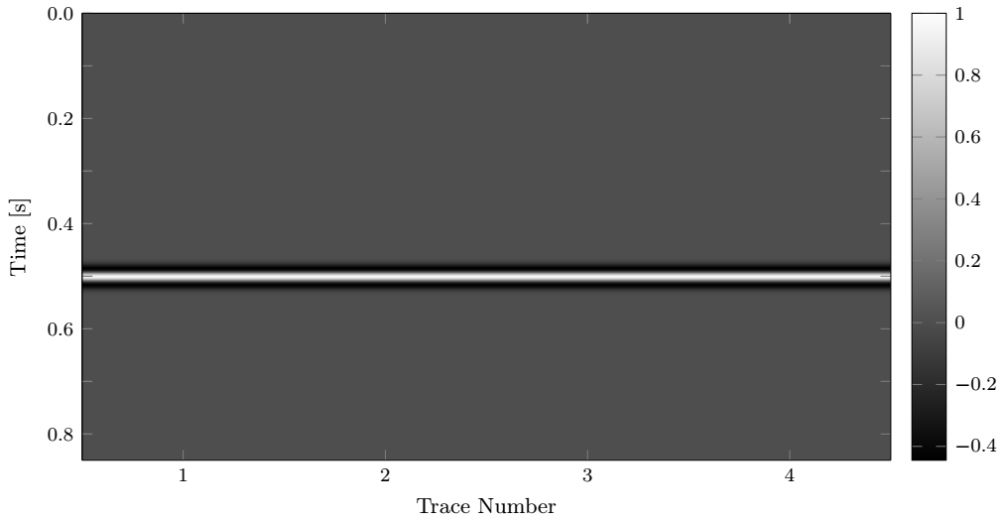
that are shown in the gather, were found with both models when using an annealing time of 1998 μs. Figure 7.2a shows perfect alignment of all the traces. Hence, the quantum annealer found the optimal solution using either of the two models.

Figure 7.2b shows the energy for different annealing times when the one-hot encoding model was used. The orange line represents the optimal solution, which is the lowest energy that can be attained. The black line shows the median energy of the 500 readouts and the red shaded area shows the area between the first and third quartiles (Q1 and Q3). The cyan line shows the minimum energy of the 500 readouts. When the annealing times increases, the median energy, first quartile and third quartile do not appear to change convincingly, which suggests that longer annealing times do not improve the quality of the solution in this experiment. Furthermore, the minimum of energies found was not equal to the solution energy when the annealing time was 10, 80, or 1280 μs. Thus, no convincing correlation between the annealing time and solution quality were found in this experiment.

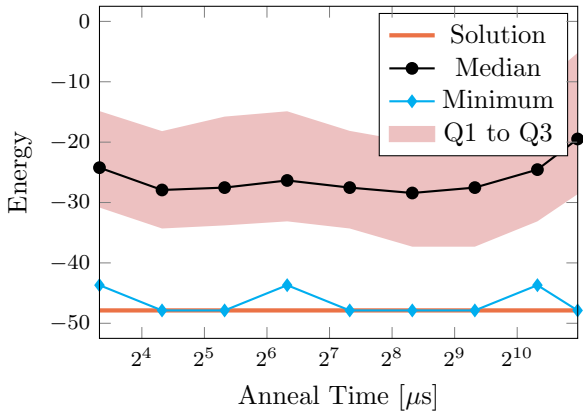
In Figure 7.2c a histogram of the number of occurrences of violated and met constraints is shown for different energies that were found with the one-hot encoding model. This histogram represents the energy distribution at the longest annealing time of 1998 μs. In the majority of the readouts, the constraints were violated. Moreover, the lowest energy solutions occurred when all constraints were met, showing that the objective function with the added penalty term does indeed favour solutions where constraints were met.

Figure 7.2d shows the energy versus the different annealing times for the binary encoding model. For all annealing times, the minimum energy of the readout is equal to the optimal solution. The median decreases only slightly for larger annealing times. However, the first quartile goes down significantly when the annealing times become larger, suggesting improvement of the solutions for larger annealing times.

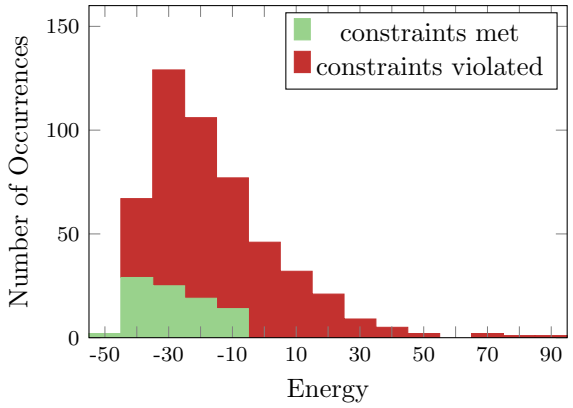
The energy histogram of the binary encoding model with the longest annealing time of 1998 μs is shown in Figure 7.2e. In most cases the constraints were met and again the lowest energies correspond with cases where all constraints were met.



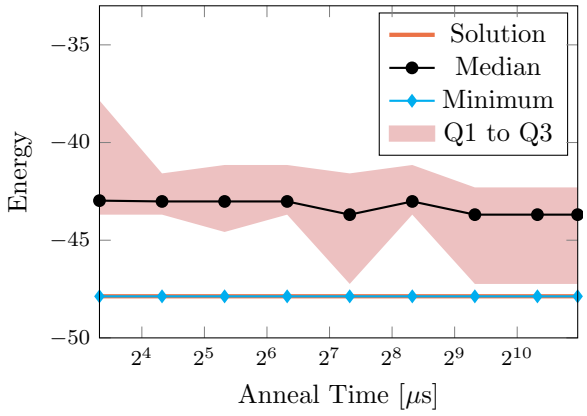
(a) Gather of the best solution at 1998 μs. It was found with both models.



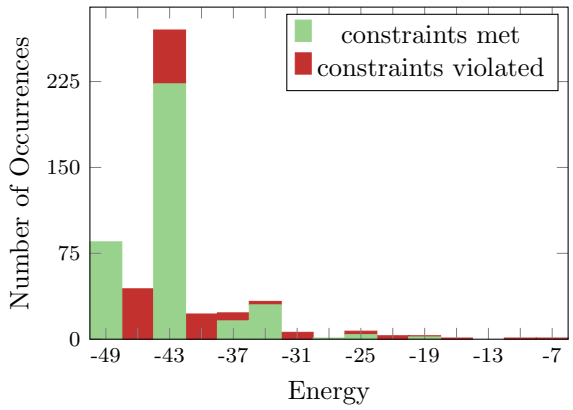
(b) Energy versus anneal time one-hot.



(c) Energy histogram at 1998 μs for one-hot.



(d) Energy versus anneal time binary.



(e) Energy histogram at 1998 μs for binary.

Figure 7.2: Results of Quantum Annealing with both the one-hot and binary encoding model.

Thus in Figure 7.2 it can be seen that in the case of 4 traces and 4 shifts the binary encoding model outperforms the one-hot encoding model. This is because the optimal solution was found at least once for every annealing time with the binary encoding model, as opposed to the one-hot encoding model, where it was not found at all for the annealing times of 10, 80, or 1280 μs. Furthermore, with the binary encoding model, the energy distribution starts to favour lower energy solutions for longer annealing times. This implies that we can increase the chance of finding the optimal solution by increasing the annealing time in the case of the binary encoding model. For the one-hot encoding model this relationship between the annealing time and the solution quality was not found. The energy distribution of the readouts did not change consistently for longer annealing times. Lastly, the one-hot encoding model violates more constraints compared to the binary encoding model, which makes interpreting results more difficult.

7.1.2 More Traces

Next, a more difficult problem with 16 traces is presented to examine if and how the results of the two models change. The gather of the 16 traces is shown in Figure 7.3. All traces have a temporal discretisation of 0.004 s, which is the same as in the previous section. However, the traces now contain many wavelets instead of just one. Every trace is an exact copy of the first trace, except for a relative shift, which are the statics we wish to estimate. The relative shifts with respect to the first trace are all either 0 s, 0.004 s, 0.008 s or 0.012 s. Therefore, the set S_K will be the same as in the previous section, i.e. $S_K = \{0,\ 0.004,\ 0.008,\ 0.012\}$. Thus for the binary quadratic models shown in this section we have $M = 16$ and $K = 4$. With S_K chosen, the binary quadratic model can be made for both the one-hot encoding and the binary encoding.

