As Accurate as Needed, as Efficient as Possible: Approximations in DD-based Quantum Circuit Simulation

Stefan Hillmich*, Richard Kueng*, Igor L. Markov†, and Robert Wille*‡
*Institute for Integrated Circuits, Johannes Kepler University Linz, Austria
†Department of EECS, University of Michigan, USA
‡Software Competence Center Hagenberg GmbH (SCCH), Austria
stefan.hillmich@jku.at, richard.kueng@jku.at, imarkov@eecs.umich.edu, robert.wille@jku.at
https://iic.jku.at/eda/research/quantum/

***Abstract*—Quantum computers promise to solve important problems faster than conventional computers. However, unleashing this power has been challenging. In particular, design automation runs into (1) the probabilistic nature of quantum computation and (2) exponential requirements for computational resources on non-quantum hardware. In quantum circuit simulation, *Decision Diagrams* (DDs) have previously shown to reduce the required memory in many important cases by exploiting redundancies in the quantum state. In this paper, we show that this reduction can be amplified by exploiting the probabilistic nature of quantum computers to achieve even more compact representations. Specifically, we propose two new DD-based simulation strategies that approximate the quantum states to attain more compact representations, while, at the same time, allowing the user to control the resulting degradation in accuracy. We also analytically prove the effect of multiple approximations on the attained accuracy and empirically show that the resulting simulation scheme enables speed-ups up to several orders of magnitudes.**

***Index Terms*—quantum computing, quantum circuit simulation, decision diagrams, approximation**

I. Introduction

Quantum computing promises significant speed-ups for solving many important computational problems. Oft-cited examples include Shor's algorithm [20] for integer factorization and Grover's search [12] for searching in unstructured databases. However, other areas such as chemistry, finance, and machine learning can benefit from quantum computing as well [10], [15], [18]. This advantage largely comes from the exploitation of quantum-mechanical effects such as *superposition* and *entanglement*, where an n-qubit state can represent the 2^n basis states at the same time and operations on one qubit can influence another qubit, respectively. Those advantages have motivated Google, IBM, Microsoft, and Intel to build quantum chips and develop design tools, while start-ups like Cambridge Quantum and Rigetti have also joined the race.

Current laboratory and commercial quantum computers are exceedingly expensive, so most of the design and validation work, e.g., simulation [2], [13], [23], compilation [3], [19], [29], and verification [7]–[9], [11], is performed on non-quantum hardware. Here, the power of quantum computing complicates design-tool development, since conventionally representing a quantum state requires an exponential amount of memory—often in the form of a vector with 2^n entries representing the corresponding *amplitudes* for each possible basis state. Moreover, the straightforward representation of quantum operations is even worse with a $2^n \times 2^n$-dimensional matrix. This problem can be mitigated in many cases by employing adaptive data structures, such as *Decision Diagrams* (DDs) [1], [17], [22], [24], [28], [30], but the worst-case complexity remains exponential in the number of qubits.

Another important quantum-mechanical effect leads to further differences compared to non-quantum computing: It is fundamentally impossible to observe the entire quantum state (i.e., the amplitudes for all basis states) without destroying the superposition and entanglement [16]. Instead, a measurement of a quantum state collapses the state into one of the possible basis states. Since such a measurement is probabilistic (and depends on the respective amplitudes for each basis state), the quantum computation as a whole is probabilistic as well, which complicates applications. And yet, our work shows how to improve design-tool efficiency using the probabilistic nature of quantum computation.

Small changes in the amplitudes of a quantum state lead to small changes in the probabilities of measurement outcomes. Accordingly, we can manipulate the quantum state to admit a more compact representation in a finely-controlled tradeoff with its accuracy. In other words, the probabilistic nature of quantum computation makes, to some degree, quantum computations resistant against such small manipulations. Depending on the simulated quantum algorithm and/or specific circuit, a low-accuracy approximation of the final state may still be suitable for non-quantum post-processing leading to the same results, perhaps, after several repeated attempts. For example, Shor's algorithm works reliably with a circuit fidelity of $50\,\%$ (as we show in Section VI), while the quantum-supremacy circuits from Google still provide meaningful results with circuit fidelity around $1\,\%$ [4], [14]. This is a truly novel feature of quantum computations compared to computations in the conventional domain which has hardly been exploited yet.

In this work, we propose to exploit error tolerance to speed up DD-based quantum circuit simulation on non-quantum hardware. We investigate applying multiple approximation rounds throughout the simulation process to attain a more compact decision diagram representing the quantum state and, thus, improve the simulation runtime. To this end, we present two dedicated methods to incorporate approximation into the simulation process. The first method is reminiscent of automatic *garbage collection* used with programming languages and triggers the approximation when the intermediate quantum state grows too large in terms of size of the decision diagram. The second method facilitates the simulation with arbitrary approximation as long as a certain minimum fidelity in the resulting quantum state is guaranteed. Empirical validation demonstrates speed-ups in quantum circuit simulation up to several orders of magnitude.

The remainder of this paper is structured as follows: Section II reviews the fundamentals of quantum computations and decision diagrams. In Section III, we describe key observations that motivate the proposed methods. In Sections IV and V, we propose two methods for approximations in DD-based quantum circuit simulation and provide a proof on the effect of multiple approximations on the overall fidelity, respectively. Section VI summarizes the empirical validation of the proposed methods. Finally, Section VII concludes the paper.

