Perspective

Grand Challenge of Quantum Applications

Ryan Babbush,* Robbie King,† Sergio Boixo, William Huggins, Tanuj Khattar, Guang Hao Low, Jarrod R. McClean, Thomas O'Brien, and Nicholas C. Rubin
Google Quantum AI, Santa Barbara, California 93111, USA

(Received 12 November 2025; revised 13 March 2026; published 15 June 2026)

This perspective outlines promising pathways and critical obstacles on the road to developing useful quantum computing applications, drawing on insights from the Google Quantum AI team. We propose a five-stage framework for this process, spanning from theoretical explorations of quantum advantage to the practicalities of compilation and resource estimation. For each stage, we discuss key trends, milestones, and inherent scientific and sociological impediments. We argue that two central stages—identifying concrete problem instances expected to exhibit quantum advantage, and connecting such problems to real-world use cases—represent essential and currently under-resourced challenges. Throughout, we touch upon related topics, including the promise of generative artificial intelligence for aspects of this research, criteria for compelling demonstrations of quantum advantage, and the future of compilation as we enter the era of early fault-tolerant quantum computing.

DOI: 10.1103/6r9l-lynr

I. INTRODUCTION

Quantum computing hardware is advancing at a remarkable rate. The theoretical basis of quantum error-correction (QEC) is solid, and several platforms are now below the error-correction threshold [1–3]. Experimentalists believe that today's technology can scale to at least hundreds of logical qubits, and while significant challenges remain, there is growing confidence that there are no fundamentally insurmountable obstacles on the path to a large fault-tolerant quantum computer.

Justifying and sustaining the investment in research, development, and infrastructure for large-scale, error-corrected quantum computing hinges on the community's ability to provide clear evidence of its future value through concrete applications. Further, in order to maintain technological momentum, it is critical to match investment growth and hardware progress with algorithmic capabilities. The time to discover quantum algorithms is now, and this provides an exciting opportunity where theory research can have enormous leverage and impact.

This perspective outlines the wide-ranging research required to create practical quantum computer applications. We divide that quest into five stages:

(1) Discovery of new quantum algorithms in an abstract setting.
(2) Identification of problem instances exhibiting quantum advantage.
(3) Establishing quantum advantage for a real-world application.
(4) Optimization, compilation, and resource estimation for a use case.
(5) Application deployment.

Along the way, we identify several key features of quantum algorithms research that we believe are crucial for delivering a useful real-world application. Our perspective focuses on the development of quantum applications realizing computational speedups, particularly those designed for fault-tolerant quantum computers. We set aside other important areas of quantum advantage—e.g., in sensing, metrology, or communication—to concentrate on practical computational applications. (For a recent review on categories of quantum advantage, seeRef. [4].)

In the following subsections, we briefly introduce several prominent theses of our perspective, which will be discussed in more depth throughout the article. We then discuss the five stages categorizing the maturity of a quantum application, key challenges for each stage, and how we can make progress. We include standalone sections on the importance of verifiability, on quantum simulation, compilation in the early-fault-tolerant regime, and the economics of quantum application discovery.

A. Focus on verifiability

It is important to distinguish between quantum algorithms that could one day provide the basis for a practically

*Contact author: babbush@google.com
†Contact author: robbieking@google.com

relevant computation, and those that will not. We argue that in the real world, computations are not useful unless they are verifiable in the sense that the quality of the solution can be efficiently checked or measured. If we can efficiently spoof the output of a quantum computation with no efficiently detectable change in performance, then such spoofing would be easier than building and running a quantum computer. If spoofing the output affects the performance in an efficiently detectable way, then we should be able to use that to verify the computation.

We believe that verifiability is a necessary (but insufficient) condition for the utility of quantum algorithms primitives. This rules out quantum advantages based on sampling from a scrambled quantum state [5,6]. While such tasks represent interesting demonstrations of the power of quantum computing and lower-bound the resources required for quantum advantage in more useful contexts, if one cannot efficiently measure how well they are solving the problem at hand, the real-world impact of that solution would also be inefficient to measure.

Efficient classical verification is the highest standard. However, we can set a lower bar for assessing the potential utility of quantum algorithms in quantum simulation: the output of a useful quantum algorithm should at least be verifiable by another quantum computer. This allows for cross-verification between quantum devices or for verifying the computation against nature itself. For instance, consider a quantum simulation algorithm that computes a physical observable. If two different quantum computers could run the simulation and get the same answer, one can at least regard this as an experimentally testable prediction about the world. This cannot be said for some sampling tasks such as random circuit sampling (RCS). We note that Refs. [7,8] have also recently called attention to the importance of verification.

B. Focus on finding hard instances and ensembles

Many research papers in quantum algorithms fail the following litmus test: if given a quantum computer tomorrow, could the quantum algorithm be implemented and realize a quantum advantage? In order to do this, one needs not only a quantum algorithm but also a way to sample a quantumly easy yet classically hard instance. Many quantum algorithms come with exponential speedups in query models or worst-case speedup guarantees such as BQP-Completeness. However, most papers lack a method to actually generate instances with quantum advantage in a computational model aside from embedding other, already known-to-be-hard problems with an advantage (e.g., factoring).

In quantum simulation there appear to be a wealth of systems that folklore holds to be intractable for classical computers yet tractable for quantum computers. However, the community does not have many candidates for verifiable quantum simulation problems with an asymptotic quantum speedup that is convincingly established in the average case, i.e., for typical problem instances drawn at random. We hope that this presents an exciting new direction for quantum algorithms researchers, one which is impactful and promising yet relatively underexplored. We note that other recent perspectives have also called attention to this specific challenge [4,7,9].

The most important reason to seek knowledge of instances with a quantum speedup is because an understanding of the type of structure that leads to advantage can help to find real-world problems where the advantage persists. Such problems are also "shovel-ready" for hardware demonstrations of quantum advantage. Finally, we discuss that if the problem has the property that a classical machine learning method would struggle to generalize from training on a small number of quantumly computed solutions, this would prevent the eventual obsolescence of quantum computing in the context of that problem.

It should be noted that a core tenet of our approach, shared by many others in the field, is that achieving practical quantum advantage in computational models in the foreseeable future (i.e., anytime in the next two decades) likely requires super-quadratic speedups [10,11]. While many problems are amenable to quadratic speedups using standard techniques like amplitude amplification, discovering algorithms with higher-order speedups is considerably more difficult. Henceforth, when we refer to a computation as having quantum "advantage," we mean a computation run at such a scale that it actually requires fewer computational resources (e.g., time) than classical competition. By contrast, when we write quantum "speedup," we are referring to the algorithm scaling. Finally, when we refer to a computation as having "utility," we mean that it is useful beyond providing a demonstration of quantum computers.

C. Focus on connecting to real-world applications

Even a quantum algorithm that is efficient for explicit classically hard instances is not yet a quantum application. Perhaps the most significant bottleneck in delivering value is the translation of abstract algorithms into solutions for real-world problems where a practical advantage holds under all physical and economic constraints. Outside of quantum simulation and cryptanalysis, this connection has proven difficult to demonstrate. This "application search" is often undervalued in the academic community compared to the discovery of new abstract algorithms, creating an imbalance: the field has a growing portfolio of theoretical tools but a scarcity of demonstrated, practical use cases.

This challenge stems from two primary difficulties. The first is technical: most algorithms with super-quadratic speedups come with a long list of stringent criteria—such

as requirements on data structure, matrix sparsity, or condition number—that are rarely met in problems of commercial interest. The second is sociological: a persistent knowledge gap exists between quantum algorithmists and real-world domain specialists. Finding a new application requires a rare, cross-disciplinary skill set to bridge this divide. We argue that a "problem-first" approach, where one starts with an industrial challenge and tries to invent a quantum solution, though attempting to more directly address an end-user's needs, is rarely fruitful. A more effective strategy is the "algorithm-first" approach: begin with a known quantum primitive (e.g., quantum simulation) that offers a clear advantage, and then search for real-world problems that map onto the required mathematical structure. We posit that generative artificial intelligence might be a useful tool for helping to bridge the knowledge gap between disparate fields.

D. Focus on architecture in early fault-tolerance

An important final step in developing quantum applications is the compilation of logical circuits and the optimization of the resources required for realization within error-correcting codes. Much has been written lately about the dawn of the "early fault-tolerant" era and an emerging important direction in the field will be to design algorithms for this era. However, in our section on early fault-tolerant compilation, we caution that simple rules of thumb such as "focus on reducing circuit depth and the number of logical qubits" are often insufficient (and sometimes actively harmful) for guiding the search for early fault-tolerant algorithms.

Just as success in the era of noisy intermediate-scale quantum devices required careful consideration of the details of the target hardware platform, we argue that success in the early fault-tolerant era demands careful consideration and perhaps even codesign of the quantum error-correction architecture. While it is not practical for all algorithmists to program at the level of stabilizer diagrams, we argue that algorithmists familiar with intermediate representations and cost models will provide unique capabilities. However, these models depend sensitively on certain assumptions of the underlying architecture that still need to be developed and optimized.

II. THE FIVE STAGES OF QUANTUM APPLICATIONS RESEARCH

We categorize the development of quantum applications into five stages, as depicted in Fig. 1. In this section, we provide a high-level summary of these stages. The remainder of the article will then offer a more detailed discussion of the challenges inherent in each.

Stage 0: "Fundamental Stage"—Fundamental research in quantum information science. The Fundamental Stage encompasses the foundational work in quantum information and complexity theory that guides the search for quantum speedups. This research often establishes fundamental limits on the power of quantum computation. For example, classical simulation algorithms [12], quantum query lower bounds [13], and no-go theorems (e.g., the "no fast-forward theorem" [14,15]) can prove that no quantum speedup is possible for certain problems, steering researchers away from dead ends. Likewise, there are many examples of quantum complexity theory teaching us that certain problems (e.g., preparing sufficiently low temperature Gibbs states of many Hamiltonians [16,17]) are sufficiently difficult in the worst case (e.g., QMA-Hard [18]) that we should not expect efficient quantum algorithms that work for all instances. Such results help quantum algorithmists focus their attention and effort on more realistic goals.

Fundamental Stage research is typically abstract, curiosity-driven, and conducted primarily within academia rather than industrial groups. While critically important,

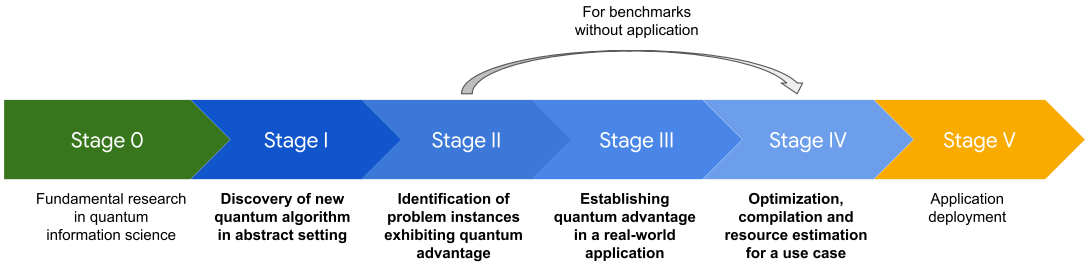

FIG. 1. The stages of quantum application development. This framework is a conceptual model of application research maturity and progression is not always linear. Some seminal works, like Shor's factoring algorithm, addressed multiple stages at once (I, II, and III), while other research paths may skip stages entirely. For example, research into "tests of quantumness" often focuses on minimizing resource requirement tasks with quantum advantage irrespective of practical applications. Ultimately, this framework serves to categorize research and identify the key bottlenecks on the path to realizing practical quantum applications.

since the Fundamental Stage is not algorithm or application development *per se*, we will not discuss it further in this perspective.

Stage I: "Algorithms Stage"—Discovery of new quantum algorithm in abstract setting. Stage I involves the discovery, design, and analysis of new quantum algorithms. In the Algorithms Stage, the goal is often to introduce a new quantum computational primitive and to give some sort of evidence for how it performs. Work at this stage often focuses on worst-case asymptotic complexity, concerned with the scaling of the algorithm for the most difficult possible instance. Algorithms that invoke blackbox oracles to solve problems in simplified computational models such as the query model would be in scope for the Algorithms Stage. This stage includes many foundational results of the quantum algorithms field. Canonical examples include Grover's algorithm [19], Simon's algorithm [20], quantum phase estimation [21], and quantum algorithms for the Abelian hidden subgroup [21]. It also encompasses broad frameworks like the quantum adiabatic algorithm [22], the quantum approximate optimization algorithm (QAOA) [23], and quantum signal processing [24,25], as well as algorithms for BQP-Complete problems such as the solving systems of linear equations [26] or estimating the Jones polynomial [27].

Stage II: "Instances Stage"—Identification of problem instances exhibiting quantum advantage. An Algorithms Stage result advances to the Instances Stage when one is able to efficiently generate concrete problem instances (even if contrived) where a quantum computer is expected to outperform the best-known classical algorithms. For example, the Instances Stage can be accomplished when a problem is shown to have an average-case super-quadratic speedup across an efficiently constructible ensemble. This is a crucial step toward practical utility. Results only demonstrating worst-case speedup are not yet in the Instances Stage, since one might not know how to generate classically difficult instances except by embedding other problems that were already known to have a quantum speedup (e.g., factoring).

Sampling problems with large quantum speedups seem to be abundant [5,28]. However, we will argue that Instances Stage algorithms with potential for utility should be either classically or quantumly verifiable and it has been considerably more difficult to advance problems with that property to this stage. Examples of quantum algorithms papers that have addressed the Instances Stage for decision problems include quantum algorithms for Gauss sums [29], tensor principal component analysis [30], the optimal polynomial intersection (OPI) problem [27], and the simulation of higher-order Out-of-Time-Order Correlators (OTOCs) on random circuits [31]. (See Ref. [32] for a formal definition of the OTOC problem.)

Instances Stage results must confront two related challenges. The first is to identify a problem instance such that the output of the quantum algorithm avoids concentration on an average input-independent value, and therefore reveals an input-dependent signal. The second challenge is the need to consider a vast array of (possibly heuristic) classical algorithms that could compete with the quantum algorithm. Beyond being a necessary step toward applications, our belief is that the Instances Stage is as much the heart of the quest for quantum advantage as the Algorithms Stage.

