

Mixed-Dimensional Qudit State Preparation Using Edge-Weighted Decision Diagrams

Kevin Mato*

Stefan Hillmich[‡]

Robert Wille*[‡]

* Chair for Design Automation, Technical University of Munich, Germany

[‡] Competence Center Hagenberg (SCCH) GmbH, Austria

kevin.mato@tum.de, stefan.hillmich@scch.at, robert.wille@tum.de

<https://www.cda.cit.tum.de/research/quantum/>

ABSTRACT

Quantum computers have the potential to solve important problems which are fundamentally intractable on a classical computer. The underlying physics of quantum computing platforms supports using multi-valued logic, which promises a boost in performance over the prevailing two-level logic. One key element to exploiting this potential is the capability to efficiently prepare quantum states for multi-valued, or *qudit*, systems. Due to the time sensitivity of quantum computers, the circuits to prepare the required states have to be as short as possible. In this paper, we investigate quantum state preparation with a focus on mixed-dimensional systems, where the individual qudits may have different dimensionalities. The proposed approach automatically realizes quantum circuits constructing a corresponding mixed-dimensional quantum state. To this end, decision diagrams are used as a compact representation of the quantum state to be realized. We further incorporate the ability to approximate the quantum state to enable a finely controlled trade-off between accuracy, memory complexity, and number of operations in the circuit. Empirical evaluations demonstrate the effectiveness of the proposed approach in facilitating fast and scalable quantum state preparation, with performance directly linked to the size of the decision diagram. The implementation is freely available as part of *Munich Quantum Toolkit* (MQT) at github.com/cda-tum/mqt-qudits.

ACM Reference Format:

Kevin Mato, Stefan Hillmich, and Robert Wille. 2024. Mixed-Dimensional Qudit State Preparation Using Edge-Weighted Decision Diagrams. In *61st ACM/IEEE Design Automation Conference (DAC '24)*, June 23–27, 2024, San Francisco, CA, USA. ACM, New York, NY, USA, 6 pages. <https://doi.org/10.1145/3649329.3656260>

1 INTRODUCTION

Quantum computing is capable of addressing problems intractable to classical computers, thanks to its distinct paradigm. Examples include, Shor's algorithm [1] to factorize integers, Grover's search [2] for unstructured data, and evaluating possible materials in quantum chemistry [3]. Worldwide, numerous academic and industrial research groups, including Google, IBM, and Microsoft, are actively exploring its potential and pushing the boundaries of its

current capabilities. Until now, most applications have centered around two-dimensional qubit systems, which has led to the under-utilization of the abundant potential offered by higher-dimensional systems present in various physical realizations of quantum computers. While constructing circuits with higher dimensions poses challenges, it also brings forth numerous benefits and serves as a more conducive platform for algorithms like quantum simulation.

The concept of higher-dimensional systems and its corresponding theory has existed for a considerable period [4]. Fundamentally, qudits offer denser information storage and a broader range of operations than qubits. This has led to the successful demonstration of basic control in physical platforms such as trapped-ions [5], to photonic systems [6] and superconducting circuits [7].

Recent developments in quantum algorithms have shown that multi-level logic is a more natural architecture for implementing complex applications [8]. Simulations of models representing fermion-boson interactions on mixed-dimensional quantum computers could enable real-time simulations of quantum electrodynamics and other field theories with continuous or larger symmetry groups [9], [10]. This is possible with the potentially reduced circuit complexity due to the temporary expansion of the Hilbert space [11].

As the arrival of new and robust multi-level quantum devices and innovative quantum algorithms is anticipated, it is reasonable to expect encountering familiar challenges similar to those of qubit systems. Accordingly, methods, e.g., for the simulation [12] or compilation [13]–[15], of mixed-dimensional quantum circuits have been proposed recently. In this context, also methods for state preparation gain relevance especially for promising applications such as quantum simulation and quantum machine learning [16]–[18], that heavily rely on specific initial states to kickstart their processes effectively. Tackling the state preparation problem is crucial not only for facilitating quantum simulations but also for gaining insights into the behavior of specific states that have not yet been extensively studied in qudit systems, including aspects like entanglement. By addressing this obstacle, the full potential of various applications for quantum technologies can be unlocked.

In this paper, we present a novel state preparation synthesis approach to compile mixed-dimensional quantum states with few operations and controls, later transpiled for an architecture-specific gate-set. To this end, the following three contributions are made:

- We explore the utilization of edge-weighted decision diagrams for a compact and efficient representation of *mixed-dimensional* states.
- We propose an automated procedure to generate quantum circuits for the construction of arbitrary quantum states.
- Further, we explore optimized circuit realizations, by leveraging the approximation of decision diagrams.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

DAC '24, June 23–27, 2024, San Francisco, CA, USA

© 2024 Copyright held by the owner/author(s). Publication rights licensed to ACM.

