

Optimizing Quantum Circuits for Arbitrary State Synthesis and Initialization

Naveed Mahmud, Andrew MacGillivray, Manu Chaudhary, and Esam El-Araby

Electrical Engineering and Computer Science, University of Kansas, Lawrence, USA

{naveed_923, amacgillivray, manu.chaudhary, and esam}@ku.edu

Abstract—For practical usage of quantum systems in solving real-world problems, large amounts of classical data are required to be transferred/encoded to the quantum domain. Arbitrary classical data is usually encoded onto quantum devices by synthesizing and initializing a corresponding quantum state. Current techniques of arbitrary state synthesis, however, produce deep and complex quantum circuits, leading to low state fidelity and possible violations of decoherence constraints. In this work, we propose an improved methodology and optimized circuits for synthesizing any arbitrary quantum state from given classical data. Compared to existing methods, the proposed methodology results in circuits with lower gate count, lower circuit depth, and high state fidelity. The proposed methods are evaluated by simulation in MATLAB and IBM *qasm*, and realistic implementation on an IBM quantum device. The experimental results show a reduction in gate count and circuit depth by a factor of two over existing methods.

Index Terms—Quantum Computing, Quantum Circuits

I. INTRODUCTION

In recent decades, significant progress has been made in the research and development of quantum circuits and algorithms [1]–[3]. While these algorithms expose the potential utility of quantum computing, one of the major challenges preventing quantum circuits and algorithms from serving a practical use is the encoding of classical information into the quantum domain [4]. Generally, when given classical data, a quantum circuit is constructed to synthesize a corresponding initial quantum state, on which further operations of the respective quantum algorithm is performed. The process of initializing the quantum state is known as arbitrary state synthesis or state initialization. On state-of-the-art quantum computers [5]–[9], it is difficult to run complex algorithms with large amounts of data, due to the high costs of the spatial and temporal complexities [10] of the state synthesis and initialization process. The state initialization process can present problems of low fidelity of the synthesized quantum state [11], [12], and even efficient initialization circuits can exceed the decoherence time of many modern NISQ devices [13]. For data-intensive applications such as quantum image processing [14] and quantum machine learning [15], it is vital to develop efficient circuits that can encode input data for the respective quantum algorithms.

Over the years, a number of algorithms and methods have been proposed for arbitrary state synthesis and initialization [16]–[19]. The most efficient methods have a spatial complexity of $O(2^{n+2})$, where n is the number of quantum bits of the corresponding state synthesis circuit. In each work, the

synthesis method has been evaluated by counting the total number of quantum gates (gate count) in the synthesis circuit. However, there has been insufficient emphasis on quantum circuit depth [10] for state synthesis. The circuit depth is defined as the number of gates or time-steps in the longest path of a circuit. The circuit depth is closely related to the temporal complexity [10] and can be used to determine whether a quantum circuit can be run within the decoherence constraints of a particular quantum device [20].

In this work, we propose an efficient methodology for synthesizing and initializing any arbitrary quantum state from given classical data. We present the corresponding optimized synthesis circuits, along with analysis of quantum gate count, and circuit depth. The proposed circuit for arbitrary state synthesis is analysed to have half of the gate counts reported in related work. In addition, the overall depth of the proposed circuit is shown to be 50% lower compared to previously reported methods. We experimentally evaluate the proposed methods by generating synthesis circuits for complex random data and grayscale image data. Accuracy of the state synthesis is evaluated by measuring the quantum state fidelity. Results are obtained from simulations in MATLAB and IBM *qasm* [20], as well as implementations on a real 15-qubit IBM-Q quantum computer [20].

The rest of the paper is organized as follows. Section II contains background concepts and survey of related work. Section III presents the proposed methodology, corresponding quantum circuits, and complexity analysis. Section IV contains experimental results and Section V is the conclusion and future research directions of this work.