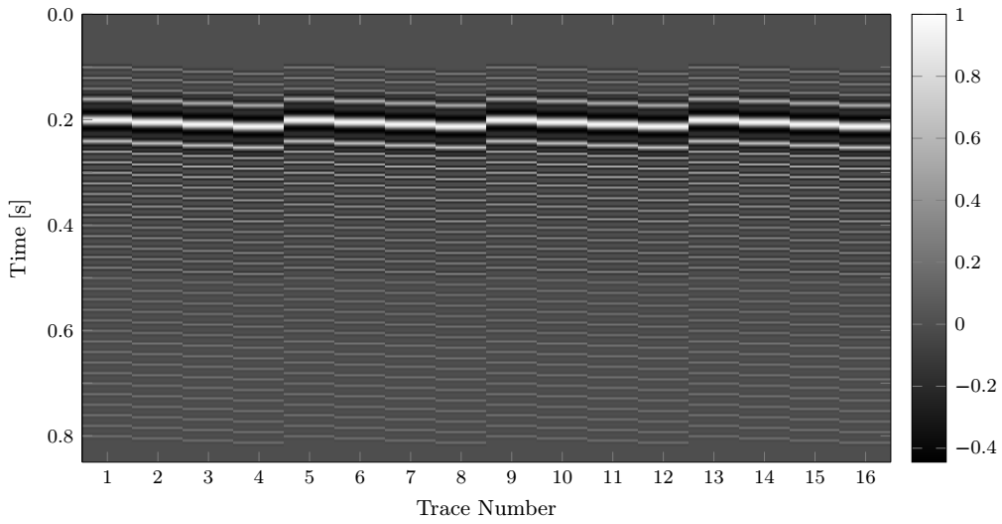
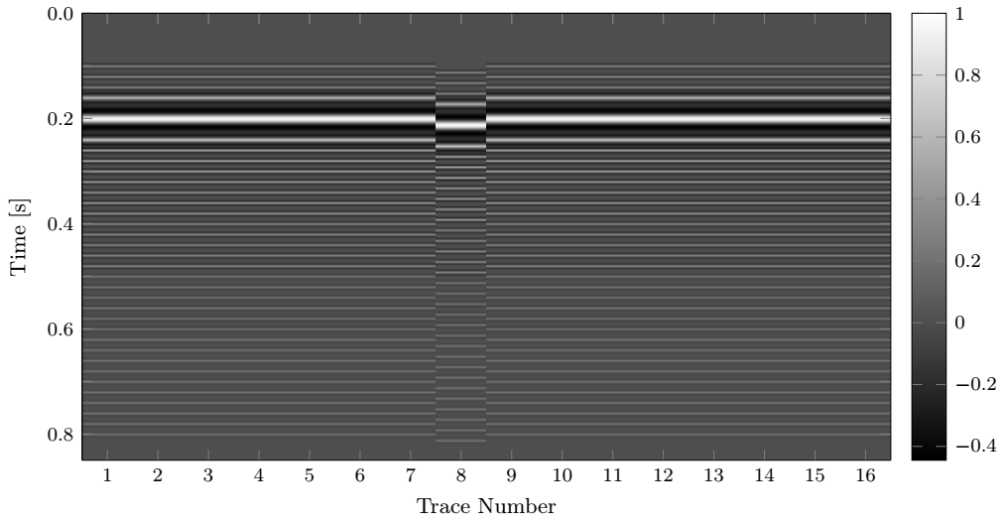
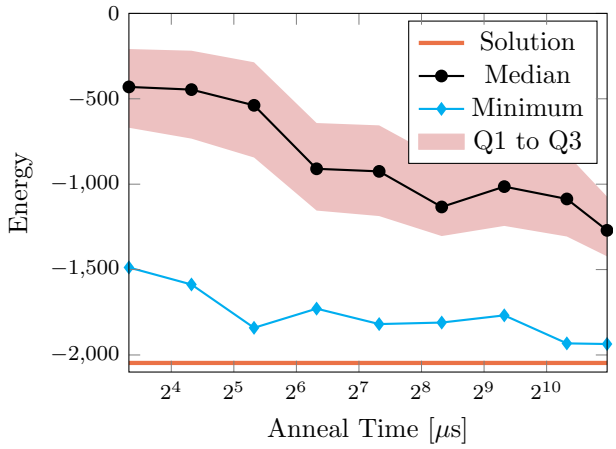


Figure 7.3: Gather of 16 traces with multiple wavelets. Each trace has a temporal discretisation of 0.004 s. With respect to the first trace, the shifts are 0 s, 0.004 s, 0.008 s or 0.012 s.

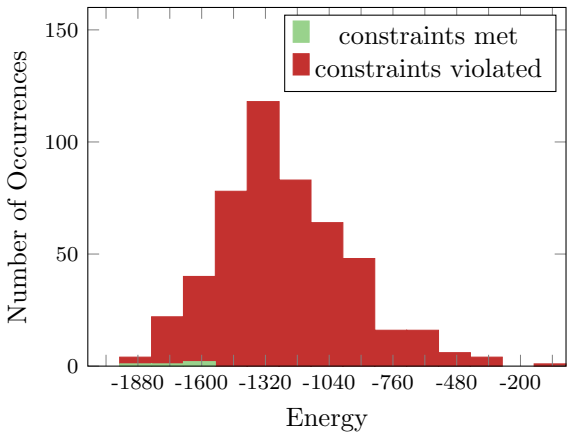
In the results of subsection 7.1.1, where 4 traces and 4 shifts were used, many readouts violated at least one constraint. When constraints are violated, the results are harder to interpret or sometimes not useful at all. Therefore, to ensure that more constraints are met, a post-processing step of steepest descent was added to the experiment of this section. Therefore, the discussion of the results has the following structure: First the results of the one-hot model without steepest descent are discussed. Next, results of the same readouts will be shown, but with the addition of the steepest descent post-processing step. Thereafter, the results are shown for the binary encoding model without steepest descent, followed by a discussing of the results where steepest descent was added in post-processing.



(a) Gather of the best solution at 1998 μs.



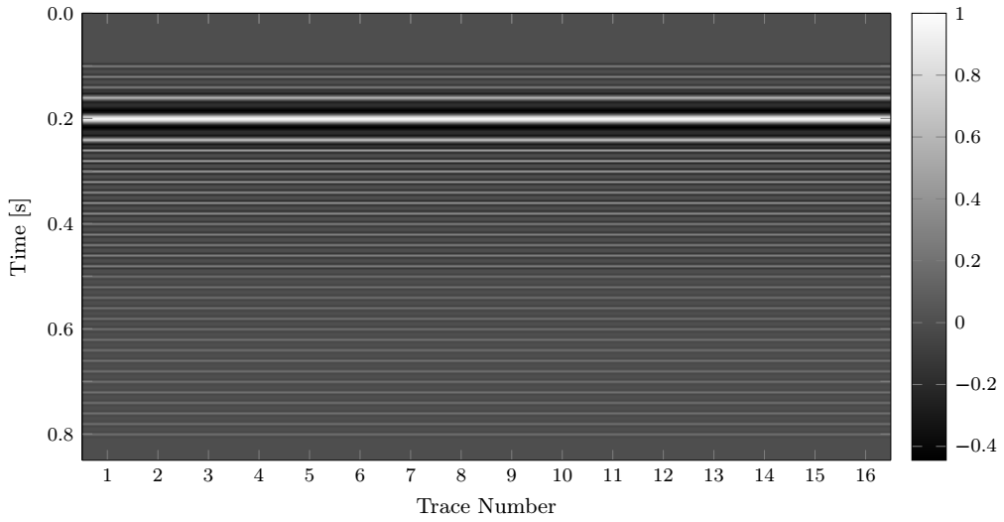
(b) Energy versus anneal time.



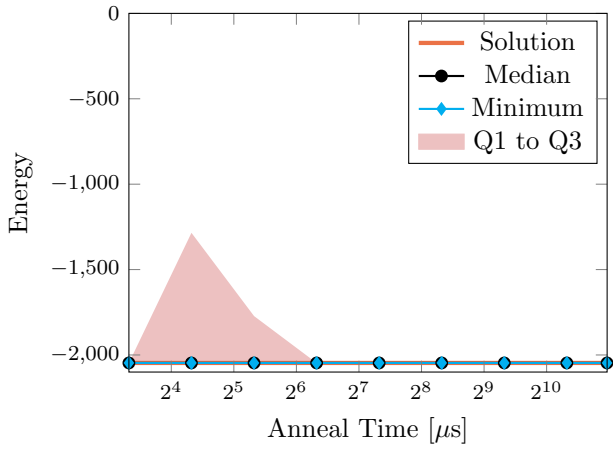
(c) Energy histogram at 1998 μs.

Figure 7.4: Results with the one-hot model *without* steepest descent in post-processing.

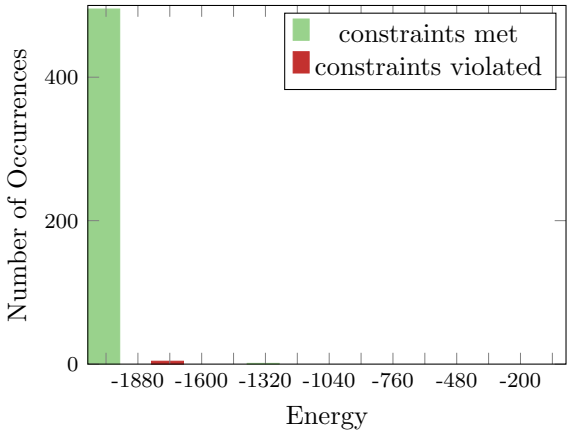
Figure 7.4 shows the results for the one-hot encoding model when no steepest descent was applied during post-processing. Figure 7.4a shows the best solution of all of the readouts. This solution was found with an annealing time of 1998 μs. Although most of the traces are aligned, the optimal solution is not found, as trace 8 is still misaligned. In Figure 7.4b, we can clearly see that the optimal solution is indeed not found as the minimum found energy does not reach the same energy as the optimal solution. The minimum found energy, as well as the median, first quartile and third quartile, all decrease in energy for larger annealing times. This means that we get a better energy distribution for larger annealing times. Lastly, Figure 7.4c shows the energy histogram. It can be seen that only a few single readouts meet the constraints. Therefore, a post-processing step of steepest descent should be very valuable in this case.



(a) Gather of the best solution at 1998 μs.