ACM ISBN 979-8-4007-0601-1/24/06

<https://doi.org/10.1145/3649329.3656260>

We demonstrate the advantages and potential of the proposed method by compiling a diverse range of quantum states, both uniform and random. These states were synthesized into quantum circuits with qudits of mixed-dimensions. For the first time, this showcases the automatic and efficient generation of mixed-dimensional quantum circuits that can construct arbitrary states. Additionally, by incorporating approximations, these circuits can be further optimized without significantly compromising their fidelity.

The remainder of this paper is organized as follows: Section 2 gives a brief background on quantum computations with qubits and qudits. Section 3 describes the considered problem, reviews the state of the art, and summarizes the contribution of this paper. Section 4 introduces the proposed approach for synthesizing quantum circuits for qubit-qudit circuits in detail. Section 5 evaluates the proposed approach. Finally, Section 6 concludes the paper.

2 BACKGROUND

In this study, we establish the groundwork for effectively preparing states in mixed-dimensional quantum systems. Initially, we offer a concise overview of quantum information, with a special emphasis on multi-level quantum logic.

In classical computations, information is encoded using *bits*, binary digits that exist in either the 0 or 1 state. When we transition to quantum computing, we introduce the concept of a *qubit* (quantum bit) as the fundamental unit of information. Unlike classical bits, qubits possess a remarkable feature—they can exist in a superposition, representing almost any linear combination of $|0\rangle$ and $|1\rangle$ (using Dirac's notation). This characteristic forms the core distinction between quantum and classical computing paradigms.

Qubits are typically engineered by confining the multi-level configuration of the underlying physical components that store information. Consequently, these systems inherently support *multi-level logic*, where the basic information unit is referred to as a *qudit* (quantum digit). A qudit represents the quantum counterpart of a d -ary digit, where $d \geq 2$. Its state can be expressed as a vector within the d -dimensional Hilbert space \mathcal{H}_d , and it can be written as a linear combination $|\psi\rangle = \alpha_0 \cdot |0\rangle + \alpha_1 \cdot |1\rangle + \dots + \alpha_{d-1} \cdot |d-1\rangle$, or simplified as vector $|\psi\rangle = [\alpha_0 \ \alpha_1 \ \dots \ \alpha_{d-1}]^T$, where $\alpha_i \in \mathbb{C}$ are the amplitudes relative to the orthonormal basis of the Hilbert space—given by the vectors $|0\rangle, |1\rangle, |2\rangle, \dots, |d-1\rangle$. The squared magnitude of an amplitude $|\alpha_i|^2$ defines the probability with which the corresponding basis state i will be observed when measuring the qudit. Since the probabilities have to add up to 1, the amplitudes have to satisfy $\sum_{i=0}^{d-1} |\alpha_i|^2 = 1$.

EXAMPLE 1. Consider a system of one qudit with only three energy levels (also referred to as qutrit). The quantum state $|\psi\rangle = \sqrt{1/3} \cdot |0\rangle + \sqrt{1/3} \cdot |1\rangle + \sqrt{1/3} \cdot |2\rangle$ is a valid state with equal probability of measuring each basis. Equivalently, the quantum state may be represented as vector $\sqrt{1/3} \cdot [1 \ 1 \ 1]^T$.

Quantum computing stands apart from classical computing due to two critical characteristics: superposition and entanglement. When we refer to a qudit being in a *superposition* of states within a specific basis, it means that at least two amplitudes, relative to this basis, are non-zero. In simpler terms, the qudit can exist in multiple states simultaneously. On the other hand, *entanglement* represents a unique kind of superposition that arises from interactions in multi-qudit systems. Entanglement entails a robust form of quantum correlation, where the entire system's state carries encoded

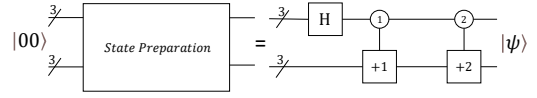


Figure 1: The state preparation algorithm for constructing $|\psi\rangle$, compiled into a sequence of single and two-qudit operations, in this case a two-qutrit GHZ state.

information, and it becomes impossible to extract information from individual qudits separately. The state of a single d -level qudit system can be manipulated by operations which are represented in terms of $d \times d$ -dimensional unitary matrices U , i.e., matrices that satisfy $U^\dagger U = U U^\dagger = I$. The state after the application of U can be determined by multiplying the corresponding input state from the left with the matrix U .

EXAMPLE 2. Consider a three-level qudit (i.e., a qutrit) initially in the state $|0\rangle$. Applying the Hadamard operation H to it yields the output state shown before in Example 1, i.e.,

$$H \cdot |0\rangle = \frac{1}{\sqrt{3}} \begin{bmatrix} 1 & 1 & 1 \\ 1 & e^{\frac{2\pi}{3}} & e^{-\frac{2\pi}{3}} \\ 1 & e^{-\frac{2\pi}{3}} & e^{\frac{2\pi}{3}} \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} = \frac{1}{\sqrt{3}} \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}.$$

3 MOTIVATION

This section revisits the problem of state preparation, particularly in mixed-dimensional quantum circuits. Then, we review relevant prior research and highlight the contribution of this work, which focuses on providing a scalable state preparation method by leveraging mixed-dimensional decision diagrams.