Stage III: "Real-World Stage"—Establishing quantum advantage in a real-world application. The Real-World Stage is the critical step of connecting an abstract quantum advantage from the Instances Stage to a specific, real-world application. A result reaches this stage when it identifies a practical problem where a quantum algorithm is expected to deliver a meaningful speedup, after accounting for all real-world constraints and overheads. This stage is challenging because the "fine print" of many quantum algorithms often prevents a direct mapping to practical problems [33].

The most famous Real-World Stage success is Shor's algorithm [34], which had the immediate application of breaking widely used cryptosystems [35,36]. Since then, quantum simulation has seen the most progress. Our team has developed Real-World Stage case studies detailing how quantum simulations could be applied to challenges in fusion energy, battery design, and drug discovery [37–42]. Outside of quantum simulation and cryptography, directly linking novel algorithms to high-value practical problems in a manner that retains a quantum speedup remains a major outstanding goal for the field.

Stage IV: "Compilation Stage"—Optimization, compilation, and resource estimation for a use case. The Compilation Stage shifts the focus from theoretical speedups to the practical engineering challenges of optimization, compilation, and resource estimation for a specific use case. This marks a significant departure from earlier stages: the emphasis moves from asymptotic analysis to concrete, finite resource costs, such as the exact number of qubits and gates required. This work meticulously analyzes and optimizes the constant factors determining an algorithm's real-world viability. Canonical examples include the long series of papers that have systematically reduced the estimated resources for simulating quantum chemistry [43] or breaking Rivest–Shamir–Adleman (RSA) encryption [44]. More broadly, the Compilation Stage encompasses the development of compilers for mapping logical circuits onto error-correcting codes, as well as the creation of software tools, like Qualtran [45], Q# [46], QREF [47], Bartiq [48], Qrisp [49], and Silq [49], designed (at least in part) to automate and streamline these painstaking resource estimates.

Stage V: "Deployment Stage"—Application deployment. The Deployment Stage represents the final phase: deploying a proven quantum solution into a practical, real-world workflow. As of today, this stage is entirely prospective, as no quantum application has yet been implemented in hardware with a conclusive advantage on a problem of real-world consequence. Importantly, the motivation for deployment will be agnostic to the fact that the underlying technology is quantum, relying entirely on the fact that it is a better solution to the problem or application at hand. Perhaps the closest current example is certified random number generation [50], though its impact to date remains limited. The Deployment Stage will likely involve creating industry-specific application programming interfaces, environments, and workflows—for example, a cloud service that allows clients to seamlessly offload tasks to a quantum computer. Although this stage is certain to have significant challenges associated with it, we elect not discuss it further here since most of this work is in the future.

III. STAGE I: "ALGORITHMS STAGE"

While discovering new algorithms is challenging, the Algorithms Stage has seen significant progress in recent years, with active research in areas including quantum machine learning, Gibbs state preparation, and quantum algorithms for differential equations among other areas [16,51–61]. See Table I for a noncomprehensive but representative list of some Algorithms Stage results with super-quadratic speedups.

TABLE I. A representative (but not at all comprehensive) list of some Algorithms Stage results with super-quadratic speedups.

Type	Result
BQP-Complete	Computing knot invariants (Jones polynomials) [27,62,63]
	Solving linear systems of equations [26]
	Coupled classical oscillator dynamics [58]
	Preparing local minima of quantum Hamiltonians [17]
	Universal quantum simulation [64]
Oracular	Simon's problem [20]
	Forrelation [65]
	Yamakawa–Zhandry [56]
	Glued trees [66]
	Abelian hidden subgroup [21]
Other	Adiabatic algorithm [22,67,68]
	Early results on QAOA [23]
	Topological data analysis [61]
	Simulating differential equations [69–72]
	Generative learning of shallow quantum circuits [54,73]

However, this does not mean the Algorithms Stage is without challenge. We observe that despite a dramatic increase in the size of the quantum computing field throughout the last decade, a smaller percentage of researchers today seem to be working on certain foundational topics in algorithms than in the past. For example, the study of quantum algorithms with algebraic structure, such as for the Hidden Subgroup Problem, has become a sparsely populated subfield. This is despite a history of profound results and a sense among practitioners that more discoveries await [29,74–80]. This trend is particularly concerning in the context of postquantum cryptography. The security of such schemes can only be established by a robust community effort to find quantum attacks. Yet, only a small fraction of the few researchers with the rare dual expertise in quantum algorithms and cryptography are actively working on this. This is alarming because a critical vulnerability discovered after postquantum cryptography has been widely deployed would be catastrophic.

That a relatively small fraction of quantum algorithms researchers appear focused on foundational, high-risk research in algorithms might be explained in part by structural incentives in academia and industry that inadvertently encourage risk-averse behavior. Finding ways of encouraging more researchers to pursue bold, foundational searches for new quantum algorithms would meaningfully expand the frontier of what our field can achieve. There is a large potential upside: a short paper with a genuinely new idea can redirect years of effort. Evaluation and funding mechanisms ought to reward originality, problem-finding, and careful exploration of hard ideas (including well-documented negative results), not only incremental improvements, which would empower more researchers to aim for breakthroughs.

Our experience suggests that new quantum algorithms rarely appear from a vacuum and instead still find some inspiration from prior (often "Fundamental Stage") work. Even Shor's algorithm was inspired by the period finding of Simon's algorithm. In turn, Shor's algorithm inspired subsequent work in period finding, phase estimation, and algorithms for the Abelian hidden subgroup problem. Many recent Algorithms Stage innovations from our own team at Google have followed a pattern of generalization and modification. For instance, our work on planted inference [81] was a generalization of prior results on tensor principal component analysis (PCA) [30]. Our algorithms for simulating classical oscillators [58], and the subsequent work on Shadow Hamiltonian simulation [82] and linear matrix equations [59], were inspired by and extended previous quantum algorithms for differential equations [69–71]. Our recent work on decoded quantum interferometry (DQI) [57] bore substantial resemblance to prior work on quantum reductions [83–86] and oracle separations [56]. This process—studying a speedup or interesting quantum subroutine in a specific context and then adapting its core

mechanism to solve a different class of problems—has been a reliable engine for Algorithms Stage discovery.

A. Verifiability or reproducibility as necessary conditions for utility

How can we tell if an abstract quantum algorithm has the potential for utility in applications? This section proposes a necessary (but insufficient) condition as such a filter: useful quantum computations must be verifiable, or at least, "quantumly verifiable."

We consider an output to be verifiable if a quantum or classical computer can efficiently check if the answer is correct (without requiring quantum communication). We argue that we cannot consider a solution to be useful if there is no way to verify it in a reasonable time. If the quality of the solution makes any difference in the real world, then the real world is effectively giving feedback on the quality of the answer and this feedback could be used for verification. For example, imagine using a quantum computer to optimize a technology. If the new design it suggests performs measurably better, that real-world success would be an informal verification of the result. It follows that tasks of relevance to the classical world ought to be classically verifiable in order to be useful.

An output is considered quantumly verifiable if the output can be verified by a trusted quantum computer. For example, any computation that is independently *reproducible* will be quantumly verifiable: running the same quantum algorithm across different quantum devices yields consistent results. Here we are referring to the final answer output by the quantum algorithm (e.g., the estimate of the expectation value or the final factors rather than the exact samples or repeated runs that were required to determine that answer). This is crucial for quantum simulations because their purpose is to make predictions about nature (which can be imagined as performing a quantum computation). For example, if a quantum computer helps design a new superconductor, the simulation would be confirmed if the material is created and observed to have the predicted properties.

We stress that the notion of quantum verification can coincide with utility even if the output is never literally verified by another quantum computer. Rather, by virtue of being quantumly verifiable in principle, we can be sure that the output is a reproducible and testable prediction about the world, unlike the output of random circuit sampling. Such a prediction has the potential for utility in the real world since Nature itself is quantum.

A task like random circuit sampling fails the conditions for utility. It is believed that the output of random circuit sampling is not efficiently verifiable or reproducible, even by another quantum computer—this is a folklore result. Of course, many sampling tasks are computationally useful. For example, Monte Carlo sampling algorithms are widely used in practice. However, applications of sampling typically use samples to extract meaningful information or specific features of the underlying distribution. For example, Monte Carlo is often used to evaluate integrals. To use a quantum example, while Shor's algorithm involves sampling from a final quantum state, classical postprocessing of the samples reveals a deterministic and classically verifiable answer. Sampling from generative AI models is only useful if the samples can be evaluated under some quality metric. In contrast, samples obtained from random quantum circuits lack any efficiently discernible features. If a quantum algorithm generates samples containing meaningful signals from which one could extract classically hard-to-compute values, then the problem ultimately being solved is not sampling but a quantumly verifiable task such as expectation value estimation with sampling as a subroutine.

Quantum computers first obtained quantum advantage in sampling against the best classical algorithms in 2019 [6]. By contrast, the first claims of quantum advantage in computing an expectation value to have withstood preliminary efforts at classical simulation seem to have occurred only in the last year. Notable claims were made by IBM [87] and D-Wave [88] in 2024 but then challenged by several groups shortly thereafter using relatively standard methods [89–93]. This year, claims were made by Quantinuum [94] and Google [31] and the Quantinuum results have recently been simulated classically [95]. Thus, even though these recent Instances Stage demonstrations are not yet solving practical problems, we believe that solving expectation value problems with quantum advantage as opposed to sampling problems represents clear progress toward more useful quantum computing.

Finally, we emphasize that we consider these conditions as guiding principles for assessing the utility of quantum algorithms, rather than as an absolute criterion. Certain computations, such as certified random number generation [50], may remain useful even when verification is inefficient, although such cases generally lie outside the computational model considered in this work.

IV. STAGE II: "INSTANCES STAGE"

We consider a quantum algorithm to have reached the Instances Stage only when one can generate concrete instances of a problem that are expected to exhibit verifiable quantum advantage. For example, this would be satisfied if one identifies an ensemble of quantumly verifiable problem instances that can be efficiently sampled for which a quantum algorithm gives a super-quadratic speedup over the best known classical algorithms, with probability vanishing at most polynomially in the problem size (e.g., they are average-case hard classically). While this is a demanding criterion, several results have met it. Examples of papers that have addressed the Instances

Stage include Shor's algorithm [34], the DQI algorithm applied to OPI [57], algorithms for tensor PCA [30], and speedups for other number-theoretic problems like Pell's equation [76]. One could also consider using a quantum computer to efficiently generate or sample hard instances, although we are not aware of any examples of this kind.

It is acceptable for Instances Stage results to target abstract or even contrived problems. Trying to contrive examples with advantage can be a good first step toward later applications because if a quantum advantage cannot be demonstrated in an idealized, hand-picked setting, the prospect of discovering one in a complex real-world application (the Real-World Stage) is probably negligible. Despite its critical role as a bridge from theory to practice, the Instances Stage is frequently overlooked by the quantum community, and its difficulty is often underestimated. However, those working at the intersection of quantum computing and cryptography often do focus on Instances Stage results because cryptosystems based on computational security usually require average case notions of problem complexity.

Many important algorithms remain in the Algorithms Stage because their advantage has only been proven in abstract settings that do not allow one to generate concrete problem instances with quantum advantage. Consider results in the query model, such as Simon's algorithm [20] or the welded trees problem [66]. While they demonstrate a proven separation from classical computing, the advantage relies on a black-box oracle. Without a method to instantiate these oracles in a way that remains classically hard, they have not addressed the Instances Stage. Similarly, a paper describing an algorithm that provides only a worst-case speedup has also failed to address the Instances Stage. The initial work on QAOA [23], for instance, was a landmark Algorithms Stage achievement that showed a better worst-case approximation ratio than was known classically at the time. However, this worst-case guarantee did not allow one to sample problem instances with quantum advantage (even before better classical algorithms were introduced). In contrast, subsequent numerical studies identifying a scaling advantage for QAOA on particular problem classes [101] made a successful push into the Instances Stage.

A common misconception is that proving BQP-Completeness implies a practical promise of quantum speedup. However, we argue that on its own, BQP-Completeness is not just a hallmark of the Algorithms Stage but a suggestion that more work must be done to get to the Instances Stage. It indicates one has found a new universal language for quantum computation, rather than new problem instances with demonstrable advantage. Of course, one can always derive ensembles of BQP-Complete problems with average-case quantum speedups by reducing already-known Instances Stage+ results (e.g., factoring) into the new framework but that does not help to elucidate new, natively hard instances. The issue is that proving a problem BQP-Complete does not inherently specify a distribution of hard inputs. Furthermore, "natural" ensembles of instances of BQP-Complete problems—like sparse random matrices for the HHL algorithm [26], random braids for knot invariants [27], or the dynamics of random oscillator networks [58]—are often classically easy on average. This occurs when quantities of interest concentrate to classically predictable values, eliminating any potential for quantum advantage. Constraining problems in a way that precludes BQP-Completeness may in some cases help to avoid this concentration phenomena.

A key distinction among Instances Stage results is their long-term vulnerability to being solved by an efficient classical machine learning algorithm given a finite amount of classical data generated by the quantum computer. This allows us to separate Instances Stage results into two groups.

The first category contains problems where there are a finite number of distinct instances with a quantum advantage. This includes important simulation targets like the FeMoco molecule or the phase diagram of the Hubbard model, which have a limited number of distinct phases or parameters. Here, a quantum computer might only be needed to provide the initial data. It is conceivable that after solving enough instances, a classical model could learn the underlying patterns and generalize to the remaining problems, reducing the need for further quantum computation. For example, in electronic structure force-field calculations one can already use a few high accuracy data points calculated using expensive (classical) methods to boost the accuracy of data taken with cheaper, less accurate methods [107]. This first class of problems is related to what complexity theorists call the complexity class P/Poly: those that can be solved in polynomial time given a polynomial amount of "advice." (When this class is restricted using efficiently computable data from a polynomial number of quantum computations it is known as BPP/Samp.) Of course, one still requires a quantum computer to provide initial data, and understanding generalization, or lack thereof, remains a matter of ongoing research [108,109].