II. BACKGROUND AND RELATED WORK

A. Fundamental Concepts in Quantum Computing

1) *Qubits and Superposition*: The quantum bit or *qubit* is the basic unit of information in a quantum computer [4]. The qubit can exist in a linear superposition of its basis states, $|0\rangle$ and $|1\rangle$, defined by the superposition equation:

$$|\psi\rangle \equiv \alpha|0\rangle + \beta|1\rangle \equiv \begin{bmatrix} \alpha \\ \beta \end{bmatrix} \quad (1)$$

where the coefficients α and β are complex numbers such that $|\alpha|^2$ and $|\beta|^2$ are the probabilities that a measurement of the qubit finds it in the basis state $|0\rangle$ or $|1\rangle$, respectively. A qubit can also be represented using a Bloch sphere [4],

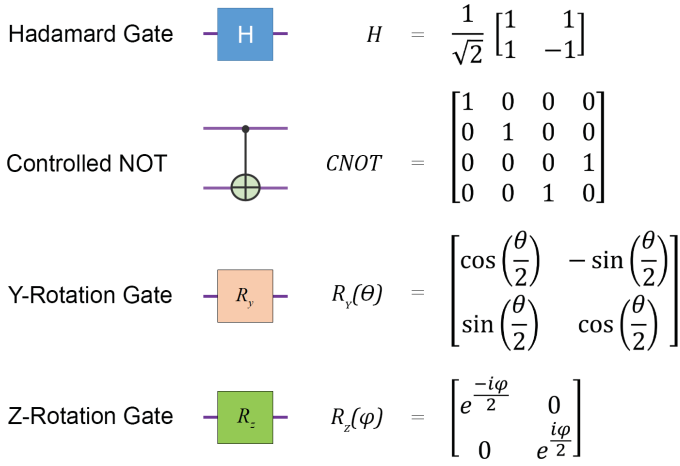


Fig. 1: Elementary quantum gates.

with the quantum state defined by the elevation and azimuth angles θ and ϕ respectively, and a global phase γ [4], see (2). The pair of angles (θ, ϕ) specify a point on the Bloch sphere that represents a pure quantum state of the qubit. For a single qubit, the global phase factor $e^{i\gamma}$ is unobservable and is usually ignored in quantum operations [4].

$$|\psi\rangle = e^{i\gamma} \left(\cos \frac{\theta}{2} |0\rangle + e^{i\phi} \sin \frac{\theta}{2} |1\rangle \right) \quad (2)$$

2) *Entanglement:* Entanglement is a unique and fundamental property of qubits. Mathematically, entanglement means that the quantum state of a qubit cannot be factored into a tensor product of the individual qubits, i.e., $|\psi\rangle = |q_{n-1}q_{n-2}\dots q_1q_0\rangle \neq |q_{n-1}\rangle \otimes |q_{n-2}\rangle \otimes \dots \otimes |q_1\rangle \otimes |q_0\rangle$. Entanglement is useful for quantum computing, because it allows operations and measurements of one qubit to affect or give information about the state of other entangled qubits [4].

3) *Decoherence:* Interactions with the environment affect the states of qubits resulting in a phenomenon known as quantum decoherence [21]. Factors including heat, radiation, magnetic and electric fields, etc. cause information to be lost and the qubit's state to become increasingly mixed [4], [21]. As time progresses past a certain limit (usually called the decoherence time constant), further quantum interference among qubits is suppressed which prevents further operations from being conducted. Ensuring that all operations can be executed within the decoherence time is a critical part of designing an optimized quantum algorithm or circuit.

B. Quantum Gates

In circuit model of quantum computing [4], a quantum gate represents a transformation on one or more qubits. Every quantum gate has a corresponding transformation matrix. Some commonly used elementary gates and their corresponding circuit symbols and matrices are shown in Fig. 1. We summarize the operations of these gates in the following section.

1) *Hadamard Gate:* The Hadamard (H) gate acts on a single qubit, where it maps the computational basis states into superposition states and vice versa. When the Hadamard gate H acts on a computational basis state $|x\rangle$, it transforms the input according to $H|x\rangle = \frac{1}{\sqrt{2}} (|0\rangle + (-1)^x|1\rangle)$ [4].

2) *Controlled-NOT Gate:* The Controlled-NOT (CNOT) gate is a 2-qubit gate, with one control qubit and one target qubit. The CNOT gate inverts the target qubit if and only if the control qubit is set to $|1\rangle$ [4].

