Noname manuscript No.
(will be inserted by the editor)

Simulated versus reduced noise quantum annealing in Maximum Independent Set solution to wireless network scheduling

Chi Wang · Edmond Jonckheere

Received: date / Accepted: date

Abstract With the introduction of Adiabatic Quantum Computation (AQC) and its implementation on D-Wave annealers, there has been a constant quest for benchmark problems that would allow for a fair comparison between such classical combinatorial optimization techniques as Simulated Annealing (SA) and AQC-based optimization. Such a benchmark case-study has been the scheduling problem to avoid interference in the very specific Dirichlet protocol in wireless networking, where it was shown that the gap expansion to retain noninterference solutions benefits AQC better than SA. Here we show that the same gap expansion allows for significant improvement of the D-Wave 2X solution compared with that of its its predecessor, the D-Wave II.

1 Introduction

Since the first proposal of the Quantum Turing Machine by David Deutsch in 1985 [52], quantum computation has seen fast evolution in recent decades [91]. It has provided a revolutionary transformation of the notion of feasible computability, with the most celebrated Shor's factoring algorithm [53] and Grover's search algorithm [54]. Adiabatic Quantum Computation (AQC), first proposed in the year 2000 [51], has been a very promising candidate for future quantum computation models. Resembling the classical metaheuristic of Simulated Annealing (SA) [39], AQC aims to

This research was supported by NSF Grant CCF-1423624.

C. Wang
Dept. of Electrical Engineering, University of Southern California, Los Angeles, CA 90089, USA
Tel.: +86 186-0035-2220 (China) +1 323-636-1777 (US)
E-mail: alpha@terraquanta.io
now with TerraQuanta, Chengdu, Sichuan, China

E. Jonckheere
Dept. of Electrical Engineering, University of Southern California, Los Angeles, CA 90089, USA
Tel.: +1 213-740-4457
Fax: +1 213-821-1109
E-mail: jonckhee@usc.edu

solve hard optimization problems with promise of utilizing quantum tunneling effects so as to more efficiently explore the search space.

With the introduction of the world's first programmable quantum annealing platform (known as D-Wave) in 2011, more research efforts have been put into such endeavors as benchmarking, error correction, quantumness test and applications, all of which led to the following hotly debated questions:

1. **Does AQC involve quantum effects?** The short answer is 'most likely yes.' In Apr. 2013, by performing experiments on D-Wave with random instances with different level of hardness [41, 42], and comparing the experimental results with classical models of simulated annealing, spin dynamics and Quantum Monte Carlo, a unique bimodal distribution of success probability of quantum annealing was observed to be in agreement of Quantum Monte Carlo. This rules out simulated annealing and spin dynamics models. In May 2014, Lanting *et al.* [55] experimentally determined the existence of entanglement inside D-Wave. In Nov. 2014, evidence of quantum tunneling was experimentally observed [56].
2. **Does AQC have speedup over classical computer?** The short answer is 'convincing evidence of speedup has been slow to show.' Definition of the term 'speedup' is more subtle than it appears to be, and speed itself is very problem dependent. Early efforts have been made at showing quantum advantage [41–43], and general speedup has not yet been detected. Katzgraber *et al.* gave a possible reason for why speedup has not been detected [57], that random Ising problems might be too easy, and that 20μs annealing time might be too long. With the introduction of D-Wave 2X with 1152 qubits in Aug. 2015, D-Wave released benchmark results claiming that time-to-target measure on D-Wave is 8 to 600 times faster than competing algorithms on all input cases tested [58]. In 2016, an arXiv preprint by Google [86] claimed that with the new D-Wave 2X, a speedup as high as a factor of 10^8 has been observed, but this work received mixed reviews for the main reason that the problem appears to be quite an artificial one, and that the annealing time might be too long for easier cases and hides the exponential scaling. More recently [88], a "first quantum speed up" was claimed when simulating Hamiltonian systems of spin rings with Heisenberg coupling subject to self-thermalization on quantum circuit models where classical simulations fail beyond 22 spins.
3. **Does AQC require error correction?** The short answer is 'at this stage most definitely yes.' In Jul. 2013, Pudenz *et al.* proposed quantum annealing correction on D-Wave [59] by using multiple qubits to represent one qubit and properly setting the penalty weights between such redundancy qubits, and observed significant improvements in experiments. Unlike traditional quantum error correction where many-body interaction is required and overhead is usually too large to be implementable on D-Wave, Pudenz code is specifically designed for D-Wave's Chimera architecture. Note, however, that it has been claimed [87] that, in order to achieve an *objective* comparison between quantum and classical computers, quantum computers should not be allowed to have error corrections. In this context, a case for *quantum supremacy* could be claimed if quantum computers could

sample the output of random quantum circuits in a manner that state-of-the-art classical computers could not achieve.

4. **What are the obstacles to AQC?**
   (a) **Minor embedding:** In Feb. 2015, Wu [63] singled out minor embedding as No. 1 challenge in applying AQC to practical problems. The hardware graph, known as 'Chimera' architecture, is a fixed graph composed of a lattice of $K_{4,4}$ bipartite cells; thus, while mapping a real world arbitrary graph into Chimera architecture, the NP-hard process of minor embedding has to be performed. This greatly increases the overall complexity of solving such problems.
   (b) **Analog control errors:** Weights on Ising Hamiltonian is problem-defined and can be anything. King *et al.* [64] claimed that the analog control errors on bias fields h and spin coupling J follow Gaussian distributions with $\sigma_h \approx 0.05$ and $\sigma_J \approx 0.035$ on the available scale of $[-2, 2]$ on D-Wave II. Such value is believed to be significantly reduced on the D-Wave 2X.

1.1 Contribution

1.1.1 Main

In the present paper, we contribute mainly to Problem #2, but with "speed-up" understood in a very specific way. More broadly speaking than "speedup," it is of overriding importance to find a problem, a Machine Learning *"killer application"* [89] that could at least potentially justify the use of a quantum annealer. By comparing it to classical methods, including exact algorithms, metaheuristics, problem-specific heuristics, the quantum annealer should at least have a practical advantage, show some *"quantum enhancement" [89]*, if not faster.

We found that the wireless network scheduling problem (Sec. 2) of moving packets from sources to destination optimally in the sense of delay and subject to interference constraints at the router nodes is such a 'good quantum problem' that shows a definite advantage over simulated annealing. More specifically, our benchmark problem involves the new Dirichlet protocol (Sec. 3), a particular case of the Heat Diffusion (HD) suite of protocols [71–75], which outperforms the original Back-Pressure (BP) protocol [22]. As already noted in [83], the gap expansion technique to move those annealing runs that satisfy the interference constraints to the bottom of the energy spectrum so that minimum energy solutions are network-relevant (Sec. 6.3) benefits quantum annealing much better than simulated annealing [83]; furthermore, as major result of the present paper, this improvement is more pronounced in D-Wave 2X than in its D-Wave II predecessor (Sec. 8).

Specifically, the improvement of D-Wave 2X over D-Wave II is two-fold. First, D-Wave 2X has 1152 qubits over 512 and, second, D-Wave 2X has better error control over the coupling and bias parameters. The latter improvement is significant and probably most relevant. It has indeed been shown that such benchmark problems as Grover search when mapped to AQC-amenable problem involve quadratic maps defined over the complex projective space that are unstable in the sense of differen-

tial topology [65, 66]. This has the consequence that, while, in theory, the spectral gap might be large enough for theoretical AQC computation, under parameter variation the spectral gap reduces dramatically making the process *diabatic* with the ineluctable consequence that the correct minimum energy solution is missed. The gap expansion technique utilized here and in [83] pushes the noninterference solutions down the energy spectrum and better error control prevents the run to deviate from the minimum energy solution making the D-Wave 2X runs much more successful than those of its predecessor.

1.1.2 Other contributions

Our main contribution to Problem #2 appears to leave the other problems in the dark. However, we feel that Problems #1 and #3 have been and are still adequately addressed by many groups. Regarding the remaining problems, we contributed to Problem #4(a) in [84, 85] using the Ollivier-Ricci curvature technique [78, 79]. The importance of Problem 4(b), as formulated in Sec. 5.2, is at least practically illustrated here; indeed, the improvement afforded by D-Wave 2X (Sec. 8) is, as we conjecture, due to better analog control errors in the latter than in D-Wave II. Along a much more theoretical line, the effect of analog control errors on the differential topology of the annealing run is considered in [65].

1.2 Summary

The workflow of this paper is depicted in Fig. 1. The top lines denotes the *classical* weighting-scheduling-forwarding sequence of the BP and HD protocols (Sec. 3). The *new, nonclassical* part of the paper is the "bypass" of the classical scheduling. This bypass comprises the embedding preprocessor (Sec. 6.2) to partially solve the minor embedding challenge of Problem 4(a) and the mapping preprocessor (Sec. 6.1) to greatly improve solution quality of quantum annealing by gap expansion (Sec. 8).