3.1 Considered Problem

Quantum state preparation is a topic of great interest, both theoretically and practically. It plays a crucial role in determining the efficiency of inputting classical data into a quantum computer. Additionally, it serves as a critical subroutine for various quantum algorithms, including well-known ones [1], [2], [19], as well as quantum machine learning [16], [20], [21], and Hamiltonian simulations [17], [18]. Formally, the task of quantum state preparation involves preparing a given quantum state $|\psi\rangle$ from an initial product state $|0\rangle^{\otimes N}$, using single and two-qudit gates of different types, as shown in Figure 1. However, quantum operations are prone to errors due to factors such as limited qudit connectivity, decoherence, and gate infidelity. These factors lead to inaccuracies in quantum computations, necessitating the development of methods and techniques that can achieve reliable results by minimizing the number of operations.

Hence, the state preparation problem is to find the shortest sequence of quantum operations that can implement a state $|\psi\rangle$, starting from the product state $|0\rangle^{\otimes N}$, while achieving a desired quantum state fidelity, in the most computationally efficient fashion.

EXAMPLE 3. Let's consider the realization of the quantum state $\frac{1}{\sqrt{3}}(|00\rangle + |11\rangle + |22\rangle)$, i.e., a GHZ state, in a qutrit-qutrit system. The state is entangled and a superposition of three basis states $|00\rangle, |11\rangle, |22\rangle$. The problem is to find a sequence of operations that can transform the state $|00\rangle$ into this state. Figure 1 shows a possible result: the qutrit Hadamard generates the superposition as shown in Example 2. Then,

controlled operations on the state of the first qutrit manipulate the second one. Each operation is controlled by the level inside the circle and imposes an increment based by “+1” or “+2”.

3.2 Related Work

While previous methods have been proposed for preparing arbitrary quantum states, their focus has been on qubit quantum circuits and tailored to use cases or particular state classes [22]–[24]. However, recent advancements in circuit depth and complexity have brought state preparation qubit circuits to a promising stage, albeit at the expense of an exponential number of ancillary qubits [25]. In the context of state preparation, decision diagrams have been utilized with success for specific classes of states, such as uniform states, cyclic states, and arbitrary states, although with the need for ancilla qubits [23], [24]. These studies demonstrate the effectiveness of decision diagrams in automating state preparation. Additionally, research has been conducted on searching for physical experiments designed to construct states directly [26]. While scalable algorithms have been presented for the case of prime-dimensional qudits w-states, or their qubit-embedded version [27], the investigation and development of scalable automated procedures, particularly optimized ones, for qudit circuits and mixed-dimensional systems, remain unexplored.

3.3 Contribution

In this paper, we investigate state preparation methods for mixed-dimensional quantum systems, facilitating various quantum algorithms on such architectures while automating complex routines involved in quantum computations. The proposed approach aims to enhance circuit efficiency in a scalable manner. To achieve this, we introduce a tool that enables automatic and efficient quantum state preparation using decision diagrams for mixed-dimensional systems.

To realize such method, we make the following three contributions:

- The first contribution of this work is the exploration of the problem of state preparation by leveraging a recently investigated data structure, called edge-weighted decision diagrams with a variable number of successors. This data structure enables a more accurate and realistic representation of diverse quantum systems, accommodating qubits and qudits of different dimensions, akin to the natural diversity observed in complex quantum simulations. This is significantly different to traditional decision diagrams [28], which were restricted to handling uniform-sized information units. Moreover, this study holds particular significance as there is no prior work on state preparation based on decision diagrams with a variable number of successors.
- The second contribution of this work focuses on the synthesis of a high-level quantum circuits, which facilitates the automated state preparation procedure in an efficient manner. The complexity of this routine is linear in the number of nodes of the decision diagram. This significant advancement plays a crucial role in constructing states that would otherwise be exceedingly difficult and time-consuming to engineer manually. By automating the state preparation process, researchers can efficiently handle complex quantum states, unlocking new possibilities and accelerating progress in quantum information processing.

- The final contribution of this work involves the application of approximation techniques, with the final goal of reducing the gate count of the generated quantum circuits. Specifically, these techniques target the elimination of operations that generate portions of the state that are practically irrelevant for the computation, as a result of the reduction in the size of the decision diagram representing the desired state. While previous work has focused on this aspect for *qubit* circuit simulations [29], the current study extends and generalizes these routines to encompass qudit systems, for a new purpose. Additionally, new reduction rules are introduced in the decision diagram simplification process. The primary outcome of these approximation techniques is the simplification of the final logic. By eliminating nodes in the decision diagram, the quantum circuit complexity is also reduced, led by the discovery of patterns of tensor operations between subspaces of the single qudit Hilbert spaces, in the form of DD subgraphs. These intricate patterns are challenging to compute and have not been explored previously. Furthermore, a secondary benefit achieved is the reduction in the number of controls needed for each operation, enabling the translation to more resource-efficient sequences of operations.