3) *Rotation Gates:* A y -rotation or $R_y(\theta)$ gate maps a state $|\psi\rangle$ to a new state $R_y(\theta)|\psi\rangle$ represented in the Bloch sphere by a rotation of an angle $\frac{\theta}{2}$ around y -axis [4]. A z -rotation $R_z(\phi)$ gate maps a state $|\psi\rangle$ to a new state $R_z(\phi)|\psi\rangle$ represented in the Bloch sphere by a rotation of an angle $\frac{\phi}{2}$ around z -axis [4].

C. Related Work

Song and Williams in [16] presented methodologies for synthesizing any n -qubit pure state or mixed state. For synthesizing a pure state, their algorithm involves first applying Gram-Schmidt procedure on a matrix that contains the input data as the leftmost column, to produce a unitary matrix. The unitary matrix is then synthesized to a quantum circuit using a recursive algebraic method that has a complexity of $O(2^{2n})$ [4], where n is the number of qubits.

The authors in [17] presented transformations for one arbitrary state $|a\rangle$ to another $|b\rangle$ using uniformly controlled rotations. From their presented circuit transformation from $|a\rangle$ to $|b\rangle$, it can be inferred that transformation from $|a\rangle$ to $|0\rangle$ (or to any basis state) would require half the reported gate count. No analysis of circuit depth was provided in their work. To compare with our proposed circuits, we considered their circuit transforming state $|a\rangle$ to $|0\rangle$ and calculated the corresponding gate count and circuit depth to be $2^{n+2} - 6$.

The work in [18] presented a method based on disentangling a qubit, i.e., producing a basis state $|0\rangle$ or $|1\rangle$ on the lowest significant qubit. The authors state that this disentangling method, which requires $2^n - 2$ CNOTs for an n -qubit circuit, can be used recursively to transform any state to a desired basis state. They reported that the resulting final transformation circuit uses $2^{n+1} - 2n$ CNOT gates, however, no detailed analysis was provided. Furthermore, only the CNOT gate count was provided, while their proposed circuit also requires single-qubit rotation gates that would double the total gate count to $2^{n+2} - 2n$.

In [19] the proposed methodology is based on applying $(n - 1)$ rotation steps with permutations on the amplitudes in-between each rotation where additional gates are required in the intermediate permutations. The total gate count reported is $2^{n+2} + 4n - 9$ with no circuit representation of their methodology. To be consistent with our analyses, we calculated their circuit depth to be $2^{n+2} + 3n - 8$.

In this paper, we propose a methodology for arbitrary state synthesis that results in a lower gate count than previous methods. We present the analytic expression for circuit gate depths that were not considered in prior work. We also present

the full and optimized quantum circuits corresponding to our methodology, and experimentally evaluate our circuits using simulation as well as implementation on a real quantum device. In addition, the state fidelity of the proposed circuits is reported for the simulations on IBM *qasm*.

III. PROPOSED METHODOLOGY AND OPTIMIZED CIRCUITS

Our proposed methodology for arbitrary circuit synthesis involves the following. The classical data to be encoded is normalized as the coefficients of a quantum state vector [4]. For each pair of coefficients, the corresponding parameters necessary to produce the coefficients from ground state are calculated. These parameters are used in a conditional quantum logic circuit [16] that will produce the desired quantum state vector. To construct the conditional quantum logic circuit, we use Hadamard gates and uniformly-controlled y and z rotation operations. Further optimizations are applied to decompose each multi-controlled rotation. Details of each step are elaborated upon in the next subsections.

