Quantum-Assisted Greedy Algorithms

Ramin Ayanzadeh
Department of CSEE
University of Maryland, Baltimore County
ayanzadeh@umbc.edu

Milton Halem
Department of CSEE
University of Maryland, Baltimore County
halem@umbc.edu

John Dorband
Department of CSEE
University of Maryland, Baltimore County
dorband@umbc.edu

Tim Finin
Department of CSEE
University of Maryland, Baltimore County
finin@umbc.edu

We show how to leverage quantum annealers to better select candidates in greedy algorithms. Unlike conventional greedy algorithms that employ problem-specific heuristics for making locally optimal choices at each stage, we use quantum annealers that sample from the ground state of Ising Hamiltonians at cryogenic temperatures and use retrieved samples to estimate the probability distribution of problem variables. More specifically, we look at each spin in the Ising model as a random variable and contract all problem variables whose corresponding uncertainties are negligible. Our empirical results on a D-Wave 2000Q quantum processor revealed that the proposed quantum-assisted greedy algorithm (QAGA) can find remarkably better solutions, compared to the state-of-the-art techniques in the realm of quantum annealing.

1 INTRODUCTION

Quantum annealing is a meta-heuristic that addresses combinatorial optimization problems (i.e., discrete optimization problems)which are intractable in the realm of classical computingby taking advantage of quantum mechanical fluctuations. Quantum annealers are a physical realization of the quantum annealing process that draws samples from the ground state of a given Ising Hamiltonian at cryogenic temperatures (i.e., near zero Kelvin) in near-constant time [Nishimori and Takada, 2017, Finnila et al., 1994, Kadowaki and Nishimori, 1998, Ohzeki and Nishimori, 2011].

The quantum processing unit (QPU), by D-Wave Systems, is a programmable quantum annealer that samples from the ground state(s) of a given Ising Hamiltonian at cryogenic temperatures [Johnson et al., 2011]. The D-Wave quantum annealer receives coefficients of an Ising Hamiltonian as input (here $\mathbf{h}$ and J) and returns the ground state of the following energy function:

$$E_{\text{Ising}}(\mathbf{z}) = \sum_{i=1}^{N} \mathbf{h}_i \mathbf{z}_i + \sum_{i=1}^{N} \sum_{j=i+1}^{N} J_{ij} \mathbf{z}_i \mathbf{z}_j, \qquad (1)$$

where N denotes the number of quantum bits (qubits). In this representation, $\mathbf{z}_i \in \{-1, +1\}$. One can apply a linear transform to map (1) to its equivalent quadratic unconstraint binary optimization (QUBO) form, and vice versa, as follows:

$$E_{\text{QUBO}}(\mathbf{x}) = \sum_{I \leq j}^{N} \mathbf{x}_i Q_{ij} \mathbf{x}_j, \qquad (2)$$

where $\mathbf{x} \in \{0, 1\}^N$ and $i, j \in \{1, 2, \cdots, N\}$. In this representation, Q includes both linear biases and quadratic couplersanalogous to $\mathbf{h}$ and J in (1), respectively.

Unlike conventional computing machines (i.e., classical and gate model quantum computers) that have a rich set of machine instructions, the D-Wave QPU executes only one quantum machine instruction (QMI)minimizing (1) or (2) [Ayanzadeh et al., 2019c]. One can reduce any problem of class NP to an NP-complete problem in polynomial-time [Garey and Johnson, 2002, Ayanzadeh et al., 2019b]. Therefore, adiabatic quantum computers can be considered universal in terms of problem-solving; however, from a programming perspective, one still needs to form an Ising Hamiltonian (executable on a QPU) whose ground state represents the optimum solution for the original problem of interest [Ayanzadeh et al., 2019c].