4 STATE PREPARATION

In this section, we propose the method to synthesize state decision diagrams into mixed-dimensional qudit circuits. To this end, we start by introducing decision diagrams, which efficiently store multi-dimensional quantum states. Then, we explain how these decision diagrams are used to create quantum circuits that generate the desired states. Lastly, we discuss how the resulting circuits can be further optimized using approximation techniques.

4.1 Decision Diagrams

Previously, decision diagrams have been demonstrated to facilitate efficient representations of exponentially large data in numerous instances [28], [30]–[32], owing to their ability to achieve compactness by capitalizing on the redundancy present in the data they depict.

More precisely, a *Decision Diagram* (DD) is a type of *Directed Acyclic Graph* (DAG) comprising nodes and directed edges. These nodes are structured in levels, where each level corresponds to a specific qudit. The edges represent the choice of value for a qudit (or node) on a level and carry additional information, e.g., complex edge weights. The fundamental concept behind using decision diagrams to describe quantum states relies on recursively breaking down the associated vector.

This following procedure describes the first contribution and part of the procedure corresponding to the first arrow in Figure 2, in other words, the representation of a quantum state as a decision diagram. Consider an n -qudit quantum state defined over a set of variables $q_{n-1}, q_{n-2}, \dots, q_0$, where q_{n-1} w.l.o.g. is assigned as the “most significant qudit”. Firstly, the state is divided into parts of equal size, based on the dimensionality of qudit q_{n-1} . Each part is associated with a successor node, which also contains the decision regarding the value of q_{n-1} . This splitting process continues for each sub-vector until the individual complex entries in the original vector are obtained. Throughout this process, the complex amplitudes of each basis state are stored in the form of weights inside the edges on the path from the root to the leaf followed during the

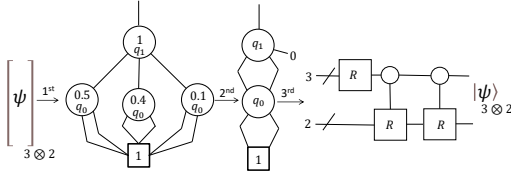


Figure 2: The three steps of state preparation.

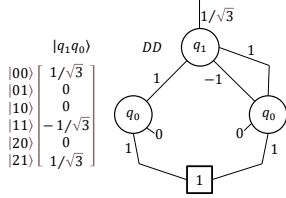


Figure 3: A state vector of circuit composed of a qutrit and a qubit and the corresponding decision diagram representation.

decomposition. To ensure consistency, the nodes are normalized such that the sum of the squared magnitudes of the out-edges from a node adds up to one. To reconstruct the amplitude for specific basis states, one can traverse the decision diagram accordingly, multiplying all edge weights along the chosen path.

The resulting decision diagram consists of a *root node* q_{n-1} , at least one node for each subsequent q_i , and ultimately a single *terminal node* without any successors. An example illustrates the procedure.

EXAMPLE 4. Consider the quantum state $\frac{1}{\sqrt{3}}(|00\rangle - |11\rangle + |21\rangle)$ in a qutrit-qubit system, represented in 2 forms in Figure 3. The vector's dimension is 6, which results from combining the local dimensionalities of the qutrit 3 and the qubit 2. The root node, denoted as q_1 , branches into 3 edges, each corresponding to a distinct level in the qutrit. Moving to the 2nd level, we encounter nodes representing the qubit, with each node having 2 edges. Notably, the 2nd and 3rd edges of the root node connect to the same qubit node, making use of redundancy. The nodes labeled as q_0 direct to the terminal node. To calculate the amplitude of a specific basis state, we multiply the weights along the path corresponding to that basis state. For instance, for the bitstring $|11\rangle$, the computation involves multiplying $1/\sqrt{3} \cdot -1 \cdot 1$, which accounts for the weights of the root node, the 1st edge of q_1 , and the 1st edge of q_0 .

4.2 Synthesis Algorithm

The algorithm initiates by recursively constructing the decision diagram that represents the quantum state, accounting for the dimensionalities of individual qudits and qubits, as described in the previous section. Due to varying dimensionalities among units of information, nodes at the same level may possess the same number of successors, while nodes at different levels could exhibit distinct dimensionality. Each connection linking a parent node and its successor is assigned a complex number weight—the normalizing factor calculated recursively from the out-edges of the successor. The normalization starts from the terminal nodes' edges. The weights are normalized by a fixed scheme to ensure canonicity in the nodes and subsequently a more compact representation after the reduction

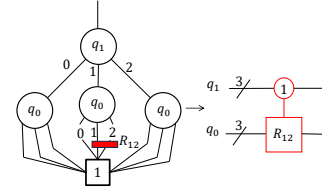


Figure 4: Example of a DD representing the state of a circuit composed of two qutrits and of a rotation synthesized from it.

step. To normalize the weights of the out-edges of a given nodes, each weight is divided by the norm, such that the sum of squared magnitudes of the edge weights equals one. The norm is then multiplied to all weights on in-edges of the considered node. At this point, the decision diagram forms a weighted tree and is fully prepared for the traversal, that will produce the quantum operations for constructing the desired quantum state.