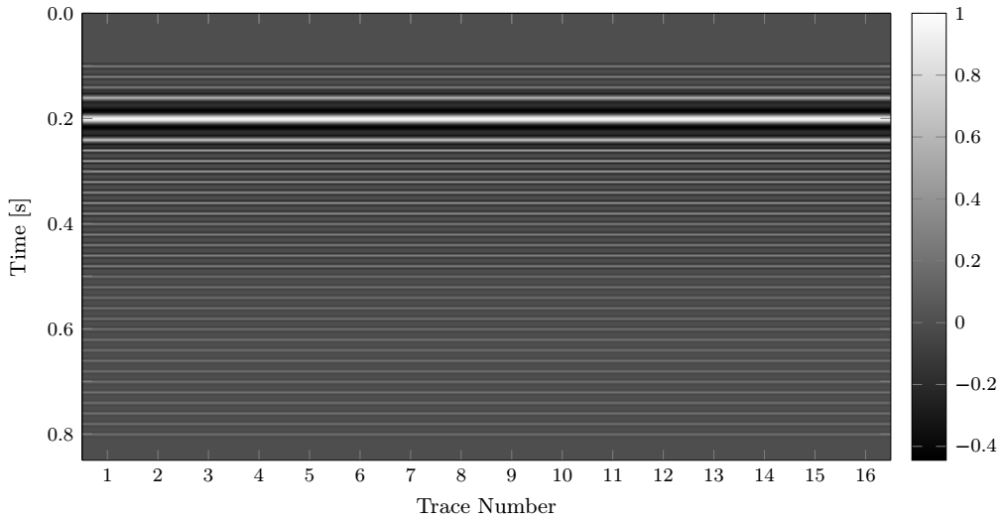
(b) Energy versus anneal time.



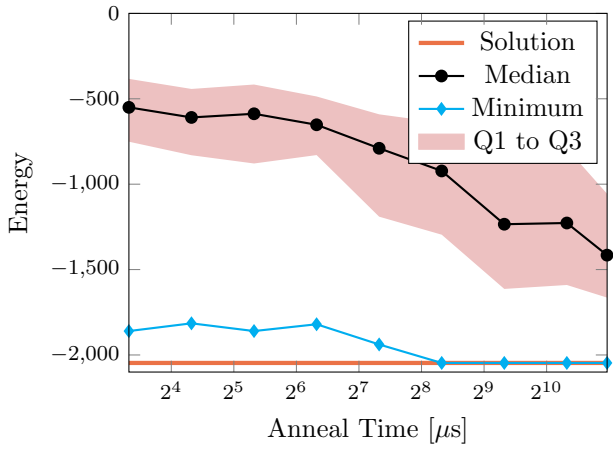
(c) Energy histogram at 1998 μs.

Figure 7.5: Results with the one-hot model *with* steepest descent in post-processing.

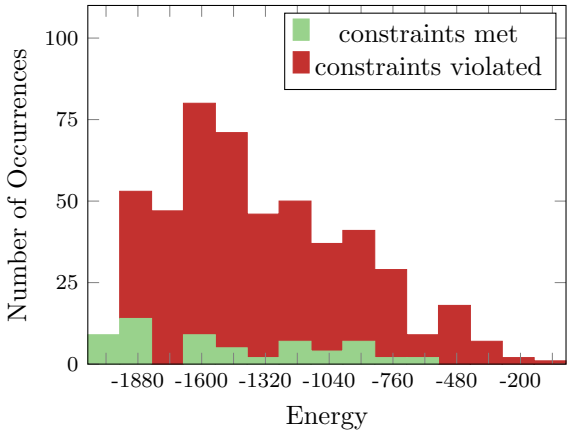
Next, steepest descent was applied to the previous readouts of which the results are shown in Figure 7.4. Figure 7.5a clearly shows that all traces are perfectly aligned and thus the optimal solution was found. Furthermore,Figure 7.5b shows that the optimal solution was found for all annealing times, as the cyan line overlaps with the solution line for all of the annealing times. Moreover, the median energy is also at the same level as the solution energy for all the different annealing times. Therefore, the optimal solution was found in at least 50 % of all the readouts. For annealing times larger or equal to 80 μs, the third quartile also meets the solution time and thus in these cases 75 % of all the readouts are the optimal solution. Figure 7.5c shows the energy histogram at an annealing time of 1998 μ, in which it can be seen that almost no constraints were violated.



(a) Gather of the best solution at 1998 μs.



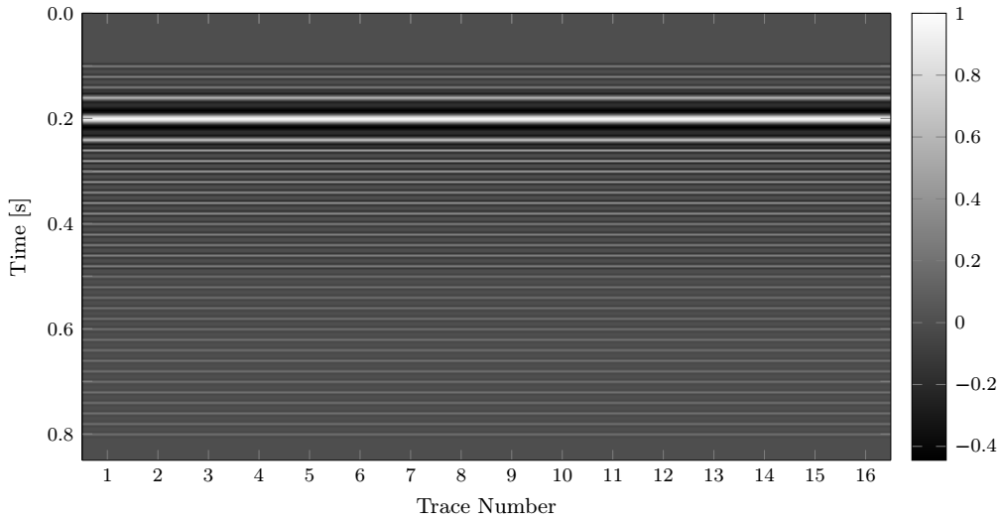
(b) Energy versus anneal time.



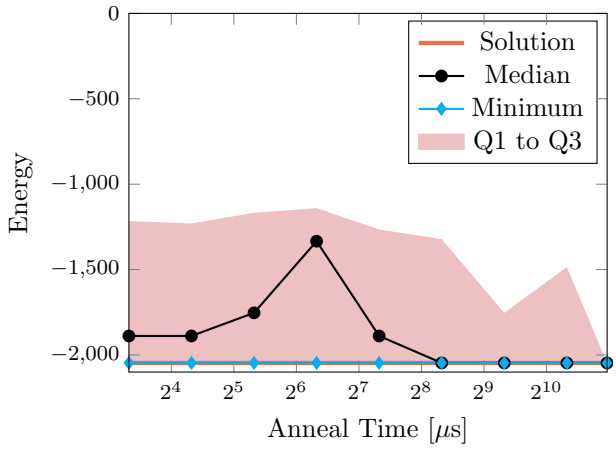
(c) Energy histogram at 1998 μs.

Figure 7.6: Results with the binary model *without* steepest descent in post-processing.

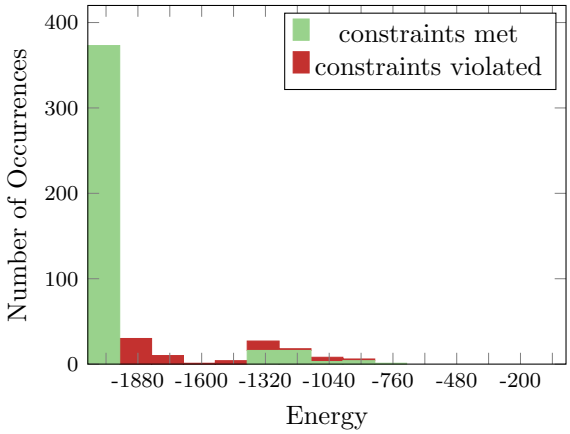
The same traces and shifts were also solved using the binary encoding model. Figure 7.6 shows the results without the addition of the steepest descent post-processing step. Figure 7.6a shows perfect alignment, thus the optimal solution was found. Furthermore, in Figure 7.6b we can see that the optimal solution was found for the longer annealing times of 320 μs and longer. Moreover, the minimum, median, first quartile and third quartile all decrease in energy for longer annealing times. Hence, longer annealing times give better results in this experiment. The energy histogram inFigure 7.6c shows that for most of the readouts the constraints were violated. However, compared to the one-hot model *without* steepest descent, a lot more readouts did not violate any constraints.



(a) Gather of the best solution at 1998 μs.



(b) Energy versus anneal time.



(c) Energy histogram at 1998 μs.

Figure 7.7: Results with the binary model *with* steepest descent in post-processing.

For the binary encoding model a post-processing step of steepest descent was added as well. The results of these experiments are shown in Figure 7.7. Since the optimal solution was already found without steepest descent, the perfect solution is also found when steepest descent is added to the post processing. This can be seen by the perfect alignment shown in Figure 7.7a. Furthermore, Figure 7.7b shows that the lowest found energy coincides with the solution for all the annealing times, showing considerable improvement. The median showed unexpected behaviour, as it first increased in energy, but decreased again with longer annealing times. We are unsure where this behaviour stems from. However, possible explanations are that this behaviour is either caused by the probabilistic nature of the quantum annealing algorithm or by faulty hardware. Nonetheless, for larger annealing times the median consistently remained at the energy of the optimal solution. Therefore, at least 50 % of the readouts consist of the optimal solution for annealing times larger or equal to 1280 μs. Furthermore, the third quartile goes down for larger annealing times, showing that the results increase for longer annealing times. Figure 7.7c shows that when steepest descent is applied, the constraints are met for almost all of the readouts.

These results show that the binary encoding model outperforms the one-hot encoding model when no steepest descent is applied in post-processing. The binary encoding model finds the optimal solution multiple times, whereas the one-hot encoding model does not find it at all. Furthermore, as most readouts of the one-hot encoding model had one or more constraints violated, the interpretation of the results is rather difficult. The situation is completely reverse when steepest descent was applied on the same readouts. In these experiments, the one-hot encoding model clearly outperforms the binary encoding model, because for all the annealing times more then 50 % of the readouts were the optimal solution, which only happens for the annealing times of 320 μs or more with the binary encoding model. In addition, with an annealing time of 1998 μs 495 out of the 500 reads found the optimal solution, whereas this was only 358 out of 500 for the binary encoding model.