The second category consists of problems where there is an exponentially large ensemble of instances that are classically hard in the average case and the solution of any polynomial sized set of problem instances provides meaningful information about only a vanishingly small fraction of other problem instances. Such fundamentally unlearnable structure implies a lack of regularity in the solution space of the problem in mapping from inputs to outputs, and the implication for the available application space is worthy of further study. For these problems, the advantage is likely to be permanent. Even after a quantum computer solves many examples, a classical machine-learning model would still be unable to generalize from that data to efficiently solve new instances. We list all results of this type

TABLE II. The short list of Instances Stage results that are currently known to the authors of this perspective for which there is an exponential-size ensemble with an average-case super-quadratic quantum speedup. This is in contrast to problems such as FeMoco or the Hubbard model where there is effectively a finite number of distinct calculations we expect to have quantum advantage. Also worth mentioning but not included in this table is work on proofs of quantumness based on trapdoor claw-free functions where verification is only efficient using hidden side information [102–106].

Algorithms Stage inspiration	Problem	Speedup	Verifiable
Period finding [20]	Factoring and discrete log [34]	Exponential	Classically
	Principal ideal and Pell's equation [76]	Exponential	Classically
	Zeta function of algebraic curves [80]	Exponential	Quantumly
	Unit and class groups [96]	Super-polynomial	Classically
	Two-variable exponential congruences [79]	Cubic	Classically
Phase estimation [21]	Gauss sums [29]	Super-polynomial	Quantumly
	Tensor principal component analysis [30]	Quartic	Classically
	Sparse learning parity with noise [81]	Quartic	Classically
	Community detection [97]	Quartic	Classically
Abelian hidden shift [74]	Legendre symbol [77]	Exponential	Classically
	Elliptic curve isogenies [78]	Super-polynomial	Classically
Regev's reduction [84]	Optimal polynomial intersection [57,98,99]	Exponential	Classically
	Yamakawa–Zhandry (instantiated) [56,100]	Exponential	Classically
Quant. approx. opt. algo. (QAOA) [23]	QAOA for 8-SAT [101]	Super-quadratic	Classically
Random circuit sampling (RCS) [28]	RCS out-of-time-order correlators [31]	Exponential	Quantumly

known to us in Table II. The second category of problems offers a potential unending need for quantum computers and value here might be more defensible for the businesses offering a quantum computer. However, the first category still forms a valuable and important application of quantum computing.

A core difficulty of the Instances Stage arises from a fundamental tension between two competing forces. On one hand, a quantum algorithm must generate highly complex states with large entanglement to be difficult for classical computers to simulate. On the other hand, these same complex states often exhibit scrambling or thermalization, causing their observables to become statistically predictable. For a scrambled quantum state, observable values tend to concentrate tightly around their average values, erasing the unique "signal" associated with the specific problem input. A trivial classical algorithm can then compete by simply outputting this predictable average value. This tension appears to be a typical behavior of randomly sampled quantum states [110,111], and states produced by the output of random quantum circuits [112]. Similarly, simple initial states that are time-evolved under Hamiltonians of generic strongly correlated systems often quickly relax to states indistinguishable from a thermal ensemble (following the so-called eigenstate thermalization hypothesis [113]) features of these ensembles are often either predictable by cluster expansions [114,115], random matrix theory [116], or from single nonrandom quantum states [117]. (This is especially true for high temperature states [118,119], or low dimensional systems [120].) A major challenge in the Instances Stage is to find problem classes where corrections are observable (i.e., not exponentially small), and not predictable by classical algorithms.

There are various rules of thumb for what make problems in many-body physics hard; e.g., disorder, systems near phase transitions, properties at low but nonzero temperature, nonequilibrium phenomena, and glassy or topologically nontrivial phases. However, significant work is required to confirm that such properties lead to instances that do not fall prey to the aforementioned issues, yielding problems that could fall into Table II above. Such results would give a better idea of asymptotically promising targets for quantum computers within many-body physics. Moreover, this search will likely expand and better our knowledge of interesting physical phenomena: previous work from the quantum computing community on many-body scars [121], random circuits [28], and higher-order out of time-ordered correlators [31] have all opened new areas of research within the strongly correlated physics community. Understanding the boundary where many-body systems become computationally complex can be viewed as a fundamental pursuit in the study of highly entangled quantum systems.

Another major challenge of the Instances Stage is that the classical competition is the constantly improving "moving target" of practical, state-of-the-art heuristics, rather than the provable, worst-case algorithms that are often the point of comparison in the Algorithms Stage. This creates a fundamental asymmetry. Today, we typically rely on theoretical proofs or limited simulations to assess a quantum algorithm's average-case performance, while classical approaches are benchmarked by their best-observed performance in practice. History shows that the

power of many of the most impactful classical algorithms—including the Metropolis algorithm, the simplex method, and modern neural networks—were difficult to assess analytically and required large-scale empirical testing for their development [122,123]. This precedent suggests that the true potential of certain quantum heuristics may likewise remain hidden until we can explore them on scaled-up quantum hardware. Moreover, central Instances Stage questions such as the extent to which expectation values concentrate in output states would be straightforward to study using large quantum computers. Hence, while the classical moving target makes the Instances Stage difficult today, future hardware may make it easier to identify promising avenues for quantum advantage.

Even without an immediate real-world application, Instances Stage results are valuable because they provide a concrete recipe for generating hard test cases, essential for validating hardware progress and demonstrating algorithmic milestones. This is the mentality underlying research into algorithmic "tests of quantumness," which treats the Instances Stage as an end in and of itself. The goal is to set aside immediate practicality and instead answer the question: What is the simplest possible problem that demonstrates quantum advantage with the absolute minimum resources? We believe that the leading proposals to date (in the sense they solve classically intractable problems with the fewest quantum resources) are Jacobi factoring [124] for a classically verifiable problem and OTOC of random quantum circuits [31] for a quantumly verifiable problem. By seeking the cheapest possible quantum advantage, such work provides a heuristic lower bound on the resources that might be needed to give advantage in a useful application. Such benchmarks also help to distill the essence of quantum advantage in a fashion which may help to sharpen intuition about where to look for necessary structures in practical contexts.

We believe the Instances Stage in an exciting opportunity for researchers. One reason is that it falls into a disciplinary gap; e.g., perhaps theorists drawn to the abstract nature of the Algorithms Stage, view the numerical and heuristic components of the Instances Stage as "too applied" while application-focused researchers see the push for contriving hard instances as an esoteric aim. Another possible reason is that parts of the community might not appreciate the difficulty, or importance of finding instances with average-case speedups within a problem known to admit worst-case ones. For example, many physicists seem to either believe that finding Instances Stage results is trivial in quantum simulation and that is why there are not many papers about it, or believe that the problem is very hard and that is why there are not many papers about it. By highlighting how few Instances Stage results have been substantiated, we hope to encourage more researchers to fill this critical gap. The Instances Stage is not merely a prerequisite for useful applications—it represents a fundamental question at the core of understanding the complexity of quantum physics.

V. STAGE III: "REAL-WORLD STAGE"

Outside of cryptanalysis and quantum simulation, the search for practical quantum applications has proven remarkably difficult. This gap represents the core challenge of Real-World Stage research: taking an algorithm with a theoretical speedup and finding a real-world problem where that advantage holds true under all practical constraints. While a broader portfolio of algorithms from the earlier stages would certainly help, the current scarcity of Real-World Stage applications highlights that connecting theory to practice is a major bottleneck in its own right.

The two primary successes of the Real-World Stage, cryptanalysis and quantum simulation, are both exceptional cases where the path from algorithm to application was unusually direct. Quantum algorithms for factoring and discrete logarithms were immediately connected to breaking RSA and elliptic curve cryptography (ECC), respectively, and quantum simulation benefits from a natural mapping between problem and computer. (We discuss quantum simulation applications further in the subsequent section.) An important goal is to identify clear real-world applications of quantum computers outside of these two areas. Current levels of investment in quantum computing anticipate eventually unlocking a total addressable market well in excess of tens of billions of dollars per year [125] and by the time large quantum computers are built, the commercial value of breaking legacy cryptosystems might be relatively limited. Investors are already quite accustomed to hearing that quantum computers will help with simulation and cryptanalysis and thus, compelling use case stories in other areas have significantly more marginal value in their ability to inspire and incentivize investment.

A significant concern is the relative lack of academic attention devoted to the Real-World Stage compared to the earlier stages. Within the research community, Algorithms and Instances Stage work is sometimes (in our view, unfairly) perceived as more "fundamental" and thus, more important than the Real-World Stage. A paper identifying a novel real-world use case for an existing quantum algorithm, for instance for linear systems [26] or topological data analysis [61], would likely not receive the same acclaim as the discovery of a new algorithm. This cultural bias creates a critical imbalance. The field has a growing portfolio of abstract algorithms but a severe scarcity of demonstrated, practical use cases—a gap that impedes the overall health of the field. Inspiring more researchers to tackle the challenges of the Real-World Stage was a primary motivation for our team's proposal and sponsorship of the Google Quantum AI XPRIZE Quantum Applications competition [126].

We posit that there are two basic reasons that the Real-World Stage is so challenging. The first (relatively obvious) reason is that Algorithms Stage and Instances Stage successes are rare and always come with a long list of criteria that must be met in order for super-quadratic quantum advantage to be possible. For example, quantum algorithms for solving linear systems $Ax = b$ [26] generally require b to be prepared on the quantum computer with polylog(N) resources, A to be sparse and have polylog(N) condition number, and these algorithms only enable sampling the entries of x proportional to their magnitude. While the original paper on this topic (the "HHL" algorithm) has over 4k citations, we are not aware of any paper identifying a promising real-world instantiation with super-quadratic speedup, or even an ensemble of average-case hard instances placing it in the Instances Stage. Additionally, many quantum algorithms pertain to problems that look somewhat esoteric, involving specific number theoretic or algebraic structure (e.g., [29,57,76–80]), and it can be difficult to find contexts in the real world where this structure naturally arises.

A second, purely sociological challenge for the Real-World Stage is the knowledge gap between quantum algorithmists and real-world domain specialists. Finding a new application is an "application search" that requires a rare, cross-disciplinary skill set to build bridges between abstract theory and practical problems. This work demands breadth over depth, which may be one reason it is often undervalued within traditional academic structures. In quantum simulation, many quantum researchers are physicists or even chemists with pre-existing domain expertise in classical many-body simulation, which effectively closed this knowledge gap from the start. In contrast, few quantum algorithmists have knowledge of fields like epidemiology or genomics, where algorithms for problems like topological data analysis [61] might apply.

Some application discovery initiatives apply a "problem-first" strategy: ask a domain expert or customer for their hardest or most important problem, then try to design a quantum algorithm to solve it. This approach is rarely fruitful because quantum speedups tend to require exceptional mathematical structure. A more effective strategy is the "algorithm-first" approach. The search should begin with a known quantum primitive that offers an advantage and then search for real-world problems that map onto that structure. To facilitate this algorithm-first search for applications, our team and collaborators are creating a "Compendium of Super-Quadratic Quantum Advantage." This document will profile several dozen algorithms outside of simulation and cryptanalysis, each with a known super-quadratic speedup. The entries will give a concise description of the abstract problem and its "regime of advantage" in simple terms, without detailing the quantum algorithm. The compendium is designed to be readable by anyone with a technical undergraduate degree, assuming no prior knowledge of quantum computing. Our goal is to empower domain experts to identify potential applications in their own fields. If an expert finds a match between a problem in the compendium and a challenge they face, they can seek collaboration with quantum specialists to assess if the advantage translates to their use case.

We observe that this sociological knowledge gap makes Real-World Stage work particularly well-suited for acceleration by generative AI. While AI is making impressive progress in advanced reasoning, our experience is that it is not yet broadly capable of the creative leaps required for most Algorithms Stage algorithm discovery. Part of its power today lies in leveraging a vast breadth of knowledge to connect disparate fields—the exact skill needed for the application search of the Real-World Stage. The agent's task would not be to invent, but to recognize: to scan its knowledge base for real-world problems that match the structure of known quantum speedups, even if they are described using different terminology in another field's literature. Our team has already seen early success with this approach using an internal Google tool based on Gemini [127].

The path through these stages of application maturity is not always linear. Sometimes, engaging directly with a Real-World Stage application can provide the necessary guidance to solve the challenges of the Instances Stage. A truly important real-world problem, like understanding a molecule with mysterious unresolved properties, often comes with a wealth of established classical research. Such work can provide a roadmap, highlighting the specific regimes where classical methods are known to fail and where phenomena like concentration are already well-understood. In such cases, the real-world application itself provides the context needed to satisfy certain Instances Stage criteria. Thus, while Real-World Stage results should always at least build on established quantum primitives from the Algorithms Stage, it may sometimes make sense to skip past the Instances Stage to the Real-World Stage and matching to applications, if doing so will help connect to literature about where the problem is actually difficult.

While it is important to proactively push the Real-World Stage forward, we conclude by noting that history teaches us to be optimistic. Graph theory, introduced by Euler in 1736, and the Fast Fourier Transform, discovered by Gauss in 1805, did not find much use until the advent of digital computing, where they became cornerstones. Number theory, an ancient art with no application whatsoever for millennia, became the bedrock of modern cryptography. The Paris Kanellakis prize given by the ACM is essentially a prize for "Real-World Stage" results in pure mathematics, not unlike the Google Quantum AI XPRIZE Quantum Applications competition. The existence of this prize in a different field speaks to a universal challenge of bridging from theory to application.

A. Quantum simulation applications

Aside from cryptanalysis, quantum simulation is the most mature domain for Real-World Stage research, benefiting from a natural mapping between the problem and the computer. However, translating this potential into real-world impact still faces several significant hurdles. First, many powerful classical algorithms already effectively model a large number of systems that do not exhibit strong correlation with a significant correlation length [128,129]. Second, it can be difficult to establish that running a particular quantum computation (e.g., at a finite basis and computational cell size and with a certain model of the environment) will resolve the important open questions about the physics of a system of interest. Finally, a successful simulation must target the correct bottleneck in a larger technological process. In drug discovery, for instance, the primary bottleneck is not generating candidate molecules but assessing their efficacy and toxicity within the complex environment of the human body, a challenge currently solvable only by expensive, high-attrition clinical trials [130]. A quantum speedup not targeting the key bottleneck, therefore, might prove useful without being truly transformative.

