

# Approximation of Quantum States Using Decision Diagrams

Alwin Zulehner<sup>1</sup>, Stefan Hillmich<sup>1</sup>, Igor L. Markov<sup>2</sup>, and Robert Wille<sup>1</sup>

<sup>1</sup>Institute for Integrated Circuits, Johannes Kepler University Linz, Austria

<sup>2</sup>Department of EECS, University of Michigan, USA

alwin.zulehner@jku.at, stefan.hillmich@jku.at, imarkov@eecs.umich.edu, robert.wille@jku.at

<http://iic.jku.at/eda/research/quantum/>

**Abstract**—The computational power of quantum computers poses major challenges to new design tools since representing pure quantum states typically requires exponentially large memory. As shown previously, decision diagrams can reduce these memory requirements by exploiting redundancies. In this work, we demonstrate further reductions by allowing for small inaccuracies in the quantum state representation. Such inaccuracies are legitimate since quantum computers themselves experience gate and measurement errors and since quantum algorithms are somewhat resistant to errors (even without error correction). We develop four dedicated schemes that exploit these observations and effectively approximate quantum states represented by decision diagrams. We empirically show that the proposed schemes reduce the size of decision diagrams by up to several orders of magnitude while controlling the fidelity of approximate quantum state representations.

## I. INTRODUCTION

Quantum computing [1] enables exponential speed-ups compared to conventional computers by exploiting quantum mechanical effects such as *superposition*, where qubits can assume a linear combination of their basis states  $|0\rangle$  and  $|1\rangle$ , and *entanglement*, where qubits may be correlated. As a result, the pure state of a quantum system composed of  $n$  qubits may represent a superposition of  $2^n$  basis states and corresponding complex amplitudes—promising a greater amount of information per qubit and greater computational power. First algorithms following this powerful computing paradigm have been developed in the last century such as Shor’s algorithm for integer factorization [2] and Grover’s algorithm for searching in unsorted databases [3]. Recently, the development of new quantum algorithms targeting various problems in physical simulation, chemistry, finance, and machine learning [4]–[6] has gained momentum since significant progress in the physical realization of quantum computers has been achieved by players such as IBM, Google, Microsoft, Intel, Rigetti [7].

The development of quantum algorithms and their successful implementation on quantum computers relies on several tasks (such as simulation, state preparation, and verification) that have to be completed on conventional computers. Here, the enormous computational power of quantum computers becomes a liability since the amplitudes describing the *wave function* of the quantum system are often represented by means of a  $2^n$ -dimensional state vector—implying exponential memory requirements.

In the past, several data structures have been developed to overcome this exponential memory complexity of representing quantum states—including *Matrix Product States* (MPS [8]), *Projected Entangled Pair States* (PEPS [8]), and even neural networks [9]. Besides that, *Decision Diagrams* (DDs [10]–[13]) have been intensely

investigated and, in the meantime, have shown great potential for the design tasks outlined above [12]–[18]—especially for quantum-circuit simulation [19]–[21]. This is in line with the successes of conventional decision diagrams such as BDDs [22], KFDDs [23], etc. which got established in numerous design tasks for conventional circuits.

Even with specialized data structures, the memory requirements for the respective representations remain a significant challenge. Although decision diagrams allow for a rather compact representation in many cases, we note that approximations of quantum states have not yet been used to reduce memory usage. To this end, quantum algorithms are resistant to errors to some extent (even without error correction) and quantum computers experience gate as well as measurement errors. Hence, allowing for small inaccuracies in the state representation does not (or hardly) affect the result of these probabilistic measurements. Moreover, current quantum computers are subject to noise [5], [24] and, thus, do not represent desired quantum states exactly anyway. These considerations suggest possibilities for memory compression which has not been investigated thus far.

In this work, we investigate the potential of approximation to obtain more compact DD-based representations of quantum states. The key idea is to eliminate nodes from the decision diagram that represent basis states with minuscule contributions to the overall state vector—setting small amplitudes to zero. More precisely, we present several dedicated approximation schemes with linear time complexity in the size of the decision diagram, which can reduce its size significantly. Tuning hyper-parameters of these schemes ensures a certain fidelity when determining approximations. Empirical evaluations confirm this behavior and show that the size of the decision diagrams can be reduced by several orders of magnitude while also controlling the fidelity of the resulting approximated state representations. Given our results, these schemes can be automatically invoked during tasks such as simulation to reduce the size of the decision diagram when it grows too large, similar to garbage collection some programming languages.

The remainder of this paper is structured as follows. Section II reviews the background on quantum states and decision diagrams. Afterwards, Section III describes the main idea of state approximation, whereas Section IV discusses four dedicated schemes to perform the approximations. Section V summarizes the evaluation of the presented schemes while Section VI concludes the paper.

## II. BACKGROUND

This section provides the background to keep the exposition self-contained. It reviews quantum states as well as decision diagrams that can represent them in conventional software.