As an illustration, we can employ the quantum annealers to address the original problem of binary compressive sensing (i.e., the ℓ_0-norm sparse recovery) [Ayanzadeh et al., 2019d]. Similarly, we can represent the problem of non-negative binary matrix factorization as finding the global minimum of (2) [O'Malley et al., 2018]. In the realm of quantum artificial intelligence and quantum machine learning, the D-Wave quantum annealer has been shown to address several problems including, but not limited to: planning [Rieffel et al., 2015], scheduling [Venturelli et al., 2015, Tran et al., 2016], constraint satisfaction problems (CSP)

[Bian et al., 2016], and training deep neural networks [Adachi and Henderson, 2015]. Moreover, the D-Wave quantum annealer has also demonstrated a capable performance in solving discrete optimization problems [Bian et al., 2014]. In addition to the optimization aspect of the quantum annealers, one can employ the D-Wave QPU to sample from high-dimensional probability distributions, which has many applications in statistics, signal processing, artificial intelligence and machine learning [Biswas et al., 2017, Adachi and Henderson, 2015].

The current generation of the D-Wave quantum annealer (the Chimera architecture) includes more than 2,000 qubits and about 6,000 couplers, while the next generation (the Advantage architecture) will include more than 5,000 qubits and about 40,000 couplers [Boothby et al., 2019]. However, the D-Wave QPU hardware does have limitations that not only restrict the process of mapping problems into an executable QMI [Ayanzadeh et al., 2019c] but also lowers the quality of its results [Ayanzadeh et al., 2019a]. In current generations of the D-Wave QPUs, any executable QMI must satisfy:

$$-2 \leq \mathbf{h}_i \leq +2, \tag{3}$$

$$-1 \leq J_{ij} \leq +1, \tag{4}$$

where $i, j \in \{1, 2, \ldots, N\}$ and $i < j$. After problem formulation,(i.e., mapping the problem to minimizing (1) whose ground state represents the optimum solution of the original problem), one needs to scale/normalize coefficients by multiplying all elements of $\mathbf{h}$ and J with a small-enough number to satisfy (3) and (4). Scaling coefficients reduce the energy gap between the ground state (global minimum) and the first excited state (a state right above the global minimum). Decreasing the energy gap between the ground and first-excited states linearly increases the required anneal time exponentially [Nishimori and Takada, 2017]. Thus, the required annealing time can quickly exceed the capacity of current D-Wave QPUs (2,000 micro-seconds), which makes the annealing process diabatic and reduces the probability of finding the global minimum exponentially [Nishimori and Takada, 2017].

In addition to the range limitations, coefficients are restricted to 45 bits precision. Consequently, after the scaling, we need to truncate the coefficients which can lower the quality/accuracy of samples (attained by the D-Wave QPU) and even make the QMI infeasible (i.e., the resulting QMI has a different ground state compared to the original Ising Hamiltonian) [Pudenz et al., 2015]. Dorband (2018) has proposed a heuristic that addresses the precision issue on the D-Wave QPUs, albeit executing many QMIs for one problem [Dorband, 2018a].

Furthermore, coupling every qubit to every other qubit is infeasible in practice, so the D-Wave QPUs have a sparse connectivity architecture. Therefore, one can entangle multiple physical qubits to form a virtual qubit with higher connectivity to other virtual qubits increasing the connectivity at the expense of a significant reduction in the number of virtual qubits relative to the actual number in the current D-Wave QPUs. As an example, 2048 qubits on the Chimera topology are equivalent to a fully-connected graph of virtual qubit size 64. It is possible to implicitly leverage the capacity of the current D-Wave QPUs at the cost of executing numerous QMIs [Okada et al., 2019]. It is crucial to emphasize that virtual qubit chains can break during the annealing (i.e., physical qubits in a chain collapses to different classical values during the measurement). Longer chains have a higher probability to become broken and finding an optimum embedding is still an open problem in the realm of programming quantum annealers. Post-quantum (or classical) heuristics can remediate broken-chains (le.g. by a majority voting scheme); however, some chains break because they represent a state with lower energy.