A common proposal is to use quantum simulation for "discovery"; it is easier to identify a classically intractable calculation than to know what one will learn from it. This is the case in both condensed matter physics and chemistry. In condensed matter, it is relatively simple to tune a model into a strongly correlated, hard-to-simulate regime, but more difficult to know which specific calculation will reveal new physics. Similarly, in chemistry, the FeMo-cofactor of nitrogenase (FeMoco)—an enzyme involved in biological nitrogen fixation (fertilizer production)—is a famous target for quantum simulation. While there is hope that studying it will lead to better industrial processes, one cannot be certain of this without first learning what such simulations reveal.

Quantum chemistry has long been widely regarded as a leading candidate for real-world impact from quantum simulation. This is because chemistry is the central science at the heart of many industries and because *ab initio* models of electronic structure aim to produce a high level of quantitative accuracy required for engineering applications of new molecules and materials. This contrasts with condensed matter physics, where models are usually designed to capture qualitative features—such as the phases of matter—rather than make precise numerical predictions about real systems. A computational chemist seeks to answer specific, quantitative questions: "if I dope this battery cathode material with a particular atom, does its conductivity increase by 1% or 3%?" A condensed matter physicist, by contrast, typically investigates broader, qualitative trends: "what type of doping might improve conductivity?" or "is this a conductor or an insulator?"

While compelling molecular use cases like FeMoco [131] and P450 [132] exist, they are relatively rare in the context of all industrially relevant chemistry. In contrast, promising solid-state applications are far more abundant, for several reasons [38]. First, many key industrial targets such as heterogeneous catalysts involve solid-state materials, and their properties are often more directly determined by their electronic structure. That is, relative to molecular chemistry, fewer material properties depend strongly on entropic contributions that would need molecular dynamics to resolve. Second, it is easier to find classically intractable problems in solid-state systems. Many phases of matter in the solid-state have longer correlation lengths than molecules, creating complex electronic structure beyond the reach of classical simulation. This complexity, however, is a double-edged sword: the same properties that make these systems hard for classical computers also make them resource-intensive for quantum computers, demanding larger and more powerful devices [38,40].

Whether a true asymptotic, exponential quantum speedup exists for chemical ground states is a subject of legitimate debate. Some argue that "natural" molecules relevant to industry may have special properties that may make them solvable in polynomial time with advanced classical methods, like tensor networks [128,129], even if many remain out-of-reach for practical methods today. Conversely, it is plausible that strongly correlated molecular ground states are not even efficiently preparable on a quantum computer in the true asymptotic limit. Yet another perspective is that the controllable errors in quantum algorithms provide a decisive advantage over classical methods where convergence guarantees may not even be feasible [41].

However, our view is that fixating on asymptotic scaling may be the wrong lens for assessing the practical value of quantum computing in chemistry. Molecules and the correlation lengths of materials are intrinsically finite; thus, an algorithm's utility depends on its performance on these specific instances, not its behavior at an infinite limit. Furthermore, while some classical methods may have favorable polynomial scaling in theory, they can be completely impractical for high-precision chemistry in practice. For example, the projected entangled pairs state (PEPS) method (the basis for claims that classical methods may efficiently solve strongly correlated electronic structure made in Ref. [128]) has never been successfully applied to molecular electronic structure due to its prohibitively high scaling. This brings us to a more pragmatic perspective. For certain high interest systems like FeMoco, we know that despite intense effort, no classical method has solved its electronic structure to the accuracy needed for definitive chemical insight [133–135]. We also know that we can efficiently prepare quantum states with a very high overlap on its true ground state [133].

Therefore, while the theoretical debate over exponential quantum speedup remains unresolved, having a large-scale quantum computer today would allow us to learn things about the molecule that are currently beyond our reach.

Our team has had success identifying specific, impactful use cases for quantum simulation by collaborating directly with domain experts. This strategy has led to concrete Real-World Stage case studies, including partnerships with pharmaceutical company Boehringer-Ingelheim on drug discovery [132], chemical company BASF on battery development [38], and Sandia National Laboratories on simulating fusion reactors [37]. In the next section, we summarize these and other similar case studies on the resources required for industrial quantum simulation problems in Table IV.

Looking further, we expect that quantum simulation will open new fields of discovery. Our modern world is built on a century of understanding equilibrium quantum mechanics, a domain often tractable by classical methods. In contrast, classical methods struggle to treat nonequilibrium dynamics and quantum computers promise to help open this field by making such simulations routine. This capability may someday allow us to design new phases of matter, like driven superconductors, that only exist far from equilibrium [136,137]. Similarly, we know that it is possible to program chemical reactions by engineering laser pulses, but simulating such a process is intractable for all but the smallest molecules [138,139]. This is a speculative, long-term vision, and it is difficult to estimate the value of as-yet-undiscovered applications. Nonetheless, we hope that the community will eventually focus as much attention on exploring and realizing such transformative applications as it has on developing techniques suitable for equilibrium systems. Realizing this potential requires the same pragmatic, collaborative work outlined above: e.g., partnering with domain experts to identify the first concrete, impactful simulation problems in this new, unexplored territory.

VI. STAGE IV: "COMPILATION STAGE"

Once a compelling problem is identified from Instances Stage or III, the focus shifts to a natural question: What are the minimum physical resources required to achieve quantum advantage? Accurate resource estimates are essential to assessing when quantum computers may deliver a return on investment. Compilation Stage research answers this by optimizing algorithms and compiling them down to a physical hardware level to produce detailed resource estimates. This process can be understood as a compilation stack with distinct layers that we outline below.

(1) *Compilation Stage(a): Algorithmic refinement.* The high-level algorithm is refined to better suit the specific use case, often by exploiting symmetries to improve asymptotic performance.
(2) *Compilation Stage(b): Compilation to logical quantum gates.* The refined algorithm is compiled into a discrete, logical gate set. At this layer, the focus is on constant factors, but may not fully account for the overhead of routing or the production of non-Clifford resources.
(3) *Compilation Stage(c): Compilation to logical QEC primitives.* The gate-level logical circuit is mapped onto a specific architecture, such as the surface code with chip modularity. This produces an instruction set of fault-tolerant operations, such as lattice surgery primitives. At this point one can make physical resource estimates (e.g., number of physical

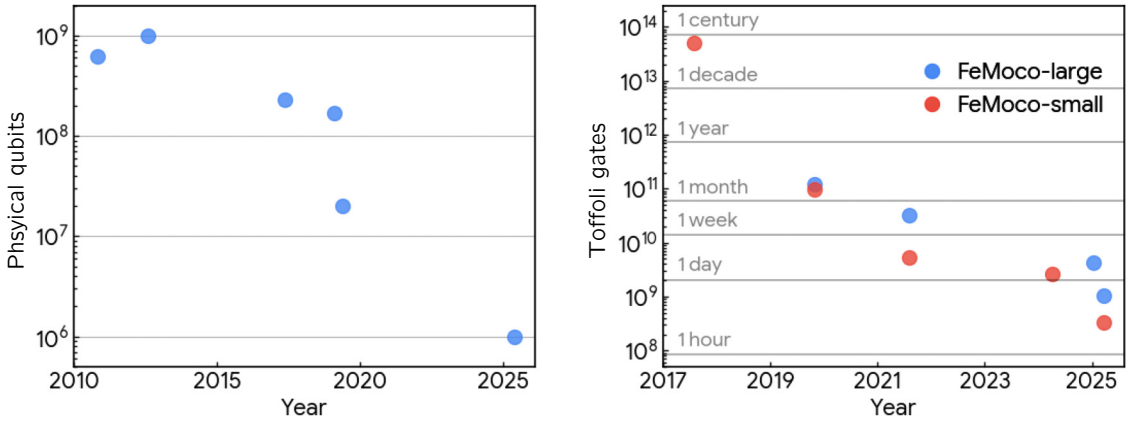

FIG. 2. These figures illustrate that Compilation Stage research has reduced the resources required to solve important problems by many orders of magnitude over the last decade as function of publication year in both space such as for breaking 2048 bit Rivest–Shamir–Adleman encryption (left), and time as approximated by the Toffoli count estimating the ground state energy of FeMoco to chemical accuracy for "small" 54-orbital [140] and "large" 76-orbital active spaces [131] (right). The left plot includes Refs. [44,141–144] and the right plot includes Refs. [41,43,140,145,146].

qubits and runtime) by using an appropriate cost model to determine code distances and cycle times.

(4) *Compilation Stage(d): Compilation to physical operations.* Final machine instructions are optimized and output by an automated compiler. This step produces exact resource counts (up to uncertainties from decoding feedback) as opposed to merely estimates. The computation is now ready to execute given sufficiently capable hardware.

Compilation Stage research is most active for the field's mature applications: quantum chemistry and cryptanalysis. Progress in these areas is evident in Fig. 2 showing the steady, year-over-year reduction in their estimated resource costs, driven by a cycle of algorithmic and compilation improvements that may dramatically shorten the expected time horizon of economic impact. Compilation Stage work sometimes commences with reasonable motivation before a real-world application is found (Real-World Stage). This is especially true for algorithms that could serve as algorithmic tests of quantumness or hardware demonstrations. However, it is premature to begin resource estimation before establishing an advantage in the first place (Instances Stage) since interesting resource estimations should be based on instances with quantum advantage.

This work often begins at Compilation Stage(a), which focuses on refining an algorithm's asymptotic scaling. Such improvements can come from integrating new techniques, like adapting block encodings to chemistry phase estimation [156]. Another common example of Compilation Stage(a) work involves adapting algorithms to more efficient representations for a target problem, such as employing a residue number system for factoring [157] or a first-quantized basis in chemistry [158–160].

To optimize resources required for an application, one must use cost models that become progressively more specific and realistic as research moves through the substages of the Compilation Stage. In Compilation Stage(b), high-level metrics like "gate count" or "gate depth" may be acceptable. A standard, more realistic step has been compiling to a minimum number of discrete logical gate set like Clifford + T, though recent advances have called into question the primacy of this cost model, diminishing the historical cost gap between T gates and Clifford gates [161].

As work advances to Compilation Stage (c), the cost models and the nature of the work itself depend critically on the specific hardware and error-correction architecture. This stage is focused on compilation to primitive fault-tolerant operations, but these operations can vary between architectures. For example, an architecture based on a 2D lattice of locally connected physical qubits relies on lattice surgery to implement multiqubit gates. Other architectures can realize long-range SWAP gates cheaply [162], implement transversal multiqubit gates [163], or even implement certain logical operators just by permuting the physical qubits [164]. The variation between these native operations will fundamentally change the targets (and the methods) of any hand or machine compilation pipeline.

Because of these differences, the work at Compilation Stage (c), and even the prospects for quantum advantage in a particular application, may be significantly platform-dependent. For instance, [148] argued that classically challenging instances of the OPI problem could be solved using approximately 6×10^6 Toffoli gates and 1900 logical qubits. Despite the low Toffoli count, the large number of logical qubits required will make it difficult to achieve quantum advantage using this approach on most small error-corrected quantum processors. However, this problem may be an early candidate for quantum advantage in photonic architectures that aim to be able to effectively trade space for time by routing qubits into long fiber-optic delay lines [162].

Once an algorithm has been compiled to a set of physically relevant abstractions, final physical qubit and runtime estimates are then obtained from high-level error-correction parameters like code distances and cycle times. These are derived from extensive numerical simulation of a physical-level error model specific to the hardware platform, and are typically out of the algorithmists' scope. Finally, in Compilation Stage (d), one uses automatic tools such as compilers to output a complete set of machine instructions for hardware execution.

Making the most effective use of quantum resources requires each stage to be performed with some understanding of cost models for later stages. For instance, it is common to compile Compilation Stage(b) to logical quantum gates with all-to-all connectivity, and apply generic qubit routing techniques or other constructive methods like the "game of surface codes" to obtain a Compilation Stage(c) compilation [165]. However, these provide resource upper bounds that are potentially quite loose. In contrast, we have found better success in approaching Compilation Stage(b) with mindfulness of gate-connectivity constraints and careful selection of non-Clifford resources [44,145]. In this regard, a deeper understanding of the compilation stack greatly benefits algorithmists seeking resource estimates representative of true hardware capabilities, especially when resources like dense quantum memories [166] and limited nonlocality are available. Currently, most Compilation Stage resource estimates are performed using a manual process for Stages IV(b) and IV(c). This can be extraordinarily tedious, difficult to verify, and imprecise. As a case in point, several researchers from the authors' team worked for well over a year on an 80-page paper [167] just to determine the constant factors in an algorithm that was introduced and described asymptotically in a preceding 7-page paper [160]. Due to the time-consuming and architecture-dependent nature of obtaining an optimized

TABLE III. Algorithms that have reached the Compilation Stage outside of quantum simulation. We summarize just one of the resource estimates from each paper. When physical resource estimates are given for multiple architectures, we focus on reporting numbers for superconducting qubits running the surface code. Still, details of the resource estimates may differ in a way that makes direct comparison challenging. For example, some resource estimates might focus on parallelization to decrease runtime at the cost of additional space. Some detailed resource estimates not included in this table because they involve only quadratic speedup are Refs. [152–155].

Horizon	Problem	Real-World Stage applications	Problem size	Compilation Stage(b)		Compilation Stage(c)	
				Logical q.	Toffolis	Phys. q.	Time
Medium	Trapdoor claw-free functions [103]	Classically verifiable quantum advantage	1024 bits	1600 [147]	8×10^6	–	–
	DQI for OPI [57]		4095 variables 70 constraints	1900	6×10^6	8×10^5 [148]	1 h
	Topological data analysis of random graphs [61]	Unknown	256 vertices 16-cliques	500 [149]	7×10^9	–	–
	Factoring [34]	Breaking RSA	2048 bits	1400	7×10^9	1×10^6 [44]	1 week
	Binary discrete log [34]	Breaking outdated cryptosystems	233 bits	3000	4×10^6	4×10^6 [36]	8 min
Furthest	Elliptic curve cryptography [34]	Breaking elliptic curve cryptography	256 bits	3000	1.1×10^8	9×10^6 [35]	4 h
	Tensor PCA [30]	Unknown	100 variables	900 [150]	1×10^{15}	–	–
	XOR-SAT refutation [81]	Unknown					
	QAOA for max 8-SAT [101]	Unknown	179 variables	28 000	5×10^{11}	7×10^7 [151]	15 h

Compilation Stage(d) compilation, it is common for many resource estimates to stop at Compilation Stage(b). Keeping in mind that the Compilation Stage is also a moving target that often lags behind the most current techniques, we present selected case studies for algorithms outside quantum simulation in Table III and inside quantum simulation in Table IV.