Now we proceed by illustrating the generation of the quantum circuit starting from the decision diagram, contribution of this work corresponding to the third arrow in Figure 2. The routine operates recursively, starting from the root node, and traverses the decision diagram. It iterates through the successor edges, beginning from the end of the list, in pairs of two, following a decreasing order. At each step, the algorithm performs a two-level rotation, known as a Givens rotation[5], on the two adjacent edges or dimensions represented by the node. The rotations between two levels i and j are expressed as

$$R_{i,j}(\theta, \phi) = \exp\left(\frac{-i\theta}{2}\left(\cos(\phi)\sigma_x^{i,j} + \sin(\phi)\sigma_y^{i,j}\right)\right),$$

where $\sigma_{\{x,y\}}$ are the Pauli- $\{X, Y\}$ matrices, θ is the rotation angle, and ϕ is the phase of the rotation. These parameters are calculated as: $\theta = 2 \cdot \arctan[|w_i/w_j|]$, $\phi = -(\pi/2 + \arg[w_j] - \arg[w_i])$. The sequence finishes with a phase rotation applied on the level 0-1, but with angle as the phase difference between the level 0, calculated from the sequence, and the father's node weight. The rotation can be decomposed into two-level rotations using the identity $Z(\theta) = R(-\frac{\pi}{2}, 0) \cdot R(\theta, \frac{\pi}{2}) \cdot R(\frac{\pi}{2}, 0)$.

Every rotation is going to be applied to the circuit with a set of qudit controls equivalent to the set of nodes encountered on the path from the root to the node of the edges considered. For each node, the control level of the operation is going to be the index of the edge taken in order to descend the decision diagram.

EXAMPLE 5. Figure 4 shows a step of the synthesis algorithm. The algorithm appends to the circuit the rotation R on levels 1 and 2, and calculates the parameters accordingly. The rotation is controlled on level 1 of the the first qutrit since the rotation was derive from the node with index 1.

4.3 Optimization and Approximation

In this section we present part of the contribution of this work that is depicted by the second arrow in Figure 2. To facilitate the synthesis of shorter circuits, we employed several optimizations during the decision diagram construction stage. The first optimization comes from studies in quantum circuit simulation. This field has exploited decision diagrams in the past and it requires efficient approaches due to exponential complexity of the problem tackled.

Approximated decision diagrams proved to be an effective solution for improving memory complexity and runtime performance [29]. However, the technique comes with a trade-off with state fidelity, requiring a careful balance between computational efficiency and accuracy in results. The technique is a generalization of [29], where it calculates the contributions in terms of fidelity of each node and then removing nodes from the decision diagram until a threshold fidelity, previously chosen for the synthesis, is reached. The contribution is calculated as the sum of the squared magnitude of each amplitude that has a path from root to leaves crossing the node.

The benefits of the technique are the following:

- The method reduces the size of the decision diagram in terms of memory.
- It reduces the synthesis time, since its complexity is tied to the size of the decision diagram.
- Lastly, the technique enables synthesizing shorter circuits for the state preparation, with guarantees on the fidelity reached by the procedure

The second optimization consists in the reduction of the decision diagram, intended as the capability of two edges pointing to the same node, whenever it represents two identical sub-trees, that would be otherwise stored twice. The advantage here lies in the logical significance of this reduction. When all edges with weights other than zero point to the same node, it resembles a tensor product operation between the two qudits representing adjacent levels in the tree. Consequently, operations in the sub-tree will not consider the father node with successors pointing to the same child as a control qudit, thereby reducing the number of entangling gates during transpilation.

EXAMPLE 6. *Figure 2 depicts the application of the optimization techniques. The successor node with lowest fidelity contribution (0.1) of the root is pruned from the decision diagram. The two remaining edges that point to nodes representing the same sub-vectors are redirected to the same shared node. This will reduce the amount of memory used for storing nodes, and due to the properties of tensor products, no controls will be synthesized.*

Overall, the method results in a state preparation procedure of three main steps, as depicted in Figure 2: starting with a representation of the state as a decision diagram, followed by an approximation routine to reduce the size of the DD, and concluding with the realization of optimized quantum circuits.

5 EXPERIMENTAL EVALUATION

To evaluate the feasibility of the proposed quantum state preparation approach, we implemented the method and evaluated it on several types of quantum states. The selected benchmarks are

- Embedded W-State [27],
- GHZ State [33],
- W-State [34], and
- Random states with amplitudes generated from a uniform distribution.

