



## Practical Quantum Computing: An Update

---

### WHITEPAPER

---

2020-09-25

#### Overview

This report presents a high-level overview of the annealing-based quantum computers manufactured by D-Wave. A survey of hundreds of application problems that have been implemented to run on D-Wave processors is also presented, together with a review of some promising early demonstrations of superior performance from hybrid quantum/classical solvers, compared to purely classical alternatives. The report presents an overview of the new Advantage™ quantum computer and of the hybrid solver service (HSS), which can be used to tackle large-scale optimization problems arising in practice. Future technological developments from D-Wave are also briefly discussed.

#### CONTACT

**Corporate Headquarters**  
3033 Beta Ave  
Burnaby, BC V5G 4M9  
Canada  
Tel. 604-630-1428

**US Office**  
2650 E Bayshore Rd  
Palo Alto, CA 94303

**Email:** [info@dwavesys.com](mailto:info@dwavesys.com)

[www.dwavesys.com](http://www.dwavesys.com)



## Notice and Disclaimer

D-Wave Systems Inc. (“D-Wave”) reserves its intellectual property rights in and to this document, any documents referenced herein, and its proprietary technology, including copyright, trademark rights, industrial design rights, and patent rights. D-Wave trademarks used herein include D-WAVE®, Leap™ quantum cloud service, Ocean™, Advantage™ quantum system, D-Wave 2000Q™, D-Wave 2X™, and the D-Wave logo (the “D-Wave Marks”). Other marks used in this document are the property of their respective owners. D-Wave does not grant any license, assignment, or other grant of interest in or to the copyright of this document or any referenced documents, the D-Wave Marks, any other marks used in this document, or any other intellectual property rights used or referred to herein, except as D-Wave may expressly provide in a written agreement.

# Summary

This report presents an overview of the annealing-based quantum computers manufactured by D-Wave; reviews some recent technological advances; surveys the variety of applications that have been successfully implemented to run on D-Wave products; and reviews what is known about performance. The report ends with some evidence-based predictions about future developments in this region of the quantum computing space. Some highlights:

- Section 2 presents a high-level introduction to the physics behind annealing-based quantum computation, the architecture of D-Wave quantum computers, and typical problem-solving workflows for users of D-Wave quantum systems.
- Section 3 describes features of two of the newest D-Wave products, the Advantage™ quantum processing unit (QPU) and the hybrid solver service (HSS):
  - Containing over 5000 qubits, the Advantage QPU is by far the largest and most powerful quantum computer available today. Not only can Advantage read inputs that are over twice as large as those that can fit on predecessor, the D-Wave 2000Q, it can also find better solutions to problems that are small enough to fit on both.
  - The HSS contains a portfolio of hybrid heuristic solvers that can leverage the best features of quantum and classical computation, to tackle inputs that are much larger than the QPU alone can handle. The latest version of HSS can read inputs with up to 20,000 variables if fully connected, or up to one million variables if not fully connected.
- Section 3 also reviews a few of the 250+ application problems that have been successfully implemented and run on D-Wave QPUs and hybrid solvers. The section concludes with a brief discussion of what is known about performance, including some promising early reports that hybrid computation methods can outperform purely classical alternatives on problems arising in real-world applications.
- The report concludes with a brief discussion of future technological developments from D-Wave.

# Contents

1	Introduction	1
2	Overview	3
2.1	The physics . . . . .	3
2.2	The quantum machine instruction . . . . .	4
2.3	Workflows for quantum problem solving . . . . .	5
3	Applications and performance update	8
3.1	Applications . . . . .	9
3.2	Performance . . . . .	11
3.3	Summary . . . . .	12
4	The future	13
5	Concluding remarks	15
	References	15

# 1 Introduction

Much of the public confusion concerning the current state of quantum computing comes from mixed messages about what is possible in theory versus what can be achieved in reality. The amazing theoretical capabilities of quantum computers — leading to more efficient drug designs, new discoveries in materials science, and revolutions in encryption technologies [1–3] — are based on having registers of qubits that can exhibit exotic properties, such as superposition and entanglement, that are not available to registers of classical bits. However, those seemingly omnipotent quantum computation resources are fragile in the face of noise and also difficult to control.

The challenge before every organization currently attempting to realize practical quantum computing is to build quantum hardware that is sufficiently robust against errors and can operate with sufficient control fidelity to outperform the world’s most powerful classical computers at tackling some of the world’s most challenging computational problems. Furthermore, the quantum computer must be large enough that the problems it can solve are truly challenging: when inputs are small, classical computation is plenty fast and there is no room for improvement. The technical hurdles to realizing robust performance at industrial-relevant scales are daunting — and some argue that they are insurmountable.

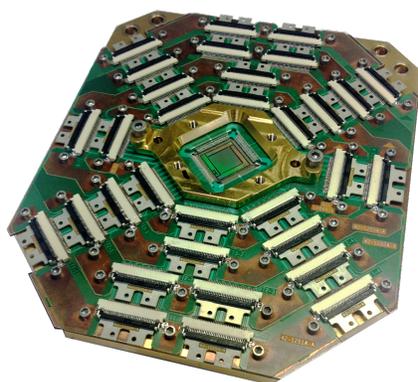
There are two dominant approaches in play for achieving practical quantum computing (QC) at scale: gate model (GM), and quantum annealing (QA).<sup>1</sup> Superficially, these approaches are often portrayed as being completely unrelated. However, careful examination reveals that the models can be equivalent in power [4]. Nonetheless, GM and QA have become markedly different approaches in practice, which has considerable bearing on their prospects for achieving practical advantage over classical computers. D-Wave is the only company that builds, sells, and sells time on annealing-based quantum computers.<sup>2</sup>

For over a decade, D-Wave has focused on delivering practical annealing-based quantum computers aimed at solving NP-hard problems. A somewhat restricted version of QA was selected for this purpose because it is more robust against noise than known approaches to GM. Although the restricted model is general enough to express any Turing-computable function, this choice meant temporarily foregoing some of the grander goals of QC, such as exactly solving the Schrödinger equation for a large number of electrons. The company intends to remove this restriction at some future date.