From a problem-solving point-of-view, executing a QMI on the D-Wave QPUs does not guarantee that we will achieve a global optimum. Applying gauge transform (or spin-reversal-transform) can reduce the impact of analog errors [Pelofske et al., 2019]. This preprocessing technique, nevertheless, is trivial because it does not alter the landscape of the Ising model.

Multi-qubit correction (MQC) is a classical post-quantum heuristic that reduces a set of samples to a sample with lower energy [Dorband, 2018b]. MQC has demonstrated to find a sample with remarkably lower energy, compared to quantum annealing with spin-reversal-transform and reverse annealing. However, applying MQC cannot guarantee that we can find the global minimum. It is worth highlighting that MQC performs well only on sparse problems.

Greedy algorithms are a problem-solving paradigm where we make locally optimal choices in each stage and expect that they yield the globally optimum solution. Although most greedy algorithms fail to achieve the global optimum, in many applications they are the best choice due to their efficiency [DeVore and Temlyakov, 1996]. As an example, greedy algorithms are widely used in sparse recovery applications, at the cost of lower recovery accuracy (compared to convex optimization methods in compressive sensing) [Mousavi et al., 2019]. It is worth noting that greedy algorithms can find globally op-

timum solutions if the problem exhibits optimal substructure.

In this study, we demonstrate that leveraging quantum annealers to optimally select candidates in each stage of the greedy algorithms significantly boosts the performance of quantum annealers to obtain optimum solutions relative to the best-known techniques for improving the results of quantum annealing. More specifically, at each stage, employ the quantum annealer to provide samples from the ground state of the problem and use these retrieved samples to estimate the probability distribution of problem variables. After fixing the variables with negligible uncertainty, one proceeds to the next stage where the quantum annealer will solve a smaller problem with sparser couplings. Our experimental results, using the D-Wave 2000Q quantum processor, showed that the proposed method can find samples with remarkably lower energy, compared to the best-known enhancements in the realm of quantum annealing.

2 METHOD

In this section, we demonstrate how to employ quantum annealers for boosting the performance of greedy algorithms in minimizing Ising Hamiltonians, shown in (1). It is crucial to highlight that minimizing the Ising model by the D-Wave quantum annealer can address a vast range of real-world applications.

Let $\mathcal{H}$ denotes the problem of minimizing a given Ising Hamiltonian as follows:

$$\mathcal{H} := \min_{\mathbf{z}} \sum_{i=1}^{N} \mathbf{h}_i \mathbf{z}_i + \sum_{i=1}^{N} \sum_{j=i+1}^{N} J_{ij} \mathbf{z}_i \mathbf{z}_j, \qquad (5)$$

where $\mathbf{h}$ and J represent linear and quadratic coefficients, respectively. The proposed quantum-assisted greedy algorithm (QAGA) starts with $\mathbf{z}^* = \{\}$ as ground state of $\mathcal{H}$. The QAGA executes $\mathcal{H}$ on a quantum annealer and requests for n samples. Quantum annealers draw samples from a problem-dependent Boltzmann distribution (i.e., Poisson distribution in which the mean parameter approaches the energy value of the ground state(s) of the given Ising Hamiltonian) at cryogenic temperatures. For a given problem, shown in (5), let $Z = \{\mathbf{z}^1, \mathbf{z}^2, \cdots, \mathbf{z}^n\}$ (where $\mathbf{z}^j \in \{-1, +1\}^N$) denotes the set of all samples attained by a quantum annealer. Here, each sample ($\mathbf{z}^j, \quad j \in \{1, 2, \cdots, n\}$) contains a measurement for every qubit. Hence, we can look at each problem variable ($\mathbf{z}_i, \quad I \in \{1, 2, \cdots, N\}$) as a random variable with Bernoulli distribution that takes its value from $\{-1, +1\}$. Note that in this representation, the value -1 in (1) is analogous to 0 in QUBO form of the same problem. Thus, we can extend all analysis to QUBO form where binary variables $\mathbf{x}_i$ takes their value from $\{0, 1\}$. After retrieving the sample set Z, we estimate the uncertainty of every variable ($\mathbf{z}_i$) as follows:

$$u(\mathbf{z}_i) = 1 - \frac{|\sum_{j=1}^{n} \mathbf{z}_i^j|}{n}. \qquad (6)$$

For any variable that the corresponding uncertainty is negligiblei.e., $u(\mathbf{z}_i) \leq \theta$, where $\theta \in [0, 1]$ specifies the threshold parameterwe fix the value of optimum solution as follows:

$$\mathbf{z}_i^* = \left[\sum_{j=1}^{n} \mathbf{z}_i^j \right]. \qquad (7)$$

Afterward, we reduce the current problem ($\mathcal{H}^t$) to a new (smaller) problem ($\mathcal{H}^{t+1}$) via removing all fixed variables. To fix the variable $\mathbf{z}_i$ in $\mathcal{H}^{t+1}$, we remove/delete the linear coefficient $\mathbf{h}_i$ and add all couplers that $\mathbf{z}_i$ is involved with to the corresponding linear coefficients. In other words, after fixing the variable $\mathbf{z}_i$, we add the value of J_{ij} (or J_{ji}) to $\mathbf{h}_j$ and remove/delete the coupler from $\mathcal{H}^{t+1}$. If $\mathcal{H}^t$ is not identical to $\mathcal{H}^{t+1}$, QAGA solves the resulting Hamiltonian on the quantum annealer and repeats the above process.

In QAGA, instead of making locally optimal choices in each stage of the greedy algorithms, we employ quantum annealers to identify candidates that yield the global optimum. The QAGA is sensitive to the threshold parameter θ. When $\theta \to 1$, the QAGA acts like a classical greedy algorithm and quickly converges to a local optimum. The case $0 \leftarrow \theta$ leads to a slower convergence process that may end prior to fixing all problem variables. Hence, if QAGA ended without fixing all problem variables, we apply the MQC method [Dorband, 2018b] on the samples from the last stage of the QAGA and assign values for the remaining problem variables. Algorithm 1 illustrates the QAGA process. Finally, we can apply a classical local search on $\mathbf{z}^*$ to increase the probability of finding the ground state.

3 RESULTS

In this section, we evaluate the performance of the proposed method. Generating random problems results in hard problems for benchmarking quantum annealers [Dorband, 2018a, Dorband, 2018b, King et al., 2019]. Hence, as a proof-of-concept, we generated random problems with arbitrary graph structures (i.e., both coefficients and graph structure are random).

Input: $\mathcal{H}, \theta$
Output: $\mathbf{z}^*$
$\mathbf{z}^* \leftarrow \{\}$
$\mathcal{H}^t \leftarrow \{\}$
$\mathcal{H}^{t+1} \leftarrow \mathcal{H}$
while $\mathcal{H}^{t+1} \neq \mathcal{H}^t$ **do**
  $\mathcal{H}^t \leftarrow \mathcal{H}^{t+1}$
  $Z \leftarrow \{\mathbf{z}^1, \mathbf{z}^2, \cdots, \mathbf{z}^n\} = \min_{\mathbf{z}} \mathcal{H}^t$
  for $i \leftarrow 0$ **to** N **do**
    if $u(\mathbf{z}_i) \leq \theta$ **then**
      $\mathbf{z}_i^* \leftarrow \left[\sum_{j=1}^n \mathbf{z}_i^j\right]$
      Remove $\left(\mathcal{H}^{t+1}, \mathbf{h}_i\right)$
      for $j \leftarrow 0$ **to** N **do**
        if $J_{ij} \in \mathcal{H}^{t+1}$ **then**
          $\mathcal{H}^{t+1}[\mathbf{h}_j] \leftarrow \mathcal{H}^{t+1}[\mathbf{h}_j] J_{ij} \mathbf{z}_i^*$
          Remove $\left(\mathcal{H}^{t+1}, J_{ij}\right)$
        end
        if $J_{ji} \in \mathcal{H}$ **then**
          $\mathcal{H}^{t+1}[\mathbf{h}_j] \leftarrow \mathcal{H}^{t+1}[\mathbf{h}_j] J_{ji} \mathbf{z}_i^*$
          Remove $\left(\mathcal{H}^{t+1}, J_{ji}\right)$
        end
      end
    end
  end
end
if $|\mathbf{z}^*| < N$ **then**
  $\hat{\mathbf{z}} \leftarrow \mathrm{MQC}(Z)$
  Append $(\mathbf{z}^*, \hat{\mathbf{z}})$
end
return $\mathbf{z}^*$