## A. Quantum States

In the realm of quantum computing, conventional bits are generalized to *quantum bits* (i.e., *qubits*). While the former can only be in exactly one of the states 0 and 1, qubits may assume two basis states (denoted  $|0\rangle$  and  $|1\rangle$ ) as well, but also any linear combination of them. This is described by  $|\psi\rangle = \alpha_0 \cdot |0\rangle + \alpha_1 \cdot |1\rangle$  with *amplitudes*  $\alpha_0, \alpha_1 \in \mathbb{C}$  and the normalization constraint  $|\alpha_0|^2 + |\alpha_1|^2 = 1$ . Qubits with  $\alpha_0$  and  $\alpha_1$  unequal to zero are referred to as being in *superposition*, i.e., they are in “both states” at the same time—one of the main characteristics of quantum computing that enables substantial speed-ups in certain applications due to the resulting potential for considering multiple states at the same time<sup>1</sup>.

Unfortunately, the respective amplitudes ( $\alpha_0$  and  $\alpha_1$ ) cannot be directly observed on a quantum computer. Instead, they dictate the probability of certain outcomes of a measurement with respect to the corresponding basis states. More precisely, measuring a single qubit in state  $|\psi\rangle = \alpha_0 \cdot |0\rangle + \alpha_1 \cdot |1\rangle$  yields the output  $|0\rangle$  with probability  $|\alpha_0|^2$  or the output  $|1\rangle$  with probability  $|\alpha_1|^2$ . After the measurement, the qubit will lose any superposition, i.e., it collapses into a basis state. For quantum systems composed of more than one qubit, the description is accordingly extended. For example, a system with two qubits has four basis states, i.e.,  $|\psi\rangle = \alpha_{00} \cdot |00\rangle + \alpha_{01} \cdot |01\rangle + \alpha_{10} \cdot |10\rangle + \alpha_{11} \cdot |11\rangle$  with the normalization constraint  $|\alpha_{00}|^2 + |\alpha_{01}|^2 + |\alpha_{10}|^2 + |\alpha_{11}|^2 = 1$ . Commonly, the description of a quantum state is shortened to a vector containing the amplitudes  $|\psi\rangle = [\alpha_{00}, \alpha_{01}, \alpha_{10}, \alpha_{11}]^T$ .

**Example 1.** Consider an arbitrary quantum system with two qubits, which are entangled and in superposition. The state is given as  $|\psi\rangle = \frac{1}{\sqrt{2}} \cdot |00\rangle + 0 \cdot |01\rangle + 0 \cdot |10\rangle + \frac{1}{\sqrt{2}} \cdot |11\rangle$ . This state is valid as the normalization constraint  $(\frac{1}{\sqrt{2}})^2 + 0^2 + 0^2 + (\frac{1}{\sqrt{2}})^2 = 1$  is satisfied. The quantum state is represented as a vector  $\psi = [\frac{1}{\sqrt{2}}, 0, 0, \frac{1}{\sqrt{2}}]^T$ .

Performing a measurement on this system yields one of the two basis states  $|00\rangle$  or  $|11\rangle$ , each with a probability of  $|\frac{1}{\sqrt{2}}|^2 = 1/2$ . After the measurement, the quantum state collapses, i.e., the state vector is either  $\psi = [1, 0, 0, 0]^T$  or  $\psi = [0, 0, 0, 1]^T$ , depending on the measured state.

## B. Decision Diagrams

Decision diagrams have been successfully utilized to drastically reduce the memory requirements for representing state vectors in quantum computing [10]–[13]. For example, simulation approaches based on decision diagrams have recently moved into the spotlight since they significantly outperform array-based simulators in cases where redundancies can be exploited (in extreme cases leading to an improvement in runtime from 30 days to 2 minutes [20]).

In the context of quantum state representation, decision diagrams identify redundancies in the state vector and provide compaction by sharing structures. The vector is split into two equal-sized sub-vectors. This process is repeated until the sub-vector contains a single element only, i.e., one split for every qubit. If identical sub-vectors occur in the process, this redundancy is exploited by re-using (sharing) the same structure in the resulting decision diagram.

More precisely, consider an arbitrary quantum system with  $n$  qubits which are labeled  $q_0, q_1, \dots, q_{n-1}$ , where  $q_0$  represents the most significant qubit of the quantum state. The state of  $q_0$  splits the state vector in half: the first  $2^{n-1}$  entries correspond to  $q_0 = |0\rangle$ , while the other entries correspond to  $q_0 = |1\rangle$ . The decision diagram

<sup>1</sup>Another important characteristic is *entanglement*, where the state of a single qubit may influence the state of another qubit.