A. Data Encoding

A quantum register of n qubits that are in ground state is given by $|\psi_0\rangle = |0\rangle^{\otimes n}$. Given a classical data set of $N = 2^n$ elements, we arrange it as an $N \times 1$ column vector for encoding onto a target input quantum state given by $|\psi\rangle$ in (3).

$$\begin{aligned} |\psi\rangle &= |q_{n-1}\rangle \otimes |q_{n-2}\rangle \otimes \dots \otimes |q_1\rangle \otimes |q_0\rangle \\ &= |q_{n-1}q_{n-2}\dots q_1q_0\rangle = \sum_{i=0}^{N-1} \alpha_i |i\rangle \end{aligned} \quad (3)$$

where α_i are the basis state coefficients of the input quantum state $|\psi\rangle$ and form the elements of the $N \times 1$ data column vector. For state initialization or synthesizing the initial arbitrary state, it is required to find a quantum circuit, U_{init} , that transforms $|\psi_0\rangle$ to $|\psi\rangle$, i.e., $|\psi\rangle = U_{init} |\psi_0\rangle$.

B. The Pauli Decomposition

Any arbitrary single-qubit gate can be decomposed as a series of R_z and R_y gates known as the ZYZ or Pauli decomposition [4], [18]. Therefore, a qubit in ground state $|0\rangle$ can be taken to any arbitrary state $|\psi\rangle$ by applying a θ rotation about y -axis followed by a ϕ rotation about z -axis:

$$|\psi\rangle = R_z(\phi) \cdot R_y(\theta) \cdot r e^{i\frac{t}{2}} \cdot |0\rangle \quad (4)$$

where $r e^{i\frac{t}{2}}$ is the unobservable global phase factor for the single qubit [4]. If the coefficients of the target state $|\psi\rangle$ are α and β , such that $|\psi\rangle = \begin{bmatrix} \alpha \\ \beta \end{bmatrix}$ and $|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$, then the parameters r , t , θ , and ϕ for the transformation given in (4) could be determined by substituting the transformation matrices of R_z , R_y , see Fig.1, in (4), and are given by:

$$\begin{aligned} r &= \sqrt{|\alpha|^2 + |\beta|^2}, \quad t = \angle\beta + \angle\alpha \\ \theta &= 2 \tan^{-1} \left(\frac{|\beta|}{|\alpha|} \right), \quad \phi = \angle\beta - \angle\alpha \end{aligned} \quad (5)$$

where,

$$\begin{aligned} |\alpha| &= \sqrt{\text{Re}^2(\alpha) + \text{Im}^2(\alpha)}, \quad \angle\alpha = \tan^{-1} \left(\frac{\text{Im}(\alpha)}{\text{Re}(\alpha)} \right), \\ |\beta| &= \sqrt{\text{Re}^2(\beta) + \text{Im}^2(\beta)}, \quad \angle\beta = \tan^{-1} \left(\frac{\text{Im}(\beta)}{\text{Re}(\beta)} \right) \end{aligned}$$

C. Conditional Quantum Logic Circuit

Using the Pauli decomposition described by (4) and the parameters obtained by (5), we derive a method for transforming an n -qubit register in the ground state $|\psi_0\rangle = |0\rangle^{\otimes n}$ to an arbitrary state $|\psi\rangle$, see Fig. 2. To synthesize the j^{th} pair of coefficients, or $|\psi_j\rangle$ in the state vector of $|\psi\rangle$, we apply U_j on a ground state $|0\rangle$, where $j = 0, 1, 2, \dots, (2^{n-1}-1)$. However, we cannot apply U_j on one qubit in the n -qubit register and realize the desired pair of coefficients without also affecting the other coefficients in $|\psi\rangle$. Hence, each transformation U_j needs to be applied conditionally to synthesize the j^{th} pair of coefficients in the output state. The resulting conditional quantum circuit can be represented by a block-diagonal matrix U , of which each diagonal block is a 2×2 transformation matrix U_j , see Fig. 2 and (6). The elements of U_j are calculated using the parameters r_j , t_j , θ_j , and ϕ_j obtained from the j^{th} pair of coefficients using (5).