A vital part of the Compilation Stage, adjacent to these substages, is further development of software tools and methods to automate and facilitate compilation and architecture-aware resource estimates. Current examples include Qualtran [45], Q# [46], QREF [47], Bartiq [48], Qrisp [49], and Silq [49] to compose and compile quantum algorithms, with automatic translation of a logical algorithm directly into a full physical instruction set. This will not only provide precise resource counts but will also enable a workflow analogous to modern software profiling. Researchers will be able to test different implementations, identify performance bottlenecks, empirically optimize their quantum code, and verify their compilations.

A. Compilation for early fault-tolerance

As quantum hardware scales, the field is rapidly approaching the era of early fault-tolerance when quantum error correction first begins to enable larger-scale quantum computation. This transition raises a crucial question: What will be the first compelling applications to run on these machines? Several key milestones will define this new era, including, e.g.:

(1) The first QEC demonstration solving a classically intractable nonverifiable problem.
(2) The first QEC demonstration that is classically intractable and quantumly verifiable.
(3) The first scientific discovery enabled using fault tolerance for quantum simulation.
(4) The first QEC demonstration of an algorithm that is classically verifiable.

There is significant confusion in the literature and little consensus on how to evaluate algorithms for this era of early fault-tolerance. Even a seemingly simple question—such as whether algorithmic technique X is better than technique Y for this era—can be divisive. For example, it is often suggested that one should aim for fewer logical qubits [193,194]. However, if this increases the number of gates, then the code distance may need to increase, which could result in a larger overall physical qubit requirement. While the primary goal is usually to

TABLE IV. Noncomprehensive compendium of Compilation Stage research progress in quantum simulation. Dynamics simulation estimates include number of repetitions required to estimate an observable unless marked by [a], which indicates that estimates are only for a single shot. Applications are roughly classified into different time horizons based on the reported number of physical 10^{-3} error-rate superconducting qubits (nearest: $\sim 10^5$, medium: $\sim 10^6$, furthest: $\geq 10^7$) needed for a first demonstration of useful quantum advantage on a surface code. One thing to note is that while there are many high-value applications requiring the simulation of *ab initio* solid-state models, such simulations are quite expensive and more research is needed to bring down costs. Where unavailable, we compare across different architectures by making a subjective classification based on the reported number of logical qubits and Toffoli gates (or equivalent), which may not capture routing overheads, opportunities to parallelize non-Clifford operations, or other space-time tradeoffs. Some other resource estimates worth mentioning include Refs. [185–192].

Horizon	Model problem	Real-World Stage applications	Problem size	Quantum advantage	Comp. Stage(b)		Comp. Stage(c)	
					Logical q.	Toffolis	Phys. q.	Time
Nearest	Condensed matter physics models	2D Ising quench[a]	121 sites	Evaluated vs. MPS [95]	140 [168]	4×10^4	6×10^4	1 s
		2D Hubbard quench[a]	128 spin orbitals		160 [169]	1×10^6	–	–
		Heisenberg quench[a]	70 sites	Evaluated vs. exact sim.	131 [170]	6×10^6	–	–
		3D uniform electron gas ground state	64 electrons 64 orbitals	Evaluated vs. QMC [171]	132 [172]	4×10^7	3×10^5	4 h
		2D uniform electron gas ground state	49 electrons 64 orbitals		140 [173]	7×10^8	–	–
Medium		NMR spectroscopy[a]	32 spins	Plausible vs. TEBD [174]	100 [174]	3×10^7	–	–
		Sachdev–Kitaev–Ye[a]	200 sites	Plausible but not assessed	300 [175]	3×10^7	–	–
	Chemical dynamics (2nd quantization + vibrations)	Singlet-fission organic solar cells[a] [176]	6 states 21 modes	Evaluated vs. MCTDH [177]	154 [176]	3×10^6	–	–
	Nuclear dynamics (1st quantization)	Pionless effective field theory[a]	40 nucleons	Plausible but not assessed	500 [178]	6×10^8	–	–
	X-ray absorption spectroscopy (2nd quantization)	Structure of Li_2MnO_3 battery cathode [179]	18 orbitals	Plausible beyond RAS methods [180]	100 [180]	5×10^{11}	–	–
	Chemical ground states (2nd quantization)	Carbon fixation by Ru-complex [41]	56 orbitals	Not assessed	900 [43]	2×10^8	–	–
		Oxidation by p450 enzyme [132]	58 orbitals	Evaluated vs. DMRG [132]	1200 [43]	5×10^8	–	–
		Nitrogen fixation by FeMo-cofactor [140]	76 orbitals	Evaluated vs. DMRG [133]	1500 [43]	1×10^9	5×10^6	9 h

TABLE IV. (Continued.)

Horizon	Model problem	Real-World Stage applications	Problem size	Quantum advantage	Comp. Stage(b)		Comp. Stage(c)	
					Logical q.	Toffolis	Phys. q.	Time
Furthest		Structure of $LiPF_6$ battery electrolyte [181]	116 orbitals	Plausible but not assessed	18 000 [181]	9×10^{11}	–	–
		Bridged Mo_2 [182]	70 orbitals	Evaluated vs. DMRG [182]	4100 [182]	3×10^{12}	6×10^7	2 years
	Chemical dynamics (1st quantization)	Reaction of NH_3 and BF_3 [183]	40 electrons	Plausible but not assessed	800 [183] w.o. ancillae	2×10^{15}	–	–
		Stopping power for inertial fusion [37]	218 electrons	Plausible beyond TDDFT	6200 [37]	1×10^{15}	–	–
	Material ground states (1st quantization)	Structure of $LiNiO_2$ battery cathode [38]	92 electrons 58 orbitals [2, 2, 1] supercell	Plausible beyond CCSD	1400 [40]	1×10^{14}	–	–
	Material ground states (2nd quantization)				80 000 [38]	1×10^{12}	9×10^7	9 months
		Heterogeneous NiO / PdO catalysis [184]	64–72 atom supercell	Plausible but not assessed	100 000 [184]	3×10^{11}	2×10^8	3 months

minimize the required number of physical qubits, achieving this can be complex. Final resource counts depend sensitively on the chosen QEC architecture, and the detailed cost models needed to truly differentiate approaches. This challenge is compounded by the fact that the underlying platform is a moving target: ideas about error-correcting codes, hardware architectures, and compilation tools are all still evolving at a rapid pace. In the era of early fault-tolerance, it will be critical to codesign quantum algorithms and the specifics of an error-correction architecture. Even hardware decisions such as how to best modularize chips can ultimately play an important role in what sort of tradeoffs are advantageous.

As an example, consider optimizing for circuit depth. Depth only translates to time if operations scheduled simultaneously in a logical layer can actually be executed in parallel. However, one quickly runs into hard limitations in the practical ability to route resource states, and to decode and implement feedback correction, as required for non-Clifford gates in most error-correcting codes. In the regime of early fault-tolerance, it is more likely that one is distilling a smaller number of resource states in the first place in order to save physical qubits, which would require non-Clifford gate serialization. Even when layers consist only of Clifford gates, these gates typically require nontrivial ancilla space to execute (e.g., in the standard surface code), and so parallelizing operations (as one is apt to do when aiming for lower circuit depth) may increase the number of ancillas required and thus, the total number of physical qubits.

As another example, consider estimating an expectation value to precision ϵ for a circuit with G gates. Amplitude estimation requires executing a single deep circuit with $\mathcal{O}(G/\epsilon)$ gates, whereas sampling requires $\mathcal{O}(1/\epsilon^2)$ repetitions of the shallow (G gates) circuit. Intuition might suggest that the shallow circuits of the sampling approach are better for early fault-tolerant hardware, but this intuition is incomplete. Amplitude estimation requires a logical gate error rate of $\mathcal{O}(\epsilon/G)$ to succeed. A naive sampling approach requires a similar error rate. At this error rate, the probability of a logical gate error in any individual sample is $\sim \epsilon$. This introduces a bias in the estimated expectation value on the order of ϵ, and any higher error rate would introduce an unacceptably larger bias. Therefore, naive sampling demands the same code distance as amplitude estimation and a higher total runtime. While error mitigation can relax this requirement, added circuit repetitions required for error mitigation may favor amplitude estimation for achieving practical runtimes. Ultimately, a detailed study is required to assess which approach is more favorable, and the conclusions will depend heavily on the specific architecture, target fidelities, and fault-tolerant error models.

A fine-grained understanding of costs in early fault-tolerance will likely require full compilation to a specific

architecture. We expect that building out precise compilation stacks will be a major focus of industrial efforts over the next few years. However, high-quality, open-source compilers will likely remain scarce for academic use, since developing these tools requires a massive software engineering effort, and compilers are necessarily tailored to their target architectures. For example, the underlying connectivity of the hardware can influence both the choice of error-correcting code (e.g., the surface code vs an LDPC code) [163,195] and the relative difficulty of implementing certain operations (e.g., increased connectivity can allow for more transversal gates) [162,195]. Photonic architectures offer another example of an interesting architecture-specific capability that any serious compilation effort would take advantage of: the ability to effectively "store" a large number of qubits using long fiber-optic delay lines without significantly increasing the cost of the hardware [196].

Despite the practical challenges of evaluating and optimizing algorithms for early fault-tolerance, we believe that this is an area of research rich with opportunities. For example, compiling to abstractions like braiding diagrams, lattice surgery, or modified ZX calculus offers a valuable middle ground between logical and physical circuits. Other efforts may make progress by focusing on specific architectural choices. Our team has constructed a rough cost model for early fault-tolerance that we call FLASQ (fluid allocation of surface code qubits) [168]. Designed to produce an estimate of surface code resources given logical circuits compiled to a 2D grid, we hope it will be useful for quantum algorithms researchers not willing to engage as deeply with the details of quantum error-correction. However, we conclude by noting that algorithmists who do embrace the messiness of fault-tolerance have a rare opportunity in the coming years to play an important role in shaping hardware through codesign by conducting studies that reveal the impact of hardware architecture (e.g., how chips are modularized) on algorithmic overhead.

VII. THE ECONOMICS OF QUANTUM APPLICATION DISCOVERY

The promise of quantum computing is tempered by the immaturity of both its hardware and applications. At this nascent stage, continued progress depends on sustained investment from public and private sources. For these stakeholders, who weigh opportunities based on risk and reward, the challenge of building scalable, fault-tolerant hardware is the primary source of risk. Consequently, tangible hardware progress (achieving higher qubit counts, lower error rates, scalable error-correction experiments, and other critical milestones) helps derisk the endeavor. Such derisking solidifies confidence that large-scale, error-corrected quantum computers are an achievable goal, reassuring investors and securing the long-term vision of the field.

On the other side of the investment equation is the "reward": the commercial or scientific value and revenue opportunities driven by applications with quantum advantage. The projected size of this market (e.g., whether billions or hundreds of billions of dollars annually) directly influences company valuations and strengthens the case for investment. Investors may currently operate under the assumption that "if we build it, apps will come." For that narrative to remain compelling, however, the rate of application discovery must accelerate as hardware matures. A perceived slowdown may be interpreted as a signal that the technology's future value is limited to our current ensemble of established quantum use cases, diminishing the expected reward.

For any stakeholder—be they a government agency, venture capitalist, or corporate executive—the "reward" in quantum's risk/reward ratio is best articulated through specific and credible application case studies. An ideal case study presents a broad use case, grounds it in at least one specific example with quantifiable value, and assesses the quantum resources required for quantum advantage. At present, the few such comprehensive case studies that have been published are largely confined to quantum simulation and cryptanalysis (we summarize much of this work in the tables of the preceding section).

These days, many papers on the quant-ph arXiv explore the topic of quantum applications by relating established quantum algorithms to industrial challenges while avoiding the singular question that relentlessly haunts our collective endeavor: is this likely to ever lead to a quantum advantage? For the most part, hype in the quantum applications space is easy for practicing researchers to identify and ignore. A bigger problem is that the noise created by such work can lead to confusion in policy and business circles. Researchers sometimes imagine that corporate executives and venture capitalists uncritically believe everything they hear about quantum computing. In reality, deep tech investors are extremely familiar with hype around new technologies. Nevertheless, our field may burn through valuable public trust if exaggerated narratives around timelines and applications of quantum computing perpetuate unchecked.

While it is important for scientists to push back on hype in the quantum field, we also caution against overcompensating and effectively "hyping in the opposite direction" (i.e., spreading misleading information, or employing hyperbole in service of combating hype). Such a strategy can lead to excessively cynical takes coming from parts of the academic community, discouraging students from working on quantum applications or besmirching researchers trying to push forward an optimistic yet scientifically grounded research vision. We need top researchers working on quantum applications and attitudes that treat

all work on quantum applications as inherently suspect are detrimental toward that end.

Finally, when it comes to the hard work required to uncover and substantiate truly promising quantum applications, we believe the quantum industry appears to face a classic collective action problem, leading to systemic under-investment in high quality work in this area. While the necessity of new applications is widely acknowledged, prevailing wisdom is that the competitive advantage of major quantum efforts lies primarily in hardware. This view is often paired with a belief that intellectual property for algorithms is either marginal or could be relatively easily circumvented. The resulting incentive structure creates an unfortunate Nash Equilibrium: the dominant strategy for any individual company is to focus an outsized share of resources on a defensible hardware moat while assuming other groups, or academics, shoulder the burden of serious application discovery.

VIII. OUTLOOK

The journey from a theoretical quantum algorithm to a practical, deployed application is a long and complex endeavor. In this perspective, we have introduced a five-stage framework to categorize and understand the maturity of applications research, arguing that progress is not uniform across all stages. While the community has seen success in the discovery of abstract algorithms (Algorithms Stage) and the detailed optimization of a few select use cases (Compilation Stage), the field's forward momentum is limited by underappreciated bottlenecks in the middle stages. Specifically, the challenges of identifying concrete problem instances with quantum advantage (Instances Stage) and matching these problems to real-world applications (Real-World Stage) represent the most significant hurdles to expanding the landscape of quantum applications.