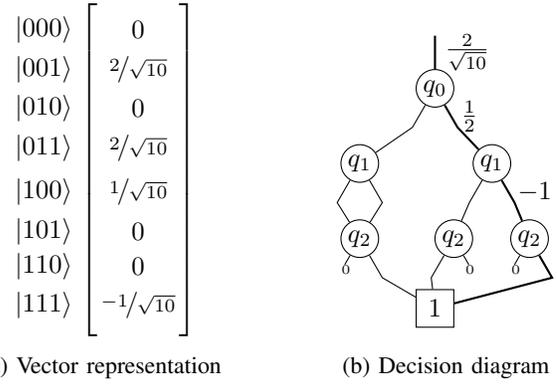


Fig. 1: Given quantum state

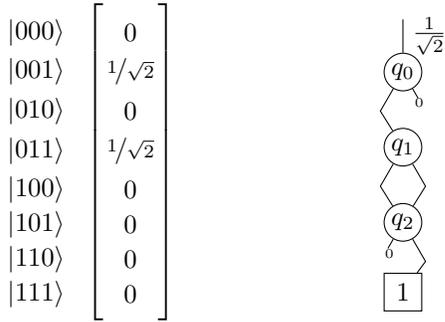
structure represents this decomposition through the node labeled  $q_0$  with two successor nodes that represent the aforementioned sub-vectors. Commonly, the left node denotes the 0-successor where as the right node denotes the 1-successor. This decomposition is recursively applied until the remaining sub-vectors have size 1, i.e., they represent a single complex number. In the process, two equivalent sub-vectors are represented by the same node, which reduces the memory requirements by exploiting redundancies. Nodes representing  $q_{n-1}$  have a single terminal for their left and right successors. This is achieved by storing common factors in the edge weights. To determine the value of an amplitude, the edge weights of the path representing said amplitude are multiplied.

**Example 2.** Consider the state vector in Fig. 1a. The corresponding basis state to each entry of the vector is written to the left. The decision diagram representing the same state as the vector is depicted in Fig. 1b. The amplitude of any state is accessed by following the path in the decision diagram and multiplying the corresponding edge weights. For the state  $|111\rangle$ , the path is bolded ( $q_0 = 1$ ,  $q_1 = 1$ ,  $q_2 = 1$ ) and the amplitude is  $\frac{2}{\sqrt{10}} \cdot \frac{1}{2} \cdot (-1) \cdot 1 = -\frac{1}{\sqrt{10}}$ .

## III. GENERAL IDEA

This section describes and illustrates the general idea of the proposed approximation approach. Naturally, designers are interested in an as compact as possible representation of the quantum states generated, e.g., by a quantum algorithm—a challenging task given that straightforward representations of states are exponential in size. As reviewed in the previous section and as already utilized in the conventional domain for decades, decision diagrams provide a well suited means for this purpose (by reducing the complexity through sharing of redundancies). In contrast to their conventional counterparts, quantum computations provide another useful property that allows for more compact representations by means of decision diagrams: Quantum algorithms are resistant to errors to some degree. Additionally, quantum computers suffer from gate as well as measurement errors anyway.

At each run, a quantum computer generates a bitstring (representing the measured basis state) which is affected by the aforementioned errors. More precisely, while the respective amplitude  $\alpha_i$  of a basis state  $|i\rangle$  dictates the probability of the measurement, small changes in the amplitude introduced by some error or approximation scheme do not (or hardly) affect the measurement due to the error resistance of the quantum computation. Because of that, perfect precision of a state representation is not necessary in most cases anyway. Besides that, quantum computing itself suffers from noise which additionally renders a precise representation unnecessary in many cases.



(a) Vector representation (b) Decision diagram

Fig. 2: Approximated quantum state

Exploiting the need of quantum computing to operate in noisy environments with gate and measurement errors offers further potential for more compact decision diagrams: Rather than aim to precisely represent the quantum state with amplitudes  $\alpha_i$  for all possible basis states  $|i\rangle$ , the quantum state is *approximated*. The DD-based representation can be approximated in two ways to become more compact. First, *similar* nodes can be merged to a single node. This is implicitly implemented by considering complex numbers with a difference below a given tolerance value as equal. Second, basis states which do not contribute significantly to the quantum state (i.e., states whose amplitudes are close to zero) are eliminated. Removing these nodes from the decision diagram sets the corresponding amplitudes to zero. Since quantum states are represented by unit-norm complex vectors, the remaining amplitudes are scaled (dividing by the magnitude of the approximated vector) such that the normalization constraint  $\sum_{i=0}^{2^n-1} |\alpha_i|^2 = 1$  remains satisfied.

**Example 3.** Assume that basis states  $|i\rangle$  with  $|\alpha_i|^2 < 1/4$  are indistinguishable from noise. Then the quantum state provided in Fig. 1a is approximated by setting the amplitudes  $\alpha_{100}$  and  $\alpha_{111}$  to 0. To normalize the state, we scale the remaining amplitudes by dividing them by  $\sqrt{|2/\sqrt{10}|^2 + |2/\sqrt{10}|^2} = 2/\sqrt{5}$ —leading to the state shown in Fig. 2a.