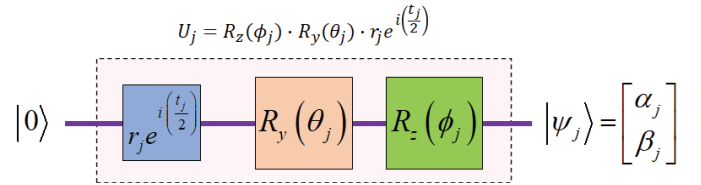
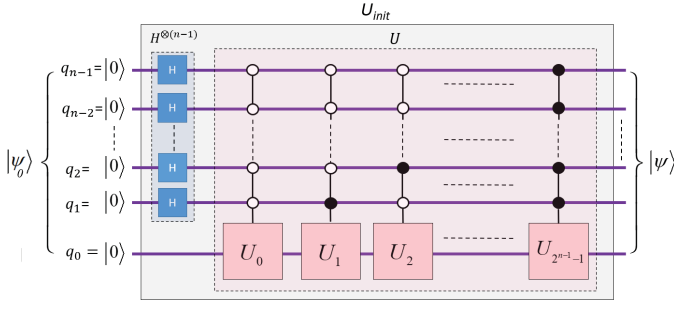


Fig. 2: Pauli decomposition for single-qubit state synthesis.

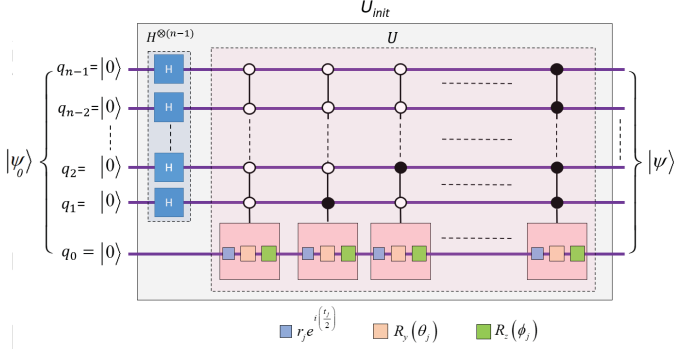
$$\begin{aligned} U &= U_0 \oplus U_1 \oplus \dots \oplus U_j \dots \oplus U_{(2^{n-1}-1)} \\ &= \text{diag}(U_0, U_1, \dots, U_j, \dots, U_{(2^{n-1}-1)}) \end{aligned} \quad (6)$$

A block-diagonal matrix such as U can be implemented as a quantum multiplexer [16], [18] with n qubits of which $(n-1)$ control qubits acting on the least significant target qubit. The corresponding circuit is shown in Fig. 3a. For each combination of the control qubits, the corresponding U_j is applied on the target qubit, where $j = 0, 1, 2, \dots, (2^{n-1}-1)$. To produce all combinations on the control qubits with equal probability, a set of H gates must be applied on the $(n-1)$ control qubits before applying the U transformation. The desired final state $|\psi\rangle$ is produced at the output with the target coefficients as a result of uniformly applying each U_j transformation on the least significant qubit. The overall transformation, U_{init} , from ground state $|\psi_0\rangle = |0\rangle^{\otimes n}$ to $|\psi\rangle$ can be expressed by (7).

$$\begin{aligned} |\psi\rangle &= U_{init} \cdot |\psi_0\rangle = U_{init} \cdot |0\rangle^{\otimes n}, \quad \text{where} \\ U_{init} &= (\sqrt{2})^{(n-1)} \cdot U \cdot (H^{\otimes(n-1)} \otimes I), \quad \text{and} \\ I &\text{ is a } 2 \times 2 \text{ identity matrix} \end{aligned} \quad (7)$$



(a) Quantum circuit for arbitrary state synthesis.



(b) Each U_j transformation can be factored into $r_j e^{i(t_j/2)}$, $R_y(\theta_j)$, and $R_z(\phi_j)$.

Fig. 3: Conditional logic based quantum circuit for arbitrary state synthesis. The white and black circles on the control qubits represent bit values of zero and one respectively.

Each U_j block is a sequence of a phase multiplication, followed by y -rotation, followed by z -rotation as shown in Fig. 3b, and U_j is calculated from the corresponding set of parameters $\{r_j, t_j, \theta_j, \phi_j\}$ obtained by (5). Since each set of operations are mutually exclusive from each other, we can separate them into uniformly controlled groups of phase multipliers, y -rotations, and z -rotations as shown in Fig. 4.