II. Background

To keep the paper self-contained, this section provides a brief overview of the basics of quantum computing and decision diagrams as a means to represent quantum states.

A. Quantum Computing

The basic unit of information in quantum computing is the *quantum bit* or *qubit* [16]. In the conventional realm, a bit can assume exactly one of the states 0 and 1. Qubits can additionally assume any linear combination of the *basis states* (denoted $|0\rangle$ and $|1\rangle$ in Dirac notation). More precisely, the state of a qubit is written as $|\psi\rangle = \alpha_0 \cdot |0\rangle + \alpha_1 \cdot |1\rangle$ with *amplitudes* $\alpha_0, \alpha_1 \in \mathbb{C}$. The squared magnitude $|\alpha_i|^2$ of an amplitude α_i defines the probability with which the corresponding basis state will be the result when measuring. Therefore, the amplitudes have to satisfy the normalization constraint $|\alpha_0|^2 + |\alpha_1|^2 = 1$. A qubit is said to be in *superposition*, when both α_0 and α_1 are non-zero. Intuitively, this means the qubits are in both states at the same time—one of the important characteristics of quantum computing. Another important characteristic is *entanglement*, where the measurement of a single qubit may influence the (future) measurement result of another qubit.

Due to the underlying physics, the exact values of the amplitudes (α_0 and α_1) are fundamentally unobservable in physical quantum computers. Instead, the only way to "look at" qubits is measuring (with the outcome probabilities dictated by the amplitudes). Measuring a qubit $|\psi\rangle = \alpha_0 \cdot |0\rangle + \alpha_1 \cdot |1\rangle$ will yield $|0\rangle$ ($|1\rangle$) with the probability $|\alpha_0|^2$ ($|\alpha_1|^2$). Further, the measurement destroys any superposition and entanglement—leaving the qubit in a basis state.

Quantum states consisting of n qubits are extended accordingly to have 2^n basis states and corresponding amplitudes. The normalizing constraint is generalized to $\sum_{i \in \{0,1\}^n} |\alpha_i|^2 = 1$. Further, quantum states are commonly described by vectors with the amplitudes as elements, e.g., a two-qubit state $|\psi\rangle$ may be denoted as $\begin{bmatrix} \alpha_{00} & \alpha_{01} & \alpha_{10} & \alpha_{11} \end{bmatrix}^{\mathrm{T}}$.

Example 1. *Consider a two-qubit quantum state $|\psi\rangle$, which is set to $1/\sqrt{2} \cdot |00\rangle + 0 \cdot |01\rangle + 0 \cdot |10\rangle + 1/\sqrt{2} \cdot |11\rangle$. This state is valid, since $|1/\sqrt{2}|^2 + |1/\sqrt{2}|^2 = 1$ satisfies the normalizing constraint. As a vector, the state is written as $|\psi\rangle = \begin{bmatrix} 1/\sqrt{2} & 0 & 0 & 1/\sqrt{2} \end{bmatrix}^{\mathrm{T}}$. Due to the superposition, measuring this state yields either of the two basis states $|00\rangle$ or $|11\rangle$ with a probability of $|1/\sqrt{2}|^2 = 1/2$ each. After the measurement, the superposition is destroyed and the quantum state is fixed to the measured state, i.e., subsequent measurements yield the same result.*

A quantum state is altered quantum operations. These operations are defined through unitary matrices, i.e., square matrices whose inverse is their conjugate transpose [16].

Example 2. *Commonly used single-qubit operations are X (negates the state of the qubit), H (sets the qubit into superposition), and Z (shifts the phase of the qubit).*

A key example for a two-qubit operation is the CNOT, which negates a target qubit iff the control qubit is in the state $|1\rangle$.

Given the vector- and matrix-based descriptions for states and operations, respectively, the effect of applying an operation to a state is defined through the matrix-vector multiplication.

Example 3. *Given a quantum state $|\psi\rangle$ with two qubits set to $|00\rangle$, first performing a Hadamard operation on the first qubit and, afterwards, a CNOT operation yields a new state $|\psi'\rangle$ defined by*

$$\underbrace{\begin{bmatrix} 1&0&0&0\\0&1&0&0\\0&0&0&1\\0&0&1&0 \end{bmatrix}}_{CNOT} \times \underbrace{\frac{1}{\sqrt{2}}\begin{bmatrix} 1&0&1&0\\0&1&0&1\\1&0&-1&0\\0&1&0&-1 \end{bmatrix}}_{\text{Hadamard on 1st qubit}} \times \underbrace{\begin{bmatrix} 1\\0\\0\\0 \end{bmatrix}}_{|\psi\rangle} = \underbrace{\frac{1}{\sqrt{2}}\begin{bmatrix} 1\\0\\0\\1 \end{bmatrix}}_{|\psi'\rangle}.$$

Possible outcomes in measuring the new state are $|00\rangle$ or $|11\rangle$—each with a probability of $50\,\%$ as in Example 1.