2 Wireless network scheduling—Weighted Maximum Independent Set

A Wireless Sensor Network (WSN) typically consists of a large number of low-power computation-capable autonomous nodes. Unlike wired networks where node-to-node delay is usually the only optimization objective, in WSNs power conservation is also a major concern. Sensor nodes usually carry generally irreplaceable power sources, densely deployed within frequently changing network topology [68]. This is also the reason why a power-conservative routing protocol is usually preferred in sensor networks instead of network flooding. However, certain applications still require lower delay, which generally entails a trade-off with power consumption. SEAD (Scalable Energy-Efficient Asynchronous Dissemination [69]) is an example of a protocol that proposes to trade-off between node-to-node delay and energy saving. Likewise, our suite of Heat Diffusion protocols [71–75] allows for *Pareto-optimal* tradeoff between routing energy (7) and delay as expressed by queue occupancy (Little's theorem) in (19). Here, however, we will focus on delay.

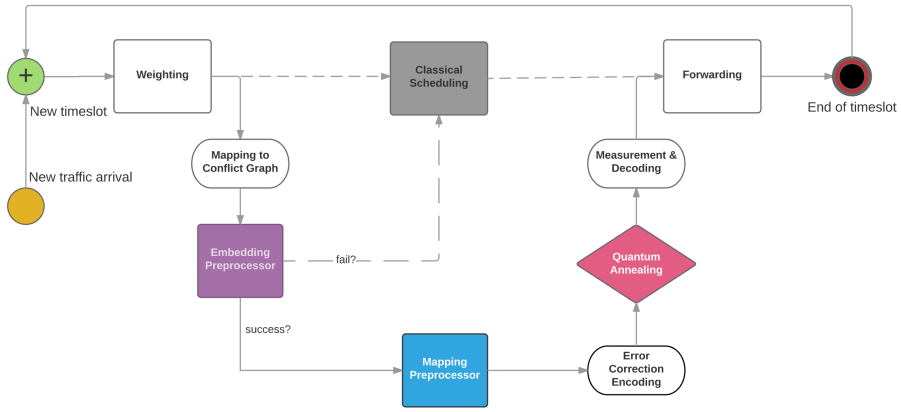

Fig. 1: Proposed workflow of quantum wireless application. The workflow features the standard 3-step process of weighting, scheduling and forwarding discussed in detail in Section 3, with the D-Wave "bypass" of classical heuristics expanded upon in Sec. 6.

2.1 Network interference

One of the fundamental problems in multi-hop wireless networks is network scheduling. Signal-to-Interference-plus-Noise Ratio (SINR) has to be maintained above a certain threshold to ensure successful decoding of information at the destination. For example, IEEE 802.11b requires minimum SINR of 4 and 10 dB corresponding to 11 and 1Mbps channel [19]. Consequently, only a subset of edges in a network can be activated at the same time, since every link transmission causes interference with nearby link transmissions. The fundamental mechanism in IEEE 802.11 uses the Distributed Coordination Function (DCF), which attempts to access wireless medium in a distributed way and backs off for a random time following exponential distribution. Despite its simplicity to implement, DCF has serious drawbacks mainly for its poor throughput [20, 21].

Different networks and protocols usually use different interference models. The most commonly used model is the 1-hop interference model (node exclusive model), in which only one link among those sharing a node in common can be activated in the same timeslot, with the restriction extended to every node.

There are two major models for analyzing network interference: one is the graph based model for solving the Weighted Maximum Independence Set (WMIS) problem on a conflict graph [14–18], the other for optimizing the geometric-based SINR [9–12]. The former is sometimes argued as being an overly idealistic assumption; however, the Maximum Independent Set (MIS) problem is still involved in the latter model [13] and is of interest in its own right. Thus, we base our abstraction on solving the MIS in a centralized scheduler in Medium Access Control (MAC) layer.

To summarize, here, the centralized scheduler, aware of the global network topology, performs best-effort scheduling on a timeslot basis in order to

1. maximize the number of simultaneous transmissions from non-interfering stations (throughput optimality);
2. minimize average network delay (19).

2.2 Weighted Maximum Independent Set (WMIS)

To give formal definitions, the network is abstracted as a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where the vertex set $\mathcal{V}$ denotes the transceiver nodes (including sources and sinks $d \in \mathcal{D} \subset \mathcal{E}$) and the edge set $\mathcal{E}$ denotes the wireless links. Let $\delta_S(u, v)$ denote the hop distance between $u, v \in \mathcal{V}$. Consider edges $e_u, e_v \in \mathcal{E}$ and let $\partial e_u = \{u_1, u_2\}, \partial e_v = \{v_1, v_2\}$. We then define

$$\delta(e_u, e_v) = \min_{i,j \in 1,2} \delta_S(u_i, v_j) \tag{1}$$

to be the distance between edges. Similar to the definition in [8], a subset of edges $\mathcal{E}'$ is said to be valid subject to the K-hop interference model if, for all $e_u, e_v \in \mathcal{E}'$ with $e_u \neq e_v$, we have $\delta(e_u, e_v) \geq K$. Let $\boldsymbol{\mathcal{S}}_K$ denote the set of subsets $\mathcal{E}' \subset \mathcal{E}$ that are K-hop valid.

Let $w_{\ell \in \mathcal{E}}$ be the wireless networking link weights. They are usually related to the queue differential at the end nodes of the link (Eqs. (5), (10)), with the convention that the destination node is a sink with vanishing queue occupancy. The weight w_ℓ therefore indicates the need to "service" the link ℓ. The weighting makes the whole difference between the BP and the various HD protocols. Regardless of the particular weighting, the network scheduling under the K-hop interference model is

$$\mathcal{E}'_{\text{opt}} = \arg \max_{\mathcal{E}' \in \boldsymbol{\mathcal{S}}_K} \sum_{\ell \in \mathcal{E}'} w_\ell. \tag{2}$$

In the $K = 1$ case, the problem is a max-weight matching problem and thus has polynomial time solution (Edmonds' blossom algorithm [38]). However, for the case $K > 1$, the problem is proved to be NP-hard and non-approximable [8]. In most cases, the network scheduling problem has to be solved in every timeslot during network operation; thus, the time complexity of the exact scheduling problem becomes critical.

Max-Weight scheduling by solving a Weighted Maximum Independence Set problem is a proved classical throughput optimal algorithm [22], that is, the scheduling set can stabilize all traffic arrival rates that are within the capacity region. However, in real-world applications, such algorithm is unrealistic due to its NP-hardness and time constraints. Instead, heuristics are widely used and well-studied, such as greedy style Longest-Queue-First (LQF) algorithm [1–6], random access algorithm [7] and classical probabilistic algorithms including simulated annealing, genetic algorithm, etc., discussed in more detail in Section 4.1. LQF algorithm, in particular, has been claimed to achieve satisfactory throughput optimality in K-hop interference model [4, 6], and is guaranteed to achieve at least $1/6$ of optimal throughput for K-hop, and $1/4$ of it for 2-hop [4] with less than 20 nodes.

3 Dirichlet protocol—specific weighting

Least Path Routing is simple to implement in wireline networks, but subject to heavy congestion at the centroid of the network if the network is negatively curved [70]. However, in wireless sensor networks, the global topology information is not generally available to every node, so that access to a global routing table is no longer a valid assumption. Thus, a dynamic routing protocol referred to as Backpressure (BP) [22] routing has been proposed. It achieves maximum throughput in the presence of varying network topology without knowing neither arrival rates nor global topology. There are, however, other protocols that are throughput optimal and that might in addition have other optimal properties.

3.1 Preamble: Heat Diffusion

Heat Diffusion (HD) protocol, originally proposed in [71], is a dynamic routing protocol with the unique feature that it mimics the heat diffusion process on a capacitated graph using information only from neighboring nodes. The graph is capacitated in the sense that the flow through the link ij is bounded by μ_{ij}, referred to as ***link capacity***. It is proved that HD stabilizes the network for any rate matrix in the interior of the capacity region. The Heat Diffusion protocol is briefly formulated as follows: At timeslot k, let $Q_i^{(d)}(k)$ denote the number of d-packets (those packets bound to destination $d \in \mathcal{V}$) queued at the network layer in node i. HD is designed along the same 3-stage process as BP: *weighting-scheduling-forwarding.*

– **HD Weighting:** At each timeslot k and for each link ij, the algorithm first finds the optimal d–packets to transmit as

$$Q_{ij}^{(d)}(k) = \max\left\{0,\ Q_i^{(d)}(k) - Q_j^{(d)}(k)\right\}, \tag{3}$$

$$d_{ij}^*(k) = \arg\max_{d\in\mathcal{D}} Q_{ij}^{(d)}(k).$$

To attribute a weight to each link, the HD algorithm performs the following:

$$\widehat{f_{ij}}(k) = \min\left\{\left\lceil 1/2\ Q_{ij}^{(d^*)}(k)\right\rceil,\ Q_i^{(d^*)}(k),\ \mu_{ij}(k)\right\}, \tag{4}$$

$$w_{ij}(k) = \left(\widehat{f_{ij}}(k)\right)\left(Q_{ij}^{(d^*)}(k)\right), \tag{5}$$

where $\widehat{f_{ij}}(k)$ denotes the number of packets that would be transmitted from i to j if the link ij were activated by the scheduling phase.

– **HD Scheduling:** After assigning the optimal weight (5) to each link, the scheduling set $\mathcal{S}(k)$ at timeslot k is chosen in a non interference set $\boldsymbol{\mathcal{S}}_{K=1}$ as in Eq. (2),

$$\mathcal{S}(k) = \arg\max_{\mathcal{E}'\in\boldsymbol{\mathcal{S}}_{K=1}} \sum_{\ell\in\mathcal{E}'} w_\ell, \tag{6}$$

where scheduling set comprises the set of links to be activated satisfying the hop constraint defined in Section 2.1.