The decision diagrams representing the original and the approximated quantum state are shown in Fig. 1b and Fig. 2b, respectively. The original paths following the right successor of the  $q_0$  node each have a probability of  $|^{\pm 1}/\sqrt{10}|^2 = 1/10$  and, therefore, are removed. This reduces the size of the decision diagram from six nodes to three, yielding an approximation that constitutes an acceptable trade off for certain applications, where probabilities below a certain threshold are useless anyway.

Approximating quantum states requires a metric to quantify how much the resulting state deviates from the original one—describing the effect of the approximation and whether the obtained results are useful. Here, we use the *fidelity metric* [1], which describes the similarity between two quantum states. The fidelity of two pure quantum states  $|\varphi\rangle$  and  $|\psi\rangle$  is defined by

$$F(|\varphi\rangle, |\psi\rangle) = |\langle\varphi|\psi\rangle|^2 = |(\varphi^*)^T \cdot \psi|^2. \quad (1)$$

The fidelity metric describes the likelihood that both quantum states produce equal outcomes after they are measured. Further, this metric is preserved when applying unitary operations. The fidelity metric can be efficiently computed in terms of decision diagrams, i.e., in a linear fashion with respect to the number of DD nodes.

**Example 3** (continued). Consider again the decision diagrams shown in Fig. 1b and Fig. 2b (representing the original and the approximated quantum state  $|\varphi\rangle$  and  $|\psi\rangle$ , respectively). The fidelity according to Eq. (1) of these two quantum states is computed as

$$F(|\varphi\rangle, |\psi\rangle) = |(\varphi^*)^T \cdot \psi|^2 = \left| \frac{2}{\sqrt{10}} \cdot \frac{1}{\sqrt{2}} + \frac{2}{\sqrt{10}} \cdot \frac{1}{\sqrt{2}} \right|^2 = \frac{4}{5}.$$

That is, in 80% of the cases, a measurement of both states yield the same result. Hence, the size of the decision diagram is reduced by 50%, while the fidelity was only reduced by 20%.

The above observations illustrate the potential of approximating quantum states to obtain more compact DD-based representations and motivate to further explore this direction. To this end, the next section proposes several approximation schemes that explicitly trade off state fidelity for memory reduction.

## IV. APPROXIMATION SCHEMES

This section introduces four schemes to approximate a quantum state based on the ideas described in Section III—two based on a repeated traversal and two based on considerations regarding the impact of the sub-vectors represented in the decision diagram. These schemes aim to identify DD-nodes that do not significantly contribute to the overall state vector, eliminate them (i.e., setting certain small amplitudes to zero), and, by this, yield more compact representations.

### A. Approximation based on Traversal

The general idea of *approximation based on traversal* is to sample  $L$  bitstrings according to the probability distribution described by the state vector. A bitstring is sampled by following a path from the root node of the decision diagram to the terminal. Taking the left (right) successor of a node results in basis state  $|0\rangle$  ( $|1\rangle$ ) for the respective qubit. To sample according to the probability distribution given by the quantum state, the *upstream probabilities* are determined for each node. The upstream probability of a node (determining the summed probability of all paths from the node to the terminal) is computed in a depth-first fashion as sum of the upstream probabilities of the successor nodes—weighted by the squared magnitude of the weight attached to the connecting edge. Then, the probability for choosing the right (left) successor is given by the upstream probability of these nodes (again weighted by the attached weight of the connecting edge). Since this is similar to measuring a quantum state (without collapsing it to a basis state), we refer to [20] for further details.

Nodes that are not traversed during this sampling procedure hardly contribute to the overall quantum state, and are hence eliminated (i.e., replaced by zero-stubs) while keeping the fidelity of the approximated state close to one (assuming a sufficiently large number of samples  $L$  is chosen). This results in an approximation scheme that requires  $\mathcal{O}(L \cdot n)$  time since each traversal visits the DD nodes from the root node to the terminal. Calculating the upstream probabilities requires to visit each node, and has to be done only once before starting to draw samples.

**Example 4.** Consider again Fig. 1b. The left (right) successor of node  $q_0$  has a probability of 0.8 (0.2) to be chosen during traversals. Assuming the number of traversals  $L = 3$ , a possible distribution of traversed paths contains basis states  $|001\rangle$  (twice) and  $|011\rangle$  (once). Therefore the nodes visited on these paths are kept, whereas the right-successor node  $q_1$  and in turn its successors labeled  $q_2$  are eliminated from the decision diagram. The traversed nodes are sketched in Fig. 3a (the eliminated nodes are shown with a dashed outline). Eliminating the remaining node results in the approximated