The strategic decision to prioritize increasing qubit and coupler counts and to develop a strong user base, over implementing the fully-general computational model, provided the impetus for subsequent developments, such as successful deployments of commercial systems starting in 2011 (with the D-Wave One), a more than 32-fold increase in qubit counts in

<sup>1</sup>The related terms *adiabatic quantum computing* and *annealing-based quantum computing* are often used to refer to the same general concept.

<sup>2</sup>Reasonable people can disagree as to whether a quantum annealer is properly called a “quantum computer.” We refer here to the full system containing quantum and classical components (since a circuit alone does not qualify as a computer), and the fact that in principle, the quantum annealing circuit implemented by D-Wave can compute any Turing-computable function.



**Figure 1:** An Advantage QPU, shown in the center of its mounting apparatus. It is a superconducting quantum circuit containing over 5000 qubits and 35,000 couplers. It operates at a temperature below 20 mK in a highly shielded chamber.

later systems (the current-generation Advantage system contains more than 5000 qubits), and the proliferation of more than 250 early applications that have been implemented and run on D-Wave systems and products (see [5]).

In September 2020, D-Wave announced general availability of the Advantage™ quantum computer, based on a quantum processing unit (QPU) that contains more than 5000 qubits and over twice the number of couplers per qubit than were available in previous systems. At the same time, D-Wave has released new solvers in the hybrid solver service (HSS), which combine quantum and classical computation methods to solve problems much larger than can fit on current-generation QPUs.

This article presents a high level introduction to the annealing-based approach to quantum computation, and provides a snapshot of where D-Wave currently stands and where we expect to go in our quest to achieve practical quantum advantage for a broad range of applications and use cases. Along the way, we highlight features of the Advantage QPU and recent upgrades to the HSS, which each represent a significant step forward in that journey:

- The Advantage QPU contains over 5,000 qubits and 35,000 couplers, more than doubling the size of inputs that can be read, compared to its predecessor, the D-Wave 2000Q. Not only can the Advantage QPU input bigger problems than the 2000Q QPU, it can also return better-quality solutions on industry-relevant application inputs. See [6] for details.
- The HSS combines quantum and classical computation methods to leverage the best properties of both, when tackling problems that are too big to fit on current-sized QPUs. The latest solvers in HSS incorporate the Advantage QPU and can solve problems with up to one million variables (or up to 20,000 variables if fully connected). Early performance studies suggest that HSS solvers can outperform state-of-the-art classical alternatives on problems that are relevant to applications practice. See [7] for more.

Section 2 presents a brief introduction to the physics of quantum annealing and a description of D-Wave quantum processors from the user's perspective. Section 3.1 describes new

features of the Advantage QPU, and new solvers in the HSS. The section also gives an overview of how D-Wave quantum computers have been put to use in practical applications, and surveys current understanding regarding their performance.

Looking to the future, Section 4 offers six evidence-based predictions pertaining to annealing-based quantum computing that we anticipate will come to pass within 2 to 5 years.

Visit the D-Wave website [dwavesystems.com](http://dwavesystems.com) to learn more about D-Wave products and systems, including the newest-generation Advantage system [6, 8]; the open source Ocean software development kit [9], and the HSS [7], part of the Leap<sup>2</sup> web portal [10]. Free public access to D-Wave quantum computers is available on a limited basis: visit the Leap website to learn more.

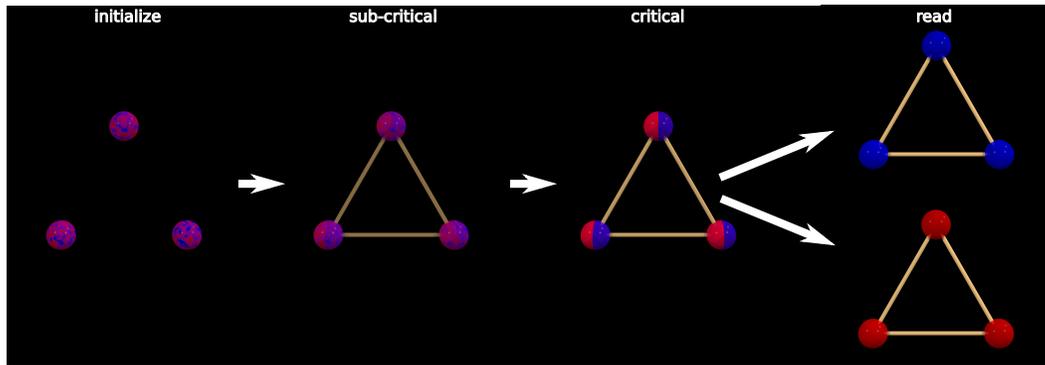
## 2 Overview

This section presents a high-level overview of the physics underlying the quantum annealing (QA) approach to computation; describes the QA architectures developed by D-Wave; and explains the quantum-solving workflow from a user's perspective.

### 2.1 The physics

Several sources are available that describe the physics of QA in detail [11]. For brevity, all that will be stated here is that QA is a means of harnessing the physics of quantum phase transitions for performing computation. Here, a phase transition is defined as a discrete change in some macroscopic property of a physical system that has been induced by tuning an external control parameter. Within the context of QA as implemented by D-Wave, the physical system is a network of qubits that are pairwise coupled and biased in a way that encodes an NP-hard problem of interest. The objective is to find a configuration of +1's and -1's that can be assigned to the set of qubits such that the cost function of the NP-hard problem is minimized. Such a configuration is referred to as a ground state.