7.1.3 More Shifts

In this section a problem with 4 traces, but a larger set S_K will be shown and discussed to examine if and how the behaviour of both the models change when there are more shifts possible. Figure 7.8 shows the gather of the 4 traces that were used for this experiment. The traces are the same as the traces of subsection 7.1.2, except for a relative shift. Relative to the first trace, the second, third and fourth trace are shifted by 0.012, 0.036 and 0.06 s respectively. The number of variables used for both of the models is roughly of the size MK. To keep the number of variables the same as in subsection 7.1.2, a set S_K will be chosen such that the size K will be exactly 16. Furthermore, S_K needs to be chosen in such a way that there is only one optimal solution. The chosen set was $S_K = \{0, 0.004, 0.008, 0.012, 0.016, 0.02, 0.024, 0.028, 0.032, 0.036, 0.04, 0.044, 0.048, 0.052, 0.056, 0.06\}$. As in subsection 7.1.2, steepest descent will be applied in the post processing step and the same structure of the discussion will be used.

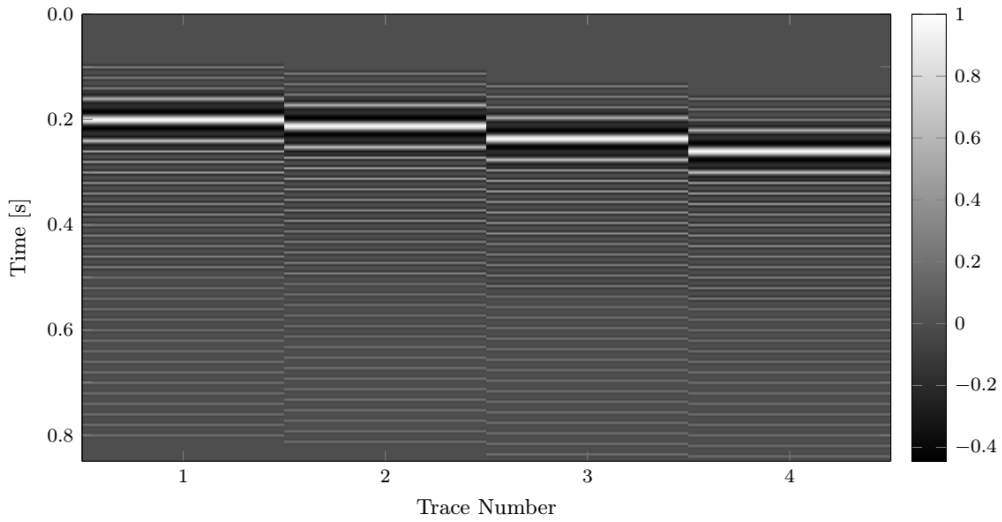
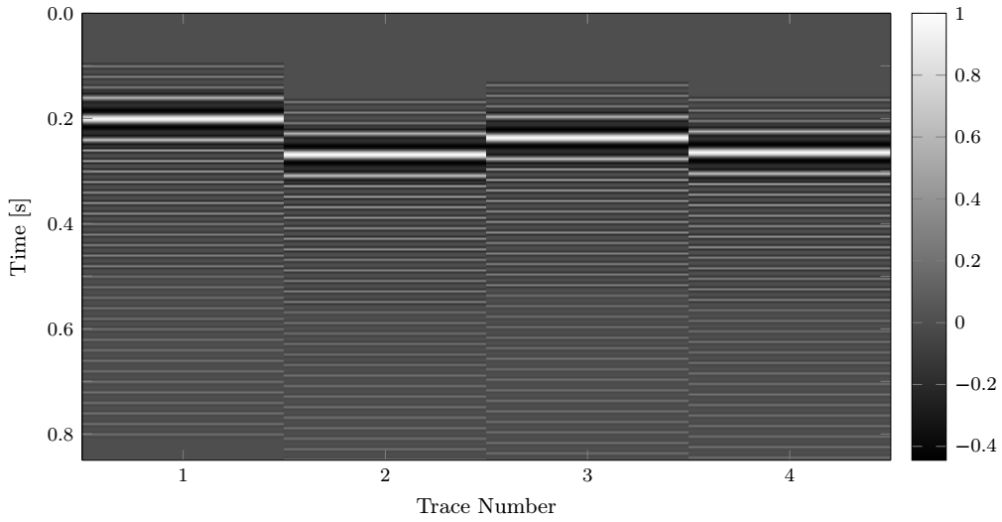
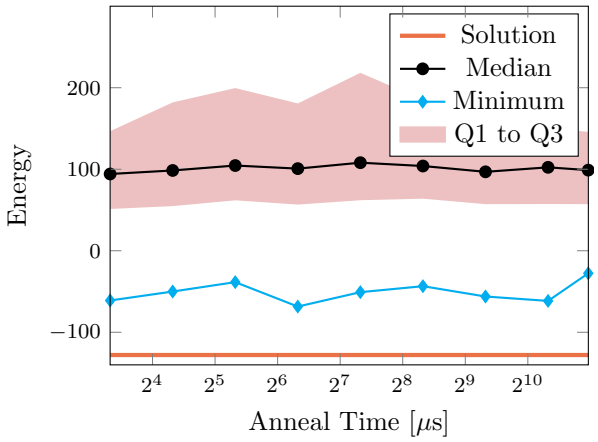


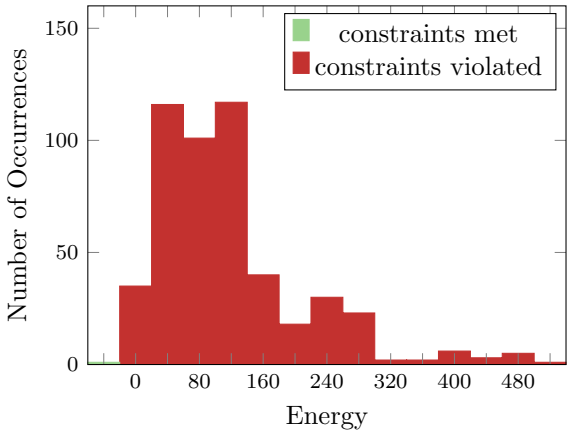
Figure 7.8: Gather of 4 traces with multiple wavelets. Each trace has a temporal discretisation of 0.004 s. With respect to the first trace, the shifts are 0.004 s, 0.008 s and 0.012 s for the second, third and fourth trace respectively.



(a) Gather of the best solution at 1998 μs.



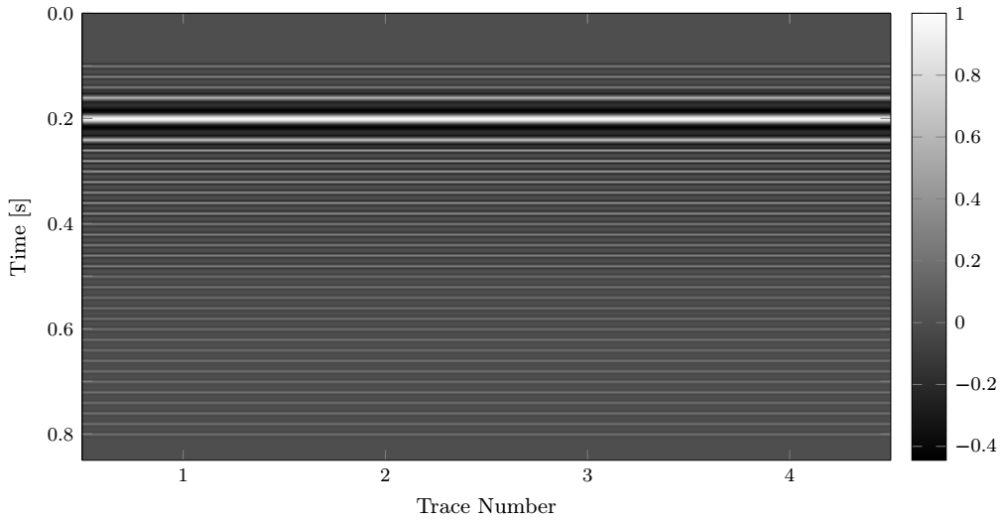
(b) Energy versus anneal time.



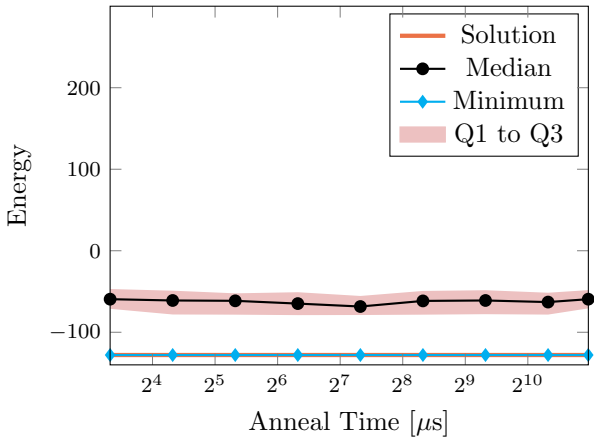
(c) Energy histogram at 1998 μs.

Figure 7.9: Results with the one-hot model *without* steepest descent in post-processing.