– **HD Forwarding:** Subsequent to the scheduling stage, each activated link transmits $\widehat{f_{ij}}\ (k)$ number of packets in accordance with (4).

3.2 Dirichlet protocol

The Dirichlet protocol (a variant of HD) was originally proposed in [72]. It involves a link cost factor $\rho_{ij}^{(d)}(k)$, the cost of transmitting a d-class packet along link ij at time slot k. Define $\mathcal{D}_{ij}(k)$ to be the set of d-classes such that $Q_i^{(d)}(k) - Q_i^{(d)}(k) =: Q_{ij}^{(d)}(k) > 0$. Specifically, the Dirichlet protocol minimizes the Dirichlet routing energy

$$\bar{R} = \limsup_{K\to\infty} \frac{1}{K} \sum_{k=0}^{K-1} \mathbb{E}\left(\sum_{ij\in\mathcal{E}} \sum_{d\in\mathcal{D}_{ij}(k)} \rho_{ij}^{(d)}(k) \left(f_{ij}^{(d)}(k)\right)^2 \right), \tag{7}$$

where $\mathbb{E}$ denotes the expectation relative to arrival statistic. As a corollary, it minimizes the average queue occupancy (Eq. 19), itself proportional, by Little's theorem, to the average delay.

– **Dirichlet Weighting:** The Dirichlet weighting proceeds from the problem of finding $\widehat{f_{ij}^{(d)}}(k)$ to minimize

$$\sum_{d\in\mathcal{D}_{ij}(k)} \left(\rho_{ij}^{(d)}(k)^{-1} Q_{ij}^{(d)}(k) - \widehat{f_{ij}^{(d)}}(k) \right)^2, \tag{8}$$

subject to

$$\sum_{d\in\mathcal{D}_{ij}(k)} \widehat{f_{ij}^{(d)}}(k) \le \mu_{ij}(k) \text{ and } 0 \le \widehat{f_{ij}^{(d)}}(k) \le Q_{ij}^{(d)}(k). \tag{9}$$

Then the weight to each class $d \in \mathcal{D}_{ij}(k)$ is assigned as:

$$w_{ij}^{(d)}(k) := 2\rho_{ij}^{(d)}(k)^{-1} Q_{ij}^{(d)}(k) \widehat{f_{ij}^{(d)}}(k) - \left(\widehat{f_{ij}^{(d)}}(k) \right)^2 \tag{10}$$

and the final link weight is

$$w_{ij}(k) = \sum_{d\in\mathcal{D}_{ij}(k)} w_{ij}^{(d)}(k). \tag{11}$$

– **Dirichlet Scheduling:** It is the same as the HD scheduling with 1-hop interference model.
– **Dirichlet Forwarding:** Subsequent to the scheduling stage, each activated link transmits a number $\widehat{f_{ij}^{(d)}}(k)$ of d-packets.

4 Simulated versus quantum annealing

4.1 Simulated Annealing (SA)

Among all classical heuristics, simulated annealing (SA) is the most important one. First proposed by Kirkpatric *et al.* [39], SA emulates the process of first melting a solid by heating it up and then slowly cooling it down to the lowest-energy state of pure lattice structure. SA is designed as a general probabilistic algorithm for searching a cost function minimum formulated as a metaphoric lowest energy state. SA is formulated as follows:

$$p_{k+1} = \begin{cases} 1 & \text{if } f(x^{k+1}) < f(x^k), \\ \exp\left(-\frac{f(x^{(k+1)}) - f(x^k)}{T(k)}\right) & \text{otherwise,} \end{cases} \tag{12}$$

where $T(k)$ denotes the temperature at iteration step k and is monotone decreasing with k, $f(x)$ denotes the cost function, and p_{k+1} denotes the probability of accepting state x^{k+1} at iteration $k+1$. A properly chosen set of parameters is essential to obtain good results from SA, with the cooling schedule $T(k)$ being one of the most important ones. SA has been well studied as applied to MIS both as a standalone problem [23, 24] and as its applications to wireless networks [25, 26]. It is claimed that simulated annealing is superior to other competing methods with experimental instances of up to $70,000$ nodes [23].

Several other classical heuristics have also been applied to the MIS problem, including neural networks [27–29], genetic algorithm [30–32], greedy randomized genetic search [33], Tabu search [34–37]. However, throughout this paper, SA will be our primary benchmark technique.

4.2 Quantum Annealing (QA)

Adiabatic Quantum Computation (AQC), a subcategory of quantum computing first proposed in [51], and later physically implemented by D-Wave, maps a Quadratic Unconstrained Binary Optimization (QUBO) problem defined as

$$\begin{gathered} \min_X E(x_1, x_2, ..., x_N) = c_0 + \sum_{i=1}^{N} c_i x_i + \sum_{1 \le i < j}^{N} c_{ij} x_i x_j, \\ x_i \in \{0, 1\}, \end{gathered} \tag{13}$$

to the problem of computing the ground state of the Ising network

$$H_{\text{Ising}} = \sum_{i=1}^{N} h_i \sigma_i^z + \sum_{1 \le i < j}^{N} J_{ij} \sigma_i^z \sigma_j^z, \tag{14}$$

where $\sigma_i^z = I^{\otimes(i-1)} \otimes \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} \otimes I^{\otimes(N-i)}$ is the computationally-relevant [67] z-Pauli operator of spin-i, J_{ij} the coupling between spin i and spin j, and h_i is the

static bias field applied to spin i. The annealer prepares an initial transverse magnetic field, an equal superposition of 2^N computational basis states, as

$$H_{\text{trans}} = -\sum_{i=1}^{N} \sigma_i^x, \tag{15}$$

where $\sigma_i^x = I^{\otimes(i-1)} \otimes \frac{1}{2}\begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix} \otimes I^{\otimes(N-i)}$. During adiabatic evolution, the Hamiltonian evolves smoothly from H_{trans} to H_{Ising} with the scheduler $s(t)$ monotonically increasing from $s(0) = 0$ to $s(t_f) = 1$,

$$H(t) = (1 - s(t))H_{\text{trans}} + s(t)H_{\text{Ising}}, \quad s \in [0, 1]. \tag{16}$$

From the adiabatic theorem [67], if the evolution is "slow enough," the system would remain in its ground state. Thus, the solution of the original QUBO problem could be obtained by measurement on the Ising problem with a certain probability of success.

QUBO is widely studied and applied in many research fields that feature optimization, graphical models, Bayesian networks, etc. One of the specific areas is the computer vision approach that involves minimizing energy functions. Felzenszwalb [40] provides an insightful survey of the applications of QUBO in computer vision. QUBO is proved to be NP-hard. There is some evidence that the D-Wave quantum computer gives a modest speed-up over classical solvers for QUBO problems, and may provide a large speed-up for *some* instances of QUBO problems [42]. Recently, on D-Wave 2X with 1152 qubits, the speedup reaches up to three orders of magnitude for a subset of scenarios in multiple query optimization problems [44].

5 Adiabatic Quantum Computation Applications

In this short review, we consider only two D-Wave machines, those on which we have evaluated the Dirichlet wireless scheduling algorithms.

5.1 D-Wave II

D-Wave launched D-Wave Two with 512 physical qubits. Two applications to quantum annealing emerged, both led by a group from NASA Ames.

One such application is Bayesian network structure learning [46], where for the first time sufficiency of suboptimal solution is proposed, with the claim that global optimum is not required. They also studied penalty weights and pointed to probable problem of analog control error caused by precision constraints. However, by claiming that only 7 logical qubits could be embedded, no experimental results were shown.

The other application is the operational planning problem [45]. For the first time, three high-level research challenges were identified, namely **(1)** Finding appropriate hard problems suitable for quantum annealing, **(2)** Mapping to QUBO with good choices of parameters, **(3)** Minor embedding into hardware. They used qubits in the

range of $8-16$, with expected annealing runs in the range of $10-10000$ to reach ground state with 99% certainty.

The same NASA Ames group applied quantum annealing to electric power system fault detection [47] on D-Wave II, and again emphasized that analog control error is a major obstacle in working with real world problem-defined graphs. They utilized as many as 340 physical qubits, reaching typically 9000 ST99 measure (expected repetitions to reach ground state with 99% certainty). They compared results with classical exact solver and claimed a comparative speedup. However, general quantum speedup was not claimed.

A group as USC also applied graph isomorphism problem to D-Wave II [48], which involves reducing baseline Hamiltonian to a more compact Hamiltonian. The solved problem size is as large as 18 logical qubits, with ST99 time around 1 second in the 18-qubit case.

5.2 D-Wave 2X

D-Wave launched D-Wave 2X with 1152 physical qubits. The first application on D-Wave 2X was database optimization [44], and they claimed to have found a subset of problems that demonstrated a speedup up to three orders of magnitude and for the first time used over 1000 qubits. However, another group at NASA while testing the new machine on deep learning application [49] claimed that no speedup was found compared to other competing classical heuristics.

Interestingly, in late 2015, Google announced a result [86] claiming a 10^8 speedup on D-Wave 2X compared to classical simulated annealing and simulated quantum annealing based on carefully crafted 'artificial problems.' However, this claim has been challenged in academia and is still in debate, for the same reason mentioned before: 1) the problem artificial nature and 2) suboptimal annealing time on easier cases may be hiding the exponential scaling.