As illustrated in Figure 2, The method involves initializing the qubits in a quantum state that broadly explores all possible solutions and then slowly ramping up the relative strength of the NP-hard problem couplings and biases. One iteration of this process is referred to as an anneal. At some *critical point* during the anneal (there may be more than one), the system can undergo a phase transition from a disordered quantum paramagnetic phase to an ordered classical magnetic phase. The final state after passing through the phase transition is then a potential solution to the NP-hard problem. The key physics that one hopes to harness in QA is the formation of entangled quantum states with long-range correlations between the individual qubit states upon approaching the phase transition, which can enhance the probability of finding the system in a ground state at the end of the anneal.



**Figure 2:** Qubits are represented by colored spheres and interactions by solid bars. QA begins with all qubits uncoupled and their states completely uncorrelated, as indicated by mixed colors (+1 = red, -1 = blue) on each sphere. As QA progresses, the interactions are slowly added and correlations between qubits become more apparent. At some critical point in the anneal, the states of the qubits become strongly correlated. In this simple example, there are two ground states corresponding to all-red (+1, +1, +1) or all-blue (-1, -1, -1). Upon approaching the critical point from the disordered phase, the system is in an equal superposition of the two ground states, indicated by spheres that are half red/half blue. In passing through the critical point, the system is forced to randomly choose to be in one of those ground states, which is subsequently read.

## 2.2 The quantum machine instruction

The QPU of a D-Wave quantum computer operates on a collection of qubits to carry out the annealing process described above. Like all quantum computers being built today, the QPU runs within a classical framework resembling the familiar von Neumann architecture containing an arithmetic logic unit (ALU), memory, I/O, and a control unit. The QPU can be thought of as an ALU-like coprocessor with an arrangement of qubits and pairwise couplers driven by a specialized control system. The network of qubits and couplers is programmed using a powerful and flexible *quantum machine instruction* (QMI) that is defined by a set of parameters that specify the desired output and how to obtain it. The output consists of a vector  $S = \{S_i\}$  of spin values  $S_i \in \{\pm 1\}$ .

The most important component of the QMI is the specification of the desired output according to input vectors  $(h, J)$ . These are inputs to the NP-hard Ising model (IM) optimization problem: Given a graph  $G = (V, E)$  with fields  $h_i$  on vertices  $V$  and interactions  $J_{ij}$  on edges  $E$ , find a spin vector  $S$  that minimizes the objective function

$$E(S) = \sum_{(i,j) \in E} J_{ij} S_i S_j + \sum_{i \in V} h_i S_i. \quad (1)$$

To invoke a QMI the user provides the values for  $h = \{h_i\}$  and  $J = \{J_{ij}\}$ , and an anneal time interval  $t_a$ . Furthermore, as with all real-world quantum computing today, there is a chance that the calculation will fail to return an optimal solution, due to physical limitations of the implementation or interference from external noise. This means that a single execution of the QMI may not return a ground state solution every time; thus it is prudent and cost-effective to repeat the anneal many times for each input. For this reason the user must also specify  $R$ , the number of solutions to be returned.

A variety of annealing protocols are supported through additional parameters to the QMI,

called *anneal path features*. For example, the anneal can be specified as a piecewise linear waveform that can be used to alter the evolution in the vicinity of a phase transition. It is also possible to adjust the individual qubit annealing schedules to a limited degree by specifying *anneal offsets*. The *reverse anneal* protocol allows the user to specify initial qubit values and explore nearby solutions. A description of the latest D-Wave QMI and variations, and of parameters for the QPU infrastructure (such as post-processing utilities) can be found at [12].

Note that in contrast to GM, programming an annealing-based QPU does not involve writing out step-by-step instructions for accomplishing the task at hand. Instead, the user specifies the desired result — an optimal spin assignment for the objective function defined by inputs  $(h, J)$  — together with parameters specifying how to accomplish the result, and the quantum algorithm implemented in hardware does the rest. This indicates that the natural programming paradigm for QA is declarative, rather than imperative, in nature.

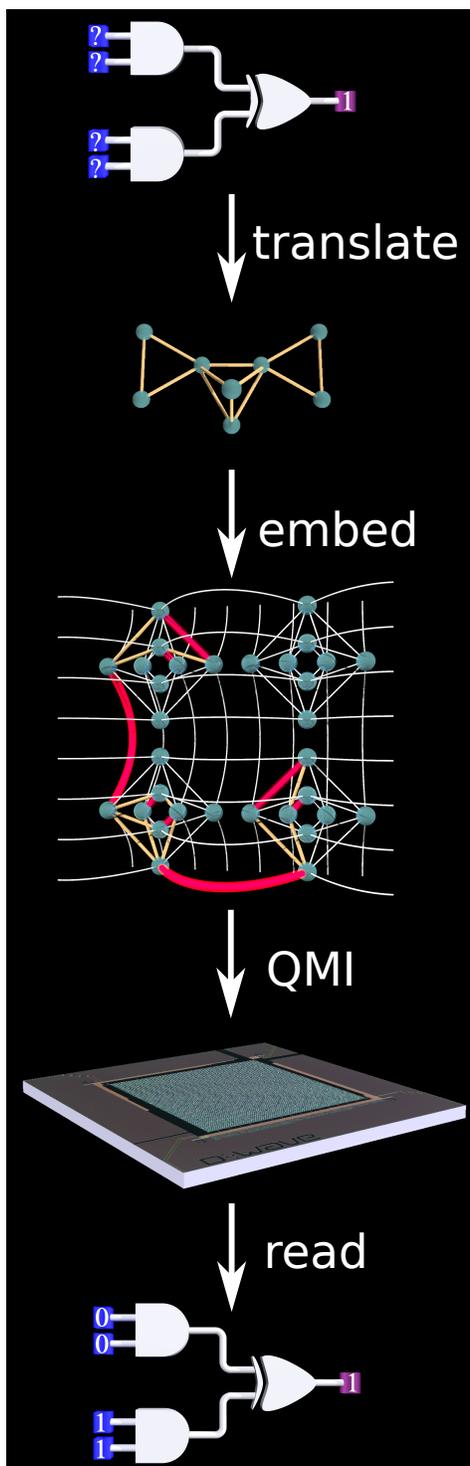
Examples of declarative programming languages in classical computation include Prolog for logic programming and SQL for computations on databases. Declarative languages such as GAMS and AMPL are often used together with optimization library packages such as CPLEX and Gurobi. Pakin [13] describes a compiler that translates programs written in Prolog into an IM form that is suitable for execution on a D-Wave platform.