To represent uniformly controlled operations as a single gate operation, we use a notation previously used in [18], where the sequence of different combinations on the control qubits are replaced with a ‘square box’ notation indicating multi-control, and the parameterized operations for each combination are replaced by a single box denoting the general operation. We use this notation to simplify the circuit in Fig. 4 and the resulting circuit representation is shown in Fig. 5.

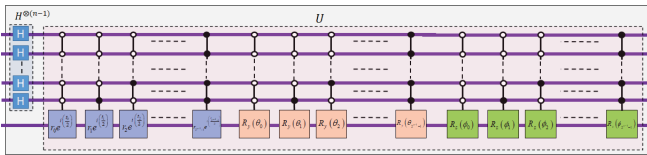


Fig. 4: Expanded full quantum circuit for arbitrary state synthesis.

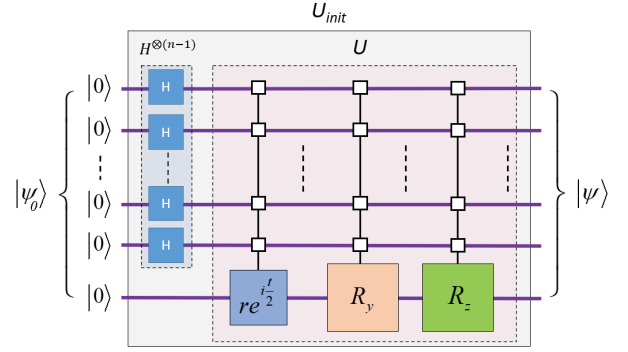


Fig. 5: Simplified full quantum circuit for arbitrary state synthesis with uniformly controlled operations.

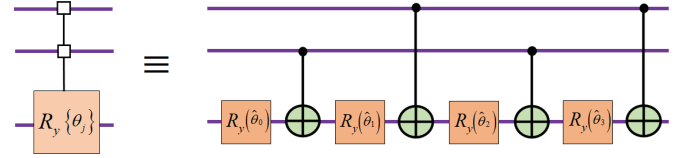


Fig. 6: Decomposition of a uniformly controlled 3-qubit R_y rotation operation.

D. Decomposition of Uniformly Controlled Rotation Circuits

The uniformly controlled R_y and R_z rotation operations in Fig. 5 can be decomposed into a sequence of CNOT and one-qubit rotation gates. A systemic decomposition method was presented in [22] which we leverage for our methodology. The method involves taking the binary reflected Gray code of the control bit sequence to determine the control qubit positions of the CNOT gates. As a demonstrative example, the decomposition for a 3-qubit controlled R_y operation with rotation angles $\{\theta_j\}$ is shown in Fig. 6. To calculate the new set of rotation angles $\{\hat{\theta}_j\}$ for the one-qubit rotations in Fig. 6, a transformation matrix $M_{ij}^k = (-1)^{b_{i-1} \cdot g_{j-1}}$ was formulated in [22]. The exponent of this matrix is the bit-wise inner product of the binary vectors for standard binary representation, b_{i-1} , and Gray code representation g_{j-1} . Applying the inverse of M_{ij}^k on the vector of angles $\{\theta_j\}$ consequently produces a vector of angles $\{\hat{\theta}_j\}$. The decomposition for one uniformly controlled rotation operations takes 2^n gates (2^{n-1} CNOTs and 2^{n-1} one-qubit rotations) in total [18], [19], [22]. We apply this decomposition for the uniformly controlled R_y and R_z rotation operations in Fig. 5.

E. Spatial and Temporal Complexity Analysis

The spatial complexity of a quantum circuit could be determined by considering the total number of gates used in the circuit construction. For our proposed n -qubit arbitrary state initialization circuit, see Fig. 5, we require $(n-1)$ H gates for creating superposition, and 2^n gates (2^{n-1} CNOTs and 2^{n-1} one-qubit rotations) for the R_y and R_z uniformly controlled operations respectively. A slight optimization [18] can be applied, where the order of the gates in the R_y rotation is reversed so that its first CNOT cancels the last CNOT from the