6 Embedding preprocessor and mapping preprocessor

Here we essentially follow the "bypass" of the classical scheduling as shown in Fig. 1. The two preprocessors implement the various steps needed to convert the wireless scheduling problem to a format amenable to AQC.

6.1 Mapping to conflict graph

Given $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, the corresponding conflict graph $\mathcal{G}_C = (\mathcal{V}_C = \mathcal{E}, \mathcal{E}_C)$ has its vertex set equal to the edge set of $\mathcal{G}$ and its edge set consisting of those pairs (e_u, e_v) of $\mathcal{E}$ edges such that their distance as defined by (1) in $\mathcal{G}$ is $\leq K$. Also, we set the vertex weight in $\mathcal{G}_C$ to be the link weight in $\mathcal{G}$ and the edge weight in $\mathcal{G}_C$ is set to 1, to denote a violation of the interference constraints. The time complexity of such conversion process is $O(|\mathcal{E}|^2)$.

Now, by solving the WMIS of the conflict graph, we solve the network scheduling problem of the original graph. As first proposed in [80], the WMIS problem can be formulated as the QUBO problem of finding the minimum of the binary function:

$$f(x_1, ..., x_N) = -\sum_{\ell \in \mathcal{V}_C} c_\ell x_\ell + \sum_{k\ell \in \mathcal{E}_C} J_{k\ell} x_k x_\ell. \tag{17}$$

The binary variable $x_\ell = 1$ if node ℓ in $\mathcal{G}_C$ (link ℓ in $\mathcal{G}$) is in the WMIS (edge ℓ of $\mathcal{G}$ is activated) and 0 otherwise. c_ℓ is set to the (Dirichlet) weight w_ℓ of link ℓ in $\mathcal{G}$. $J_{k\ell}$ is a penalty for allowing links k and ℓ of $\mathcal{G}$ to be simultaneously activated.

It is proved [80] that a sufficient condition for the ground state of the Hamiltonian to be the optimal solution to the WMIS problem is that $J_{k\ell} > \min(c_k, c_\ell)$ for $k\ell \in \mathcal{E}_C$. Thus, the energy spectrum of the QUBO problem relates to the spectrum of the corresponding Ising formulation in Eq. (14), with the ground state energy of the Ising problem giving the solution to the network scheduling problem.

Note that the network scheduling problem has an Ising Hamiltonian with natural two-body interaction. Thus, unlike some other applications [40, 50], additional reduction is not needed. Such many-body reductions would cause sizable overhead, resulting in the practical problem size to be very small due to hardware constraints.

6.2 Minor embedding preprocessor

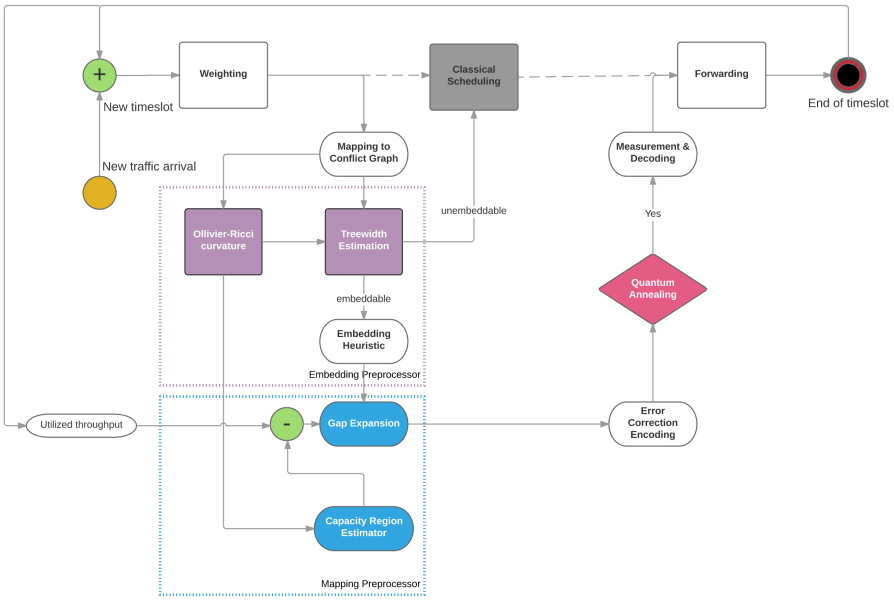

Fig. 2: Complete picture of proposed framework with amplification of the embedding preprocessor and the mapping preprocessor

The next block in the scheduling bypass of Fig. 1 is the embedding preprocessor, which in its simplest interpretation would embed the problem graph $\mathcal{G}_C$ in the hardware architecture graph $\mathcal{H}$. The latter requires that the problem graph be a subgraph of the architecture graph. For general problems, this is a very strong requirement, since the hardware graph is fixed. In the D-Wave architecture, *minor embedding* instead of *subgraph embedding* is used to allow 1-to-many vertex mapping [80]. By properly adjusting the coupling strengths of particular edges and nodes [81], more than one physical qubits can represent the same logical qubit, thus greatly increasing the range of graphs that can be minor embedded to a fixed hardware graph, at the cost of using more resources (more physical qubits).

The ***minor embedding*** $\phi : \mathcal{G}_C \to \mathcal{H}$ is defined such that (i) each vertex v in $\mathcal{G}_C$ is mapped to a connected subtree T_v of $\mathcal{H}$; (ii) for each $vw \in \mathcal{E}_C$, there are corresponding $i_v \in T_v$ and $i_w \in T_w$ with $i_v i_w \in \mathcal{E}_\mathcal{H}$. Crucially related to minor embedding is the concept of ***tree decomposition*** T of $\mathcal{G}_C$. Each vertex $i \in I$ of the tree T abstracts a subset $\mathcal{V}_i$, called a "bag," of vertices of $\mathcal{G}_C$ such that (i) $\cup_{i\in I}\mathcal{V}_i = \mathcal{V}_C$; (ii) for any $vw \in \mathcal{E}_C$, there is a $i \in I$ such that $v, w \in \mathcal{V}_i$; (iii) for any $v \in \mathcal{V}_C$ the set $\{i \in I : v \in \mathcal{V}_i\}$ forms a connected subtree of T. The ***width*** of a tree decomposition is $\max_i(|\mathcal{V}_i| - 1)$. The ***treewidth*** τ is the minimum width over all tree decompositions. A theorem, crucial to rule out some candidate embeddings, says that necessary for existence of an embedding $\mathcal{G}_C \to \mathcal{H}$ is that $\tau(\mathcal{G}_C) \leq \tau(\mathcal{H})$. Since the tree width is difficult to compute, we derived an estimate of it based on the Ollivier-Ricci curvature [84, 85]. The embedding preprocessor block of Fig. 1 is amplified in Fig. 2, which shows how tree width and Ollivier-Ricci curvature in particular lead to a new heuristic embedding.

Different heuristic embeddings and different runs of them, if successful, will lead to different minor embedded graphs. Different embeddings do not significantly affect the final results, since the gap expansion—instrumental in making the method work—is decoupled from the minor-embedding. Thus, multiple embeddings were tried and results remained similar.

6.2.1 Error correction

In Jul. 2015, Vinci *et al.* proposed error correction in conjunction with minor embedding [60]. By solving encoded problems experimentally, significant improvements on minor-embedded instances are detected. There also exist other error correction efforts on adiabatic quantum computing, in particular Vinci [61] and Mishra [62] both in late 2015.

6.3 Gap expansion to favor noninterference solutions

Here, by "gap," we do not specifically mean the classical minimum gap between the ground energy level and the first excited one along the adiabatic evolution, but rather the gap between energy evolution curves satisfying the interference constraints and those violating such constraints. This is with the hope that the ground energy (max weight) curve would be nonviolating and well separated from the violating

curves. Another motivation for this separation is to make the computation insensitive to analog errors.

We try to achieve the aforementioned goals by properly setting the h_ℓ and $J_{k\ell}$ terms in Eq. (17). We introduce a scaling factor $\beta_{k\ell}$ that multiplies the quadratic part of the QUBO formulation and as such scales the various terms so as to put more penalty weight on the independence constraints if $\beta_{k\ell} > 1$:

$$f(x_1, ..., x_N) = -\sum_{\ell \in \mathcal{V}_C} c_\ell x_\ell + \sum_{k\ell \in \mathcal{E}_C} \beta_{k\ell} J_{k\ell} x_k x_\ell. \tag{18}$$

In theory, $\beta_{k\ell} = 1$ would suffice as long as $J_{k\ell} > \min(c_k, c_\ell)$, as the ground state has already encoded the correct solution of the WMIS problem [80]. However, since measuring the ground state correctly is not guaranteed, increasing $\beta_{k\ell}$ becomes necessary to enforce the independence constraints, so that the energy spectrum of the non-independence states is raised to the upper energy spectrum and the feasible energy states are compressed to the lower spectrum.

Figure 5 of [83] illustrates this concept via the intuitive idea of plotting the energy levels versus s. However, as emphasized in [65], the deeper justification of the separation of the energy levels is to be found in the numerical range of $H_{\text{trans}} + \jmath H_{\text{Ising}}(\beta)$. For small β, the numerical range is highly singular, mixing the various energy levels, whereas as β increases, the numerical range becomes "dis-singularized" and the energy levels are better separated.