The declarative paradigm makes possible a simple and scalable interface for using the QPU, which is an important component of D-Wave's efforts to make quantum computers widely accessible.

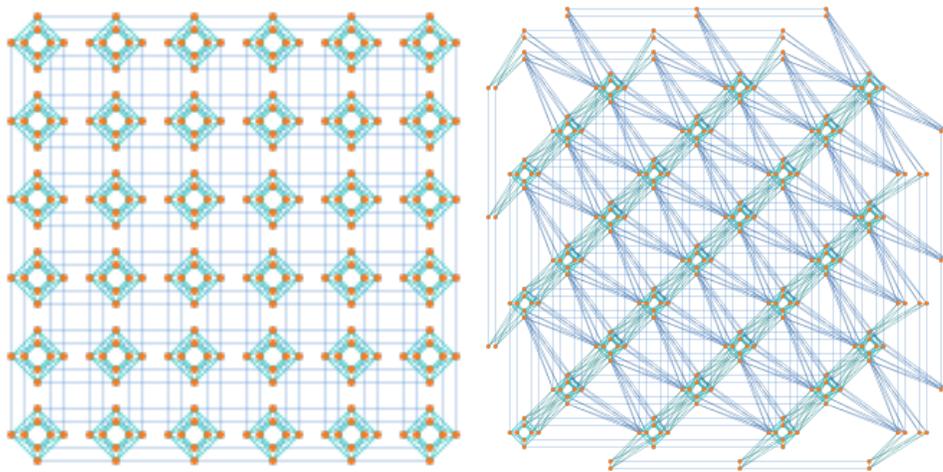
## 2.3 Workflows for quantum problem solving

The core operation of the QPU is to return a sample of  $R$  solutions to a given IM input  $(h, J)$ . Any NP-hard problem can be translated to IM using well-understood methods from NP-completeness theory, which means that in principle this core operation can be applied to those problems as well. Solving a given application problem  $P$  using the quantum processor involves a few discrete tasks, as described below and illustrated in Fig. 3.

- **Translate inputs for  $P$  to inputs for IM or QUBO.** The IM problem is defined above for spin values  $S_i = \pm 1$ ; the quadratic unconstrained binary optimization (QUBO) problem is equivalent but defined on binary values  $B_i \in \{0, 1\}$ . The QPU interface accepts either format. Note that some problems are more suitable for this approach than others, depending on how much input expansion is created by the translation.
- **Decompose or distill the problem.** If a translated input turns out to be too big to fit on the QPU, the problem can be decomposed by breaking into pieces that are solved separately, or else distilled by merging nodes of a larger graph to obtain a smaller one with similar global structure. This approach does have its limits, in the sense that a given problem instance may be less suitable if only a small part of its overall structure can be represented on the quantum hardware.
- **Minor-embed the input.** An IM or QUBO input is defined by weights  $(h, J)$  assigned to the vertices and edges of general graph  $G = (V, E)$ . These weights must be mapped to the qubits (vertices) and couplers (edges) inside a D-Wave QPU, which do not have general connectivity.



**Figure 3:** Steps in the problem-solving workflow. A input for the NP-hard *circuit satisfaction problem* is a description of a logic circuit: the goal is to find an assignment of inputs that makes the circuit evaluate to 1. This input is translated to a graph representation for IM or QUBO (weights are not shown here); then it is minor-embedded (a Chimera embedding is shown here). The extra red arcs are created during embedding. The QPU sends back a sample of results, which are translated into solutions to the original problem.



**Figure 4:** The Chimera (left) and Pegasus (right) graph topologies. The left panel shows a C6 Chimera graph, a  $6 \times 6$  grid of unit cells, containing 288 qubits. The right panel shows a P4 Pegasus graph, with unit cells on a diagonal grid, containing 264 qubits. Although both have about the same number of qubits, the more complex connection structure of Pegasus is clearly seen. The 2000Q QPU contains a C16 with over 2000 qubits and the Advantage QPU contains a P16 with over 5,000 qubits.

Mapping  $G$  onto an equivalent representation in the QPU involves a process called minor-embedding. Tools for minor-embedding are available in the D-Wave software library, although problem-specific custom embedders can sometimes give better results.

Note that previous generations of D-Wave QPUs have used the *Chimera* connection topology shown in Figure 3. The Advantage QPU uses the *Pegasus* topology: these connection structures are shown side-by-side in Figure 4. Pegasus has more couplers-per-qubit Chimera, which means that minor-embedded inputs can fit more compactly into the QPU, with therefore fewer auxiliary components (such as the extra arcs marked in red in Figure 3) needed to represent the problem. As a result, the Advantage QPU can input larger problems, and can return better-quality solutions, than the 2000Q QPU.