R_z rotation circuit. The controlled $re^{i\frac{\pi}{2}}$ operation represents different scalar multiplications of the global phase factors for every value of the select qubits, and can be implemented as a diagonal operator [18] acting on the select qubits. In earlier studies, the global phase factor was usually not considered in the circuit complexity as it is physically unobservable or undetectable. The total gate count G_{total} for our optimized circuit is given by (8). The temporal complexity of the circuit can be analyzed by considering the circuit depth (number of time-steps). For our final circuit in Fig. 5, the H gates can be applied in parallel in one step with the first R_y rotation gate, and therefore do not contribute any additional depth. The R_y and R_z uniformly controlled operations combined consist of $2^n - 2$ CNOT gates and 2^n one-qubit rotation gates, each of which contribute a single unit in the longest circuit path. The overall circuit depth d_{total} is given by (8).

$$G_{total} = 2^{n+1} + n - 3, \quad d_{total} = 2^{n+1} - 2 \quad (8)$$

For applications with real data, the encoded state coefficients will be real numbers. Since they have no imaginary component, the R_z rotations become identity operations and therefore do not contribute to the gate depth. The total gate count for such applications is reduced to $2^n + n - 3$ and the circuit depth is reduced to $2^n - 2$.

TABLE I: Comparison of proposed method to current methods for arbitrary state synthesis.

Method	Gate Count	Circuit Depth
Mottonen [17], 2004	$2^{n+2} - 6$	$2^{n+2} - 6$
Shende [18], 2006	$2^{n+2} - 2n$	$2^{n+2} - 2n$
Niemann [19], 2016	$2^{n+2} + 4n - 9$	$2^{n+2} + 3n - 8$
Proposed	$2^{n+1} + n - 3$	$2^{n+1} - 2$

We analyzed the complexities of the methods presented in prior works related to arbitrary state synthesis. A quantitative comparison of those methods with our proposed work, in terms of the theoretical gate counts and circuit depth, is shown in Table I. In general, the previous methods proposed using uniformly controlled rotation operations recursively to disentangle each qubit, which results in larger gate count and depth. In our proposed method, in combination with Hadamard gates on the select qubits, we can synthesize the desired state using a single set of uniformly controlled R_z and R_y rotations on the target qubit. This results in the reduction of gate count and circuit depth by at least a factor of two, see Table I.

IV. EXPERIMENTAL WORK

Our proposed circuits have been modelled and verified using two highly-productive and easy-to-use environments, i.e., MATLAB for noise-free qubits, and IBM-Q [20] for noisy qubits on a real Noisy Intermediate-Scale Quantum (NISQ) device. Our methods are also portable to other devices/platforms [6]–[9]. In the IBM-Q environment, simulation was performed using the IBM *qasm* simulator, while real










TABLE II: Simulation and implementation of proposed synthesis circuits using IBM-Q.

	Number of data items (N)	Number of qubits (n)	Simulation (MATLAB / ibmq_qasm)		Implementation (ibmq_16_melbourne)	
			Gate Count	Circuit Depth	Gate Count	Circuit Depth
Complex randomized data	4.00E+00	2	7.00E+00	6.00E+00	1.70E+01	1.50E+01
	1.60E+01	4	3.30E+01	3.00E+01	9.20E+01	7.20E+01
	6.40E+01	6	1.31E+02	1.26E+02	3.38E+02	2.76E+02
	2.56E+02	8	5.17E+02	5.10E+02	NA due to hardware limitations of IBM-Q	
	1.02E+03	10	2.06E+03	2.05E+03		
	4.10E+03	12	8.20E+03	8.19E+03		
Image data	1.64E+04	14	3.28E+04	3.28E+04	NA due to hardware limitations of IBM-Q	
	2 x 2 pixels	2	3.00E+00	2.00E+00	1.30E+01	1.10E+01
	4 x 4 pixels	4	1.70E+01	1.40E+01	5.80E+01	4.70E+01
	8 x 8 pixels	6	6.70E+01	6.20E+01	2.38E+02	1.97E+02
	16 x 16 pixels	8	2.61E+02	2.54E+02	8.71E+02	7.46E+02
	32 x 32 pixels	10	1.03E+03	1.02E+03	NA due to hardware limitations of IBM-Q	
	64 x 64 pixels	12	4.11E+03	4.09E+03		
	128 x 128 pixels	14	1.64E+04	1.64E+04		