B. Decision Diagrams

Naïve representations of quantum states and operations rely on exponentially large vectors and matrices of size 2^n and $2^n \times 2^n$, respectively (with n denoting the number of qubits). *Decision diagrams* (DDs) are tried and tested alternative representations that exploit redundancies in order to reduce these complexities [1], [5], [17], [22], [24], [30]. In the task of simulation, utilizing decision diagrams instead of vectors/matrices increases the performance by several orders of magnitude for certain algorithms [30]. In the best case, this resulted in a runtime of under two minutes with decision diagrams compared to 30 days with matrix-vector multiplication.

Redundancies are exploited by using shared structures whenever possible. To this end, e.g., the vector is split into upper and lower sub-vectors. This splitting is repeated for the sub-vectors until the result is a single element. Since each split halves the size of the vector, there are n levels of splitting for an n-qubit state. During this process, identical sub-vectors are detected and represented by a single shared structure for the identical sub-vectors. The decision diagram is then normalized to ensure a canonical representation. To determine the value of an amplitude, the edge weights of the path representing said amplitude are multiplied. Analogously, this is done for matrices.

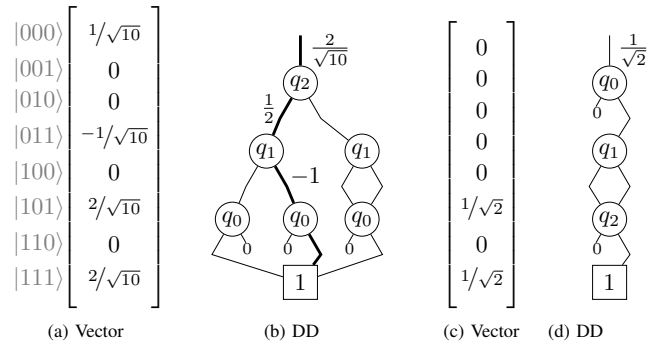

Fig. 1. Two quantum states with vector and DD representation, respectively

Example 4. *Consider the state vector in Fig. 1a. The annotations on the left denote the basis state each amplitude corresponds to. In Fig. 1b, a decision diagram representing the same state is depicted. To access the amplitude of basis state $|011\rangle$, the bolded path in the decision diagram has to be traversed and the edge weights along this path have to be multiplied, i.e., ($q_2 = 0$, $q_1 = 1$, $q_0 = 1$) yielding $2/\sqrt{10} \cdot 1/2 \cdot (-1) \cdot 1 = -1/\sqrt{10}$.*

III. Motivation

In this work, we investigate how to efficiently simulate quantum computations on non-quantum machines. Computational efficiency aside, this task is straightforward to solve and boils down to a series of matrix-vector multiplications as reviewed in Section II-A. However, this approach requires an exponential

amount of memory to store the corresponding vectors and matrices. Decision diagrams as reviewed in Section II-B may cope with that for many instances, but eventually have the same worst-case complexity. We propose to improve upon that state of the art by exploiting approximations of the respectively considered decision diagrams.

Physical quantum computers are inherently probabilistic in nature, i.e., they do not output an explicit vector of amplitudes but rather a bitstring representing a single basis state which is probabilistically selected during measurement. While the results of such measurements still depend on the respective amplitudes, $100\,\%$ accuracy in the amplitudes is not always necessary for that. For example, near-zero amplitudes imply near-zero probabilities and, thus, may be ignored in many quantum circuit applications without much impact on the final result. In this work, we are going to exploit this observation. To this end, we first require a metric to quantify proximity for quantum states.

The *fidelity metric* of two quantum states measures their proximity [16], [25]. It expresses the likelihood that measuring two quantum states results in the same outcome.

Definition 1 (Fidelity)**.** *For pure quantum states as used in this work, the* fidelity F *is defined as* $F(|\psi\rangle, |\phi\rangle) = |\langle\psi|\phi\rangle|^2 = |\,(\psi^*)^{\mathrm{T}} \cdot \phi|^2$ *for any quantum states* $|\psi\rangle$ *and* $|\phi\rangle$ *[25]. Further,* $0 \leq F(|\psi\rangle, |\phi\rangle) \leq 1$ *holds, where* $F(|\psi\rangle, |\phi\rangle) = 1$ *implies* $|\psi\rangle = |\phi\rangle$.

Example 5. *Given the two quantum states* $|\psi\rangle = 1/2[\,1\;1\;1\;1\,]^{\mathrm{T}}$ *and* $|\phi\rangle = 1/\sqrt{2}[\,1\;0\;0\;1\,]^{\mathrm{T}}$*, their fidelity is calculated as* $F(|\psi\rangle, |\phi\rangle) = |1/2 \cdot 1/\sqrt{2} + 1/2 \cdot 1/\sqrt{2}|^2 = 1/2$*. Intuitively, this states that the probability of measuring the same outcome from both quantum states is* $50\,\%$ *(which indeed is the case for these two quantum states).*

We use the fidelity metric to determine the accuracy of approximation, i.e., how far the original state is from its approximation obtained by zeroing out some of its small amplitudes. In this context, the fidelity metric has an important property: it is preserved under unitary transformations, i.e., the quantum operations we use in simulation. More precisely, given a quantum operation U and two quantum states $|\psi\rangle$ and $|\phi\rangle$, applying U to each state does not further decrease their fidelity, i.e., $F(U|\psi\rangle, U|\phi\rangle) = F(|\psi\rangle, |\phi\rangle)$ [25]. Therefore, small amplitudes can be zeroed out during the simulation while maintaining a reasonable end-to-end accuracy.