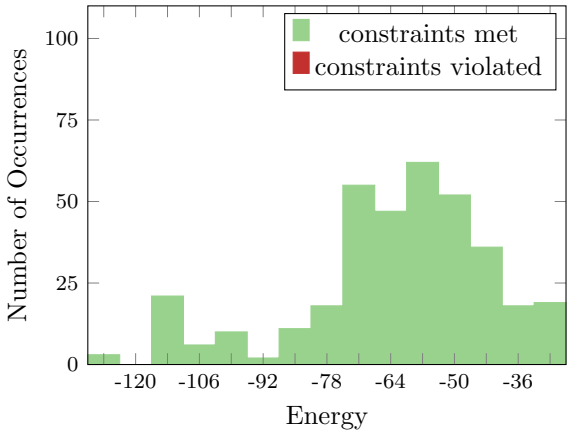
In Figure 7.9 the results of the one-hot model *without* steepest descent applied in the post-processing step are shown. The gather of the best found statics are shown in Figure 7.9a. This happened at 1998 μs and as can be seen the solution was not found. Although trace 2 and 4 are almost aligned, the other traces are not aligned at all. Figure 7.9b also shows that the optimal solution was not found, as the minimum found energy does not reach the energy of the optimal solution. Furthermore, the minimum, median, first quartile and third quartile do not convincingly decrease for longer annealing times. Therefore, the quality of the results does not improve with longer annealing times, as was also the case with the results of subsection 7.1.1. It must be noted that even though the found minimum energy was lower for all other annealing times than 1998 μs, all of these solutions violated at least one constraint. Hence, the gather shown in Figure 7.9a is the lowest energy solution that meets the constraints, even though other solutions with lower energies were found. This is not surprising when we take a closer look at the energy distribution shown in Figure 7.9c. Almost all readouts violate at least one constraint. In fact, the solution shown in Figure 7.9a, shows the only readout where all constraints are met.



(a) Gather of the best solution at 1998 μs.



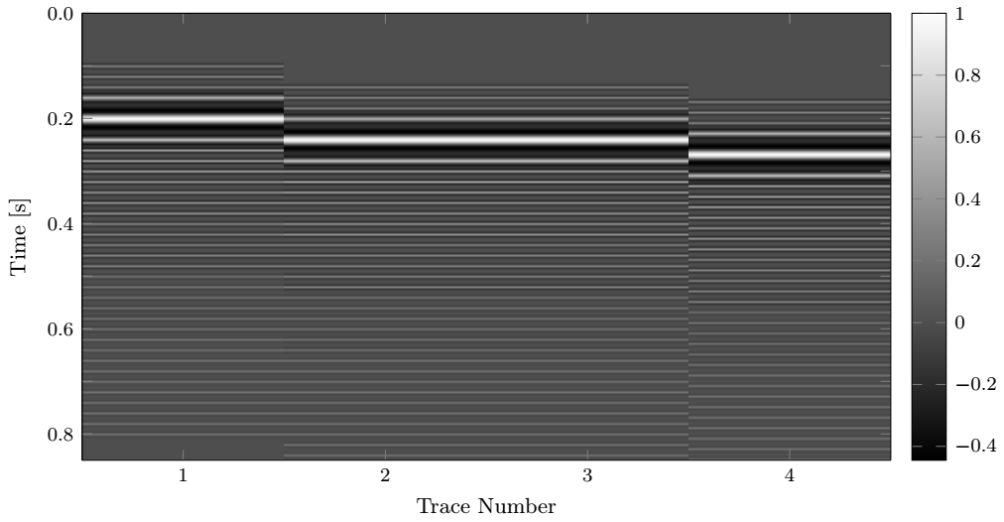
(b) Energy versus anneal time.



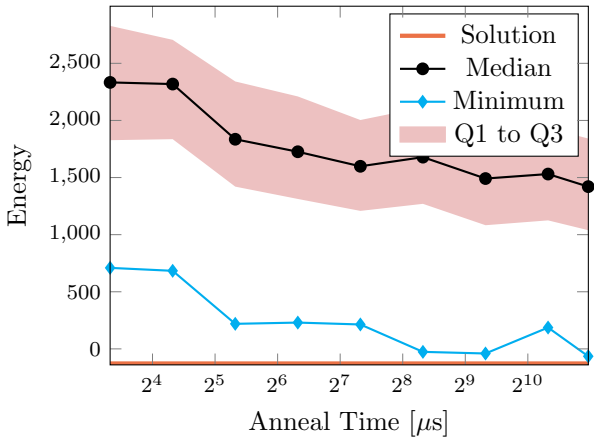
(c) Energy histogram at 1998 μs.

Figure 7.10: Results with the one-hot model *with* steepest descent in post-processing.

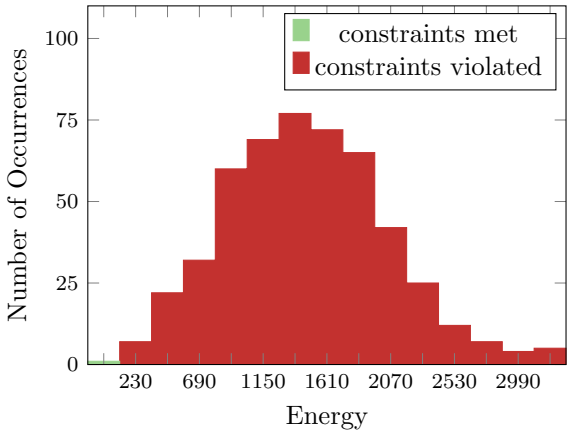
Next, steepest descent was applied to the previous readouts and the results are shown in Figure 7.10. The best gather at an annealing time of 1998 μs is shown in Figure 7.10a. Clearly, all the traces are perfectly aligned. Therefore, we can infer that the optimal solution was found. Figure 7.10b shows that for all of the annealing times, the minimum found energy corresponds to the optimal solutions. Hence, the optimal solution was found for all annealing times. After applying steepest descent, the minimum, median first quartile and third quartile did not change significantly for different annealing times. Longer annealing times did thus not improve the solution quality for this combination of traces and shifts. Figure 7.10 shows the energy histogram at 1998 μs. There are no readouts that violate constraints and we can see that the optimal solution was found 3 times. This is considerably better than the results without the use of a steepest descent post-processing step.



(a) Gather of the best solution at 1998 μs.



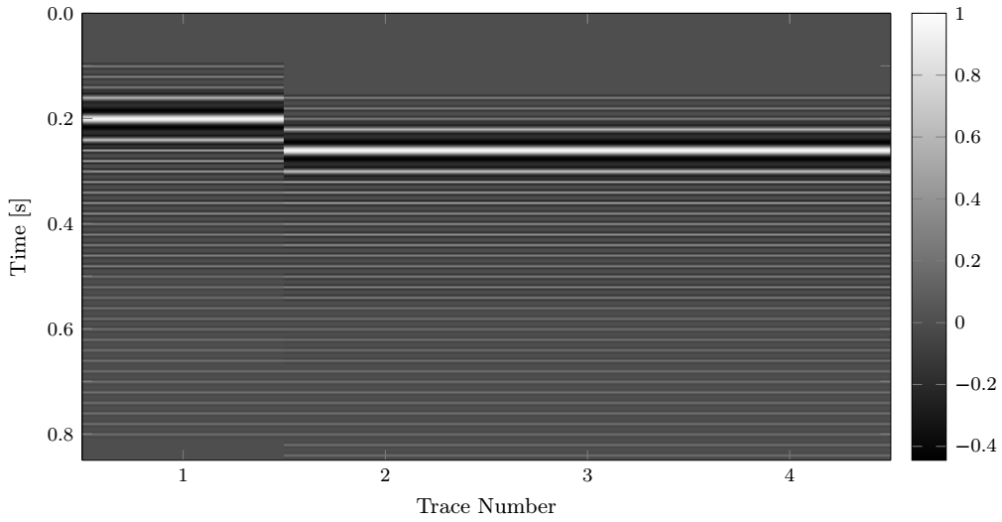
(b) Energy versus anneal time.



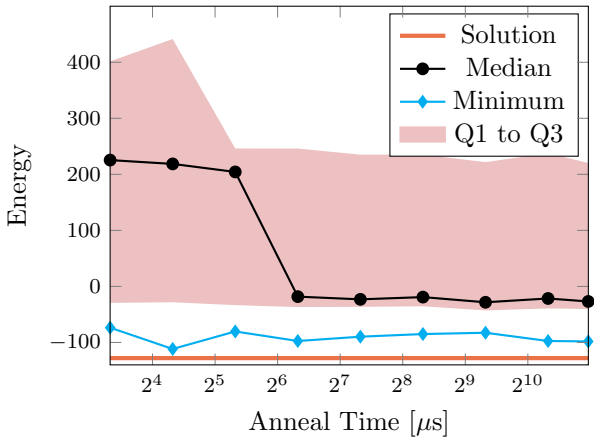
(c) Energy histogram at 1998 μs.

Figure 7.11: Results with the binary model *without* steepest descent in post-processing.

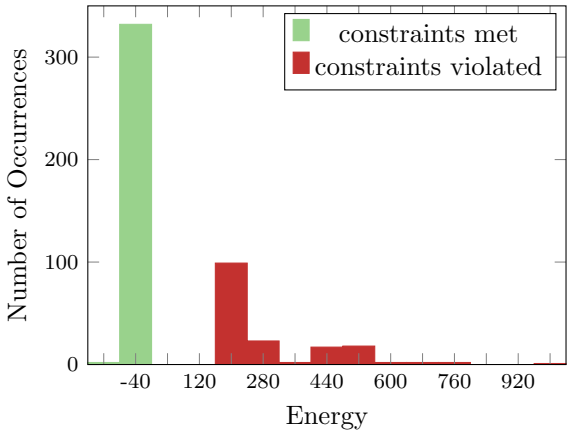
Figure 7.11 shows the results when the binary encoding model was used, but no steepest descent was applied in post-processing. Figure 7.11a shows the best found gather at an annealing time of 1998 μs. Trace 2 and 3 are perfectly aligned. Furthermore, the small wavelets of trace 1 are aligned with the small wavelets of trace 2 and 3. These are the thin white lines in the figure. While the alignment is better as compared to the one-hot model without steepest descent, it is still far off from being the optimal solution. The energy distribution for different annealing times is visualised in Figure 7.11b. Here we can confirm that the solution was not found as the minimum energy found did not reach the energy of the optimal solution. However, as opposed to the one-hot encoding model, we see that the minimum, median, first quartile and third quartile lower in energy as the annealing time increases. This suggests that the solution quality increases with the annealing time in this experiment. The energy for an annealing time of 1998 μs is further visualised with the energy histogram in Figure 7.11c. Almost all the readouts violate at least one constraint. In fact, the solution shown in Figure 7.11a is the only readout which does not violate a constraint.