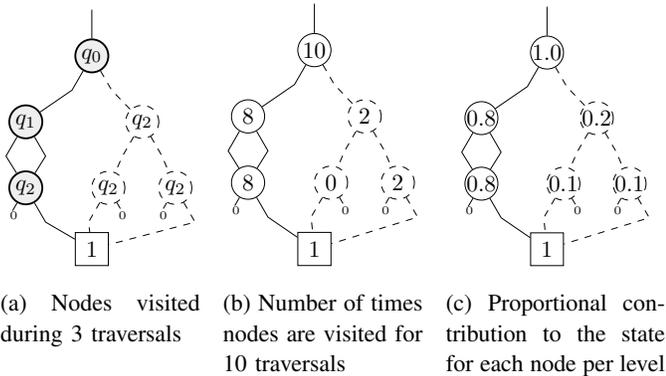


Fig. 3: Illustration of approximation schemes with nodes to be eliminated shown with a dashed outline

decision diagram shown in Fig. 2b and a fidelity of 80% with respect to the original quantum state.

### B. Approximation based on Traversal with Threshold

The traversal scheme above can be generalized by counting how many times a node has been visited. Additionally, we introduce a threshold  $\tau$  and only keep nodes that are visited more than  $\tau$  times, so that nodes visited  $\tau$  times or less are eliminated. The idea of the threshold is to filter “outliers” that do not contribute significantly to the quantum state but are nonetheless encountered by chance during the traversals.

**Example 5.** Consider again Fig. 1b and Example 4. Increasing the number of traversals to  $L = 10$  may lead to the possible distribution of following the left (right) successor of  $q_0$  8 times (2 times) as illustrated in Fig. 3b. This distribution is closer to the actual probabilities but also includes the right-hand nodes  $q_1$  and  $q_2$ . Introducing a threshold on how many times a node has to be visited during the traversals serves as a cut-off point for nodes with a lower probabilities. Selecting a threshold of  $\tau = 3$  leads to the elimination of the dashed nodes and, hence, to the approximation depicted in Fig. 2b.

### C. Approximation with a Target Fidelity

As discussed in the previous section, it is desirable to approximate quantum states with the fidelity desired in a given application context. The traversal-based approximation schemes so far do not provide this feature. Hence, we present further schemes that guarantee the minimum fidelity of the approximated state.

To this end, the entire decision diagram is traversed twice: once in a depth-first fashion to compute the *upstream probabilities* (as discussed above) and once in a breath-first fashion to compute the *downstream probabilities* (i.e., the summed probabilities of all paths from the root node to the considered one). The downstream probability of a node is recursively computed as a sum of the downstream probabilities of its parents—weighted by the squared magnitude of the weight attached to the connecting edge.

To guarantee a target fidelity  $f$ , a level is chosen and the nodes of this level are sorted by their contribution for the overall state vector in ascending order. The contribution is calculated by multiplying the downstream probabilities of the nodes with the summed upstream probabilities of the successors (considering again the weight attached to the connecting edge). These numbers sum up to 1 since a quantum state has a norm of one. An approximated state vector with fidelity  $f$  is now determined by summing up the sorted significance of the

nodes until the sum exceeds  $1 - f$ , followed by the elimination of all nodes included in the sum so far. Eliminating only these nodes guarantees that the approximated state vector has at least fidelity  $f$ .

The difficulty in this approach is to choose the level from which the nodes shall be eliminated in order to obtain the most compact decision diagram. One possibility to tackle this issue is to precompute how many nodes can be eliminated at each level. However, this still does not allow predicting how the elimination of nodes affects nodes below the chosen level. In fact, computing how the elimination of certain nodes affects the DD size requires determining all nodes to be removed.

**Example 6.** Consider again Fig. 1b. The calculated contributions based on upstream and downstream probabilities are depicted in Fig. 3c. Given a target fidelity of 0.5 and choosing the second level (i.e., the nodes labeled  $q_1$ ), the probabilities in ascending order are 0.2 and 0.8. The sum  $0.2 + 0.8$  exceeds 0.5, therefore only the  $q_1$  node on the right will be eliminated including its both successors (shown with a dashed outline)—leading to the approximation depicted in Fig. 2b.

### D. Approximation with a Target Fidelity per Level

Approximating a quantum state by targeting the fidelity of a single level requires determining a suitable level to eliminate nodes. As mentioned earlier, a possible solution is to calculate the fidelities for each level and choose one. Assuming that the target fidelity  $f$  is not required to be the absolute minimum fidelity acceptable, the visited nodes of each level can be eliminated instead of only eliminating the nodes for one level. This cannot guarantee fidelity  $f$ , but the resulting fidelity admits a lower bound  $f^{n-1}$  with  $n$  denoting the number of qubits. Notably, this bound was hardly reached in our experiments.