Furthermore, even though zeroing out small amplitudes a single time during simulation might be sufficient, multiple applications may produce better size-fidelity tradeoffs—in particular, when larger instances of quantum algorithms are considered. This raises the problem of how to calculate the fidelity between the exact state (where no amplitudes were ignored) and the modified state (where amplitudes were ignored multiple times during simulation).

Example 6. *Given the original state* $|\psi\rangle = 1/2[\,1\;1\;1\;1\,]^{\mathrm{T}}$*, we are generating two states with successively more amplitudes set to zero (as an exaggerated example). We obtain* $|\psi'\rangle = 1/\sqrt{2}[\,1\;0\;0\;1\,]^{\mathrm{T}}$ *in the first round and* $|\psi''\rangle = [\,0\;0\;0\;1\,]^{\mathrm{T}}$ *in the second. The pairwise fidelities are* $F(|\psi\rangle, |\psi'\rangle) = 1/2$*,* $F(|\psi'\rangle, |\psi''\rangle) = 1/2$*, and* $F(|\psi\rangle, |\psi''\rangle) = 1/4$.

The proposed approximation method of zeroing out small amplitudes and rescaling the vector constitutes a worst case for the fidelity metric. In Section V, we back up the intuition that successive applications lower the overall fidelity multiplicatively by a formal statement with a proof. This way we can repeatedly decrease the size of the representation of a quantum state during simulation while, at the same time, being able to precisely keep track of the resulting accuracy.

IV. Approximating DD-based Quantum Circuit Simulation

The discussions in Section III eventually provide the basis for a DD-based quantum circuit simulation approach which allows the user to simplify (to *approximate*) decision diagrams (and, by that, to accelerate the simulation process), while controlling the resulting accuracy. In this section, we describe a corresponding implementation of such an approach. To this end, we first review how approximation of decision diagrams can actually be employed. Afterwards, two strategies that incorporate the approximation into the simulation process are presented.

A. Constructively Approximating DDs

Before introducing our approximation techniques, we briefly review how decision diagrams are used in simulation. Performing quantum circuit simulations based on decision diagrams is conceptionally similar to the matrix-vector-based approach discussed in Section II-A. Decision diagrams can represent quantum operations and states just like matrices and vectors, respectively. Additionally, decision diagrams support the same operations, especially matix-vector multiplication, which are required for circuit simulation [17], [22], [30], [31]. Hence, on an abstract level, a decision diagram representing a basis state is constructed and, afterwards, the quantum operations (again, represented by decision diagrams) are successively applied to the quantum state.

A method to efficiently employ approximation as motivated in the previous section must be able to (1) remove nodes from the decision diagrams while, at the same time, (2) efficiently determine the effect of that removal on the accuracy (i.e., the fidelity). Moreover, a corresponding method should ideally guarantee that the accuracy/fidelity has a lower bound, which can be defined by the user. These objectives can be accomplished by numerically estimating the *norm contribution* of each node to the overall accuracy and selecting appropriate nodes for removal [27].

Definition 2 (Norm Contribution of a Node)**.** *The paths through a decision diagram from top to bottom encode the amplitudes in the edge weights. More precisely, the product of edge weights along a path yields the corresponding amplitude. The* norm contribution of a node *(or* contribution *for short) is the sum of squared magnitudes of amplitudes for each path passing through that node. From this, it follows that for each level* i *in the decision diagrams, the contributions of nodes* q_i *on this level add up to 1.*

Example 7. *Consider again the decision diagram shown in Fig. 1b. Since all paths of the decision diagram go through the root node labeled* q_2*, this node has a contribution of 1. The nodes labeled* q_1 *and* q_0 *on the right-hand side have a contribution of 0.8 each, because the squared magnitude of amplitudes for paths passing through is* $2 \cdot |2/\sqrt{10}|^2 = 0.8$*. The node labeled* q_1 *on left-hand side has a contribution of 0.2 and its two* q_0*-successors have a contribution of 0.1 each.*

Having calculated the contributions of each node, the effect of removing a node from the decision diagram on the resulting fidelity can be determined: since the removal of a node corresponds to setting the relevant amplitudes to zero, the fidelity decreases additively by the sum of squared magnitude of the zeroed amplitudes. Hence, the contribution of a node directly translates into the fidelity lost on removal. This facilitates a lower bound on the resulting fidelity by only removing nodes with a small enough contribution.

Example 8. *Consider again the decision diagram shown in Fig. 1b and the corresponding node contributions as determined above. Removing the root node labeled q_2 (and, by this, the entire decision diagram) would lead to a fidelity of 0—accordingly reflected by the node's contribution of 1. Instead, removing the left-hand side node labeled q_1 (which has a contribution of 0.2) results in the state shown in Fig. 1d as discussed before—yielding a much more compact representation while maintaining a fidelity of 0.8.*

The above technique can be used at any point during simulation to evaluate the effect of removing individual nodes in terms of implied accuracy loss. Based on that, we can now focus on specific strategies to ensure good accuracy-efficiency tradeoffs. These are described next.