Algorithm 1: Quantum-assisted greedy algorithm for minimizing the Ising model

3.1 Experiment A

For every problem in this experiment, we first generated a random graph of size 50 with a specified sparsity rate ($s \in \{0.05, 0.25, 0.5, 0.75, 1.0\}$). More specifically, we randomly selected edges from a complete graph with $N = 50$ nodes where the sparsity rate s denoted the probability of selecting edges. Afterward, we set values of the corresponding biases and couplers randomly. In this study, we generated three different types of benchmark problems: (a) we picked values of biases and couplers randomly from $\{-1, +1\}$ (binary coefficients); (b) we used random numbers in $[-1, +1]$ with uniform distribution to assign values of $\mathbf{h}$ and J (uniform coefficients); and (c) coefficients values were drawn randomly from a standard normal distribution (normal coefficients). As a proof-of-concept, we compared the proposed quantum assisted greedy algorithm (QAGA) with two methods:

- quantum annealing (QA) with spin-reversal-transforms [Pelofske et al., 2019];
- multi-qubit correction (MQC) which is the best known post-quantum heuristic for quantum annealers [Dorband, 2018b].

In this experiment, we requested for 1,000 samples for all methods. The uncertainty threshold in QAGA was $\theta = 0.0$i.e., all spins must have the same value so QAGA can fix them for the next stage. In QA, we enabled the inter-sample delay to reduce the sample-to-sample correlations in successive reads/measurements. In addition, the number of spin-reversal-transforms was 10. Since randomly generated problems are not compatible with the working graph of the current D-Wave quantum processors, we used the minor-embedding heuristic [Cai et al., 2014] for embedding the arbitrary random graphs to the Chimera topology of the D-Wave 2000Q quantum processors.

3.2 Experiment B

In this experiment, we aim to analyze the convergence of the QAGA, empirically. To this end, we measured the average number of stages/iterations that QAGA takes to solve the given problems. Table 1 illustrates the average number of iterations that QAGA takes to solve 100 random benchmark problems with $N = 50$ variables and normal coefficients.

Table 1: Average number of iterations for QAGA in solving 100 random benchmark problems with different thresholds

	sparsity rate (s)				
θ	0.05	0.25	0.50	0.75	1.00
0.25	3.25	2.80	3.15	3.40	3.30
0.15	3.60	3.40	3.35	3.45	3.65
0.05	4.10	4.45	4.30	3.20	4.05
0.00	5.50	6.20	2.10	2.05	2.30

4 DISCUSSION

In this study, we proposed a novel approach for combinatorial optimization problems, so-called quantum-assisted greedy algorithms (QAGA), that employs the quantum annealers for making globally optimum decisions in each stage of a greedy algorithm. To this end, we looked at the quantum annealers as a physical process that naturally draws samples from the ground state of Ising Hamiltonians (i.e., a problem-dependent Boltzmann distribution) at cryogenic temperatures. From a problem-solving point-of-view, conjugating quantum annealers and greedy algorithms addresses drawbacks of both methods and results in re-