For each state a mixed-dimensional system is used as target architecture. The implementation is written in Python 3, excluding third-party dependencies. The evaluations were performed on a server running GNU/Linux using an AMD Ryzen Threadripper PRO 5955WX (at 4 GHz) and 125 GiB main memory. The implementation

used in this evaluation is freely available (as part of the Munich Quantum Toolkit, MQT) at github.com/cda-tum/mqt-qudits.

The results are presented in Table 1. The synthesis procedure is executed 40 times to average possible fluctuations. The initial group of columns provides key information about the benchmark, including the algorithm’s name, the number of qudits involved, and the qudit dimensions represented as $Count \times Dimension$, randomly selected to further prove the efficacy of the method. For example, an entry 2×4 signifies two qudits, each with a dimension of four. The following two column groups present details on the resulting qudit circuits for both “Exact” (circuits that compile the target state with fidelity 1) and “Approximated 98 %” (compiling the target state with fidelity of at least 0.98). The columns “Nodes” and “DistinctC” represent the average number of nodes and unique complex numbers in the decision diagram representing the state. These are the main metrics for evaluating the efficiency in the decision diagram for storing the mixed-dimensional quantum states. The “Operations” column indicates the average number of multi-controlled operations synthesized for a state on a specific mixed-dimensional architecture, relevant for both exact and approximated synthesis. Additionally, the column “Controls” refers to the average median number of controls present in multi-controlled operations synthesized for a state on the mixed-dimensional architecture. The use of controlled operations as a primary metric is justified by the fact that the circuit can later be transposed into a sequence of local and two-qudit operations [35], with also linear complexity in terms of depth, as demonstrated in [36]. Finally, the “Time” column denotes the elapsed time during the approximation and synthesis process. The method is efficient, since the synthesis routine has time complexity linear in the number of nodes of the DD. Moreover, the approximation technique leads to a linear improvement in the number of quantum gates, with the reduction of control nodes during the synthesis.

Both approaches, “Exact” and “Approximated” have been shown to be capable of preparing a given quantum state in a mixed-dimensional system. Due to the regular structure of the first three (non-random) benchmarks, the approximation shows no effect. In contrast, approximation decreases the number of operations (and controls) by about 5 % while losing only 1 % fidelity. Regardless of the approximation step, the individual runs of the benchmarks finish in less than one second.

Overall, the results confirm the successful development of the first automated method for synthesizing quantum circuits that can construct arbitrary states for mixed-dimensional quantum systems. Furthermore, the outcomes demonstrate the method’s efficiency and the capabilities in optimizing the circuits, preparing the desired state without significantly affecting its fidelity.

6 CONCLUSION

In this work, we proposed a novel synthesis method for the generation of quantum circuits, performing the state preparation of arbitrary states for mixed-dimensional quantum systems. This is the first developed automated method for such an application on quantum architectures with qubits and qudits of different sizes.

The approach uses edge-weighted decision diagrams with a variable number of successors for the representation of the state and for the synthesis of the quantum circuits. Extended approximation techniques applied to the data-structure optimizes the number of

Table 1: Evaluation of the proposed approach comparing the average results over 40 runs of the synthesis method per benchmark