- **Query the quantum processor.** An input of suitable format and size can be sent to the QPU together with appropriate parameter settings; note that well-chosen parameter settings can sometimes dramatically improve solution quality. Besides this one-shot approach to problem solving, the QPU can be incorporated in a hybrid approach that involves an iterated sequence of queries to the QPU, with classical computation that modifies  $(h, J)$  between queries. This approach is used, for example, during the training cycle in machine learning (ML) applications.
- **Return the results.** The raw solutions from the QPU are (in typical use) improved by a quick post-processing step and mapped back to their unembedded form. Solutions in IM/QUBO representation must also be translated back to their original problem formulation.

## 3 Applications and performance update

This section starts with a brief overview of the Advantage QPU as well as new solvers in the HSS, both available starting in September 2020. More information about these new products may be found in two companion technical reports [6, 7], which are summarized below.

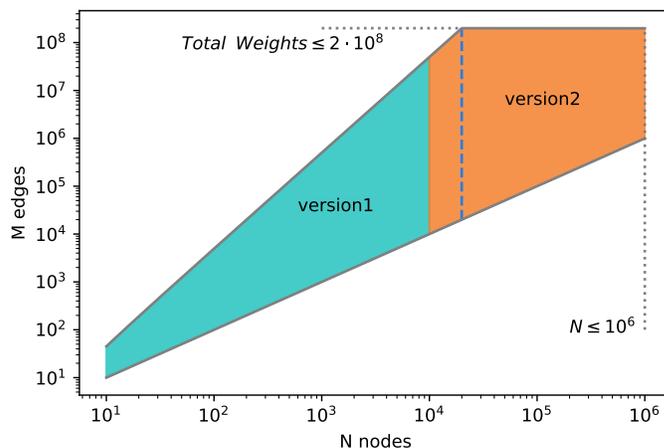
Section 3.1 describes the variety of real-world application problems and use cases that have been demonstrated to run on D-Wave QPUs. Section 3.2 gives a brief overview of what is known about performance of the quantum and hybrid solvers in D-Wave's portfolio.

**Advantage new features** The most important differences between the Advantage QPU and its predecessor, the 2000Q QPU, are the increased qubit counts and the upgrade from Chimera to the Pegasus connection topology, as shown in Figure 4. These changes make Advantage by far the largest and most powerful quantum computer in existence, a fact that has important consequences for practical use (see [6, 14] for details):

- The Advantage QPU contains at least 5,000 qubits and about 2.5 times more couplers per qubit than the 2000Q QPU. Depending on problem structure, Advantage can hold inputs that are typically between 2 and 4 times larger than those on the 2000Q system.
- Not only can Advantage read larger application inputs, it can find better solutions to inputs that are small enough to fit on both. In one case study, the Advantage system found optimal solutions 6 times faster in wall clock time (or 30 times faster if considering pure anneal time); in other studies it found better solutions between 7 and 16 times more often than the 2000Q QPU.

**Hybrid solver service** The HSS represents another step in implementing our strategy of building quantum systems that are both widely accessible and applications-relevant. Some features of HSS as of September 2020 are listed below, see [7] for details.

- HSS contains a portfolio of hybrid solvers that combine quantum and classical solution methods to leverage the best features of each computing paradigm. Its easy-to-use interface is designed to lower barriers to usability of D-Wave quantum computers.
- Hybrid computation can sometimes exhibit *quantum acceleration*, whereby queries to the quantum processor are used to guide a classical heuristic towards promising regions of the search space. This allows the heuristic to find better solutions faster than would otherwise be possible.
- The newest solvers in HSS can read much larger inputs than those in the original version, as illustrated in Figure 5. Inputs are represented as graphs containing  $n$  nodes (equal to the number of variables), and some number of  $m$  edges. The original solvers could read inputs with up to 10,000 variables, as shown by the teal region. Solvers in the new version of HSS (orange region) can read inputs with up to one million variables (or 20,000 variables if the graph is fully connected, marked by the blue dotted line). The total number of nonzero weights on nodes and edges is at most 200 million.



**Figure 5:** Input graph sizes for hybrid solvers, plotted as the number of edges  $M$  versus the number of nodes  $N$  in the input graph.

- A comparison to state-of-the-art solvers in a public repository shows that the hybrid solvers in HSS can significantly outperform classical approaches. In one case study the newest solver found solutions of better or equal quality than the best of 27 classical solvers, in 84 percent of inputs tested.

## 3.1 Applications

To date, over 250 application problems have been demonstrated to run on D-Wave quantum systems as well as hybrid solvers; Table 1 highlights about 50 examples from those appearing in scores of proof-of-concept papers and presentations at D-Wave user group meetings; see [5] for details.

Table 1 also illustrates the wide variety of use cases that are compatible with this approach:

- Some cases (e.g., factoring and 3-Satisfiability) require solutions that are optimal.
- Some (e.g., traffic flow optimization and tsunami evacuation routing) require good-quality solutions in short time frames (not necessarily optimal).
- Some (e.g., portfolio selection) require diverse samples of good-quality solutions.
- Applications in machine learning (marked ML in the table) require samples drawn from a Boltzmann distribution on the solution set.
- Applications in quantum simulation (e.g., phase transitions in quantum systems) require distributions sampled from entangled states mid-anneal.