B. Memory-driven Approximation

First, we propose a reactive strategy which focuses on efficiency, i.e., caters to the use case, where the size of the decision diagram (and, hence, the required memory requirements) should be kept low—even if this means risking an unsuitable loss in fidelity. To this end, we evaluate after each simulation step (i.e., after each application of a quantum operation to the current state) whether the size of the decision diagram (i.e., its number of nodes) exceeds a certain threshold (defined by the user). This is similar to typical garbage collection methods in which memory is freed once certain thresholds are exceeded (the difference here is that we do this at the expense of accuracy). Since the fidelity decreases with each application of such an *approximation round*, the threshold is dynamically increased as well to avoid that the number of applied approximations gets too large.

We distill the ideas proposed so far into the following procedure: in the simulation, the quantum operations are successively applied to the quantum state. After each quantum operation, the size of the decision diagram representing the state is compared to the user-defined threshold. If that threshold is exceeded, the quantum state is approximated targeting the *single-round fidelity* (which is also given by the user). Additionally, the threshold is doubled after each approximation round to avoid too many approximations.

Due to its simple structure, this strategy is appropriate as a reactive protection against out-of-memory conditions, especially when simulating a new circuit type and not knowing what to expect.

Example 9. *The quantum-supremacy circuits [6] are designed so that they posses little to no redundancy, which is an exceptionally challenging case for quantum state representation by decision diagrams. Therefore, the size of the decision diagram will increase rapidly to the point were it significantly slows down the simulation process. At that point, the approximation scheme proposed here kicks in and trades off some accuracy for a smaller representation and subsequently faster simulation. This process repeats when the decision diagram reaches the new threshold size.*

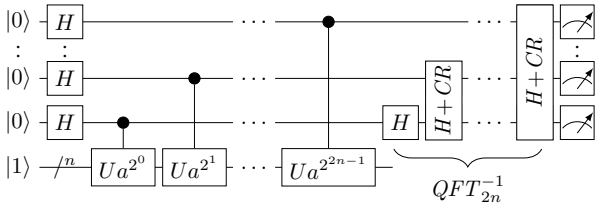

Fig. 2. Circuit blocks of Shor's algorithm

Underestimating the hyper-parameters for the threshold and single-round fidelity may render the simulation result meaningless if the final state fidelity is too low. Nonetheless, when carefully considering the algorithm at hand, memory-driven approximation may enable successful simulations previously blocked by insufficient memory.

C. Fidelity-driven Approximation

To complement the *reactive* strategy from Section IV-B, we now propose a *proactive* strategy based on accuracy. This strategy caters to the use case where the fidelity must not drop below a certain lower bound but, beyond that, the decision diagram can be approximated as much as possible. Indeed, in many applications, it is sufficient if the resulting quantum state is not exact but employs a certain lower-bound accuracy. For example, Shor's algorithm [20] frequently works with a circuit fidelity of around $50\,\%$ and still determines the factors of an integer correctly as we show in Section VI. Since the approximation method described in Section IV-A, i.e., the removal of nodes of a decision diagram, can guarantee a certain lower bound for fidelity, this can easily be realized.

Given a minimum required final fidelity f_{final} for the quantum state after the simulation, the number of times the approximation is applied (*approximation rounds*) is proactively calculated for the circuit. Due to the multiplicative property of the fidelity metric, the maximum number of approximation rounds is $\lfloor \log_{f_{\text{round}}}(f_{\text{final}}) \rfloor$, where f_{round} is the target fidelity for each approximation round. This requires choosing a tradeoff between (1) few approximation rounds with low single-round fidelity and (2) many approximation rounds with high single-round fidelity. The first choice keeps the overhead of the approximation rounds low, but in between the size of the decision diagram may grow too much. The second choice, on the other hand, is much more likely to limit the growth of the decision diagram while imposing the overhead of performing many approximation rounds. The optimal selection heavily depends on the quantum algorithm.

In addition to deciding the number of approximation rounds, the locations in the quantum circuit at which to approximate have to be determined. From a high-level perspective, promising candidates for such locations are between circuit blocks of the algorithm. When no such circuit blocks can be identified, e.g., after certain types of circuit optimization, the individual approximation rounds are evenly spaced out through the circuit.

Example 10. *Shor's algorithm [20] exhibits distinctive circuit blocks as illustrated in Fig. 2. It consists of Hadamard operations, a series of modular multiplications (Ua^x), and an inverse QFT, which itself is further split into Hadamard operations and controlled rotations (CR). Exploiting this knowledge facilitates approximation rounds after each modular multiplication and after the controlled rotations during the inverse QFT. Following the proposed strategy, the number of approximation rounds and*

their positions are determined before the simulation to lower-bound the fidelity of the quantum state after the simulation.

As with the reactive strategy, suitable hyper-parameter selection is paramount, because producing results with unnecessarily high fidelity may require greater computational resources.

V. Effect of Multiple Approximations

The effect of the applied approximations is crucial to the eventual accuracy and, hence, the approaches proposed above. In the motivation in Section III and the approaches proposed in Section IV, we established the contribution of nodes to directly calculate the effect on the fidelity if said nodes are removed. Moreover, we claimed that even applying several approximation rounds can be reflected by multiplying the correspondingly resulting fidelities. However, while this claim may seem intuitive, it lacks a rigorous underpinning thus far. We now provide theoretical basis for this claim.