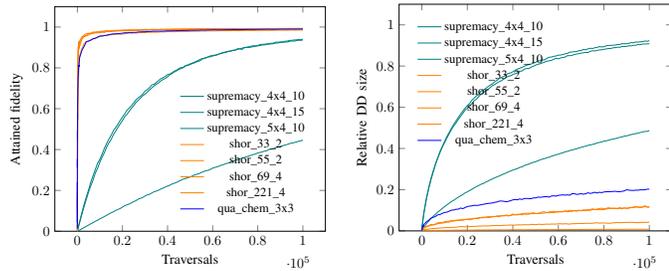
**Example 7.** Consider again Fig. 1b and the associated contributions to the quantum state depicted in Fig. 3c. To apply the fidelity-per-level scheme, the downstream probabilities are summed up for each level until reaching the target of 0.5 and all nodes visited up to this point are eliminated. More precisely, for the first level  $q_0$  the single node cannot be eliminated as it contributes to every basis state. On the second level the right node  $q_1$  is eliminated as in the previous example. Finally, on the third level the two nodes  $q_2$  on the right are eliminated as they only contribute 20% to this level.<sup>2</sup> These eliminations (i.e., the nodes drawn with a dashed outline) result in the approximation depicted in Fig. 2b.

## V. EVALUATION

In order to evaluate the potential of approximation for obtaining more compact DD-based representations of quantum states, we implemented the four schemes discussed in Section IV (on top of [25]). Afterwards, the respectively approximated decision diagrams have been evaluated regarding their resulting fidelity and the size compared to the original (i.e., non-approximated) decision diagrams. To this end, quantum states generated by representative quantum functionalities from the following groups have been considered as benchmarks:

- Quantum circuits provided by researchers from Google [26] as candidates to establish quantum-computational supremacy using controlled-phase gates (denoted “supremacy\_ $A \times B_C$ ”, representing a circuit on an  $A \times B$  surface with depth  $C$ ),

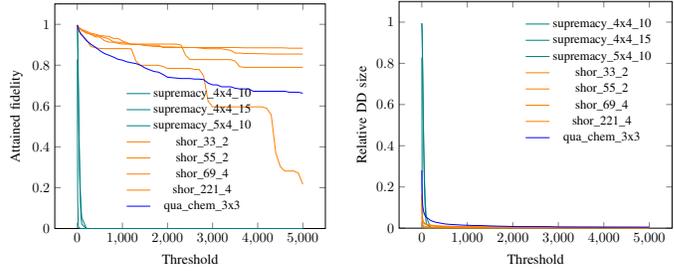
<sup>2</sup>This step is kept in the example as these nodes would have been eliminated automatically along with the  $q_1$  node.



(a) Attained fidelity v. #traversals

(b) Size v. #traversals

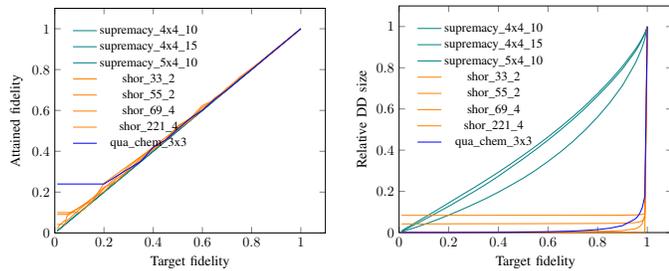
Fig. 4: Approximation based on the number of traversals



(a) Attained fidelity v. threshold

(b) Size v. threshold

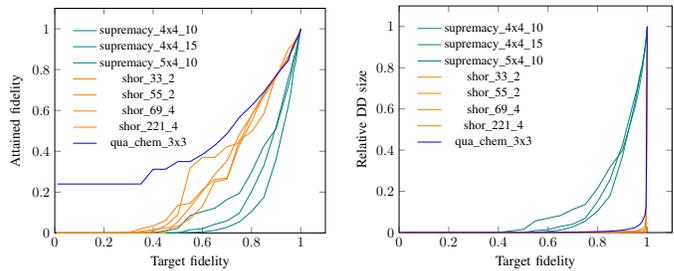
Fig. 5: Approximation based on traversals with a threshold



(a) Attained fidelity v. target

(b) Size v. target fidelity

Fig. 6: Approximation for a target fidelity



(a) Attained fidelity v. target

(b) Size v. target fidelity

Fig. 7: Approximation for a target fidelity per level

- quantum circuits realizing Shor’s algorithm to factorize integers [2] (denoted “shor\_A\_B” for factorizing  $A$  with the coprime value  $B$ ), and
- a quantum circuit simulating the uniform electron gas (taken from [27] and denoted “qua\_chem\_3x3”).

In the following, we present our main empirical results and put them in perspective.