(a) Gather of the best solution at 1998 μs.



(b) Energy versus anneal time.



(c) Energy histogram at 1998 μs.

Figure 7.12: Results with the binary model *with* steepest descent in post-processing.

Steepest descent was also applied to the readouts of the binary encoding model, of which the results are shown in Figure 7.12. The gather of the best solution is shown in Figure 7.12a. Trace 2,3 and 4 are perfectly aligned. Furthermore, the small wavelets of trace 1 are aligned with the small wavelets of the other traces. Nonetheless, this is still not the optimal solution. Figure 7.12b confirms that the optimal solution was not found for any annealing time, as the minimum found energy does not reach the solution energy for any of the annealing times. The energy distribution steadily improved as the annealing times increased when no steepest descent was applied. When steepest descent was applied, there was a sudden jump in the energy after which no significant improvement can be observed. The energy histogram in Figure 7.12c shows that roughly half of the readouts do meet all the constraints.

From the results in this section we can conclude that the binary encoding model outperforms the one-hot encoding model when steepest descent was not added in the post-processing step. This is illustrated by the fact that better solutions are found with the binary encoding model and the fact that the solution quality improves when the annealing time increases. However, when the steepest descent is added in post-processing the one-hot encoding model outperforms the binary encoding model. It finds the optimal solution for all annealing times, whereas the binary encoding model does not find the optimal solution at all. For both the models it also observed that an increase in the number of traces M is easier to solve then an increase of the number of allowed shifts K.

7.2 The D-Wave Hybrid Solver

In this section the statics of a larger gather with 108 traces will be estimated. To this end the D-Wave Hybrid solver will be used as these problems are too large to run directly on the quantum annealers. Only results of the one-hot encoding will be discussed, as a number of problems arose with the binary encoding model for this experiment. First of all, the pre-processing steps took significantly longer compared to the one-hot encoding model. For the one-hot encoding model, the pre-processing steps took only a couple of seconds, whereas for the binary encoding model this took roughly half an hour, most of the time was used for the quadratisation of the higher order model. This is unacceptable, as the optimisation procedure with the Hybrid solver itself only took 10 seconds. Furthermore, the long pre-processing time made it very challenging to determine a suitable weight for the penalty term needed for the quadratisation. Consequently, when the Hybrid solver was used with the binary model, the output gathers that looked random and had smaller stack power compared to the input. This implies that the output was considerably worse than the input.

7.2.1 The Arid Model

Producing realistic synthetic data with this great amount of traces is a challenging task. Therefore, an outside source was used to gather realistic synthetic data to test the Hybrid solver. The Arid Model [45] is a large and realistic synthetic model that is widely used for subsurface imaging. A figure of the model is shown in Figure 7.13. In some parts of the model, large near-surface anomalies are present [45]. These anomalies cause large residual statics and thus make signal processing a challenging task. Because these large residual statics are exactly what we try to estimate, a selection of traces from the Arid model were used to put the Hybrid solver to the test. Before continuing with the results, an important note must be made with regards to the traces. Contrary to our more theoretical synthetic data of the previous sections, the Arid model gives a more realistic idea of the inconsistent nature of the traces one would find in field experiments. Therefore, the exact optimal solution is harder to distinguish as it is not a pixel for pixel alignment like we saw before.

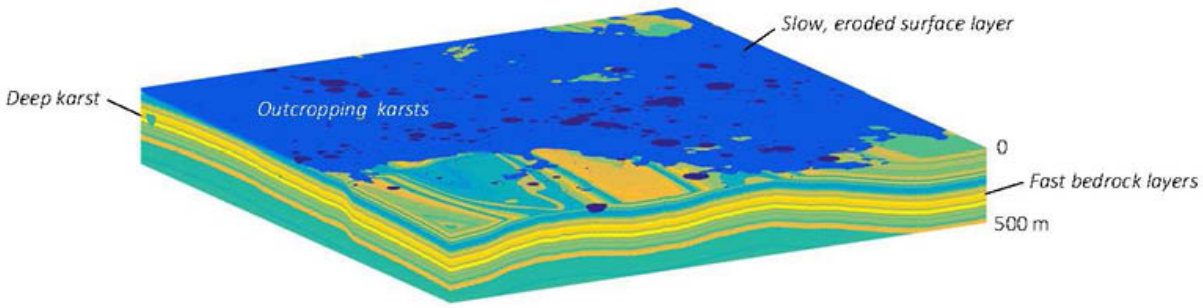


Figure 7.13: 3D View of the Arid Model [45]

From the Arid model a gather of 108 traces, shown in Figure 7.14, was selected to be optimised by the

D-Wave Hybrid Solver using the one-hot encoding model. All the traces consist of 690 datapoints and have a temporal discretisation of 0.006 s. The gather shows significant misalignment between the traces.

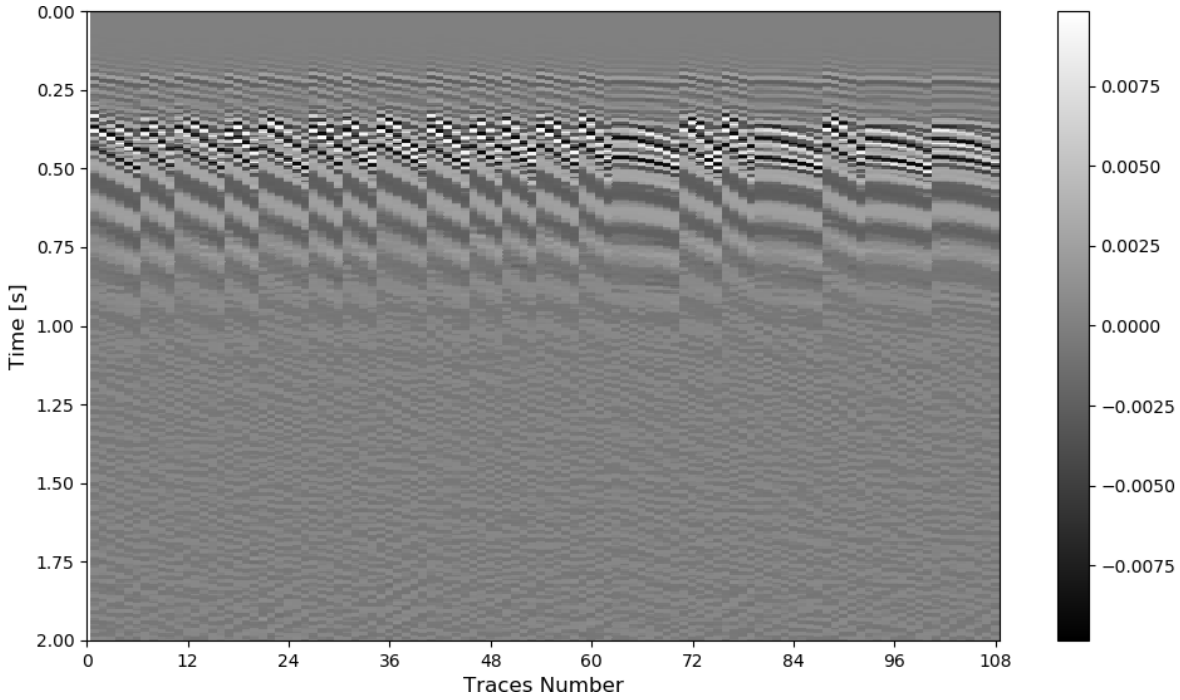


Figure 7.14: Gather of 108 misaligned traces from the Arid model containing 690 datapoints.

7.2.2 Deterministic Corrected Traces

First the results of an industry standard deterministic cross-correlation solver will be discussed. For this purpose, a single iteration of Algorithm 1 was used, where the reference trace was the average of all the traces. Figure 7.15 shows the gather of the corrected traces. The gather shows much more alignment compared to the input and therefore the quality of the gather improved significantly. However, some traces are still misaligned. Nonetheless, this is an impressive result considering the simplicity of the algorithm used and the size of the data set. The most clear misalignment is noticeable at trace 1 and 2. However, some of the other traces also show some slight misalignments. For example, trace 53 and 92 show some minor misalignments with the rest of the gather.

7.2.3 Hybrid Corrected Traces

The same gather was also given to the D-Wave Hybrid solver. In this case a K of 15 was chosen and $S_K = \{0,\ 0.006,\ 0.012,\ 0.018,\ 0.024,\ 0.03,\ 0.036,\ 0.042,\ 0.048,\ 0.054,\ 0.06,\ 0.066,\ 0.072,\ 0.078,\ 0.084\}$. The penalty term was determined by performing quick runs with simulated annealing until a suitable value was found. The gather shown in Figure 7.16 shows almost perfect alignment, indicating that the solution is at the optimal solution or very close to it. Furthermore, no large misalignments are visible, indicating that the solution is better than the solution found by the deterministic cross-correlation solver. Lastly, the stack power of the gather found with the 10.7 % larger compared to the stack power found with the deterministic solver. This indeed confirms the observation that the gather found with the Hybrid solver is of higher quality compared to the gather of the deterministic solver.