The availability of the Advantage QPU and HSS has stimulated interest in quantum annealing and hybrid approaches to solving challenging optimization problems. We look forward

Applications for Annealing-based Quantum Computers	
3-Satisfiability	3D image tomography
Bayesian inference in imaging	Binary matrix factorization
Budget pacing in auctions	Calibrating transmissions
Capacitated vehicle routing	Chemical structure analysis
Designing metamaterials	Display advertisement optimization
Donor-patient matching	Election modeling
Factoring	Factory vehicle scheduling
Fault diagnosis	Fault tree analysis
Financial stress analysis	Flight gate assignment
IMRT beamlet optimization	Job-shop scheduling
Linear least squares	List order optimization
Detecting LHC particle collisions	MIMO processing in radio networks
Minimizing polynomials	Modeling molecular dynamics
Model predictive control	Modeling terrorist networks
Number partitioning	Peptide design
Phylogenetics	Portfolio optimization
Satellite scheduling	Simulating KT phase transitions
Simulating material structures	Simulating atomic magnetometers
Simulating quantum lattice transitions	Stock market forecasting
Telecommunications network design	Topological data analysis
Traffic flow optimization	Traffic signal optimization
Tsunami evacuation routing	Waste collection optimization
ML: accelerating deep learning	ML: classification in cancer research
ML: classification in DNA analysis	ML: quantum boosting
ML: training neural networks	ML: reinforcement learning
ML: unsupervised learning	

**Table 1:** Some of the 250+ real-world applications that have been successfully implemented to run on D-Wave QPUs and on quantum annealing-based hybrid systems.

to hearing more about an ever-widening scope of application problems that are amenable to the quantum approach.

## 3.2 Performance

While the expanding list of early-phase applications for QA is encouraging, its existence alone does not answer the most important question: can annealing-based quantum processors outperform classical computers at solving NP-hard problems? Progress on this question is very briefly summarized here.

We first note that research communities in different areas of quantum science, and in academia versus industry, have different expectations about what constitutes a demonstration of superior quantum performance. Our status report addresses the interests of these communities in turn.

### Of interest to academia

- Do quantum algorithms offer an asymptotic speedup (preferably exponential) over classical algorithms? Most complexity-theory versions of the performance question remain open. We note that performance results in complexity theory tend to have little relevance to practice, since they assume noise-free computation and worst-case bounds. Conversely, empirical results about performance of real-world quantum systems cannot be used to settle hypotheses from complexity theory, since the former are necessarily finite, and the latter contain universal quantifiers over infinite sets.
- A physics-based approach to performance looks for a specific property known as *quantum speedup*, by comparing the scaling curves of classical and quantum algorithms under best-case (optimally tuned) conditions [15]. An observation of “limited quantum speedup” has been reported [16] but the general question remains open. Note that quantum speedup results also tend to have little relevance to practice, since the synthetic inputs do not resemble real-world inputs and the scaling analysis ignores constant-factor speedups, sometimes favoring the QPU by many orders of magnitude [17], that are of interest to the practitioner. Conversely, real world inputs typically do not lend themselves to scaling analysis, which requires close control of problem size.

**Performance in applications practice** Answering the performance question from the practitioner’s point of view requires identifying realistic input sets, reasonable alternative solution methods, and application-relevant performance metrics. This work can involve the QPU used alone or in some hybrid framework.

The list below describes five promising reports of superior performance found using D-Wave’s annealing-based quantum computers: the first two involve hybrid use of the Advantage system, the second two use hybrid solvers with the D-Wave 2000Q, and the last considers a D-Wave 2000Q QPU outside the hybrid framework.

- Researchers at Menten AI describe a project to leverage quantum annealing to tackle a problem in computational protein design. Using a hybrid solver with a D-Wave Ad-

vantage QPU, they found better solutions faster than the industry-standard method. This led to the development of the world's first quantum-designed molecule, which was subsequently studied in wet-lab experiments. They plan to continue using hybrid methods to reduce the cost and time required to engineer proteins for drug design. [18].

- Save-On-Foods, a western Canadian grocery retailer, has announced the successful use of a hybrid solver incorporating the Advantage QPU to find solutions to optimization problems in grocery logistics. The company reports that they were able to reduce the time needed for one optimization task from 25 hours to just 2 minutes per week [19].
- Mugel et al. [20] considered the problem of dynamic portfolio optimization, i.e. determining the optimal trading trajectory for an investment portfolio over a period of time. They compared a `dwave.hybrid` solver<sup>3</sup> (DWH), based on the D-Wave 2000Q, to three classical solvers and two alternative quantum approaches. Only DWH and a classical Tensor Networks (TN) solver could handle the largest inputs in this test: on those inputs the DWH found better solutions and ran 100 times faster than TN.
- Nishimura et al. [21] reported that a D-Wave 2000Q QPU used together with `qbsolv` (an open source hybrid tool [22]) found better solutions faster than their industry-standard approach, when applied to a problem that involved finding optimal listing orders for online search engine results.
- Inoue et al. [23] developed a method for direct (nonhybrid) use of a D-Wave 2000Q system in an application involving global optimization of a grid of traffic signals to minimize imbalance of traffic flows. They show that global control delivers better results than standard local control methods, and that the solutions obtained by the 2000Q are better than those obtained by a well known classical approach.

While these results are encouraging, they must be tempered by the knowledge that superior classical methods for solving any particular problems may possibly exist. On the other hand, industrial users of QC technology tend to prefer having the means in hand to quickly solve the problem, as opposed to spending time and money searching for better classical algorithms that may or may not be found.

Overall, we might roughly summarize these and similar reports as providing encouraging evidence that annealing-based quantum computers — especially when incorporated into hybrid frameworks — are making good progress on the road to demonstrating widespread quantum advantage on real-world application problems and use cases.

### 3.3 Summary

Our survey of the research literature and reports of users' experience gives rise to the following general observations about performance of D-Wave quantum processors.

---

<sup>3</sup>Note that `dwave.hybrid` is different from the HSS: the former is an open source toolset that provides support for users to develop their own hybrid solution methods, while the latter is an online service that finds good-quality solutions to application inputs submitted by servers.

**QPUs exhibit fast convergence to good solutions.** A D-Wave QPU routinely returns near-optimal solutions within a few anneals, sometimes much faster than known classical competition, but then may require a significantly larger number of anneals to find at least one ground state. The reason for this phenomenon may be a combination of physical limitations of the system and finite precision representation of the IM parameters on-chip. Ongoing research programs at D-Wave are focused on removing or reducing these limitations in future quantum technologies.