### A. Comparison of Approximation Schemes

In a first series of evaluations, the different approximation schemes are compared against each other. To this end, Figures 4 to 7 show the respectively *attained fidelity* and the *relative DD size* of the approximated decision diagrams (compared to the original decision diagrams) for each of the four approximation schemes. Here, the plots nicely show the overall trade-off between the attained fidelity and the DD size as well as the effect of the respective parameter associated with each scheme. More precisely, Fig. 4 shows how the number of traversals relates to the attained fidelity and relative DD size (more traversals yield more accurate decision diagrams which increase the fidelity but also the resulting DD size). Fig. 5 demonstrates the same effect with a decreasing threshold, i.e., the smaller the threshold the more accurate (but also the larger) the decision diagrams.<sup>3</sup> Fig. 6 shows that the scheme in Section IV-C always guaranteed the target fidelity (besides that, the smaller the target fidelity is given the smaller decision diagrams are obtained). Finally, Fig. 7 shows that the scheme in Section IV-D might not always guarantee the target fidelity but still yields fidelities which are close to the targeted one.<sup>4</sup>

<sup>3</sup>One million traversals have been carried out before the threshold has been applied.

<sup>4</sup>Fig. 6 and Fig. 7 confirm that the corresponding schemes indeed approximate a quantum state towards a desired fidelity.

Overall, these results show that for *all* schemes the DD size is greatly reduced with moderate effects on the fidelity in many cases. The actual effects significantly depend on the respectively considered quantum functionality. Quantum states obtained from *quantum-supremacy* circuits are the hardest to approximate (but some schemes at least lead to improvements which are proportional to the fidelity losses). This can be explained by the fact that *most* amplitudes have comparable magnitudes in these cases and, hence, approximations harm the fidelity in the same proportion as the size is reduced. In contrast, quantum states arising in the remaining benchmarks frequently have many amplitudes which, probabilistically, are negligible and, hence, can easily be approximated. Eventually, this reduces the DD sizes to a fraction of its original size while controlling the attained fidelity for almost all schemes. This validates the initial idea proposed in this paper that approximating quantum states using decision diagrams is an effective technique for more compact representations.

### B. Application from a Designer’s Perspective

In a second series of evaluations, we additionally evaluated how the idea and findings proposed in this work explicitly aid quantum circuit designers and/or the development of corresponding design automation tools for quantum computing. Due to space limitations, we consider a representative scenario in which the acceptable fidelity is specified by the application context.<sup>5</sup> More precisely, assume that a fidelity of 0.5 is acceptable. Then, the designer can easily decide when to apply the scheme from Section IV-C to obtain an approximated quantum state, e.g., each time the decision diagram grows to big during simulation, in a fashion similar to garbage collection. Further, since the fidelity metric of a sufficiently entangled quantum state is multiplicative, the overall fidelity after each step can be easily

<sup>5</sup>Assumptions about the “acceptable fidelity” of a result obtained from a quantum computer can be made based on the considered functionality [28].

TABLE I: Target fidelity scheme targeting 0.5 fidelity

Benchmark	Orig. size	Approx. size	Compression	Fidelity
supremacy_4x4_10	65 071	23 305	0.3581	0.5000
supremacy_4x4_15	65 536	24 592	0.3752	0.5000
supremacy_5x4_10	482 408	127 412	0.2641	0.5000
shor_33_2	48 980	2 115	0.0432	0.5249
shor_55_2	93 541	344	0.0037	0.5064
shor_69_4	196 234	16 462	0.0839	0.5117
shor_221_4	1 048 575	198	0.0002	0.5197
qua_chem_3x3	59 594	147	0.0025	0.5098

tracked. Table I shows the corresponding characteristics of those decision diagrams (here, the name of the benchmark, the original DD size, the DD size of the approximated decision diagram, the relative size, and the fidelity of the approximated decision diagram is given). These results show that exact quantum state representations can be replaced with approximations in practice (e.g., for purposes of simulation, verification, synthesis)—leading to reductions of the DD size of several orders of magnitude while, at the same time, controlling the attained fidelity. In the best case, this reduces the DD size from over a million nodes to just under 200 nodes.

## VI. CONCLUSION

In this work, we proposed and evaluated the idea of approximating quantum states using decision diagrams. To this end, we exploited the inherent error resistance in quantum algorithms, i.e., small inaccuracies of the state representation do not (or hardly) affect the measurement outcome, in order to approximate the state of a quantum system. We applied this idea to decision diagrams by carefully analyzing their structure to quantify the contribution of each node to the quantum state. This resulted in four dedicated approximation schemes—leading to a reduction of several orders of magnitude in DD size while, at the same time, controlling the obtained fidelity. Moreover, our best approximation schemes closely match the desired quantum fidelity or even guarantee it. The findings of this work provide the basis for several design automation tools, e.g., for simulation, for verification, and for synthesis, that previously suffered from unaffordable memory requirements and may become practical at greater scales through the use of approximate quantum state representations in a fashion similar to garbage collection.

## ACKNOWLEDGMENTS

This work has partially been supported by the LIT Secure and Correct Systems Lab funded by the State of Upper Austria.