| Name | Benchmark | | Exact (Averaged) | | | | | Approximated 98% (Averaged) | | | | | |
|--------------|-----------|--|------------------|-----------|------------|-----------|----------|-----------------------------|-----------|------------|-----------|----------|----------|
| | #Qudits | Qudits | Nodes | DistinctC | Operations | #Controls | Time [s] | Nodes | DistinctC | Operations | #Controls | Time [s] | Fidelity |
| Emb. W-State | 3 | $[1 \times 3, 1 \times 6, 1 \times 2]$ | 58.0 | 5.0 | 21.0 | 2.0 | 0.00 | 22.0 | 6.0 | 21.0 | 2.0 | 0.00 | 1.00 |
| | 4 | $[1 \times 9, 1 \times 5, 1 \times 6, 1 \times 3]$ | 1135.0 | 7.0 | 49.0 | 3.0 | 0.01 | 50.0 | 8.0 | 49.0 | 3.0 | 0.01 | 1.00 |
| | 6 | $[3 \times 4, 1 \times 7, 1 \times 3, 1 \times 5]$ | 8657.0 | 12.0 | 91.0 | 3.0 | 0.03 | 92.0 | 12.0 | 91.0 | 3.0 | 0.03 | 1.00 |
| GHZ State | 3 | $[1 \times 3, 1 \times 6, 1 \times 2]$ | 58.0 | 3.0 | 19.0 | 2.0 | 0.00 | 20.0 | 3.0 | 19.0 | 2.0 | 0.00 | 1.00 |
| | 4 | $[1 \times 9, 1 \times 5, 1 \times 6, 1 \times 3]$ | 1135.0 | 3.0 | 51.0 | 2.0 | 0.01 | 52.0 | 3.0 | 51.0 | 2.0 | 0.01 | 1.00 |
| | 6 | $[3 \times 4, 1 \times 7, 1 \times 3, 1 \times 5]$ | 8657.0 | 3.0 | 73.0 | 2.0 | 0.04 | 74.0 | 3.0 | 73.0 | 2.0 | 0.05 | 1.00 |
| W-State | 3 | $[1 \times 3, 1 \times 6, 1 \times 2]$ | 58.0 | 5.0 | 37.0 | 2.0 | 0.00 | 38.0 | 6.0 | 37.0 | 2.0 | 0.00 | 1.00 |
| | 4 | $[1 \times 9, 1 \times 5, 1 \times 6, 1 \times 3]$ | 1135.0 | 11.0 | 186.0 | 2.0 | 0.03 | 185.0 | 10.0 | 186.0 | 2.0 | 0.03 | 1.00 |
| | 6 | $[3 \times 4, 1 \times 7, 1 \times 3, 1 \times 5]$ | 8657.0 | 14.0 | 262.0 | 4.0 | 0.06 | 259.0 | 14.0 | 262.0 | 4.0 | 0.06 | 1.00 |
| Random State | 3 | $[1 \times 3, 1 \times 6, 1 \times 2]$ | 58.0 | 58.0 | 57.0 | 2.0 | 0.00 | 40.2 | 52.9 | 54.05 | 1.9 | 0.00 | 0.99 |
| | 4 | $[1 \times 9, 1 \times 5, 1 \times 6, 1 \times 3]$ | 1135.0 | 1135.0 | 1134.0 | 2.0 | 0.12 | 573.92 | 1071.97 | 1084.28 | 2.82 | 0.10 | 0.99 |
| | 5 | $[2 \times 6, 1 \times 5, 2 \times 3]$ | 2383.0 | 2383.0 | 2382.0 | 4.0 | 0.14 | 1255.22 | 2276.62 | 2287.07 | 3.75 | 0.16 | 0.99 |
| | 6 | $[3 \times 5, 1 \times 4, 2 \times 2]$ | 3266.0 | 3266.0 | 3265.0 | 5.0 | 0.18 | 2187.15 | 3121.65 | 3136.9 | 4.79 | 0.19 | 0.99 |
| | 6 | $[3 \times 4, 1 \times 7, 1 \times 3, 1 \times 5]$ | 8657.0 | 8657.0 | 8656.0 | 5.0 | 0.44 | 3176.65 | 8336.8 | 8357.35 | 4.78 | 0.39 | 0.99 |

operations in the circuits. The evaluations conducted in the study showcased the method's efficiency and versatility in preparing arbitrary quantum states for any mixed-dimensional quantum architecture. Through optimizations, we achieved a notable reduction in the number of operations and controls required, while maintaining a high fidelity. Potential future work could involve further refining and exploring qudit quantum circuit optimization and approximation techniques by taking the capabilities of the targeted quantum hardware in account.

ACKNOWLEDGMENT

This work has received funding from the European Union's Horizon 2020 research and innovation programme under the ERC Consolidator Grant (agreement No 101001318) and the NeQST Grant (agreement No 101080086). It is part of the Munich Quantum Valley, which is supported by the Bavarian state government with funds from the Hightech Agenda Bayern Plus and was partially supported by the BMK, BMDW, and the State of Upper Austria in the frame of the COMET program (managed by the FFG).