There exist several strategies in setting heavier penalty weights to expand the gap. Set

$$J_{k\ell} := \max(c_k, c_\ell),\ \forall k\ell \in \mathcal{E}_C; \quad J_{\max} := \max_{k\ell \in \mathcal{E}_C} J_{k\ell}.$$

Then define

- **Global gap expansion:** Pick β_{global} and set $\beta_{k\ell} J_{k\ell} = \beta_{\text{global}} J_{\max}, \forall k\ell \in \mathcal{E}_C$.
- **Local gap expansion:** Pick $\beta_{k\ell} > 1, \forall k\ell \in \mathcal{E}_C$.

In the local adjustment [80], the constraint on $J_{k\ell}$ depends only on the fields at $\partial k\ell$, whereas in the global adjustment, contrary to [80], $J_{k\ell}$ depends on *all* fields.

The D-Wave II is subject to an Internal Control Error (ICE) that gives Gaussian errors with standard deviations $\sigma_{h_\ell} \approx 0.05$ and $\sigma_{J_{k\ell}} \approx 0.035$ [64]. Putting too large a $\beta_{k\ell}$ penalty would incur two problems: 1) The local fields would become indistinguishable and 2) The minimum evolution gap would become too small. Accordingly, a few parameter values have been tried out and the results are shown in Table 4, which corroborates the experimental results of Section 8 indicating that gap expansion would significantly influence the optimality of the returned solution.

7 Quality metrics

Among the wireless network protocols that have been demonstrated to be throughput-optimal (e.g., Backpressure and Heat-Diffusion), network delay came out as a metric that can be optimized subject to throughput optimality.

(**Average Network Delay**) Since Poisson arrival rate is commonly assumed in wireless network studies, the Dirichlet protocol minimizes the expected long-term time-averaged total queue congestion

$$\bar{Q} = \limsup_{K\to\infty} \frac{1}{K} \sum_{k=0}^{K-1} \mathbb{E}\left(\sum_{i\in\mathcal{V}} \sum_{d\in\mathcal{D}} Q_i^{(d)}(k) \right), \tag{19}$$

where $Q_i^{(d)}(k)$ is the d-packets occupancy at queue i at time slot k. By Little's theorem, $\bar{Q}$ is proportional to the long-term averaged node-to-node network delay. Thus, it is sufficient to work with average queue occupancy over all nodes in the network.

(**Extended Throughput Optimality**) The classical solver is exact, that is, it has no interference constraint violations and it achieves the true maximum throughput. The quantum solver may or may not satisfy the interference constraints. In the former case (no violations), its quality factor is defined as the ratio of the quantum throughput and the exact throughput and is ≤ 1. In the latter case (violations), the quantum solver may or may not have its quality factor ≤ 1 and its definition is split into two cases:

$$\begin{cases} \text{no violations} \begin{cases} \text{Avg Opt} = \frac{\sum_{ij\in\mathcal{E}'_{\mathcal{S}}} f_{ij}}{\sum_{kl\in\mathcal{E}'_{\text{opt}}} f_{kl}} \leq 1, \end{cases} \\ \text{violations} \begin{cases} \text{Avg Opt} = \frac{\sum_{ij\in\mathcal{E}''_{\mathcal{S}}} f_{ij}}{\sum_{kl\in\mathcal{E}'_{\text{opt}}} f_{kl}} \leq 1, \\ \text{Avg Opt} = 1 - \frac{\sum_{ij\in\mathcal{E}''_{\mathcal{S}}} f_{ij}}{\sum_{kl\in\mathcal{E}'_{\text{opt}}} f_{kl}} \leq 0, \end{cases} \end{cases} \tag{20}$$

where f_{ij} denotes the forwarding amount as defined in Section 2.2, $\mathcal{E}'_{\mathcal{S}}$ denotes the set of edges in the scheduling set $\mathcal{S}$ computed by QUBO without violations, $\mathcal{E}''_{\mathcal{S}}$ the same set but with violations, and $\mathcal{E}'_{\text{opt}}$ denotes the set of edges in the *optimal* scheduling set solved by the exact solver, without violations.

(**ST99[OPT]**) Along the line of other benchmarking methods [41–43], we define a slight variant of speed measure, ST99(OPT), as the expected number of repetitions to reach at least a certain optimality level OPT with 99% certainty,

$$ST99[\mathrm{OPT}] = \frac{\log(1-0.99)}{\log(1-P_{\mathrm{OPT}})}, \tag{21}$$

where P_{OPT} is the probability of reaching a state with at least OPT optimality. Note that this is of practical significance for time-sensitive problems like wireless network scheduling, where enough time might not be available within a timeslot for the quantum annealer to reach the ground state.

8 Results: D-Wave II versus D-Wave 2X

8.1 Experimental setup

It is known that SA can boost the overall performance of classical heuristic algorithms in solving WMIS problems [23]. Thus, it is worth comparing the latter with

Table 1: Search range for best parameters for simulated annealing

	Range
Number of sweeps	400-10000
Number of repetitions	300-5000
Initial temperature	0.1-3
Final temerature	3-13
Scheduling Type	linear/exponential

Table 2: Problem size of Erdös-Rényi random graphs used in experiment. The link capacities were setup as $\mu_{ij} = \infty, \forall ij$. The third column represents the actual physical qubits used after minor embedding.

	Problem Size	QUBO Size	Physical Qubits
Graph1	15	31	164
Graph2	20	57	405
Graph3	25	68	759
Graph4	30	84	783

our QA results. Here, we adapt a highly optimized Simulated Annealing algorithm (an_ss_ge_fi_vdeg) from reference [82], compiling the C++ source code with gcc 4.8.4 with MATLAB® C-mex API. Additionally, a wide range of parameters, shown in Table 1, were tested to ensure near optimal performance of the algorithm within a reasonable run time.

We performed experiments on both D-Wave II and D-Wave 2X, and showed that QA has an advantage over SA after gap expansion—thus a potential general quantum speedup. We also showed that performance is improved significantly in D-Wave 2X compared to its predecessor. At the time the experiment was done, lower level API was not made available; thus, parameters like anneal length were all default values set by D-Wave.

Table 2 shows the parameters of the 4 randomly generated Erdös-Rényi graphs being tested on D-Wave II and D-Wave 2X, where the link capacities μ_{ij} were all set to ∞. The "problem size" is the order $|\mathcal{V}|$ of the problem (wireless network) graph, the "QUBO size" represents $|\mathcal{E}|$, and the "physical qubits" are those nodes utilized after minor embedding in the architecture $\mathcal{H}$. With the effect of minor-embedding, a significantly large proportion of all available qubits are utilized (402 out of 502 on D-Wave II for Graph 2 and 783 out of 1098 on D-Wave 2X for Graph 4).

8.2 Average network delay

In Figure 3, we show that D-Wave 2X gains an advantage over D-Wave II, which itself gains an advantage over SA in terms of average network delay after gap expansions in Graph 2 test case. (The conflict graphs of Graph 3 and Graph 4 could not be

minor embedded in the limited architecture of D-Wave II.) In the experiment, should the lowest energy solution not conform to the K-hop interference model, we skip the timeslot and no forwarding is allowed. The benefit afforded by D-Wave 2X is conjectured to be its better analog error control.

In Fig. 4, we compare various solutions in terms of delay for Graph 2 and Graph 3 on D-Wave 2X. Again, gap expansion appears essential for QA and SA to be competitive with the exact solver. After gap expansion, QA shows an advantage over SA.

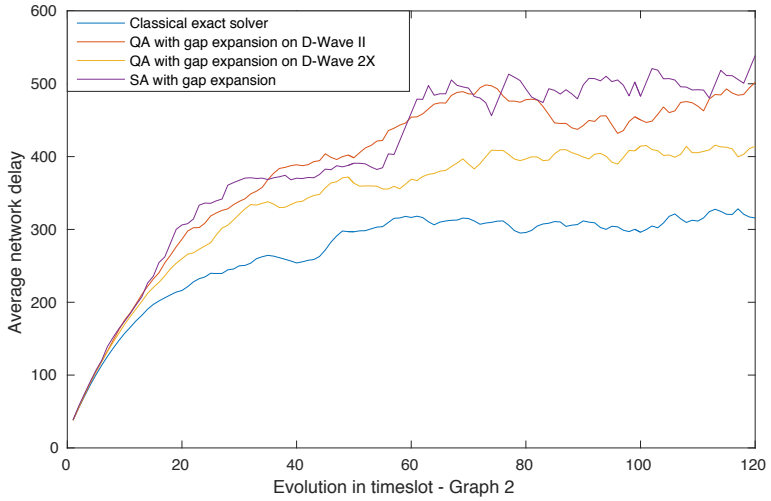

Fig. 3: D-Wave 2X noise improvement compared to D-Wave II. Noise reduction is sufficient enough to demonstrate a quantum advantage over SA after gap expansion.

8.3 Throughput optimality

As already said, the solutions given by the QA and SA solvers may or may not satisfy the independent constraints and, in the latter case, the transmission is skipped. It is of vital importance to determine how this affects throughput optimality as defined in Section 7. The ratio of Eq. (20) may exceed 1 because solutions that do not satisfy the independent condition may have a bigger total weight, in which case the Avg Opt is redefined as negative as a warning. We refer to those solutions do not satisfy the independent condition as violations. From Table 3, we can see that energy compression directly results in less violations of the independent condition in the case of QA rather than SA, especially with D-Wave 2X. This could explain the performance improvement afforded by D-Wave 2X.