The *fidelity metric* quantifies the proximity of quantum states. Recall Definition 1: Given two quantum states $|\psi\rangle, |\phi\rangle \in \mathbb{C}^D$: $F\left(|\psi\rangle, |\phi\rangle\right) = |\langle\psi|\phi\rangle|^2 \in [0,1]$ for $D = 2^n$-dimensional states. In the following, we show that the fidelity metric behaves as one might expect under successive approximations. More precisely, we consider a simple *truncation procedure*, which is a generalization of the proposed approximation scheme for decision diagrams. Fix a subset $I \subset \{0, \ldots, D-1\}$ of relevant coordinates and consider the following truncation:

$$|\psi_I\rangle = \frac{P_I|\psi\rangle}{\|P_I|\psi\rangle\|_{\ell_2}} \quad \text{where} \quad P_I = \sum_{i \in I} |i\rangle\langle i|. \tag{1}$$

This procedure zeros out every amplitude that does *not* belong to the set I and subsequently re-scales the vector to have unit length again. Hence, it has the same effect as eliminating nodes in a decision diagram (with subsequent re-scaling) if I is chosen appropriately. The following mathematical statement asserts that that fidelity behaves nicely under such truncation procedures.

Lemma 1. *Fix two quantum states $|\psi\rangle, |\phi\rangle$ and a truncation procedure* (1)*. Then,*

$$F(|\psi\rangle, |\phi_I\rangle) = F(|\psi\rangle, |\psi_I\rangle) \cdot F(|\psi_I\rangle, |\phi_I\rangle).$$

To spell it out, the fidelity between a target state $|\psi\rangle$ and an approximation of $|\phi\rangle$ factorizes into the fidelity of loss incurred by approximating the target times the fidelity between both approximations.

Proof. The statement readily follows from combining the following properties of a truncation procedure (1):

$$P_I|\psi_I\rangle = |\psi_I\rangle \quad \text{and} \quad \|P_I|\psi\rangle\|_{\ell_2} = \sqrt{F\left(|\psi\rangle, |\psi_I\rangle\right)}.$$

The first equation states that projecting twice does not change the state, while the second relation expresses the scaling parameter in terms of fidelity. Combining both yields

$$\begin{aligned} F\left(|\psi\rangle, |\phi_I\rangle\right) &= |\langle\psi|P_I|\phi_I\rangle|^2 \\ &= |\sqrt{F(|\psi\rangle, |\psi_I\rangle)} \cdot \langle\psi_I|\phi_I\rangle|^2 \\ &= F(|\psi\rangle, |\psi_I\rangle) \cdot F(|\psi_I\rangle, |\phi_I\rangle). \end{aligned}$$

□

This observation addresses the (possibly) multiple approximation rounds in our simulation approach with a starting state $|\chi\rangle$ and unitaries U_i:

$$\begin{aligned} |o\rangle &= U_3U_2U_1|\chi\rangle && \text{(true target)}, \\ |o'\rangle &= U_3|(U_2U_1\chi)_I\rangle && \text{(one approximation)}, \\ |o''\rangle &= U_3|(U_2(U_1\chi)_J)_I\rangle && \text{(two approximations)}. \end{aligned}$$

Unitary invariance of the fidelity function will allow us to ignore U_3. Set

$$\begin{aligned} |\psi\rangle &= U_3^\dagger|o\rangle = U_2U_1|\chi\rangle, & |\psi_I\rangle &= |(U_2U_1\chi)_I\rangle = U_3^\dagger|o'\rangle, \\ |\phi\rangle &= U_2|(U_1\chi)_J\rangle, & |\phi_I\rangle &= |(U_2(U_1\chi)_J)_I\rangle = U_3^\dagger|o''\rangle \end{aligned}$$

and observe

$$\begin{aligned} F(|o\rangle, |o''\rangle) &= F(U_3|\psi\rangle, U_3|\phi_I\rangle) = F(|\psi\rangle, |\phi_I\rangle) \\ &= F(|\psi\rangle, |\psi_I\rangle) \cdot F(|\psi_I\rangle, |\phi_I\rangle) \\ &= F(U_3^\dagger|o\rangle, U_3^\dagger|o'\rangle) \cdot F(U_3^\dagger|o'\rangle, U_3^\dagger|o''\rangle) \\ &= F(|o\rangle, |o'\rangle) \cdot F(|o'\rangle, |o''\rangle). \end{aligned}$$

Hence, we have shown that the overall fidelity after multiple approximation rounds can be calculated by simply multiplying the fidelities between the individual rounds.

VI. Empirical Validation

To validate the impact of approximation in DD-based quantum circuit simulation, we implemented both methods proposed in Section IV on top the simulator provided at https://iic.jku.at/eda/research/quantum_simulation [28], [30] as part of the JKQ toolset [26]. We used the simulation without approximation as a reference to quantify the effects of approximations with respect to maximum DD size and runtime when approximately simulating the exact circuits. The simulations were performed on a server running GNU/Linux with a 4.2 GHz (8 cores) CPU and 32 GiB main memory. GCC 7.5.0 served as compiler and GNU parallel [21] was used to control experiment execution.