Successfully navigating these stages requires confronting both technical and sociological challenges. The technical difficulty of the Instances Stage often lies in the delicate balance of finding problem instances that cause the quantum algorithm to transition through states complex enough to be classically intractable, yet structured enough to prevent the quantum computer's output from concentrating to trivial, classically predictable values. The primary challenge of the Real-World Stage is due in part to the narrow window of quantum advantage that survives the first two stages and is in part sociological: a knowledge gap exists between the experts who understand the nuanced requirements for quantum advantage and the domain specialists who know where such problem structures might appear in the real world.

We believe, however, that these challenges are surmountable with a strategic and collaborative effort. Ultimately, the continued justification for the investment required to build fault-tolerant quantum computers depends on our ability to articulate a growing portfolio of valuable, achievable, and specific applications. By harnessing the community's ingenuity, we can accelerate the discovery of these applications and bring the transformative potential of quantum computing into clearer view.

ACKNOWLEDGMENTS

The authors thank the following people for helpful discussions or feedback that helped shape our perspective on these matters: Garnet Chan, David Gosset, Matt Harrigan, Sid Jain, Stephen Jordan, Robin Kothari, Tony Metger, Danial Motlagh, Hartmut Neven, Matt Reagor, Juan Miguel Arrazola, Alexander Schmidhuber, Vadim Smelyanskiy, Rolando Somma, Guifre Vidal, Alec White, Adam Zalcman, and Mark Zhandry.

DATA AVAILABILITY

There are no publicly available research data or software supporting this manuscript. Requests for further information or data should be sent to the authors.

[1] D. Bluvstein, S. J. Evered, A. A. Geim, S. H. Li, H. Zhou, T. Manovitz, S. Ebadi, M. Cain, M. Kalinowski, D. Hangleiter, *et al*., Logical quantum processor based on reconfigurable atom arrays, Nature **626**, 58 (2024).

[2] V. V. Sivak, A. Eickbusch, B. Royer, S. Singh, I. Tsioutsios, S. Ganjam, A. Miano, B. Brock, A. Ding, L. Frunzio, *et al*., Real-time quantum error correction beyond break-even, Nature **616**, 50 (2023).

[3] Google Quantum AI, Suppressing quantum errors by scaling a surface code logical qubit, Nature **614**, 676 (2023).

[4] H.-Y. Huang, S. Choi, J. R. McClean, and J. Preskill, The vast world of quantum advantage, arXiv:2508.05720.

[5] S. Aaronson and A. Arkhipov, in *Proceedings of the Forty-Third Annual ACM Symposium on Theory of Computing* (2011), pp. 333–342.

[6] F. Arute, K. Arya, R. Babbush, D. Bacon, J. C. Bardin, R. Barends, R. Biswas, S. Boixo, F. G. Brandao, D. A. Buell, *et al*., Quantum supremacy using a programmable superconducting processor, Nature **574**, 505 (2019).

[7] QuantumFrontiers, Quantum Algorithms: A Call to Action, https://quantumfrontiers.com/2025/04/20/quantum-algorithms-a-call-to-action/ (2025).

[8] O. Lanes, M. Beji, A. D. Corcoles, C. Dalyac, J. M. Gambetta, L. Henriet, A. Javadi-Abhari, A. Kandala, A. Mezzacapo, and C. Porter, A framework for quantum advantage, arXiv:2506.20658.

[9] H. Buhrman, N. Galke, and K. Meichanetzidis, Formal framework for quantum advantage, arXiv:2510.01953.

[10] R. Babbush, J. R. McClean, M. Newman, C. Gidney, S. Boixo, and H. Neven, Focus beyond quadratic speedups for error-corrected quantum advantage, PRX Quantum **2**, 010103 (2021).

[11] T. Hoefler, T. Häner, and M. Troyer, Disentangling hype from practicality: On realistically achieving quantum advantage, Commun. ACM **66**, 82 (2023).

[12] D. Gottesman, in *Proceedings of the XXII International Colloquium on Group Theoretical Methods in Physics*, edited by S. P. Corney, R. Delbourgo, and P. D. Jarvis (International Press, Cambridge, MA, 1999), pp. 32–43.

[13] R. Beals, H. Buhrman, R. Cleve, M. Mosca, and R. De Wolf, Quantum lower bounds by polynomials, J. ACM **48**, 778 (2001).

[14] D. W. Berry, G. Ahokas, R. Cleve, and B. C. Sanders, Efficient quantum algorithms for simulating sparse Hamiltonians, Commun. Math. Phys. **270**, 359 (2007).

[15] Y. Atia and D. Aharonov, Fast-forwarding of Hamiltonians and exponentially precise measurements, Nat. Commun. **8**, 1572 (2017).

[16] C.-F. Chen, M. J. Kastoryano, F. G. Brandão, and A. Gilyén, Quantum thermal state preparation, arXiv:2303.18224.

[17] C.-F. Chen, H.-Y. Huang, J. Preskill, and L. Zhou, in *Proceedings of the 56th Annual ACM Symposium on Theory of Computing* (2024), pp. 1323–1330.

[18] A. Y. Kitaev, A. Shen, and M. N. Vyalyi, *Classical and Quantum Computation* (American Mathematical Soc., 2002), Vol. 47.

[19] L. K. Grover, in *Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing* (1996), pp. 212–219.

[20] D. R. Simon, On the power of quantum computation, SIAM J. Comput. **26**, 1474 (1997).

[21] A. Y. Kitaev, Quantum measurements and the Abelian stabilizer problem, arXiv:quant-ph/9511026.

[22] E. Farhi, J. Goldstone, S. Gutmann, and M. Sipser, Quantum computation by adiabatic evolution, arXiv:quant-ph/0001106.

[23] E. Farhi, J. Goldstone, and S. Gutmann, A quantum approximate optimization algorithm, arXiv:1411.4028.

[24] G. H. Low and I. L. Chuang, Optimal Hamiltonian simulation by quantum signal processing, Phys. Rev. Lett. **118**, 010501 (2017).

[25] A. Gilyén, Y. Su, G. H. Low, and N. Wiebe, in *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing* (2019), pp. 193–204.

[26] A. W. Harrow, A. Hassidim, and S. Lloyd, Quantum algorithm for linear systems of equations, Phys. Rev. Lett. **103**, 150502 (2009).

[27] D. Aharonov, V. Jones, and Z. Landau, in *Proceedings of the Thirty-Eighth Annual ACM Symposium on Theory of Computing* (2006), pp. 427–436.

[28] S. Boixo, S. V. Isakov, V. N. Smelyanskiy, R. Babbush, N. Ding, Z. Jiang, M. J. Bremner, J. M. Martinis, and H. Neven, Characterizing quantum supremacy in near-term devices, Nat. Phys. **14**, 595 (2018).

[29] W. Van Dam and G. Seroussi, Efficient quantum algorithms for estimating gauss sums, arXiv:quant-ph/0207131.

[30] M. B. Hastings, Classical and quantum algorithms for tensor principal component analysis, Quantum **4**, 237 (2020).

[31] Google Quantum AI and Collaborators, Observation of constructive interference at the edge of quantum ergodicity, Nature **646**, 825 (2025).

[32] R. King, R. Kothari, R. Babbush, S. Boixo, K. Kechedzhi, T. E. O'Brien, and V. Smelyanskiy, A simplified version of the quantum OTOC$^{(2)}$ problem, arXiv:2510.19751.

[33] S. Aaronson, Read the fine print, Nat. Phys. **11**, 291 (2015).

[34] P. W. Shor, Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer, SIAM Rev. **41**, 303 (1999).

[35] D. Litinski, How to compute a 256-bit elliptic curve private key with only 50 million toffoli gates, arXiv:2306.08585.

[36] M. Garn and A. Kan, Quantum resource estimates for computing binary elliptic curve discrete logarithms, IEEE Trans. Quantum Eng. **6**, 1 (2025).

[37] N. C. Rubin, D. W. Berry, A. Kononov, F. D. Malone, T. Khattar, A. White, J. Lee, H. Neven, R. Babbush, and A. D. Baczewski, Quantum computation of stopping power for inertial fusion target design, Proc. Natl. Acad. Sci. **121**, e2317772121 (2024).

[38] N. C. Rubin, D. W. Berry, F. D. Malone, A. F. White, T. Khattar, A. E. DePrince III, S. Sicolo, M. Küehn, M. Kaicher, J. Lee, *et al*., Fault-tolerant quantum simulation of materials using Bloch orbitals, PRX Quantum **4**, 040303 (2023).

[39] R. Santagati, A. Aspuru-Guzik, R. Babbush, M. Degroote, L. González, E. Kyoseva, N. Moll, M. Oppel, R. M. Parrish, N. C. Rubin, *et al*., Drug design on quantum computers, Nat. Phys. **20**, 549 (2024).

[40] D. W. Berry, N. C. Rubin, A. O. Elnabawy, G. Ahlers, A. E. DePrince III, J. Lee, C. Gogolin, and R. Babbush, Quantum simulation of realistic materials in first quantization using non-local pseudopotentials, npj Quantum Inf. **10**, 130 (2024).

[41] V. von Burg, G. H. Low, T. Häner, D. S. Steiger, M. Reiher, M. Roetteler, and M. Troyer, Quantum computing enhanced computational catalysis, Phys. Rev. Res. **3**, 033055 (2021).

[42] A. Caesura, C. L. Cortes, W. Pol, S. Sim, M. Steudtner, G.-L. R. Anselmetti, M. Degroote, N. Moll, R. Santagati, M. Streif, *et al*., Faster quantum chemistry simulations on a quantum computer with improved tensor factorization and active volume compilation, PRX Quantum **6**, 030337 (2025).

[43] G. H. Low, R. King, D. W. Berry, Q. Han, A. E. DePrince III, A. F. White, R. Babbush, R. D. Somma, and N. C. Rubin, Fast quantum simulation of electronic structure by spectral amplification, Phys. Rev. X **15**, 041016 (2025).

[44] C. Gidney, How to factor 2048 bit RSA integers with less than a million noisy qubits, arXiv:2505.15917.

[45] M. P. Harrigan, T. Khattar, C. Yuan, A. Peduri, N. Yosri, F. D. Malone, R. Babbush, and N. C. Rubin, Expressing and analyzing quantum algorithms with qualtran, arXiv:2409.04643.

[46] K. Singhal, K. Hietala, S. Marshall, and R. Rand, Q# as a quantum algorithmic language, EPTCS **394**, 170 (2023).

[47] PsiQ, QREF GitHub Repository, https://github.com/PsiQ/QREF (a).

[48] PsiQ, Bartiq GitHub Repository, https://github.com/PsiQ/bartiq (b).

[49] R. Seidel, S. Bock, R. Zander, M. Petrič, N. Steinmann, N. Tcholtchev, and M. Hauswirth, Qrisp: A framework for compilable high-level programming of gate-based quantum computers, arXiv:2406.14792.

[50] S. Aaronson and S.-H. Hung, in *Proceedings of the 55th Annual ACM Symposium on Theory of Computing* (2023), pp. 933–944.

[51] H.-Y. Huang, M. Broughton, J. Cotler, S. Chen, J. Li, M. Mohseni, H. Neven, R. Babbush, R. Kueng, J. Preskill, *et al*., Quantum advantage in learning from experiments, Science **376**, 1182 (2022).

[52] H.-Y. Huang, Y. Tong, D. Fang, and Y. Su, Learning many-body Hamiltonians with Heisenberg-limited scaling, Phys. Rev. Lett. **130**, 200403 (2023).

[53] H.-Y. Huang, S. Chen, and J. Preskill, Learning to predict arbitrary quantum processes, PRX Quantum **4**, 040337 (2023).

[54] H.-Y. Huang, Y. Liu, M. Broughton, I. Kim, A. Anshu, Z. Landau, and J. R. McClean, in *Proceedings of the 56th Annual ACM Symposium on Theory of Computing* (2024), pp. 1343–1351.

[55] R. R. Allen, F. Machado, I. L. Chuang, H.-Y. Huang, and S. Choi, Quantum computing enhanced sensing, arXiv:2501.07625.

[56] T. Yamakawa and M. Zhandry, Verifiable quantum advantage without structure, J. ACM **71**, 1 (2024).

[57] S. P. Jordan, N. Shutty, M. Wootters, A. Zalcman, A. Schmidhuber, R. King, S. V. Isakov, T. Khattar, and R. Babbush, Optimization by decoded quantum interferometry, Nature **646**, 831 (2025).

[58] R. Babbush, D. W. Berry, R. Kothari, R. D. Somma, and N. Wiebe, Exponential quantum speedup in simulating coupled classical oscillators, Phys. Rev. X **13**, 041041 (2023).

[59] R. D. Somma, G. H. Low, D. W. Berry, and R. Babbush, Quantum algorithm for linear matrix equations, arXiv:2508.02822.

[60] M. B. Hastings, A short path quantum algorithm for exact optimization, Quantum **2**, 78 (2018).

[61] S. Lloyd, S. Garnerone, and P. Zanardi, Quantum algorithms for topological and geometric analysis of data, Nat. Commun. **7**, 10138 (2016).

[62] D. Aharonov and I. Arad, The BQP-hardness of approximating the Jones polynomial, New J. Phys. **13**, 035019 (2011).

[63] P. W. Shor and S. P. Jordan, Estimating Jones polynomials is a complete problem for one clean qubit, Quantum Inf. Comput. **8**, 681 (2008).

[64] S. Lloyd, Universal quantum simulators, Science **273**, 1073 (1996).

[65] S. Aaronson and A. Ambainis, in *Proceedings of the Forty-Seventh Annual ACM Symposium on Theory of Computing* (2015), pp. 307–316.

[66] A. M. Childs, R. Cleve, E. Deotto, E. Farhi, S. Gutmann, and D. A. Spielman, in *Proceedings of the Thirty-Fifth Annual ACM Symposium on Theory of Computing* (2003), pp. 59–68.