There is a relatively large number of violations in SA even after gap expansion. This agrees with our previous knowledge about SA that it is designed to find the ground state and it is not optimized to search for the sub-optimal results. This is

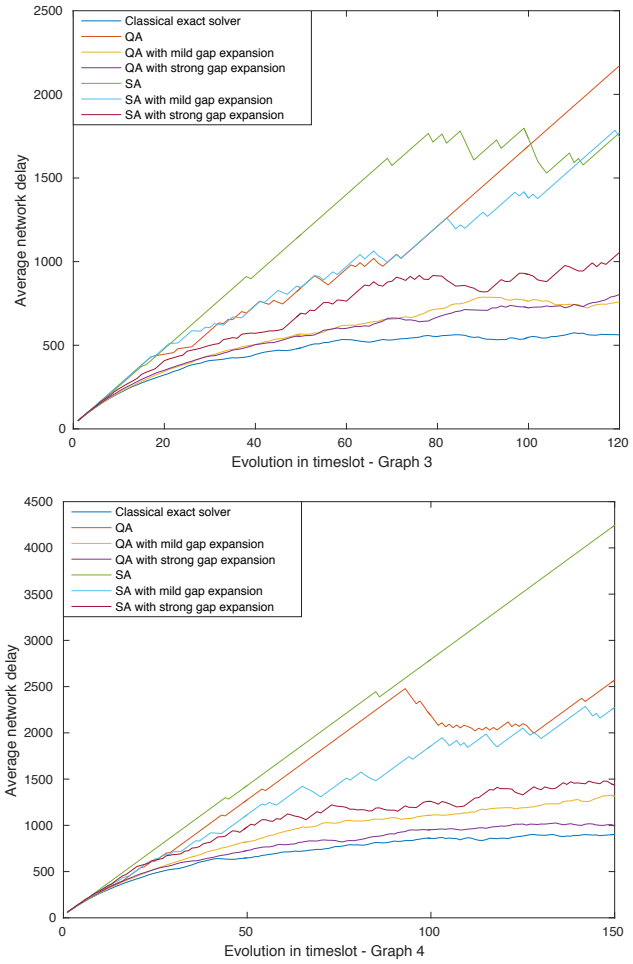

Fig. 4: D-Wave 2X: Network delay of classical exact algorithm, quantum annealing, quantum annealing after gap expansion with mild and strong strategies (representing local and global adjustment, respectively). Simulated annealing are performed on two networks randomly generated also with mild and strong strategies. The top one is a graph with 25 nodes and 68 edges, the bottom one is with 30 nodes and 84 edges, both randomly generated using Erdös-Rényi model.

especially true for larger Graph 4, of which SA violates the independence constraint in almost all cases (146 out of 150).

8.4 ST99[OPT]

From optimality data returned from D-Wave, we plot ST99 related to level of optimality as shown in Fig. 5. Note that our ST99 definition relies on optimality level, which could typically be 80% or 90% depending on user's needs. In network setup,

Table 3: Optimality & violation performance after gap expansion for Graphs 2,3, and 4

	G2 Number of violations (out of 120)	G2 Average optimality (without violations)
QA on D-Wave II	13	0.8129
QA on D-Wave 2X	0	0.8746
SA	31	0.9935
	G3 Number of violations (out of 120)	G3 Average optimality (without violations)
QA on D-Wave 2X	0	0.8737
SA	98	0.7434
	G4 Number of violations (out of 150)	G4 Average optimality (without violations)
QA on D-Wave 2X	0	0.8714
SA	146	0.7613

this optimality of classical heuristic relies heavily on topology (Ollivier-Ricci curvature [76–79]) of such network and traffic rate model. Again, gap expansion is indispensable to obtain competitive results.

8.5 Throughput optimality versus delay versus gap expansion

Although gap expansion in itself is a classical method, we showed that such procedure can help QA improve its results, and thus help demonstrate potential quantum advantage over SA. In Table 4, we show how setting the penalty weight β in local or global approach would affect the quality of the returned solutions. We found that setting $\beta_{\mathrm{global}} = 1$ would yield the best performance so far. We do not have a quantitative explanation for the wrong solutions; potential explanations on small problems have been discussed in [41–43]. Intuitively, as the problem size grows, it is much more difficult even to find close to ground states; indeed, as the penalty weight grows too large, the local fields begin to vanish, thus making the problem effectively more difficult since all weights have to be scaled to $[-2, +2]$. The problem of quantitative connection among quality measure, network stability, and throughput optimality remains open.

9 Conclusion

We have further developed the complete Adiabatic Quantum Computation (AQC) approach to wireless network scheduling, already proposed in [83] and depicted in Fig. 2, by comparing the newer D-Wave 2X versus the older D-Wave II results. The newer machine allows for simulations on higher order graphs (G3, G4) afforded by

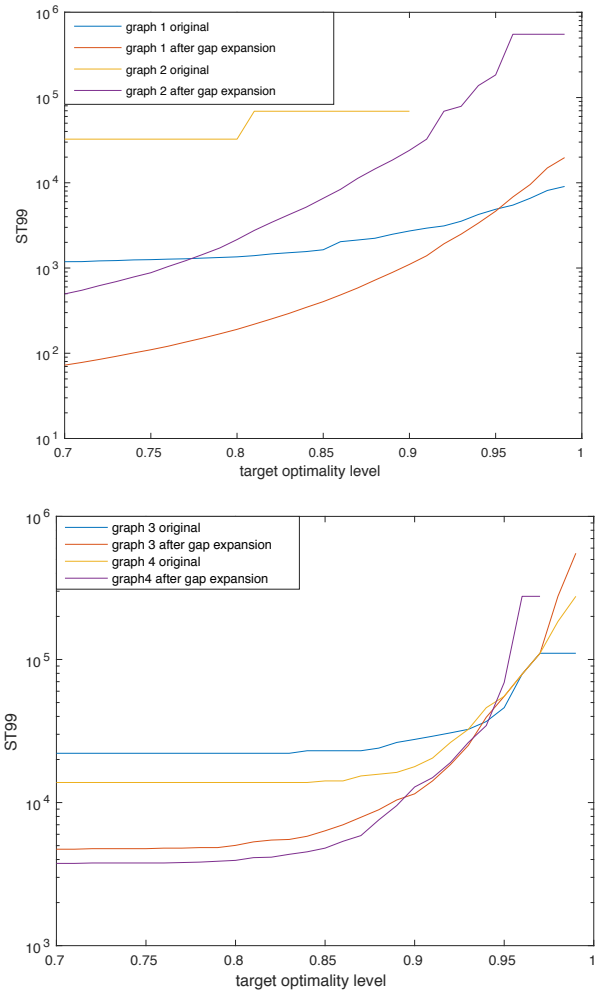

Fig. 5: ST99, defined as $\frac{\log(1-0.99)}{\log(1-P_{\mathrm{OPT}})}$ relative to required optimality specified by network, before and after gap expansion on four test graphs. Note that probability is calculated based on all solutions returned by D-Wave; thus ST99 of 10^3 corresponds to one set of annealing runs, which in our case costs 20ms. The curve for graph 2 ends at 0.9 optimality because there is no solution that satisfies such optimality after a total of $120,000$ annealing runs. Also note that the top figure is run on D-Wave II while the bottom figure is run on D-Wave 2X.

the 1152 qubits versus the 512 qubits of the older machine. Remarkably, even on higher order graphs (G3, G4), the "gap expanded" Quantum Annealing (QA) runs on D-Wave 2X result in no interference constraint violations, while on smaller graphs (G1,G2) D-Wave II still had some violations. Across the board (G2-G4), Simulated Annealing (SA) experienced violations. On a small order graph (G2), SA was showing some optimality level advantages over QA, which disappeared on higher order

Table 4: Penalty weight with different β setup and resulting quality measure of returned solution averaged over 120 timeslots. Graph 2 is a larger size instance than Graph 1. Delay refers to average network delay in steady state, and NC refers to the non-convergent case where steady state is not reached within tested time span.

	Avg. Opt. - G1	Delay - G1	Avg. Opt. - G2	Delay - G2
$\beta_{kl} = 1 + \epsilon$	0.781	384.9	0.077	NC
$\beta_{kl} = 2$	0.928	376.3	0.312	1213
$\beta_{\text{global}} = 1$	0.974	268	0.741	566.1
$\beta_{\text{global}} = 1.5$	-0.165	NC	-0.331	NC
$\beta_{\text{global}} = 2$	-0.185	NC	-0.356	NC
	Avg. Opt. - G3	Delay - G3	Avg. Opt. - G4	Delay - G4
$\beta_{kl} = 1 + \epsilon$	0.8585	743	0.8478	1168.9
$\beta_{\text{global}} = 1$	0.8737	762.9	0.8714	975.8

graphs on D-Wave 2X. Probably the better analog control error on the newer contributes to this significant improvement.

Fundamentally, the scheduling part of this AQC approach to wireless network scheduling aims at solving the Weighted Maximum Independent Set (WMIS) problem *in general,* and thus can be trivially applied to other problems involving WMIS.