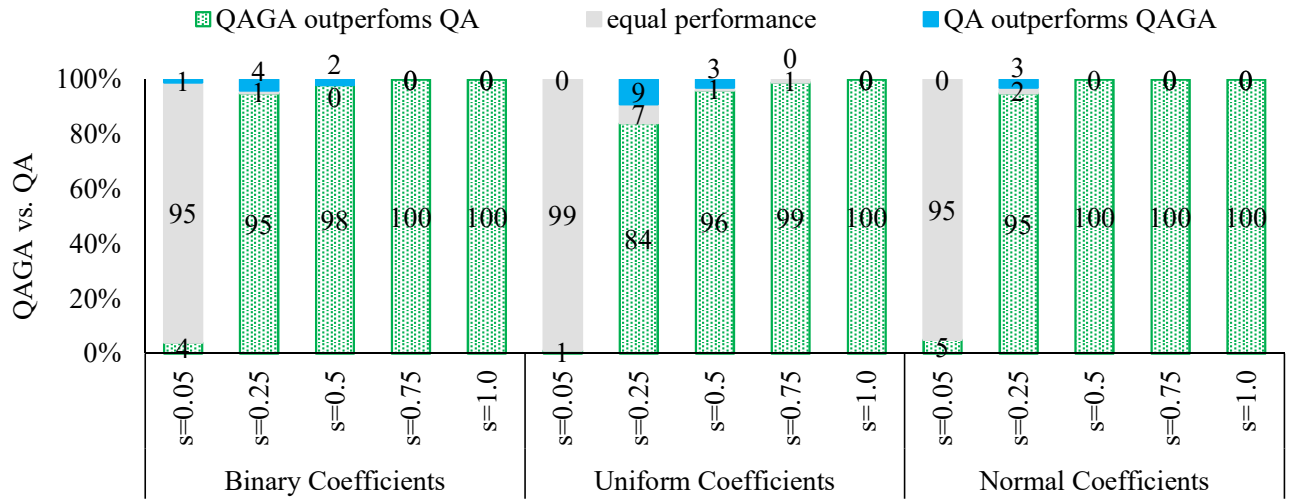

Figure 1: Performance comparison between QAGA and QA (with spin-reversal-transforms and inter-sample delays) in solving 100 random benchmark problems with $N = 50$ spin variables and different sparsity rates s

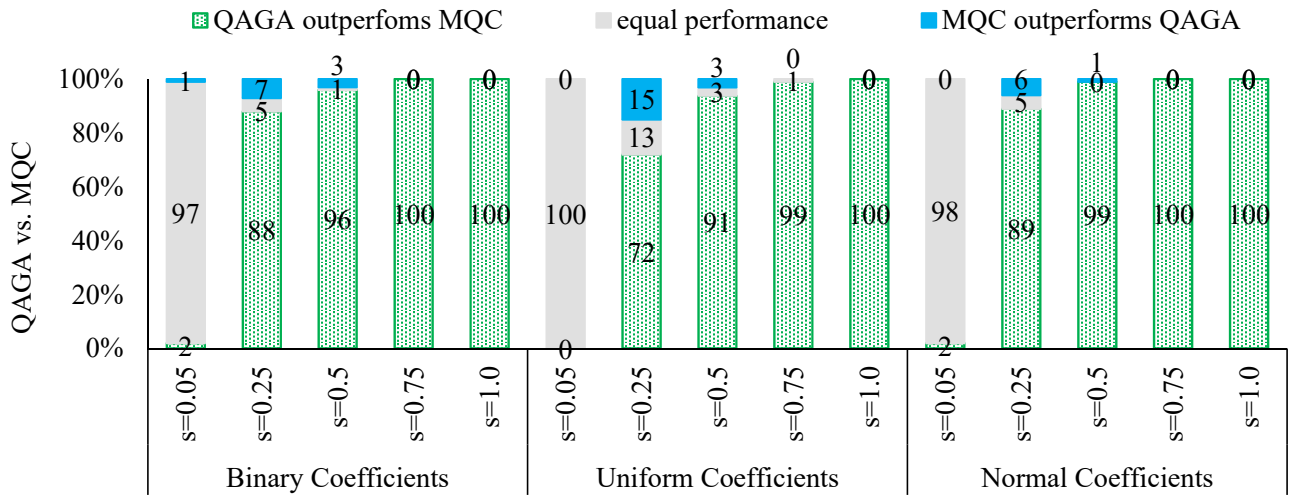

Figure 2: Performance comparison between QAGA and MQC in solving 100 random benchmark problems with $N = 50$ spin variables and different sparsity rates s

markably better solutions, at the cost of executing multiple QMIs for one problem.