[67] D. Aharonov, W. Van Dam, J. Kempe, Z. Landau, S. Lloyd, and O. Regev, Adiabatic quantum computation is equivalent to standard quantum computation, SIAM Rev. **50**, 755 (2008).

[68] A. Gilyén, M. B. Hastings, and U. Vazirani, in *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing* (2021), pp. 1357–1369.

[69] P. C. Costa, S. Jordan, and A. Ostrander, Quantum algorithm for simulating the wave equation, Phys. Rev. A **99**, 012323 (2019).

[70] D. W. Berry, High-order quantum algorithm for solving linear differential equations, J. Phys. A: Math. Theor. **47**, 105301 (2014).

[71] J.-P. Liu, H. Ø. Kolden, H. K. Krovi, N. F. Loureiro, K. Trivisa, and A. M. Childs, Efficient quantum algorithm for dissipative nonlinear differential equations, Proc. Natl. Acad. Sci. **118**, e2026805118 (2021).

[72] G. H. Low and Y. Su, in *2024 IEEE 65th Annual Symposium on Foundations of Computer Science (FOCS)* (IEEE, 2024), pp. 1051–1062.

[73] H.-Y. Huang, M. Broughton, N. Eassa, H. Neven, R. Babbush, and J. R. McClean, Generative quantum advantage for classical and quantum problems, arXiv:2509.09033.

[74] G. Kuperberg, A subexponential-time quantum algorithm for the dihedral hidden subgroup problem, SIAM J. Comput. **35**, 170 (2005).

[75] K. Friedl, G. Ivanyos, F. Magniez, M. Santha, and P. Sen, Hidden translation and translating coset in quantum computing, SIAM J. Comput. **43**, 1 (2014).

[76] S. Hallgren, Polynomial-time quantum algorithms for Pell's equation and the principal ideal problem, J. ACM **54**, 1 (2007).

[77] W. Van Dam, S. Hallgren, and L. Ip, Quantum algorithms for some hidden shift problems, SIAM J. Comput. **36**, 763 (2006).

[78] A. M. Childs, D. Jao, and V. Soukharev, Constructing elliptic curve isogenies in quantum subexponential time, J. Math. Cryptol. **8**, 1 (2014).

[79] W. Van Dam and I. E. Shparlinski, in *Workshop on Quantum Computation, Communication, and Cryptography* (Springer, 2008), pp. 1–10.

[80] K. S. Kedlaya, Quantum computation of zeta functions of curves, Comput. Complex. **15**, 1 (2006).

[81] A. Schmidhuber, R. O'Donnell, R. Kothari, and R. Babbush, Quartic quantum speedups for planted inference, Phys. Rev. X **15**, 021077 (2025).

[82] R. D. Somma, R. King, R. Kothari, T. E. O'Brien, and R. Babbush, Shadow Hamiltonian simulation, Nat. Commun. **16**, 2690 (2025).

[83] D. Aharonov and A. Ta-Shma, in *Proceedings of the Thirty-Fifth Annual ACM Symposium on Theory of Computing* (2003), pp. 20–29.

[84] O. Regev, Quantum computation and lattice problems, SIAM J. Comput. **33**, 738 (2004).

[85] D. Aharonov and O. Regev, Lattice problems in NP ∩ coNP, J. ACM **52**, 749 (2005).

[86] O. Regev, On lattices, learning with errors, random linear codes, and cryptography, J. ACM **56**, 1 (2009).

[87] Y. Kim, A. Eddins, S. Anand, K. X. Wei, E. Van Den Berg, S. Rosenblatt, H. Nayfeh, Y. Wu, M. Zaletel, K. Temme, *et al*., Evidence for the utility of quantum computing before fault tolerance, Nature **618**, 500 (2023).

[88] A. D. King, A. Nocera, M. M. Rams, J. Dziarmaga, R. Wiersema, W. Bernoudy, J. Raymond, N. Kaushal, N.

Heinsdorf, R. Harris, *et al*., Beyond-classical computation in quantum simulation, Science **388**, 199 (2025).
[89] T. Begušić, J. Gray, and G. K.-L. Chan, Fast and converged classical simulations of evidence for the utility of quantum computing before fault tolerance, Sci. Adv. **10**, eadk4321 (2024).
[90] K. Kechedzhi, S. V. Isakov, S. Mandrà, B. Villalonga, X. Mi, S. Boixo, and V. Smelyanskiy, Effective quantum volume, fidelity and computational cost of noisy quantum processing experiments, Fut. Gen. Comput. Syst. **153**, 431 (2024).
[91] J. Tindall, M. Fishman, E. M. Stoudenmire, and D. Sels, Efficient tensor network simulation of IBM's eagle kicked Ising experiment, PRX Quantum **5**, 010308 (2024).
[92] J. Tindall, A. Mello, M. Fishman, M. Stoudenmire, and D. Sels, Dynamics of disordered quantum systems with two-and three-dimensional tensor networks, arXiv:2503.05693.
[93] L. Mauron and G. Carleo, Challenging the quantum advantage frontier with large-scale classical simulations of annealing dynamics, arXiv:2503.08247.
[94] R. Haghshenas, E. Chertkov, M. Mills, W. Kadow, S.-H. Lin, Y.-H. Chen, C. Cade, I. Niesen, T. Begušić, M. S. Rudolph, *et al*., Digital quantum magnetism at the frontier of classical simulations, arXiv:2503.20870.
[95] S. Mandrá, N. Astrakhantsev, S. Isakov, B. Villalonga, B. Ware, T. Westerhout, and K. Kechedzhi, A heuristic for matrix product state simulation of out-of-equilibrium dynamics of two-dimensional transverse-field Ising models, arXiv:2511.23438.
[96] S. Hallgren, in *Proceedings of the Thirty-Seventh Annual ACM Symposium on Theory of Computing* (2005), pp. 468–474.
[97] A. Schmidhuber and A. Zlokapa, Quartic quantum speedups for community detection, arXiv:2510.08494.
[98] A. Gu and S. P. Jordan, Algebraic geometry codes and decoded quantum interferometry, arXiv:2510.06603.
[99] D. C. Hillel, Optimization of quadratic constraints by decoded quantum interferometry, arXiv:2510.08061.
[100] P. Briaud, I. Dinur, R. Ghosal, A. Jain, P. Lou, and A. Sahai, Quantum advantage via solving multivariate polynomials, arXiv:2509.07276.
[101] S. Boulebnane and A. Montanaro, Solving boolean satisfiability problems with the quantum approximate optimization algorithm, PRX Quantum **5**, 030348 (2024).
[102] Z. Brakerski, P. Christiano, U. Mahadev, U. Vazirani, and T. Vidick, A cryptographic test of quantumness and certifiable randomness from a single quantum device, J. ACM **68**, 1 (2021).
[103] G. D. Kahanamoku-Meyer, S. Choi, U. V. Vazirani, and N. Y. Yao, Classically verifiable quantum advantage from a computational bell test, Nat. Phys. **18**, 918 (2022).
[104] Z. Brakerski, V. Koppula, U. Vazirani, and T. Vidick, Simpler proofs of quantumness, arXiv:2005.04826.
[105] P. Arabadjieva, A. Gheorghiu, V. Gitton, and T. Metger, Single-round proofs of quantumness from knowledge assumptions, arXiv:2405.15736.
[106] D. Zhu, G. D. Kahanamoku-Meyer, L. Lewis, C. Noel, O. Katz, B. Harraz, Q. Wang, A. Risinger, L. Feng, D. Biswas, *et al*., Interactive cryptographic proofs of quantumness using mid-circuit measurements, Nat. Phys. **19**, 1725 (2023).
[107] O. T. Unke, S. Chmiela, H. E. Sauceda, M. Gastegger, I. Poltavsky, K. T. Schutt, A. Tkatchenko, and K.-R. Muller, Machine learning force fields, Chem. Rev. **121**, 10142 (2021).
[108] H.-Y. Huang, M. Broughton, M. Mohseni, R. Babbush, S. Boixo, H. Neven, and J. R. McClean, Power of data in quantum machine learning, Nat. Commun. **12**, 2631 (2021).
[109] H.-Y. Huang, R. Kueng, G. Torlai, V. V. Albert, and J. Preskill, Provably efficient machine learning for quantum many-body problems, Science **377**, eabk3333 (2022).
[110] M. J. Bremner, C. Mora, and A. Winter, Are random pure states useful for quantum computation? Phys. Rev. Lett. **102**, 190502 (2009).
[111] D. Gross, S. T. Flammia, and J. Eisert, Most quantum states are too entangled to be useful as computational resources, Phys. Rev. Lett. **102**, 190501 (2009).
[112] A. Angrisani, A. Schmidhuber, M. S. Rudolph, M. Cerezo, Z. Holmes, and H.-Y. Huang, Classically estimating observables of noiseless quantum circuits, Phys. Rev. Lett. **135**, 170602 (2025).
[113] M. Srednicki, Chaos and quantum thermalization, Phys. Rev. E **50**, 888 (1994).
[114] D. S. Wild and Á. M. Alhambra, Classical simulation of short-time quantum dynamics, PRX Quantum **4**, 020340 (2023).
[115] R. L. Mann and R. M. Minko, Algorithmic cluster expansions for quantum problems, PRX Quantum **5**, 010305 (2024).
[116] M. L. Mehta, *Random Matrices* (Elsevier, 2004), Vol. 142.
[117] J. S. Cotler, D. K. Mark, H.-Y. Huang, F. Hernández, J. Choi, A. L. Shaw, M. Endres, and S. Choi, Emergent quantum state designs from individual many-body wave functions, PRX Quantum **4**, 010311 (2023).
[118] M. Kliesch, C. Gogolin, M. Kastoryano, A. Riera, and J. Eisert, Locality of temperature, Phys. Rev. X **4**, 031019 (2014).
[119] A. Bakshi, A. Liu, A. Moitra, and E. Tang, High-temperature gibbs states are unentangled and efficiently preparable, arXiv:2403.16850.
[120] A. Bakshi, S. Choi, and S. Pilatowsky-Cameo, Entanglement in quantum spin chains is strictly finite at any temperature, arXiv:2601.22286.
[121] D. Bluvstein, A. Omran, H. Levine, A. Keesling, G. Semeghini, S. Ebadi, T. T. Wang, A. A. Michailidis, N. Maskara, W. W. Ho, *et al*., Controlling quantum many-body dynamics in driven Rydberg atom arrays, Science **371**, 1355 (2021).
[122] D. A. Spielman and S.-H. Teng, Smoothed analysis of algorithms: Why the simplex algorithm usually takes polynomial time, J. ACM **51**, 385 (2004).
[123] T. Roughgarden, Beyond worst-case analysis, Commun. ACM **62**, 88 (2019).
[124] G. D. Kahanamoku-Meyer, S. Ragavan, V. Vaikuntanathan, and K. Van Kirk, in *Proceedings of the 57th Annual ACM Symposium on Theory of Computing* (2025), pp. 1496–1507.