**Size matters.** If an input is too small or too easy, a classical solver using nanosecond-scale instruction sets can find solutions in times well below 10 ms, the time needed to set up the problem on a current-generation QPU. On the other hand, if the input is too large to fit on the QPU, it must be decomposed or distilled, as described earlier. Doing so adds classical overhead time and may impact solution quality, and can therefore only be effective if raw performance on QPU-sized problems is differentiated enough to pay for that overhead. This suggests the existence of a sweet spot with respect to input size: An ideal problem must be small enough such that substantial portions of it fit on current hardware but also big enough (and hard enough) that it cannot be solved quickly by purely classical means.

Tests of smaller previous-generation QPUs on real-world application problems revealed very few cases that meet both criteria; however, as mentioned above, there is evidence that 2000-qubit systems are large enough to break even with or slightly outperform classical counterparts on problems involving finding near-optimal solutions. We are optimistic that wider user experience with the Advantage system will give rise to many more such reports.

**Favorable phase transitions lead to better performance.** As a necessary condition for an annealing-based quantum computer to perform well, the input should have features that elicit the special capabilities of the QPU that are not available to classical solvers. This observation has implications for identifying input sets that are most suitable for demonstrations of quantum speedup, as well as for demonstrating quantum performance advantage on applications. Much work is needed to identify large application inputs that exhibit such features, and to understand how to tune system parameters for best performance. Of course, one prerequisite for this work is having QPUs large enough to study inputs of suitable size.

**Infrastructure is key.** As discussed above, to be considered useful for practical applications, an annealing-based quantum computer must run within a classical support framework that provides tools for formulating and transforming inputs and postprocessing outputs. Recent developments suggest that hybrid tools that can cope with larger-than-chip inputs are critically important as well. Tools for performing these tasks and more are available in D-Wave software libraries and GitHub repository, but there is much room for improvement, both in terms of performance and of application scope.

## 4 The future

The current-generation Advantage quantum systems manufactured by D-Wave are but a step on the path to realization of industrially-relevant quantum computing, and beyond that, to implementation of annealing-based quantum computers that are fully general in their computational power (known as universal quantum computers). While the amazing innovations mentioned in the beginning of this paper may be considered strong motivation



for making the attempt in the first place, we do not know when or even whether they might someday be realized.

However, based on ongoing empirical work and our current understanding of past and current D-Wave systems, describe some improvements to next-generation QA technologies that we are confident will come to pass within the next two-to-five years.

**Next-generation architectures will solve much larger problems.** D-Wave will continue to develop new quantum architectures that contain more qubits and more couplers per qubit, together with the infrastructure needed to make best use of their unique capabilities. We believe that expanding use of the Advantage system with over 5000 qubits will yield demonstrations of differentiated performance at solving an even wider range of industry-relevant problems, and that this trend will continue moving forward.

**Error and noise suppression will keep pace with increasing qubit counts.** We are not aware of any fundamental impediments to building larger QA-based QPUs in the long-term future. However, increasing processor size and complexity is only worth doing if control precision and noise suppression can also be scaled up. D-Wave has made a long-term commitment to a continuously running materials science program to reduce noise that is closely linked to our mainstream QPU fabrication program. Every successive generation of D-Wave QPUs to date has made use of new technologies that improve control precision, and work is on track to do so again with future QPUs. We believe that these innovations will further broaden the set of use-cases where demonstration of a quantum performance advantage is possible.

**New QMI parameters will lead to better performance and wider applicability.** Much of the recent excitement surrounding D-Wave QPUs has been related to their use in the context of quantum simulation [24–26]. When used in this mode, the task for the QPU is not to solve a classical problem, rather it is to expose the physics of a given quantum system via measurement of qubit states at intermediate points during the anneal. Expanding this capability required development of new features and new QMI parameters.

In the longer term, D-Wave will continue to research means to build additional controls into the QPU that will enhance and expand upon current capabilities. We anticipate developing a better understanding of how to use these new capabilities by making them available to a diverse and rapidly-growing user community. We believe that these features will enable better ways to use the QPU to solve industrially relevant problems as well as performing quantum simulation experiments of interest to science.

**Software infrastructure will continue to improve.** Discovery of better translation and embedding methods, introduction of new and better postprocessing utilities, and automatic choice of QMI parameters, will be effective at changing that tipping point from “competes with” to “outperforms” classical alternatives for an ever-expanding range of application problems. Work on developing better classical support tools surrounding the QMI is ongoing, and better tools will continue to be made available.

While many real-world use cases will be able to find quantum advantage with loose integration of classical and quantum processing components (i.e. allowing interactions across a network), some applications will need tighter integration. In particular, effective hybrid solvers will require low-latency couplings between the classical and quantum portions, leading to geographic co-location of classical and quantum processing components, with appropriate security and scheduling mechanisms in place.

D-Wave is working on developing new techniques to support tighter integration of classical/quantum processors, which should yield improvements in overall workflow efficiency.

**Efficient hybrid methods will outperform purely classical or quantum methods in a growing number of cases.** We have found that many industrially-relevant problems too big to fit on near-term QPUs, and thus see an urgent need for development of hybrid solvers that use quantum queries to guide classical searches for good-quality solutions to general optimization problems, as discussed in Section 3.

To this end, open source tools to support users in developing their own hybrid solutions have been made part of the Ocean developers' toolkit [22]. As well, the D-Wave hybrid solver service (HSS) is available to provide users with immediate solutions to bigger-than-chip optimization problems.

Finding new and better hybrid methods is the subject of much current research both internally at D-Wave and by the growing user community, and we anticipate rapid improvements in near-term time frames. Together with improvements in the size and performance of the QPU, increased flexibility of the QMI, and better infrastructure support, we anticipate an upsurge in the development of efficient and effective hybrid algorithms that leverage the best features of quantum and classical computation.