By comparing QA with SA, though omitting Quantum Monte Carlo (QMC), we have strengthened our earlier finding [83], claiming a potential comparison point where QA outperforms SA in the sense of benefit from gap expansion. Although, as seen from Table 3, SA has an optimality advantage in non-violation cases in test case G2, it is however interesting to observe that among suboptimal solution cases QA has less violations than SA. For larger graphs tested on D-Wave 2X, SA lost such only advantage in terms of optimality.

It is also important to notice that the algorithm on D-Wave 2X "scales up" better than on its predecessor. In Table 3, the optimality level for test graphs ranging from size 405 to 783 remains at the very close to optimality level of 0.87.

Despite encouraging results, due to the very limited experimental data, we cannot positively assert a general sizable advantage of QA against SA. However, it is our hope that such study could be the inspiration for future general speedup demonstrations.

References

1. Dimakis, A., Walrand, J.: Sufficient conditions for stability of Longest-Queue-First scheduling: Second-order properties using fluid limits. Advances in Applied Probability. 38(2), 505521 (2006)
2. Joo, C., Lin, X., Shroff, N.: Understanding the capacity region of the greedy maximal scheduling algorithm in multi-hop wireless networks. IEEE/ACM Trans. Netw. 17(4), 1132–1145 (2009)
3. Zussman, G., Brzezinski, A., Modiano, E.: Multihop local pooling for distributed throughput maximization in wireless networks. INFOCOM'08, Phoenix, Arizona (2008)
4. Leconte, M., Ni, J., Srikant, R.: Improved bounds on the throughput efficiency of greedy maximal scheduling in wireless networks. MOBIHOC'09, 165-174 (2009)

5. Li, B., Boyaci, C., Xia, Y.: A refined performance characterization of longest-queue-first policy in wireless networks. ACM MOBIHOC, New York, NY, USA, 6574 (2009)
6. Brzezinski, A., Zussman, G., Modiano, E.: Distributed throughput maximization in wireless mesh networks via pre-Partitioning. IEEE/ACM Trans. Netw. 16(6), 14061419 (2008)
7. Proutiere, A., Yi Y., Chiang, M.: Throughput of random access without message passing. 42nd Annual Conference on Information Sciences and Systems, Princeton, NJ, USA, 509-514 (2008)
8. Sharma, G., Mazumdar, R., Shroff, N.: On the complexity of scheduling in wireless networks. MobiCom'06, Proceedings of the 12th Annual International Conference on Mobile Computing and Networking, Los Angeles, CA, 227-238 (2006)
9. Blough, D.M., Resta, G., Sant, P.: Approximation algorithms for wireless link scheduling with SINR-based interference. IEEE Transactions on Networking, 18(6), 1701-1712 (2010)
10. Chafekar, D., Anil Kumar, V.S., Marathe, M.V., Parthasarathy, S., Srinivasan, A.: Capacity of wireless networks under SINR interference constraints. Wireless Networks, 17, 1605-1624 (2011)
11. Moscibroda, T., Wattenhofer, R., Zollinger, A.: Topology control meets SINR: the scheduling complexity of arbitrary topologies. MobiHoc'06, ACM, Florence, Italy, 310-321 (2006)
12. Gupta, P., Kumar, P.R., The capacity of wireless networks. IEEE Transactions on Information Theory, 46(2), 388-404 (2000)
13. Andrews, M., Dinitz, M.: Maximizing capacity in arbitrary wireless networks in the SINR model: complexity and game theory. INFOCOM'09, Rio de Janeiro, 1332-1340 (2009)
14. Jain, K., Padhey, J., Padmanabhan, V.N., Qiu, L.: Impact of interference on multi-hop wireless network performance. MobiCom '03, ACM, San Diego, California, USA, 66-80 (2003)
15. Alicherry, A., Bhatia, R., Li, L.E.: Joint channel assignment and routing for throughput optimization in multiradio wireless mesh networks. IEEE Journal on Selected Areas in Communications, 24(11), 1960-1971 (2006)
16. Kodialam, M., Nandagopal, T.: Characterizing the capacity region in multi-radio multi-channel wireless mesh networks. MobiCom'05, ACM, Cologne, Germany, 73-87 (2005)
17. Sanghavi, S.S., Bui, L., Srikant, R.: Distributed link scheduling with constant overhead. ACM SIGMETRICS, 35(1), 313-324 (2007)
18. Wan, P.J.: Multiflows in multihop wireless networks. MobiHoc'09, New Orleans, LA, USA, 85-94 (2009)
19. Official homepage of the IEEE 802.11 working group, http://www.ieee802.org/11
20. Bianchi, G.: Performance analysis of the IEEE 802.11 distributed coordination function. IEEE Journal on Selected Areas in Communications, 18, 535547 (2000)
21. Cali, F.: Dynamic tuning of the IEEE 802.11 protocol to achieve a theoretical throughput limit. IEEE/ACM Transactions on Networking, 8, 785799 (2000)
22. Tassiulas, L., Ephremides, A.: Stability properties of constrained queuing systems and scheduling policies for maximal throughput in multihop radio networks. IEEE Trans. Autom. Control, 37(12), 19361948 (1992)
23. Homer, S., Peinado, M.: Experiments with polynomial-time clique approximation algorithms on very large graphs. Cliques, Coloring, and Satisfiability: Second DIMACS Implementation Challenge, volume 26 of DIMACS Series. American Mathematical Society, Providence, RI, (1996)
24. Xu, X., Ma, J., An, H.W.: Improved simulated annealing algorithm for the Maximum Independent Set problem. Intelligent Computing, Volume 4113 of the series Lecture Notes in Computer Science, 822-831 (2006)
25. Kim, Y.G., Lee, M.G.: Scheduling multi-channel and multi-timeslot in time constrained wireless sensor networks via simulated annealing and particle swarm optimization. IEEE Communications Magazine, 52(1), 122-129 (2014)
26. Mappar, M., Rahmani, A.M., Ashtari, A.H.: A new approach for sensor scheduling in wireless sensor networks using simulated annealing. ICCIT '09. Fourth International Conference on Computer Sciences and Convergence Information Technology, Seoul, Korea, 746-750 (2009)
27. Grossman, T.: Applying the INN model to the max clique problem. Cliques, Coloring, and Satisfiability: Second DIMACS Implementation Challenge, volume 26 of DIMACS Series. American Mathematical Society, Providence, RI, (1996)
28. Jagota, A., Approximating maximum clique with a Hopfield network. IEEE Trans. Neural Networks, 6, 724735 (1995)
29. Jagota, A., Sanchis, L., Ganesan, R.: Approximately solving maximum clique using neural networks and related heuristics. Cliques, Coloring, and Satisfiability: Second DIMACS Implementation Challenge, volume 26 of DIMACS Series. American Mathematical Society, Providence, RI (1996)