Our empirical results on several randomly generated benchmark problems demonstrated that QAGA finds samples with remarkably lower energies, compared to the best-known techniques in the realm of quantum annealingnamely spin-reversal-transforms (or gauge transform) and multi-qubit correction (MQC). For sparse problems (i.e., the structure of the problem is close to the Chimera architecture), the performance of the QAGA approaches to MQC. When the sparsity decreases, however, QAGA shows supremacy in terms of finding samples with lower energy. In other words, QAGA also addresses the embedding-related issues, specifically the broken-chains. It is worth noting that QAGA has a remarkably higher convergence rate, compared to the classical greedy algorithms.

Acknowledgements

This research has been supported by NASA grant (#NNH16ZDA001N-AIST 16-0091), NIH-NIGMS Initiative for Maximizing Student Development Grant (2 R25-GM55036), and the Google Lime scholarship. We would like to thank the D-Wave Systems management team, namely Rene Copeland, for granting access to the D-Wave 2000Q quantum annealer.

References

[Adachi and Henderson, 2015] Adachi, S. H. and Henderson, M. P. (2015). Application of quantum an-

nealing to training of deep neural networks. *arXiv preprint arXiv:1510.06356*.

[Ayanzadeh et al., 2019a] Ayanzadeh, R., Halem, M., and Finin, T. (2019a). Reinforcement quantum annealing. *University of Maryland, Baltimore County*.

[Ayanzadeh et al., 2019b] Ayanzadeh, R., Halem, M., and Finin, T. (2019b). SAT-based compressive sensing. *arXiv preprint arXiv:1903.03650*.

[Ayanzadeh et al., 2019c] Ayanzadeh, R., Halem, M., and Finin, T. (2019c). An SAT-based quantum programming paradigm. *University of Maryland, Baltimore County*.

[Ayanzadeh et al., 2019d] Ayanzadeh, R., Mousavi, S., Halem, M., and Finin, T. (2019d). Quantum annealing based binary compressive sensing with matrix uncertainty. *arXiv preprint arXiv:1901.00088*.

[Bian et al., 2014] Bian, Z., Chudak, F., Israel, R., Lackey, B., Macready, W. G., and Roy, A. (2014). Discrete optimization using quantum annealing on sparse ising models. *Frontiers in Physics*, 2:56.

[Bian et al., 2016] Bian, Z., Chudak, F., Israel, R. B., Lackey, B., Macready, W. G., and Roy, A. (2016). Mapping constrained optimization problems to quantum annealing with application to fault diagnosis. *Frontiers in ICT*, 3:14.

[Biswas et al., 2017] Biswas, R., Jiang, Z., Kechezhi, K., Knysh, S., Mandra, S., O'Gorman, B., Perdomo-Ortiz, A., Petukhov, A., Realpe-Gómez, J., Rieffel, E., et al. (2017). A nasa perspective on quantum computing: Opportunities and challenges. *Parallel Computing*, 64:81–98.

[Boothby et al., 2019] Boothby, K., Bunyk, P., Raymond, J., and Roy, A. (2019). Next-generation topology of d-wave quantum processors. Technical report, Technical report.

[Cai et al., 2014] Cai, J., Macready, W. G., and Roy, A. (2014). A practical heuristic for finding graph minors. *arXiv preprint arXiv:1406.2741*.

[DeVore and Temlyakov, 1996] DeVore, R. A. and Temlyakov, V. N. (1996). Some remarks on greedy algorithms. *Advances in computational Mathematics*, 5(1):173–187.