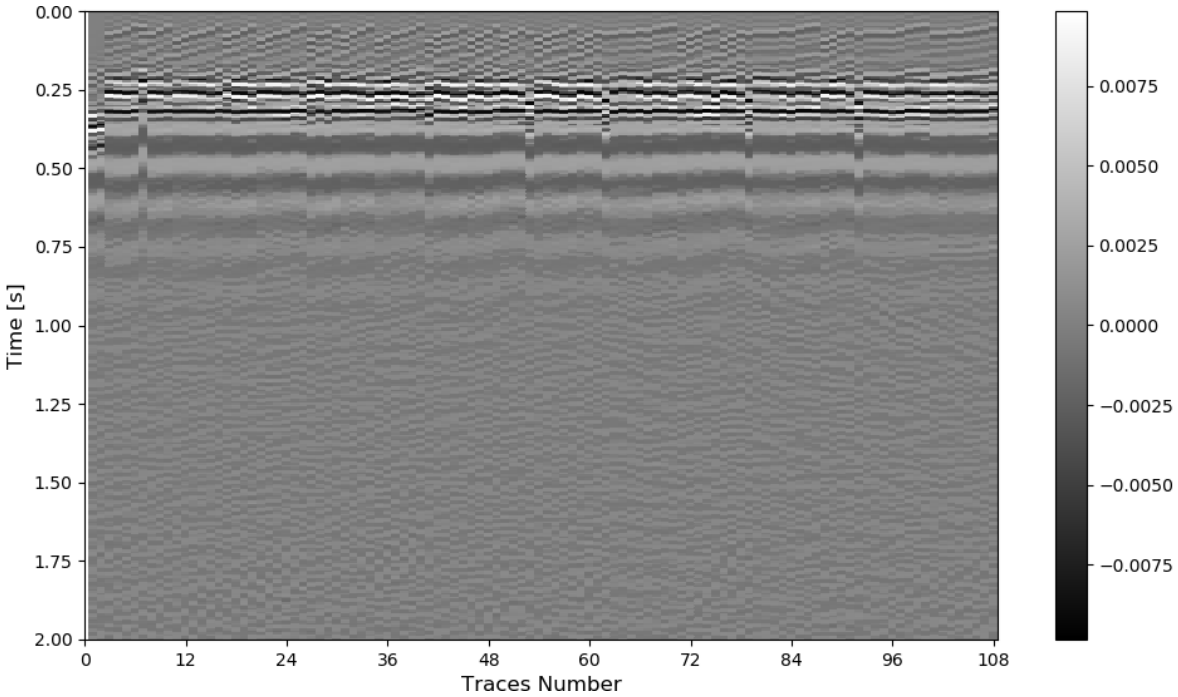


Figure 7.15: Gather of the 108 traces from Figure 7.14, which have been corrected using an industry standard deterministic cross-correlation solver.

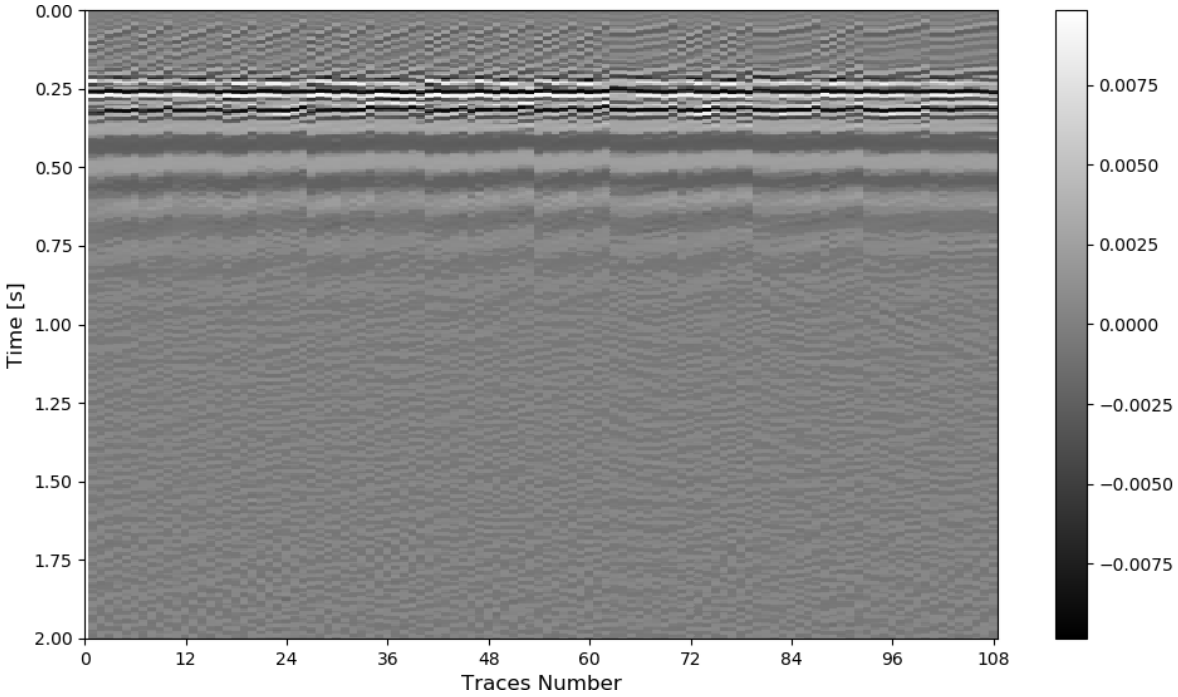


Figure 7.16: Gather of the 108 traces from Figure 7.14, which have been corrected by using the one-hot encoding model together with the D-Wave Hybrid Solver.

8 Conclusion and Outlook

Residual statics estimation is a subproblem of geophysical subsurface imaging. With residuals statics estimation the negative effects of large near-surface anomalies are estimated and corrected to produce a subsurface image which is of higher resolution. These higher resolution images are of importance for a wide variety of applications and therefore much research has been done to effectively estimate these residual statics with classical computers. However, these methods have many shortcomings that can deem them ineffective. For example, solvers can get stuck in a local minimum of a multimodal objective, the solver parameters might be difficult to estimate or the method is simply too computationally expensive. With quantum annealing we hope to overcome these problems.

8.1 Conclusion

In this thesis it is shown that the residual statics estimation problem can be solved using a quantum annealer and a Hybrid solver. First, a binary quadratic model describing the residual statics estimation needed to be formulated. For this purpose two different binary quadratic models were developed: (1) the one-hot encoding model and (2) the binary encoding model. A summary of some of their properties is given below. Some of the results from the one-hot encoding model are not usable, whereas the binary encoding model always gives a meaningful result. The pre-processing of the one-hot encoding model is relatively easy to implement and low in computational complexity, whereas the computational complexity of the binary encoding model is much larger. Moreover, the pre-processing costs of the binary encoding model are so huge, that its current implementation cannot be used for problems with more than 100 traces. No significant difference was found in the scaling of the number of variables before embedding with empirical testing. However, with a smaller number of allowed shifts, the embedding of the binary encoding model is much smaller, resulting in the ability to solve problems with a larger amount of traces directly on the quantum annealer.

When working directly on the quantum annealer with no additional post-processing steps, the binary encoding model outperforms the one-hot encoding model. The binary encoding model finds the optimal solution more often and improves the energy distribution as the annealing time increases. However, when steepest descent is applied in post-processing, the one-hot encoding model clearly outperforms the binary encoding model. In all of the experiments the one-hot encoding model with steepest descent found the optimal solution regardless of the annealing time. Furthermore, it is also observed that an increase in the number of allowed shifts results in a more difficult problem than an increase in the number of traces, even when the binary search space in approximately the same.

As a final test, the Arid model was used to test a large and realistically complex problem using the D-Wave Hybrid Solver. The pre-processing cost of the binary encoding model proved too large to efficiently estimate the residual statics. However, the one-hot encoding model found very good residual statics, even outperforming the current industry standard deterministic cross-correlation solver.

8.2 Outlook

The weight of the penalty term is by far the most influential parameter to determine. When it is too low, the binary quadratic models do not correspond to the original residual statics estimation problem anymore. On the other hand, when the weight is set too high, the quantum annealing algorithm performs poorly and the optimal solution is not found. For this research, the weight of the penalty term was determined using some quick runs of simulated annealing and comparing the results for different penalty weights. However, a more robust, less subjective and less labour intensive method would be preferred. In general there is no known way to efficiently calculate the optimal weight of the penalty term, except for the general rule of thumb that it should be in the order of magnitude of the absolute value of the largest coefficient in the binary quadratic model. However, during the course of this MSc research project, we observed that there seems to be a correlation between the frequency of the wavelets and the size of the penalty terms. Smaller frequencies seem to correlate with larger penalty terms. More research is needed to confirm this observation. When this observation is confirmed, the preprocessing will be less labour-intensive and more can be done more subjectively. This would be a very promising development.

The maximum problem size than can be solved directly on the current state of the art quantum annealers was estimated in Section 6.3 . However, when an extra pre-processing step is added to find the roof duality, the maximum problem size could be larger. The roof dual of a quadratic pseudo-Boolean function $f : \mathbb{B}^n \to \mathbb{R}$ is linear pseudo-Boolean function $l : \mathbb{B}^n \to \mathbb{R}$ which majorises f in every point of $\mathbb{B}^n$ [46]. Roof duality has many applications, but as pointed out in [47], the most important are

1. Possibly dividing the binary quadratic model into multiple independent subproblems.
2. Possibly finding variables that can remain fixed at either 0 or 1.

Both can significantly reduce the number of variables in the binary quadratic model and thus allow for larger problems to run directly on the quantum annealers. However, finding the roof dual of the binary quadratic models is a computationally intensive task with a computational complexity of $\mathcal{O}(M^3K^3)$ [48].

In this thesis, the one-hot encoding and binary encoding were used to encode different combinations. A recent study [49] suggests that a different encoding can increase the solution quality. The so called *domain-wall encoding* was first introduced by Chancellor [50]. Next to the possible increase of solution quality, it also offers benefits in terms of the number of variables and number of quadratic terms. Whereas the one-hot encoding model has MK variables and $\frac{1}{2}MK(MK-1)$ quadratic terms, the domain-wall encoding would have $M(K-1)$ variables and $\frac{1}{2}M(K-1)(M(K-1)-1-\frac{K}{M})$ quadratic terms [50]. Hence, using the domain-wall encoding would allow for the computation of even larger problem sizes, because of the lower number of variables and quadratic terms. This new type of encoding could prove useful and is worth exploring.