[125] H. Soller, M. Gschwendtner, S. Shabani, and W. Svejstrup, *The Year of Quantum: From Concept to Reality in 2025*, 4th Annual Edition.Quantum Technology Monitor (McKinsey & Company, 2025),
[126] XPrize, XPrize Quantum-Centric Applications, https://www.xprize.org/prizes/qc-apps (2024).
[127] J. Gottweis and V. Natarajan, Accelerating scientific breakthroughs with an AI co-scientist, Google Research Blog (2025).
[128] S. Lee, J. Lee, H. Zhai, Y. Tong, A. M. Dalzell, A. Kumar, P. Helms, J. Gray, Z.-H. Cui, W. Liu, *et al*., Evaluating the evidence for exponential quantum advantage in ground-state quantum chemistry, Nat. Commun. **14**, 1952 (2023).
[129] G. K.-L. Chan, Spiers memorial lecture: Quantum chemistry, classical heuristics, and quantum advantage, Faraday Discuss. **254**, 11 (2024).
[130] J. W. Scannell, A. Blanckley, H. Boldon, and B. Warrington, Diagnosing the decline in pharmaceutical R&D efficiency, Nat. Rev. Drug Discov. **11**, 191 (2012).
[131] Z. Li, J. Li, N. S. Dattani, C. Umrigar, and G. K. Chan, The electronic complexity of the ground-state of the Femo cofactor of nitrogenase as relevant to quantum simulations, J. Chem. Phys. **150**, 024302 (2019).
[132] J. J. Goings, A. White, J. Lee, C. S. Tautermann, M. Degroote, C. Gidney, T. Shiozaki, R. Babbush, and N. C. Rubin, Reliably assessing the electronic structure of cytochrome p450 on today's classical computers and tomorrow's quantum computers, Proc. Natl. Acad. Sci. **119**, e2203533119 (2022).
[133] D. W. Berry, Y. Tong, T. Khattar, A. White, T. I. Kim, G. H. Low, S. Boixo, Z. Ding, L. Lin, S. Lee, *et al*., Rapid initial-state preparation for the quantum simulation of strongly correlated molecules, PRX Quantum **6**, 020327 (2025).
[134] H. Zhai and G. K. Chan, Low communication high performance ab initio density matrix renormalization group algorithms, J. Chem. Phys. **154**, 224116 (2021).
[135] C. Xiang, W. Jia, W.-H. Fang, and Z. Li, Distributed multi-GPU ab initio density matrix renormalization group algorithm with applications to the p-cluster of nitrogenase, J. Chem. Theory Comput. **20**, 775 (2024).
[136] D. Fausti, R. Tobey, N. Dean, S. Kaiser, A. Dienst, M. C. Hoffmann, S. Pyon, T. Takayama, H. Takagi, and A. Cavalleri, Light-induced superconductivity in a stripe-ordered cuprate, Science **331**, 189 (2011).
[137] X. Mi, M. Ippoliti, C. Quintana, A. Greene, Z. Chen, J. Gross, F. Arute, K. Arya, J. Atalaya, R. Babbush, *et al*., Time-crystalline eigenstate order on a quantum processor, Nature **601**, 531 (2022).
[138] H. Rabitz, R. de Vivie-Riedle, M. Motzkus, and K. Kompa, Whither the future of controlling quantum phenomena? Science **288**, 824 (2000).
[139] A. H. Zewail, Femtochemistry: Atomic-scale dynamics of the chemical bond using ultrafast lasers (nobel lecture), Angew. Chem. Int. Ed. **39**, 2586 (2000).
[140] M. Reiher, N. Wiebe, K. M. Svore, D. Wecker, and M. Troyer, Elucidating reaction mechanisms on quantum computers, Proc. Natl. Acad. Sci. **114**, 7555 (2017).
[141] N. C. Jones, R. Van Meter, A. G. Fowler, P. L. McMahon, J. Kim, T. D. Ladd, and Y. Yamamoto, Layered architecture for quantum computing, Phys. Rev. X **2**, 031007 (2012).
[142] A. G. Fowler, M. Mariantoni, J. M. Martinis, and A. N. Cleland, Surface codes: Towards practical large-scale quantum computation, Phys. Rev. A:At., Mol. Opt. Phys. **86**, 032324 (2012).
[143] J. O'Gorman and E. T. Campbell, Quantum computation with realistic magic-state factories, Phys. Rev. A **95**, 032338 (2017).
[144] V. Gheorghiu and M. Mosca, Benchmarking the quantum cryptanalysis of symmetric, public-key and hash-based cryptographic schemes, arXiv:1902.02332.
[145] J. Lee, D. W. Berry, C. Gidney, W. J. Huggins, J. R. McClean, N. Wiebe, and R. Babbush, Even more efficient quantum computations of chemistry through tensor hypercontraction, PRX Quantum **2**, 030305 (2021).
[146] D. Rocca, C. L. Cortes, J. F. Gonthier, P. J. Ollitrault, R. M. Parrish, G.-L. Anselmetti, M. Degroote, N. Moll, R. Santagati, and M. Streif, Reducing the runtime of fault-tolerant quantum simulations in chemistry through symmetry-compressed double factorization, J. Chem. Theory Comput. **20**, 4639 (2024).
[147] G. D. Kahanamoku-Meyer and N. Y. Yao, Fast quantum integer multiplication with zero ancillas, arXiv:2403.18006.
[148] T. Khattar, N. Shutty, C. Gidney, A. Zalcman, N. Yosri, D. Maslov, R. Babbush, and S. P. Jordan, Verifiable quantum advantage via optimized DQI circuits, arXiv:2510.10967.
[149] D. W. Berry, Y. Su, C. Gyurik, R. King, J. Basso, A. D. T. Barba, A. Rajput, N. Wiebe, V. Dunjko, and R. Babbush, Analyzing prospects for quantum advantage in topological data analysis, PRX Quantum **5**, 010319 (2024).
[150] E. Fontana, S. Omanakuttan, J. L. Kim, J. Sullivan, M. Perlin, R. Shaydulin, and S. Chakrabarti, End-to-end quantum algorithms for tensor problems, arXiv:2510.07273.
[151] S. Omanakuttan, Z. He, Z. Zhang, T. Hao, A. Babakhani, S. Boulebnane, S. Chakrabarti, D. Herman, J. Sullivan, M. A. Perlin, *et al*., Threshold for fault-tolerant quantum advantage with the quantum approximate optimization algorithm, arXiv:2504.01897.
[152] S. Chakrabarti, R. Krishnakumar, G. Mazzola, N. Stamatopoulos, S. Woerner, and W. J. Zeng, A threshold for quantum advantage in derivative pricing, Quantum **5**, 463 (2021).
[153] Y. R. Sanders, D. W. Berry, P. C. Costa, L. W. Tessler, N. Wiebe, C. Gidney, H. Neven, and R. Babbush, Compilation of fault-tolerant quantum heuristics for combinatorial optimization, PRX Quantum **1**, 020312 (2020).
[154] J. Penuel, A. Katabarwa, P. D. Johnson, C. Farquhar, Y. Cao, and M. C. Garrett, Feasibility of accelerating incompressible computational fluid dynamics simulations with fault-tolerant quantum computers, arXiv:2406.06323.
[155] A. M. Dalzell, B. D. Clader, G. Salton, M. Berta, C. Y.-Y. Lin, D. A. Bader, N. Stamatopoulos, M. J. Schuetz, F. G. Brandão, H. G. Katzgraber, *et al*., End-to-end resource analysis for quantum interior-point methods and portfolio optimization, PRX Quantum **4**, 040325 (2023).
[156] D. W. Berry, C. Gidney, M. Motta, J. R. McClean, and R. Babbush, Qubitization of arbitrary basis quantum

chemistry leveraging sparsity and low rank factorization, Quantum **3**, 208 (2019).
[157] C. Chevignard, P.-A. Fouque, and A. Schrottenloher, in *Annual International Cryptology Conference* (Springer, 2025), pp. 384–415.
[158] R. Babbush, N. Wiebe, J. McClean, J. McClain, H. Neven, and G. K.-L. Chan, Low-depth quantum simulation of materials, Phys. Rev. X **8**, 011044 (2018).
[159] M. Motta, E. Ye, J. R. McClean, Z. Li, A. J. Minnich, R. Babbush, and G. K.-L. Chan, Low rank representations for quantum simulation of electronic structure, npj Quantum Inf. **7**, 83 (2021).
[160] R. Babbush, D. W. Berry, J. R. McClean, and H. Neven, Quantum simulation of chemistry with sublinear scaling in basis size, npj Quantum Inf. **5**, 92 (2019).
[161] C. Gidney, N. Shutty, and C. Jones, Magic state cultivation: growing T states as cheap as CNOT gates, arXiv:2409.17595.
[162] D. Litinski and N. Nickerson, Active volume: An architecture for efficient fault-tolerant quantum computers with limited non-local connections, arXiv:2211.15465.
[163] S. Bravyi, A. W. Cross, J. M. Gambetta, D. Maslov, P. Rall, and T. J. Yoder, High-threshold and low-overhead fault-tolerant quantum memory, Nature **627**, 778 (2024).
[164] H. Zhou, C. Zhao, M. Cain, D. Bluvstein, N. Maskara, C. Duckering, H.-Y. Hu, S.-T. Wang, A. Kubica, and M. D. Lukin, Low-overhead transversal fault tolerance for universal quantum computation, Nature **646**, 303 (2025).
[165] D. Litinski, A game of surface codes: Large-scale quantum computing with lattice surgery, Quantum **3**, 128 (2019).
[166] C. Gidney, M. Newman, P. Brooks, and C. Jones, Yoked surface codes, Nat. Commun. **16**, 4498 (2025).
[167] Y. Su, D. W. Berry, N. Wiebe, N. Rubin, and R. Babbush, Fault-tolerant quantum simulations of chemistry in first quantization, PRX Quantum **2**, 040332 (2021).
[168] W. J. Huggins, T. Khattar, A. Xu, M. Harrigan, C. Kang, G. H. Low, A. Fowler, N. C. Rubin, and R. Babbush, The FLuid Allocation of Surface code Qubits (FLASQ) cost model for early fault-tolerant quantum algorithms, arXiv:2511.08508.
[169] A. Kan and B. C. Symons, Resource-optimized fault-tolerant simulation of the fermi-hubbard model and high-temperature superconductor models, npj Quantum Inf. **11**, 138 (2025).
[170] Y. Nam and D. Maslov, Low-cost quantum circuits for classically intractable instances of the Hamiltonian dynamics simulation problem, npj Quantum Inf. **5**, 44 (2019).
[171] R. Babbush, C. Gidney, D. W. Berry, N. Wiebe, J. McClean, A. Paler, A. Fowler, and H. Neven, Encoding electronic spectra in quantum circuits with linear t complexity, Phys. Rev. X **8**, 041015 (2018).
[172] I. D. Kivlichan, C. Gidney, D. W. Berry, N. Wiebe, J. McClean, W. Sun, Z. Jiang, N. Rubin, A. Fowler, A. Aspuru-Guzik, *et al.*, Improved fault-tolerant quantum simulation of condensed-phase correlated electrons via trotterization, Quantum **4**, 296 (2020).
[173] S. McArdle, E. Campbell, and Y. Su, Exploiting fermion number in factorized decompositions of the electronic structure Hamiltonian, Phys. Rev. A **105**, 012403 (2022).
[174] J. E. Elenewski, C. M. Camara, and A. Kalev, Prospects for nmr spectral prediction on fault-tolerant quantum computers, arXiv:2406.09340.
[175] R. Babbush, D. W. Berry, and H. Neven, Quantum simulation of the Sachdev-Ye-Kitaev model by asymmetric qubitization, Phys. Rev. A **99**, 040301(R) (2019).
[176] D. Motlagh, R. A. Lang, P. Jain, J. A. Campos-Gonzalez-Angulo, W. Maxwell, T. Zeng, A. Aspuru-Guzik, and J. M. Arrazola, Quantum algorithm for vibronic dynamics: Case study on singlet fission solar cell design, Quantum Sci. Technol. **10**, 045048 (2025).
[177] Y. Xie, J. Zheng, and Z. Lan, Full-dimensional multilayer multiconfigurational time-dependent Hartree study of electron transfer dynamics in the anthracene/C60 complex, J. Chem. Phys. **142**, 084706 (2015).
[178] L. Spagnoli, C. Lissoni, and A. Roggero, Quantum simulation of nuclear dynamics in first quantization, arXiv:2507.22814.
[179] A. Kunitsa, D. Dhawan, S. Fomichev, J. M. Arrazola, M. Zhang, and T. F. Stetina, Quantum simulation of electron energy loss spectroscopy for battery materials, arXiv:2508.15935.
[180] S. Fomichev, P. A. Casares, J. Soni, U. Azad, A. Kunitsa, A.-C. Voigt, J. E. Mueller, and J. M. Arrazola, Fast simulations of x-ray absorption spectroscopy for battery materials on a quantum computer, arXiv:2506.15784.
[181] I. H. Kim, Y.-H. Liu, S. Pallister, W. Pol, S. Roberts, and E. Lee, Fault-tolerant resource estimate for quantum chemical simulations: Case study on Li-ion battery electrolyte molecules, Phys. Rev. Res. **4**, 023019 (2022).
[182] N. Bellonzi, A. Kunitsa, J. T. Cantin, J. A. Campos-Gonzalez-Angulo, M. D. Radin, Y. Zhou, P. D. Johnson, L. A. Martínez-Martínez, M. R. Jangrouei, A. S. Brahmachari, *et al.*, Feasibility of accelerating homogeneous catalyst discovery with fault-tolerant quantum computers, arXiv:2406.06335.
[183] F. H. da Jornada, M. Lostaglio, S. Pallister, B. Şahinoğlu, and K. I. Seetharam, A comprehensive framework to simulate real-time chemical dynamics on a fault-tolerant quantum computer, arXiv:2504.06348.
[184] A. V. Ivanov, C. Sünderhauf, N. Holzmann, T. Ellaby, R. N. Kerber, G. Jones, and J. Camps, Quantum computation for periodic solids in second quantization, Phys. Rev. Res. **5**, 013200 (2023).
[185] A. A. Agrawal, J. Job, T. L. Wilson, S. Saadatmand, M. J. Hodson, J. Y. Mutus, A. Caesura, P. D. Johnson, J. E. Elenewski, K. J. Morrell, *et al.*, Quantifying fault tolerant simulation of strongly correlated systems using the Fermi-Hubbard model, arXiv:2406.06511.
[186] N. Nguyen, T. W. Watts, B. Link, K. S. Williams, Y. R. Sanders, S. J. Elman, M. Kieferova, M. J. Bremner, K. J. Morrell, J. Elenewski, *et al.*, Quantum computing for corrosion-resistant materials and anti-corrosive coatings design, arXiv:2406.18759.
[187] T. W. Watts, M. Otten, J. T. Necaise, N. Nguyen, B. Link, K. S. Williams, Y. R. Sanders, S. J. Elman, M. Kieferova, M. J. Bremner, *et al.*, Fullerene-encapsulated cyclic

ozone for the next generation of nano-sized propellants via quantum computation, arXiv:2408.13244.

[188] L. Clinton, T. Cubitt, B. Flynn, F. M. Gambetta, J. Klassen, A. Montanaro, S. Piddock, R. A. Santos, and E. Sheridan, Towards near-term quantum simulation of materials, Nat. Commun. **15**, 211 (2024).

[189] S. Malpathak, S. D. Kallullathil, I. Loaiza, S. Fomichev, J. M. Arrazola, and A. F. Izmaylov, Trotter simulation of vibrational Hamiltonians on a quantum computer, J. Chem. Theory Comput. **22**, 95 (2026).

[190] J. S. Baker, P. A. M. Casares, M. S. Zini, J. Thik, D. Banerjee, C. Ling, A. Delgado, and J. M. Arrazola, Simulating optically active spin defects with a quantum computer, Phys. Rev. A **110**, 032606 (2024).

[191] I. Loaiza, S. Fomichev, D. Motlagh, P. A. M. Casares, D. H. Menendez, S. Shum, A. Delgado, and J. M. Arrazola, Simulating near-infrared spectroscopy on a quantum computer for enhanced chemical detection, arXiv:2504.10602.

[192] P. A. M. Casares, Y. Zhou, U. Azad, S. Fomichev, J. S. Baker, C. Ling, D. Banerjee, A. Delgado, and J. M. Arrazola, Quantum algorithms to detect ODMR-active defects for quantum sensing applications, arXiv:2508.13281.

[193] L. Lin and Y. Tong, Heisenberg-limited ground-state energy estimation for early fault-tolerant quantum computers, PRX Quantum **3**, 010318 (2022).

[194] A. Katabarwa, K. Gratsea, A. Caesura, and P. D. Johnson, Early fault-tolerant quantum computing, PRX Quantum **5**, 020101 (2024).

[195] D. Bluvstein, H. Levine, G. Semeghini, T. T. Wang, S. Ebadi, M. Kalinowski, A. Keesling, N. Maskara, H. Pichler, M. Greiner, *et al.*, A quantum processor based on coherent transport of entangled atom arrays, Nature **604**, 451 (2022).

[196] S. Bartolucci, P. Birchall, H. Bombin, H. Cable, C. Dawson, M. Gimeno-Segovia, E. Johnston, K. Kieling, N. Nickerson, M. Pant, *et al.*, Fusion-based quantum computation, Nat. Commun. **14**, 912 (2023).