[Dorband, 2018a] Dorband, J. E. (2018a). Extending the d-wave with support for higher precision coefficients. *arXiv preprint arXiv:1807.05244*.

[Dorband, 2018b] Dorband, J. E. (2018b). A method of finding a lower energy solution to a qubo/ising objective function. *arXiv preprint arXiv:1801.04849*.

[Finnila et al., 1994] Finnila, A., Gomez, M., Sebenik, C., Stenson, C., and Doll, J. (1994). Quantum annealing: a new method for minimizing multidimensional functions. *Chemical physics letters*, 219(5-6):343–348.

[Garey and Johnson, 2002] Garey, M. R. and Johnson, D. S. (2002). *Computers and intractability*, volume 29. wh freeman New York.

[Johnson et al., 2011] Johnson, M. W., Amin, M. H., Gildert, S., Lanting, T., Hamze, F., Dickson, N., Harris, R., Berkley, A. J., Johansson, J., Bunyk, P., et al. (2011). Quantum annealing with manufactured spins. *Nature*, 473(7346):194.

[Kadowaki and Nishimori, 1998] Kadowaki, T. and Nishimori, H. (1998). Quantum annealing in the transverse ising model. *Physical Review E*, 58(5):5355.

[King et al., 2019] King, J., Mohseni, M., Bernoudy, W., Fréchette, A., Sadeghi, H., Isakov, S. V., Neven, H., and Amin, M. H. (2019). Quantum-assisted genetic algorithm. *arXiv preprint arXiv:1907.00707*.

[Mousavi et al., 2019] Mousavi, S., Taghiabadi, M. M. R., and Ayanzadeh, R. (2019). A survey on compressive sensing: Classical results and recent advancements. *arXiv preprint arXiv:1908.01014*.

[Nishimori and Takada, 2017] Nishimori, H. and Takada, K. (2017). Exponential enhancement of the efficiency of quantum annealing by non-stoquastic hamiltonians. *Frontiers in ICT*, 4:2.

[Ohzeki and Nishimori, 2011] Ohzeki, M. and Nishimori, H. (2011). Quantum annealing: An introduction and new developments. *Journal of Computational and Theoretical Nanoscience*, 8(6):963–971.

[Okada et al., 2019] Okada, S., Ohzeki, M., Terabe, M., and Taguchi, S. (2019). Improving solutions by embedding larger subproblems in a d-wave quantum annealer. *Scientific reports*, 9(1):2098.

[O'Malley et al., 2018] O'Malley, D., Vesselinov, V. V., Alexandrov, B. S., and Alexandrov, L. B. (2018). Nonnegative/binary matrix factorization with a d-wave quantum annealer. *PloS one*, 13(12):e0206653.

[Pelofske et al., 2019] Pelofske, E., Hahn, G., and Djidjev, H. (2019). Optimizing the spin reversal transform on the d-wave 2000q. *arXiv preprint arXiv:1906.10955*.

[Pudenz et al., 2015] Pudenz, K. L., Albash, T., and Lidar, D. A. (2015). Quantum annealing correction for random ising problems. *Physical Review A*, 91(4):042302.

[Rieffel et al., 2015] Rieffel, E. G., Venturelli, D., O'Gorman, B., Do, M. B., Prystay, E. M., and Smelyanskiy, V. N. (2015). A case study in programming a quantum annealer for hard operational planning problems. *Quantum Information Processing*, 14(1):1–36.

[Tran et al., 2016] Tran, T. T., Do, M., Rieffel, E. G., Frank, J., Wang, Z., O'Gorman, B., Venturelli, D., and Beck, J. C. (2016). A hybrid quantum-classical approach to solving scheduling problems. In *Ninth annual symposium on combinatorial search*.

[Venturelli et al., 2015] Venturelli, D., Marchand, D. J., and Rojo, G. (2015). Quantum annealing implementation of job-shop scheduling. *arXiv preprint arXiv:1506.08479*.