REFERENCES

- [1] P. W. Shor, "Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer," *SIAM Jour. of Comp.*, 1997.
- [2] L. K. Grover, "A fast quantum mechanical algorithm for database search," in *Symp. on Theory of Computing*, 1996.
- [3] Y. Cao, J. Romero, *et al.*, "Quantum chemistry in the age of quantum computing," *Chemical Reviews*, 2019.
- [4] Y. Wang, Z. Hu, B. C. Sanders, and S. Kais, "Qudits and high-dimensional quantum computing," *Frontiers in Physics*, 2020.
- [5] M. Ringbauer, M. Meth, *et al.*, "A universal qudit quantum processor with trapped ions," *Nature Physics*, 2022.
- [6] X.-M. Hu, Y. Guo, B.-H. Liu, Y.-F. Huang, C.-F. Li, and G.-C. Guo, "Beating the channel capacity limit for superdense coding with entangled ququarts," *Sci. Adv.*, 2018.
- [7] M. Kononenko, M. Yurtalan, S. Ren, J. Shi, S. Ashhab, and A. Lupascu, "Characterization of control in a superconducting qutrit using randomized benchmarking," *Philips Res. Repts.*, 2021.
- [8] Y. Deller, S. Schmitt, *et al.*, "Quantum approximate optimization algorithm for qudit systems," *Phys. Rev. A*, 2023.
- [9] E. J. Gustafson, "Prospects for simulating a qudit-based model of $(1 + 1)D$ scalar qed," *Phys. Rev. D*, 2021.
- [10] L. Lamata, A. Mezzacapo, J. Casanova, and E. Solano, "Efficient quantum simulation of fermionic and bosonic models in trapped ions," *EPJ Quantum Technology*, 2014.
- [11] B. Lanyon, M. Barbieri, *et al.*, "Quantum computing using shortcuts through higher dimensions," *Nature Physics*, 2008.
- [12] K. Mato, S. Hillmich, and R. Wille, "Mixed-dimensional quantum circuit simulation with decision diagrams," in *Int'l Conf. on Quantum Computing and Engineering*, 2023.
- [13] K. Mato, M. Ringbauer, S. Hillmich, and R. Wille, "Adaptive compilation of multi-level quantum operations," in *Int'l Conf. on Quantum Computing and Engineering*, 2022.
- [14] K. Mato, M. Ringbauer, S. Hillmich, and R. Wille, "Compilation of entangling gates for high-dimensional quantum systems," in *Asia and South Pacific Design Automation Conf.*, 2023.
- [15] K. Mato, S. Hillmich, and R. Wille, "Compression of Qubit Circuits: Mapping to Mixed-Dimensional Quantum Systems," in *Int'l Conf. on Quantum Software*, 2023.
- [16] X. Wang, L. Gu, H.-W. Lee, and G. Zhang, "Quantum context-aware recommendation systems based on tensor singular value decomposition," *Quantum Information Processing*, 2021.
- [17] G. H. Low and I. L. Chuang, "Hamiltonian simulation by qubitization," *Quantum*, 2019.
- [18] A. M. Childs, D. Maslov, Y. Nam, N. J. Ross, and Y. Su, "Toward the first quantum simulation with quantum speedup," *Proceedings of the National Academy of Sciences*, 2018.
- [19] A. Y. Kitaev, "Quantum measurements and the abelian stabilizer problem," *Electron. Colloquium Comput. Complex.*, 1995.
- [20] A. W. Harrow, A. Hassidim, and S. Lloyd, "Quantum algorithm for linear systems of equations," *Phys. Rev. Lett.*, 2009.
- [21] S. Lloyd, M. Mohseni, and P. Rebentrost, "Quantum principal component analysis," *Nature Physics*, 2014.
- [22] P. Niemann, R. Datta, and R. Wille, "Logic synthesis for quantum state generation," *Int'l Symp. on Multi-Valued Logic*, 2016.
- [23] F. Mozafari, G. D. Micheli, and Y. Yang, "Efficient deterministic preparation of quantum states using decision diagrams," *Phys. Rev. A*, 2022.
- [24] F. Mozafari, Y. Yang, and G. D. Micheli, "Efficient preparation of cyclic quantum states," in *Asia and South Pacific Design Automation Conf.*, 2022.
- [25] X.-M. Zhang, T. Li, and X. Yuan, "Quantum state preparation with optimal circuit depth: Implementations and applications," *Phys. Rev. Lett.*, 2022.
- [26] M. Krenn, M. Malik, R. Fickler, R. Lapkiewicz, and A. Zeilinger, "Automated search for new quantum experiments," *Phys. Rev. Lett.*, 2016.
- [27] L. Yeh, "Scaling w state circuits in the qudit clifford hierarchy," in *ACM Companion Proceedings of the 7th International Conference on the Art, Science, and Engineering of Programming*, 2023.
- [28] D. Miller and M. Thornton, "QMDD: A decision diagram structure for reversible and quantum circuits," in *Int'l Symp. on Multi-Valued Logic*, 2006.
- [29] S. Hillmich, A. Zulehner, R. Kueng, I. L. Markov, and R. Wille, "Approximating decision diagrams for quantum circuit simulation," *ACM Transactions on Quantum Computing*, 2022.
- [30] G. F. Viamontes, I. L. Markov, and J. P. Hayes, *Quantum Circuit Simulation*. Springer, 2009.
- [31] P. Niemann, R. Wille, D. M. Miller, M. A. Thornton, and R. Drechsler, "QMDDs: Efficient quantum function representation and manipulation," *IEEE Trans. Comput. Aided Des. Integr. Circuits Syst.*, 2016.
- [32] 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.
- [33] D. M. Greenberger, M. A. Horne, and A. Zeilinger, "Going beyond Bell's theorem," in *Bell's Theorem, Quantum Theory and Conceptions of the Universe*. Springer, 1989.
- [34] A. Cabello, "Bell's theorem with and without inequalities for the three-qubit Greenberger-Horne-Zeilinger and W states," *Phys. Rev. A*, 2002.
- [35] Y.-M. Di and H.-R. Wei, "Synthesis of multivalued quantum logic circuits by elementary gates," *Phys. Rev. A*, 2013.
- [36] W. Zi, Q. Li, and X. Sun, *Optimal synthesis of multi-controlled qudit gates*, 2023. arXiv: 2303.12979 [quant-ph].