30. Bui, T.N., Eppley, P.H.: A hybrid genetic algorithm for the maximum clique problem. In Proceedings of the 6th International Conference on Genetic Algorithms, Pittsburgh, PA, 478484 (1995)
31. Hifi, M.: A genetic algorithm - based heuristic for solving the weighted maximum independent set and some equivalent problems. J. Oper. Res. Soc., 48, 612622 (1997)
32. Marchiori, E.: Genetic, iterated and multistart local search for the maximum clique problem. In Applications of Evolutionary Computing, volume 2279 of Lecture Notes in Computer Science, 112121. Springer-Verlag, Berlin, (2002)
33. Feo, T.A., Resende, M.: A greedy randomized adaptive search procedure for maximum independent set. Operations Research, 42, 860878 (1994)
34. Battiti, R., Protasi, M.: Reactive local search for the maximum clique problem. Algorithmica, 29, 610637 (2001)
35. Friden, C., Hertz, A., de Werra, D.: Stabulus: A technique for finding stable sets in large graphs with tabu search. Computing, 42, 3544 (1989)
36. Mannino, C., Stefanutti, E.: An augmentation algorithm for the maximum weighted stable set problem. Computational Optimization and Applications, 14, 367381 (1999)
37. Soriano, P., Gendreau, M.: Tabu search algorithms for the maximum clique problem. Cliques, Coloring, and Satisfiability: Second DIMACS Implementation Challenge, volume 26 of DIMACS Series. American Mathematical Society, Providence, RI, 1996.
38. Edmonds, J.: Paths, trees, and flowers. Canad. J. Math. 17, 449467 (1965)
39. Kirkpatrick, S., Gelatt Jr, C.D., Vecchi, M.P.: Optimization by simulated annealing. Science 220 (4598), 671680 (1983)
40. Felzenszwalb, P.F.: Dynamic programming and graph algorithms in computer vision. IEEE Transactions on Pattern Analysis and Machine Intelligence, 33(4), 721-740 (2011)
41. Boixo, S., Rnnow, T.F., Isakov, S.V., Wang, Z., Wecker, D., Lidar, D.A., Martinis, J.M., Troyer, M.: Quantum annealing with more than one hundred qubits. Nature Phys. 10(3) (2013)
42. Rnnow, T.F., Wang, Z., Job, J., Boixo, S., Isakov, S.V., Wecker, D., Martinis, J.M., Lidar, D.A., Troyer, M.: Defining and detecting quantum speedup. Science 345(6195), 420-424 (2014)
43. Hen, I., Job, J., Albash, T., Rnnow, T.F., Troyer, M., Lidar, D.A.: Probing for quantum speedup in spin glass problems with planted solutions. Physical Review A 92(4), 042325 (2015)
44. Trummer, I., Koch, C.: Multiple query optimization on the D-Wave 2X adiabatic quantum computer. Proceedings of the VLDB Endowment, 9(9), 648-659 (2016)
45. Rieffel, E.G., Venturelli, D., O'Gorman, B., Do, M.B., Prystay, E., Smelyanskiy, V.N.: A case study in programming a quantum annealer for hard operational planning problems. Quantum Information Processing, 14(1), 1-36 (2015)
46. O'Gorman, B., Babbush, R., Perdomo-Ortiz, A., Aspuru-Guzik, A., Smelyanskiy, V.: Bayesian network structure learning using quantum annealing: The European Physical Journal Special Topics, 224(1), 163-188 (2015)
47. Perdomo-Ortiz, A., Fluegemann, J., Narasimhan, S., Biswas, R., Smelyanskiy, V.N.: A quantum annealing approach for fault detection and diagnosis of graph-based systems. The European Physical Journal Special topics, 224(1), 131-148 (2015)
48. Zick, K.M., Shehab, O., French, M.: Experimental quantum annealing: case study involving the graph isomorphism problem. Scientific Reports (2015). http://dx.doi.org/10.1038/srep11168
49. Benedetti, M., Realpe-Gmez, J., Biswas, R., Perdomo-Ortiz, A.: Estimation of effective temperatures in a quantum annealer and its impact in sampling applications: A case study towards deep learning applications. Physical Review A 94(2), 022308 (2016)
50. Bian, Z., Chudak, F., Macready, W.G., Clark L., Gaitan, F.: Experimental Determination of Ramsey Numbers. Phys. Rev. Lett. 111, 130505 (2013)
51. Farhi, E., Goldstone, J., Gutmann, S., Sipser, M.: Quantum computation by adiabatic evolution. arXiv:quant-ph/0001106 (2000)
52. Deutsch, D.: Quantum theory, the Church-Turing principle and the universal quantum computer. Proceedings of the Royal Society of London A 400, 97117 (1985)
53. Shor, P.: Algorithms for quantum computation: Discrete logarithms and factoring. SIAM Journal on Computing, 26(5), 14841509 (1997)
54. Grover, L.: A fast quantum mechanical algorithm for database search. Proceedings of the 28th Annual ACM Symposium on the Theory of Computing, Philadelphia, PA, 212219 (1996)
55. Lanting, T., et al.: Entanglement in a quantum annealing processor. Phys. Rev. X 4, 021041, May (2014)
56. Boixo, S., et al.: Computational multiqubit tunnelling in programmable quantum annealers. Nature Communications 7, 10327 (2016)

57. Katzgraber, H., Hamze, F., Andrist, R.: Glassy chimeras could be blind to quantum speedup: Designing better benchmarks for quantum annealing machines. Phys. Rev. X 4, 021008 (2014)
58. King, J., et al.: Benchmarking a quantum annealing processor with the time-to-target metric. arXiv:1508.05087 (2015)
59. Pudenz, K., Albash, T., Lidar, D.A.: Error corrected quantum annealing with hundreds of qubits. Nature Comm. 5, 3243 (2014)
60. Vinci, W., et al.: Quantum annealing correction with minor embedding. Phys. Rev. A 92, 042310 (2015)
61. Vinci, W., Albash, T., Lidar, D.A.: Nested quantum annealing correction. npj, Quantum Information 2, 16017 (2016)
62. Mishra, A., Albash, T., Lidar, D.A.: Performance of two different quantum annealing correction codes. Quantum Information Processing, 15(2), 609-636 (2016)
63. Wu, K.J.: Solving practical problems with quantum computing hardware. ASCR Work-shop on Quantum Computing for Science, (2015) DOI: 10.13140/RG.2.1.3656.5200
64. King, A.D., McGeoch, C.C., Algorithm engineering for a quantum annealing platform. arXiv: 1410.2628 (2014)
65. Jonckheere, E.A., Rezakhani, A.T., Ahmad, F.: Differential topology of adiabatically controlled quantum processes. Quantum Information Processing, Special Issue on Quantum Control 12(3), 1515-1538 (2013)
66. Jonckheere, E.A., Ahmad, F., Gutkin, E.: Differential topology of numerical range. Linear Algebra and Its Applications, 279/1-3, 227-254 (1998)
67. Reichardt, B.W.: The quantum adiabatic optimization algorithm and local minima. STOC '04, Proceedings of the thirty-sixth Annual ACM Symposium on Theory of Computing, Chicago, IL, 502-510 (2004)
68. Akyildiz, I.F., Su, W., Sankarasubramaniam, Y., Cayirci, E.: A Survey on Sensor Network. IEEE Communication Magazine, 40(8), 102-114 (2002)
69. Karp, B., Kung, H.T.: GPSR: Greedy perimeter stateless routing for wireless networks. Proceedings ACM MobiCom'00, Boston, MA, 243-254 (2000)
70. Jonckheere, E., Lou, M., Bonahon, F., Baryshnikov, Y.: Euclidean versus hyperbolic congestion in idealized versus experimental networks. Internet Mathematics, 7(1), 1-27 (2011)
71. Banirazi, R., Jonckheere, E., Krishnamachari, B.: Heat diffusion algorithm for resource allocation and routing in multihop wireless networks. GLOBECOM, Anaheim, California, USA, 5915-5920 (2012)
72. Banirazi, R., Jonckheere, E., Krishnamachari, B.: Dirichlet's principle on multiclass multihop wireless networks: Minimum cost routing subject to stability. ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems, Montreal, Canada, 31-40 (2014)
73. Ghosh, P., Ren, He, Banirazi, R., Krishnamachari, B., Jonckheere, E.: Empirical evaluation of the Heat-Diffusion collection protocol for wireless sensor networks. Computer Networks (COMNET), 127, 217-232 (2017)
74. Banirazi, R., Jonckheere, E., Krishnamachari, B., Minimum delay in class of throughput-optimal control policies on wireless networks. American Control Conference (ACC), Portland, OR, 2668-2675 (2014)
75. Banirazi, R., Jonckheere, E., and Krishnamachari, B.: Heat diffusion optimal dynamic routing for multiclass multihop wireless networks. INFOCOM, Toronto, Canada, 325-333 (2014)
76. Ollivier, Y.: Ricci curvature on Markov chains on metric spaces. J. Funct. Anal. 256(3), 810-864 (2009)
77. Bauer, F., Jost, J., Liu, S.: Ollivier-Ricci curvature and the spectrum of the normalized graph Laplace operator. Mathematical Research Letters 19(6), 1185-1205 (2012)
78. Wang, C., Jonckheere, E., Banirazi, R.: Wireless network capacity versus Ollivier-Ricci curvature under Heat Diffusion (HD) protocol. American Control Conference (ACC 2014), Portland, OR, 3536-3541 (2014)
79. Wang, C., Jonckheere, E., Banirazi, R.: Interference constrained network performance control based on curvature control. 2016 American Control Conference, Boston, USA, 6036-6041 (2016)
80. Choi, V.: Minor-embedding in adiabatic quantum computation: I. The parameter setting problem. Quantum Information Processing, 7, 193–209 (2008)
81. Choi, V.: Minor-embedding in adiabatic quantum computation: II. Minor-universal graph design. Quantum Information Processing, 10(3), 343-353 (2011)
82. Isakov, S.V., Zintchenko, I.N., Rnnow, T.F., Troyer, M.: Optimized simulated annealing for Ising spin glasses. Computer Physics Communications, 192, 265-271 (2015)

83. Wang, C., Chen, H., Jonckheere, E.: Quantum versus simulated annealing in wireless interference network optimization. Scientific Reports, 6, 25797 (2016)
84. Wang, C., Jonckheere, E., Brun, T.: Ollivier-Ricci curvature and fast approximation to tree-width in embeddability of QUBO problems. ISCCSP, Athens, Greece, 639-642 (2014)
85. Wang, C., Jonckheere, E., Brun, T.: Differential geometric treewidth estimation in adiabatic quantum computation. Quantum Information Processing, 15(10), 39513966, (2016)
86. Denchev, V., et al.: What is the computational value of finite range tunneling. Physical Review X, 6, 031015 (2016)
87. Boxio, S., et al.: Characterizing quantum supremacy in near-term devices. Nature Physics, (2018) https://doi.org/10.1038/s41567-018-0124-x
88. Childs, A.M., Maslov, D., Yunseong Nam, Ross, N.J., Yuan Su: Towards the first quantum simulation with quantum speedup. arXiv:1711.10980v1 [quant-ph] 29 Nov 2017].
89. Perdomo-Ortiz, A., Benedetti, M., Realpe-Gomez, J., Biswas, R.: Opportunities and challenges for quantum-assisted machine learning in near-term quantum computers. arXiv:1708.09757v2 [quant-ph] 19 Mar 2018.
90. Farhi, E., Goldstone, J., Gutmann, S., Neven, H.: Quantum algorithms for fixed qubit architecture. arXiv:1703:06199v1 [quant-ph] 17 Mar 2017.
91. Albash T., and Lidar, D.A.: Adiabatic quantum computation. Rev. Mod. Phys. 90, 015002 (2018)