Table I shows the validation results for the memory- as well as the fidelity-driven approximate simulation method. The results show that suitably chosen hyper-parameters enable significant improvements for quantum-supremacy circuits from Google and speed-ups of several orders of magnitudes for Shor's algorithm.

The memory-driven approach was validated using the quantum-supremacy circuits from Google using conditional phase-gates [6] (denoted "qsup_AxB_C" with $A \times B$ being the grid size and C being the depth). Since these circuits are designed to be hard to simulate (exactly and approximately) and have little to no redundancy, they are exceptionally challenging for decision diagrams. Nonetheless, the approximation scheme with a sensible threshold halves the runtime in the best case, while maintaining a fidelity above 10 %, which (depending on the research context) is still acceptable and, in fact, better than the results from a physical quantum computer [4], [14]. However, the validation also highlights that the parameters have to be carefully selected or there is risk of performance degradation.

The fidelity-driven approach is ideal if the user has knowledge of the required accuracy as it enables to guarantee a lower bound. In the validation, we focused on Shor's algorithm [31] (denoted "shor_A_B" with the number to factorize A and coprime B). Using the proposed approximate simulation, we were able to simulate Shor's algorithm with a speed-up of several orders of magnitude while setting 50 % as minimum fidelity. Notably, the benchmark shor_1157_8 (33 qubits) timed out in three hours, while the approximate simulation completed in just under two minutes. In the experiment, we exploited the knowledge that

TABLE I
RESULTS OF THE EMPIRICAL VALIDATION

Approach	Benchmark		Non-Approximating		Proposed Approach				
	Name	Qubits	Max. DD Size	Runtime [s]	Max. DD Size	Rounds	f_{round}	Runtime [s]	f_{final}
Memory-driven	qsup_4x5_15_0	20	2 097 150	3 666.87	1 963 906	91	0.99	5 512.16	0.401
					1 810 948	90	0.975	3 340.89	0.102
					1 471 425	89	0.95	1 341.53	0.010
	qsup_4x5_15_1	20	2 097 150	2 024.83	1 417 398	84	0.99	1 066.64	0.430
					932 915	84	0.975	697.40	0.119
					799 830	49	0.95	361.01	0.081
	qsup_4x5_15_2	20	2 097 150	2 090.09	1 963 347	81	0.99	3 208.59	0.443
					1 823 513	83	0.975	2 349.31	0.122
					1 562 367	84	0.95	1 227.10	0.013
Fidelity-driven (target 50 %)	shor_33_5	18	73 736	0.50	8 135	6	0.9	0.33	0.567
	shor_55_2	18	131 254	0.57	5 637	6	0.9	0.20	0.559
	shor_69_2	21	523 410	8.50	52 726	4	0.9	1.87	0.661
	shor_221_4	24	1 472 942	12.56	7 647	5	0.9	0.19	0.616
	shor_323_8	27	11 829 160	807.52	13 706	6	0.9	0.79	0.571
	shor_629_8	30	–	*Timeout*	57 710	5	0.9	2.07	0.596
	shor_1157_8	33	–	*Timeout*	535 001	5	0.9	117.19	0.610

The runtime *Timeout* indicates the experiment was terminated after 3 h.

the inverse *Quantum Fourier Transformation* (QFT) at the end of the algorithm required by far the most time to simulate and, hence, applied the approximation rounds during the inverse QFT. While 50 % fidelity seems low, we were able to correctly factorize the numbers given in the benchmarks by performing the non-quantum postprocessing steps of Shor's algorithm.

Our results clearly demonstrate the potential of DD-based simulation with approximation: it enables users to improve runtime performance by up to several orders of magnitude, while keeping the decrease in the accuracy of the resulting quantum state at a level which still delivers meaningful results.

VII. CONCLUSIONS

Quantum circuit simulation is an important pillar of design automation for quantum computing. In this work, we proposed two new methods for simulation that facilitate an increased performance in a tradeoff with the accuracy of the resulting state. More precisely, we have investigated quantum circuit simulation with multiple approximation rounds to reduce the size of the decision diagram representing the quantum state. The first method focuses on efficiency and reactively sacrifices accuracy for a more compact representation. In the second method, given a minimal required accuracy, the approximation rounds are proactively configured to guarantee the required accuracy. We further provided an analytical proof for the intuition that the end-to-end fidelity after multiple approximation rounds is the product of the fidelity for each round. The empirical validation of both methods showed significant increases in performance up to orders of magnitude in the best-case scenario, while still producing a result with suitable accuracy.

ACKNOWLEDGMENTS

This work has partially been supported by the LIT Secure and Correct Systems Lab funded by the State of Upper Austria as well as by BMK, BMDW, and the State of Upper Austria in the frame of the COMET Programme managed by FFG.

REFERENCES

[1] A. Abdollahi and M. Pedram. Analysis and synthesis of quantum circuits by using quantum decision diagrams. In *Design, Automation and Test in Europe*, pages 317–322, 2006.

[2] H. Abraham et al. Qiskit: An open-source framework for quantum computing, 2019.