## REFERENCES

- [1] M. Nielsen and I. Chuang, *Quantum Computation and Quantum Information*. Cambridge Univ. Press, 2000.
- [2] P. W. Shor, “Algorithms for quantum computation: discrete logarithms and factoring,” *Foundations of Computer Science*, pp. 124–134, 1994.
- [3] L. K. Grover, “A fast quantum mechanical algorithm for database search,” in *Symp. on Theory of Computing*, 1996, pp. 212–219.
- [4] A. Montanaro, “Quantum algorithms: an overview,” *npj Quantum Information*, vol. 2, p. 15023, 2016.
- [5] J. Preskill, “Quantum computing in the NISQ era and beyond,” *Quantum*, vol. 2, p. 79, 2018.
- [6] 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.
- [7] L. Gomes, “Quantum computing: Both here and not here,” *IEEE Spectrum*, 2018.

- [8] R. Orús, “A practical introduction to tensor networks: Matrix product states and projected entangled pair states,” *Annals of Physics*, vol. 349, pp. 117–158, 2014.
- [9] M. Schuld, I. Sinayskiy, and F. Petruccione, “The quest for a quantum neural network,” *Quantum Information Processing*, vol. 13, no. 11, pp. 2567–2586, 2014.
- [10] G. F. Viamontes, I. L. Markov, and J. P. Hayes, “Improving gate-level simulation of quantum circuits,” *Quantum Information Processing*, vol. 2, no. 5, pp. 347–380, 2003.
- [11] 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*, vol. 35, no. 1, pp. 86–99, 2016.
- [12] A. Abdollahi and M. Pedram, “Analysis and synthesis of quantum circuits by using quantum decision diagrams,” in *Design, Automation and Test in Europe*, 2006, pp. 317–322.
- [13] S.-A. Wang, C.-Y. Lu, I.-M. Tsai, and S.-Y. Kuo, “An XQDD-based verification method for quantum circuits,” *IEICE Trans. Fundamentals*, vol. 91-A, no. 2, pp. 584–594, 2008.
- [14] G. F. Viamontes, I. L. Markov, and J. P. Hayes, “Checking equivalence of quantum circuits and states,” in *Int’l Conf. on CAD*, 2007, pp. 69–74.
- [15] P. Niemann, R. Wille, and R. Drechsler, “Equivalence checking in multi-level quantum systems,” in *Int’l Conf. of Reversible Computation*, 2014, pp. 201–215.
- [16] P. Niemann, R. Wille, and R. Drechsler, “Improved synthesis of Clifford+T quantum functionality,” *Design, Automation and Test in Europe*, 2018.
- [17] R. Wille, D. Große, D. M. Miller, and R. Drechsler, “Equivalence checking of reversible circuits,” in *Int’l Symp. on Multi-Valued Logic*, 2009, pp. 324–330.
- [18] A. Zulehner and R. Wille, “One-pass design of reversible circuits: Combining embedding and synthesis for reversible logic,” *IEEE Trans. on CAD of Integrated Circuits and Systems*, vol. 37, no. 5, pp. 996–1008, 2018.
- [19] G. F. Viamontes, I. L. Markov, and J. P. Hayes, *Quantum Circuit Simulation*. Springer, 2009.
- [20] A. Zulehner and R. Wille, “Advanced simulation of quantum computations,” *IEEE Trans. on CAD of Integrated Circuits and Systems*, vol. 38, no. 5, pp. 848–859, 2018.
- [21] 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*, 2019.
- [22] R. E. Bryant, “Graph-based algorithms for Boolean function manipulation,” *IEEE Trans. on Computers*, vol. 35, no. 8, pp. 677–691, 1986.
- [23] R. Drechsler and B. Becker, “Ordered Kronecker functional decision diagrams—a data structure for representation and manipulation of boolean functions,” *IEEE Trans. on CAD of Integrated Circuits and Systems*, vol. 17, no. 10, pp. 965–973, 1998.
- [24] J. M. Gambetta, J. M. Chow, and M. Steffen, “Building logical qubits in a superconducting quantum computing system,” *npj Quantum Information*, vol. 3, no. 1, p. 2, 2017.
- [25] A. Zulehner, S. Hillmich, and R. Wille, “How to efficiently handle complex values? implementing decision diagrams for quantum computing,” *Int’l Conf. on CAD*, 2019, The implementation is available at [http://iic.jku.at/eda/research/quantum\\_dd/](http://iic.jku.at/eda/research/quantum_dd/).
- [26] 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*, vol. 14, no. 6, p. 595, 2018.
- [27] R. Babbush, N. Wiebe, J. McClean, J. McClain, H. Neven, and G. K.-L. Chan, “Low-depth quantum simulation of materials,” *Phys. Rev. X*, vol. 8, p. 011044, 2018.
- [28] I. L. Markov, A. Fatima, S. V. Isakov, and S. Boixo, “Quantum supremacy is both closer and farther than it appears,” *CoRR*, vol. abs/1807.10749, 2018.