Section 3 describes a few cases where hybrid solvers incorporating D-Wave QPUs have already demonstrated superior performance to industry-standard alternatives. We believe that the hybrid approach shows promise to deliver differentiated performance on ever-widening sets of problem domains and use cases, as D-Wave quantum computers, software infrastructure, and familiarity with their unique capabilities continue to grow and improve.

## 5 Concluding remarks

Of all the quantum computing platforms currently under development, annealing-based quantum computing offers the most viable way forward to connect quantum hardware to real-world applications. Providing annealing-based QPUs as part of a complete computing platform within a declarative paradigm has made this nascent technology accessible for a large number of users who, in turn, are helping to drive this approach forward. The development of a QA user ecosystem can be directly attributed to D-Wave's strategic decision to develop a technological model that puts applications first, and that is capable of solving real-world problems now.

The D-Wave technology development program works in a tight feedback loop with practitioners and users, which provides a foundation upon which to build next-generation systems. Because of this approach, we see a clear path forward to improving performance, which we believe will allow us to demonstrate a wider scope of industrially-relevant performance advantages over classical computing within the next few years.

## References

- <sup>1</sup> A. Ambainis, "What can we do with a quantum computer?" Institute for Advanced Studies Ideas, <http://www.ias.edu/ideas/2014/ambainis-quantum-computing> (accessed November 2014).

- <sup>2</sup> W. Knight, "Serious quantum computers are finally here. What are we going to do with them?" MIT Technology Review (2018).
- <sup>3</sup> R. Beall and M. Reynolds, "What are quantum computers and how do they work?" WIRED Explains, Wired Magazine (2018).
- <sup>4</sup> D. Aharonov, "Adiabatic quantum computation is equivalent to standard computation," *SIAM JCOMP* **V37**, 166–194 (2007).
- <sup>5</sup> Presentations at current and past QUBITS user-group meetings may be found at [qubits.com](http://qubits.com). A searchable repository of applications that have run on D-Wave processors may be found at [dwavesys.com/applications](http://dwavesys.com/applications) (2020).
- <sup>6</sup> C. McGeoch and P. Farré, *The D-Wave Advantage System: An Overview*, <http://dwavesys.com/resources/publications> (D-Wave Technical Report 14-1049A-A, 2020).
- <sup>7</sup> C. McGeoch and P. Farré, *Hybrid Solver Service+Advantage: Technology Update*, <http://dwavesys.com/resources/publications> (D-Wave Technical Report 14-1048A-A, 2020).
- <sup>8</sup> "D-Wave Technology Overview," <https://www.dwavesys.com/quantum-computing> (2020).
- <sup>9</sup> "D-Wave's Ocean Software," <http://ocean.dwavesys.com> (2020).
- <sup>10</sup> "D-Wave Systems: Leap 2 Web Portal," <http://cloud.dwavesys.com/leap> (2020).
- <sup>11</sup> S. Tanaka, R. Tamura, and B. Chakrabarti, *Quantum Spin Glasses, Annealing, and Computation* (Cambridge University Press, 2017).
- <sup>12</sup> "D-Wave Ocean Software Documentation," <http://docs.ocean.dwavesys.com/en/latest/index.html> (2020).
- <sup>13</sup> S. Pakin, "Performing fully parallel constraint logic programming on a quantum annealer," *Theory and Practice of Logic Programming: Special Issue 5-6* **18**, arXiv:1904.00036, 928–949 (2018).
- <sup>14</sup> K. Boothby et al., *Next-Generation Topology of D-Wave Quantum Processors*, [dwavesys.com/resources/publications](http://dwavesys.com/resources/publications) (D-Wave Technical Report 14-1026A-C, 2019).
- <sup>15</sup> T. Roennow et al., "Defining and detecting quantum speedup," *Science* **345**, 6195, 420–424 (2014).
- <sup>16</sup> T. Albash and D. A. Lidar, "Demonstration of a scaling advantage for a quantum annealer over simulated annealing," *Phys. Rev. X* **8**, 031016 (2018).
- <sup>17</sup> V. Denchev, "What is the computational value of finite range tunneling?" *Physical Review X* **6**, 031015 (2016).
- <sup>18</sup> "How Menten AI is Reimagining Biology with Quantum-Powered Protein Design," <http://dwavesys.com/application> search filter Drug & Protein Design (to appear), also see <http://menten.ai/publications> (2020).
- <sup>19</sup> "Save-on-Foods press release," (link to appear) (2020).
- <sup>20</sup> S. Mugel et al., "Dynamic portfolio optimization with real datasets using quantum processors and quantum-inspired tensor networks," arXiv 2007.00017 (2020).
- <sup>21</sup> N. Nishimura et al., "Item listing optimization for E-commerce websites based on diversity," *Front. Comput. Sci.*, (also see <http://dwavesys.com/application> search filter E-Commerce) (2019).
- <sup>22</sup> "D-wave systems dwave-hybrid," <http://github.com/dwavesystems/dwave-hybrid> (2020).
- <sup>23</sup> D. Inoue et al., "Traffic signal optimization on a square lattice using the D-Wave quantum annealer," arXiv: 2003.07527 [math.OC], (also see <http://dwavesys.com/applications> search filter Transportation) (2019).
- <sup>24</sup> R. Harris, "Phase transitions in a programmable quantum spin glass simulator," *Science* **316**, 162–165 (2018).
- <sup>25</sup> A. King et al., "Observation of topological phenomenon in a programmable lattice of 1,800 qubits," *Nature* **560**, 456–460 (2018).
- <sup>26</sup> P. Kairys et al., "Simulating the Shastry-Sutherland Ising model using quantum annealing," arXiv:2003.01019 [quant-ph] (2020).