In [51] it is shown how an initial solution can be encoded in the annealing scheme of the D-Wave quantum annealers. When residual statics are large, initial solutions to the original residual static problems are often of poor quality [15]. However, for the one-hot encoding model an initial guess of all zeros could be useful, as it is always M bit flips away from the optimal solution. Hence, the annealing procedure is started relatively close to the optimal solution. Therefore this is also an interesting idea to explore.

Bibliography

[1] P. Benioff. The computer as a physical system: A microscopic quantum mechanical hamiltonian model of computers as represented by turing machines. *Journal of statistical physics*, 22(5):563–591, 1980.

[2] N.D. Mermin. *Quantum computer science: an introduction*. Cambridge University Press, 2007.

[3] C. McGeoch. Adiabatic quantum computation and quantum annealing: Theory and practice. *Synthesis Lectures on Quantum Computing*, 5(2):1–93, 2014.

[4] S.E. Venegas-Andraca, W. Cruz-Santos, C. McGeoch, and M. Lanzagorta. A cross-disciplinary introduction to quantum annealing-based algorithms. *Contemporary Physics*, 59(2):174–197, 2018.

[5] M. Borowski, P. Gora, K. Karnas, M. Błajda, K. Król, A. Matyjasek, D. Burczyk, M. Szewczyk, and M. Kutwin. New hybrid quantum annealing algorithms for solving vehicle routing problem. In *International Conference on Computational Science*, pages 546–561. Springer, 2020.

[6] F. Phillipson and I. Chiscop. Multimodal container planning: A qubo formulation and implementation on a quantum annealer. In *International Conference on Computational Science*, pages 30–44. Springer, 2021.

[7] S. Yarkoni, A. Alekseyenko, M. Streif, D. Von Dollen, F. Neukart, and T. Bäck. Multi-car paint shop optimization with quantum annealing. *arXiv preprint arXiv:2109.07876*, 2021.

[8] V.K. Mulligan, H. Melo, H.I. Merritt, S. Slocum, B.D. Weitzner, A.M. Watkins, P.D. Renfrew, C. Pelissier, P.S. Arora, and R. Bonneau. Designing peptides on a quantum computer. *bioRxiv*, page 752485, 2020.

[9] K. Mato, R. Mengoni, D. Ottaviani, and G. Palermo. Quantum molecular unfolding. *arXiv preprint arXiv:2107.13607*, 2021.

[10] F. Phillipson, R.S. Wezeman, and I. Chiscop. Indoor–outdoor detection in mobile networks using quantum machine learning approaches. *Computers*, 10(6), 2021.

[11] S.H. Adachi and M.P. Henderson. Application of quantum annealing to training of deep neural networks. *arXiv preprint arXiv:1510.06356*, 2015.

[12] A.M. Souza, L.J.L. Cirto, E.O. Martins, N.L. Hollanda, I. Roditi, M.D. Correia, N. Sá, R.S. Sarthour, and I.S. Oliveira. An application of quantum annealing computing to seismic inversion. *arXiv preprint arXiv:2005.02846*, 2020.

[13] S. Greer and D. O'Malley. An approach to seismic inversion with quantum annealing. In *SEG Technical Program Expanded Abstracts 2020*, pages 2845–2849. Society of Exploration Geophysicists, 2020.

[14] D. O'Malley. An approach to quantum-computational hydrologic inverse analysis. *Scientific reports*, 8(1):1–9, 2018.

[15] D.H. Rothman. *Large near-surface anomalies, seismic reflection data, and simulated annealing.* PhD thesis, Stanford University, 1985.

[16] J. Ronen and J. Claerbout. by stack-power maximization. *Geophysics*, 50(12), 1985.

[17] S. Pierini, M. Aleardi, and A. Sajeva. Comparisons of recent global optimization algorithms: Tests on analytic objective functions and residual statics corrections. In *81st EAGE Conference and Exhibition 2019*, pages 1–5. European Association of Geoscientists & Engineers, 2019.

[18] Clarity: a roadmap for the future of quantum computing, 2021. `https://www.dwavesys.com/media/xvjpraig/clarity-roadmap_digital_v2.pdf`.

[19] C. McGeoch, P. Farré, and W. Bernoudy. D-wave hybrid solver service + advantage: Technology update. Technical Report 14-1041A-A, D-Wave: The Quantum Computing Company, 2650 E Bayshore Rd, Palo Alto, September 2020.

[20] I.F. Jones. *An introduction to: Velocity model building.* EAGE publications, 2016.

[21] G.G. Drijkoningen. *Introduction to Reflection Seismology.* TU Delft, 2012.

[22] M.T. Taner, F. Koehler, and K.A. Alhilali. Estimation and correction of near-surface time anomalies. *Geophysics*, 39(4):441–463, 1974.

[23] K.H. Waters. *Reflection seismology: A tool for energy resource exploration.* Wiley New York, 1981.

[24] S. Kirkpatrick, C.D. Gelatt, and M.P. Vecchi. Optimization by simulated annealing. *science*, 220(4598):671–680, 1983.

[25] R.W. Eglese. Simulated annealing: A tool for operational research. *European Journal of Operational Research*, 46(3):271–281, 1990.

[26] D. Henderson, S.H. Jacobson, and A.W. Johnson. The theory and practice of simulated annealing. In *Handbook of metaheuristics*, pages 287–319. Springer, 2003.

[27] N. Rey-Villamizar and O. Yilmaz. Automatic global optimal estimation of large residual statics. In *SEG Technical Program Expanded Abstracts 2017*, pages 2701–2705. Society of Exploration Geophysicists, 2017.

[28] S. Geman and D. Geman. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-6(6):721–741, 1984.

[29] X.S. Yang and X. He. Firefly algorithm: recent advances and applications. *International journal of swarm intelligence*, 1(1):36–50, 2013.

[30] B. Zeng, L. Gao, and X. Li. Whale swarm algorithm for function optimization. In *International Conference on Intelligent Computing*, pages 624–639. Springer, 2017.

[31] J. Preskill. Quantum computing 40 years later. *arXiv preprint arXiv:2106.10522*, 2021.

[32] K. Igeta and Y. Yamamoto. Quantum mechanical computers with single atom and photon fields. In *International Quantum Electronics Conference*. Optical Society of America, 1988.

[33] M.A. Nielsen and I. Chuang. Quantum computation and quantum information, 2002.

[34] J. Preskill. Lecture notes for physics 229: Quantum information and computation. *California Institute of Technology*, 16(1):1–8, 1998.

[35] T. Kadowaki and H. Nishimori. Quantum annealing in the transverse ising model. *Phys. Rev. E*, 58:5355–5363, Nov 1998.

[36] D-Wave: The Quantum Computing Company. What is quantum annealing? `https://docs.dwavesys.com/docs/latest/c_gs_2.html`. Accessed: 2021-10-08.

[37] Fred Glover, Gary Kochenberger, and Yu Du. A tutorial on formulating and using qubo models. *arXiv preprint arXiv:1811.11538*, 2018.

[38] A. Lucas. Ising formulations of many np problems. *Frontiers in physics*, 2:5, 2014.

[39] D. Freedman and P. Drineas. Energy minimization via graph cuts: Settling what is possible. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 2, pages 939–946. IEEE, 2005.

[40] H. Ishikawa. Transformation of general binary mrf minimization to the first-order case. *IEEE transactions on pattern analysis and machine intelligence*, 33(6):1234–1249, 2010.

[41] N.S. Dattani and N. Bryans. Quantum factorization of 56153 with only 4 qubits. *arXiv preprint arXiv:1411.6758*, 2014.

[42] R. Tanburn, E. Okada, and N. Dattani. Reducing multi-qubit interactions in adiabatic quantum computation without adding auxiliary qubits. part 1: The "deduc-reduc" method and its application to quantum factorization of numbers. *arXiv preprint arXiv:1508.04816*, 2015.

[43] D-Wave: The Quantum Computing Company. Programming the d-wave qpu: Setting the chain strength. Technical Report 14-1041A-A, D-Wave: The Quantum Computing Company, 2650 E Bayshore Rd, Palo Alto, April 2020.

[44] D.C. Hankerson, G. Hoffman, D.A. Leonard, C.C. Lindner, K.T. Phelps, C.A. Rodger, and J.R. Wall. *Coding Theory and Cryptography: The Essentials*. Chapman & Hall/CRC Pure and Applied Mathematics. Taylor & Francis, 2000.

[45] M. Oristaglio. Seam update: The arid model—seismic exploration in desert terrains. *The Leading Edge*, 34(4):466–468, 2015.

[46] E. Boros and P.L. Hammer. Pseudo-boolean optimization. *Discrete applied mathematics*, 123(1-3):155–225, 2002.

[47] E. Boros, P. Hammer, and G. Tavares. Preprocessing of unconstrained quadratic binary optimization. *RUTCOR Research Report*, 01 2006.

[48] E. Boros, P.L. Hammer, R. Sun, and G. Tavares. A max-flow approach to improved lower bounds for quadratic unconstrained binary optimization (qubo). *Discrete Optimization*, 5(2):501–529, 2008.

[49] J. Chen, T. Stollenwerk, and N. Chancellor. Performance of domain-wall encoding for quantum annealing. *arXiv preprint arXiv:2102.12224*, 2021.

[50] N. Chancellor. Domain wall encoding of discrete variables for quantum annealing and qaoa. *Quantum Science and Technology*, 4(4):045004, 2019.

[51] E. Pelofske, G. Hahn, and H.N. Djidjev. Advanced anneal paths for improved quantum annealing. In *2020 IEEE International Conference on Quantum Computing and Engineering (QCE)*, pages 256–266. IEEE, 2020.