[3] M. Amy, D. Maslov, M. Mosca, and M. Roetteler. A meet-in-the-middle algorithm for fast synthesis of depth-optimal quantum circuits. *IEEE Trans. on CAD of Integrated Circuits and Systems*, 32(6):818–830, 2013.

[4] 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(7779):505–510, 2019.

[5] R. I. Bahar, E. A. Frohm, C. M. Gaona, G. D. Hachtel, E. Macii, A. Pardo, and F. Somenzi. Algebraic decision diagrams and their applications. In *Int'l Conf. on CAD*, pages 188–191, 1993.

[6] 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. *Nature Physics*, 14(6):595, 2018.

[7] Z. Brakerski, P. Christiano, U. Mahadev, U. Vazirani, and T. Vidick. A cryptographic test of quantumness and certifiable randomness from a single quantum device. In *Foundations of Computer Science*, pages 320–331, 2018.

[8] L. Burgholzer, R. Raymond, and R. Wille. Verifying results of the IBM Qiskit quantum circuit compilation flow. In *Int'l Conf. on Quantum Computing and Engineering*, 2020.

[9] L. Burgholzer and R. Wille. Advanced equivalence checking of quantum circuits. *IEEE Trans. on CAD of Integrated Circuits and Systems*, 2020.

[10] P. J. Coles, S. Eidenbenz, S. Pakin, A. Adedoyin, J. Ambrosiano, P. Anisimov, W. Casper, G. Chennupati, C. Coffrin, H. Djidjev, et al. Quantum algorithm implementations for beginners. *arXiv:1804.03719*, 2018.

[11] A. Gheorghiu, T. Kapourniotis, and E. Kashefi. Verification of quantum computation: An overview of existing approaches. *Theory of Computing Systems*, 63(4):715–808, 2019.

[12] L. K. Grover. A fast quantum mechanical algorithm for database search. In *Theory of computing*, pages 212–219, 1996.

[13] T. Jones, A. Brown, I. Bush, and S. C. Benjamin. QuEST and high performance simulation of quantum computers. *Scientific reports*, 9(1):1–11, 2019.

[14] I. L. Markov, A. Fatima, S. V. Isakov, and S. Boixo. Massively parallel approximate simulation of quantum circuits. In *Design Automation Conf.*, 2020.

[15] A. Montanaro. Quantum algorithms: An overview. *npj Quantum Information*, 2:15023, 2016.

[16] M. Nielsen and I. Chuang. *Quantum Computation and Quantum Information*. Cambridge Univ. Press, 2000.

[17] P. Niemann, R. Wille, D. M. Miller, M. A. Thornton, and R. Drechsler. QMDDs: Efficient quantum function representation and manipulation. *IEEE Trans. on CAD of Integrated Circuits and Systems*, 35(1):86–99, 2016.

[18] J. Preskill. Quantum computing in the NISQ era and beyond. *Quantum*, 2:79, 2018.

[19] E. A. Sete, W. J. Zeng, and C. T. Rigetti. A functional architecture for scalable quantum computing. In *Int'l. Conf. on Rebooting Computing*, 2016.

[20] P. W. Shor. Algorithms for quantum computation: Discrete logarithms and factoring. *Foundations of Computer Science*, pages 124–134, 1994.

[21] O. Tange. Gnu parallel - the command-line power tool. *;login: The USENIX Magazine*, 36(1):42–47, 2011.

[22] G. F. Viamontes, I. L. Markov, and J. P. Hayes. *Quantum Circuit Simulation*. Springer, 2009.

[23] B. Villalonga, S. Boixo, B. Nelson, C. Henze, E. Rieffel, R. Biswas, and S. Mandrà. A flexible high-performance simulator for verifying and benchmarking quantum circuits implemented on real hardware. *npj Quantum Information*, 5(1):1–16, 2019.

[24] S.-A. Wang, C.-Y. Lu, I.-M. Tsai, and S.-Y. Kuo. An XQDD-based verification method for quantum circuits. *IEICE Trans. Fundamentals*, 91-A(2):584–594, 2008.

[25] J. Watrous. *The Theory of Quantum Information*. Cambridge University Press, 2018.

[26] R. Wille, S. Hillmich, and L. Burgholzer. JKQ: JKU tools for quantum computing. In *Int'l Conf. on CAD*, 2020.

[27] A. Zulehner, S. Hillmich, I. L. Markov, and R. Wille. Approximation of quantum states using decision diagrams. In *Asia and South Pacific Design Automation Conf.*, pages 121–126, 2020.

[28] A. Zulehner, S. Hillmich, and R. Wille. How to efficiently handle complex values? Implementing decision diagrams for quantum computing. In *Int'l Conf. on CAD*, 2019.

[29] A. Zulehner, A. Paler, and R. Wille. An efficient methodology for mapping quantum circuits to the IBM QX architectures. 38(7):1226–1236, 2019.

[30] A. Zulehner and R. Wille. Advanced simulation of quantum computations. *IEEE Trans. on CAD of Integrated Circuits and Systems*, 38(5):848–859, 2019.

[31] A. Zulehner and R. Wille. Matrix-vector vs. matrix-matrix multiplication: Potential in DD-based simulation of quantum computations. In *Design, Automation and Test in Europe*, pages 90–95, 2019.