implementation was performed on the 15-qubit real quantum processor, *ibmq_16_melbourne*. We tested the synthesis of two types of target data: (1) complex randomized data, and (2) real grayscale image data. The experimental results are presented in Table II. For complex randomized data, the gate count and circuit depth reach the theoretical upper bounds derived earlier in (8) as the full synthesis circuit is required. For real image data, the gate counts and circuit depths of the circuits were reduced by at least a factor of two, as there are no imaginary components ($t_j = \phi_j = 0$) in the data, and thus both the uniformly-controlled R_z operations, i.e., $R_z(\phi_j) = I$, and their corresponding CNOT operations are eliminated. Results obtained from *ibmq_qasm* simulations of up to 14-qubit circuits were consistent with our theoretical expectations for gate count and circuit depth, see Table I and Table II.

Due to hardware constraints for the *ibmq_16_melbourne* device [20], gate counts and circuit depths were obtained for circuits up to only 6 qubits (complex randomized data) and 8 qubits (real image data). Several of the gates used in our proposed circuit, such as H , CNOT and R_y , are not physically realizable on the *ibmq_16_melbourne* device and are instead replaced in a transpilation process using a different subset of universal gates that are native to the IBM-Q platform. The transpilation step resulted in higher gate counts and circuit depths for the implementations, compared to our theoretical expectations, see Table I and Table II. For larger data sets that require a large number of qubits, and consequently larger synthesis circuits, the system decoherence time on *ibmq_16_melbourne* was exceeded [20], limiting implementations to only 6 qubits (complex randomized data) and 8 qubits (real image data). For simulations and implementations on IBM-Q, the circuits were executed with 8000 shots (iterations) to measure the probability distributions of the output states.

To verify the correctness of the proposed methodology and circuits, the encoded images were reconstructed from the synthesized state coefficients and the fidelity of the synthesized state was calculated. The state fidelity is a measure for the similarity of the measured output state $|\psi_{measured}\rangle$, observed in simulation or implementation, to the theoretical or expected state $|\psi_{expected}\rangle$. The Uhlmann-Jozsa fidelity for pure states

Original image	Reconstructed image	
	MATLAB (noise-free)	IBM-QASM (NISQ [†] devices)
 16 x 16 pixels (8 qubits)		
	Fidelity = 100 %	Fidelity = 99.1644 %
 32 x 32 pixels (10 qubits)		
	Fidelity = 100 %	Fidelity = 96.5429 %
 64 x 64 pixels (12 qubits)		
	Fidelity = 100 %	Fidelity = 94.0894 %

[†]NISQ = Noisy Intermediate-Scale Quantum

Fig. 7: Original and reconstructed images from synthesized quantum states.

[11], [12], given in (9), is used for our experiments.

$$F = |\langle \psi_{expected} | \psi_{measured} \rangle|^2 \quad (9)$$

Figure 7 shows 16×16 , 32×32 , and 64×64 -pixel grayscale images encoded using 8-qubit, 10-qubit, and 12-qubit synthesis circuits respectively in both MATLAB and IBM-Q. The reconstructed images from the synthesized state are also shown along with the corresponding state fidelity between the original data and the reconstructed data. When the images were encoded as pure states using noise-free qubits in MATLAB, the reconstructed images were identical to the original images, i.e., $F = 100\%$, see Fig. 7. For simulation on realistic Noisy Intermediate Scale Quantum (NISQ) devices, such as the *ibmq_16_melbourne*, the reconstructed images were partially corrupted by device noise. The state fidelity between the original data and the reconstructed data was 99.1644%, 96.5429%, and 94.0894% for 16×16 , 32×32 , and 64×64 -pixel images respectively, see Fig. 7.

V. CONCLUSIONS AND FUTURE WORK

To efficiently utilize quantum computers for real-world applications, it is important to investigate efficient methods of encoding classical data on to the quantum system. In this work we presented a methodology for synthesizing and initializing any arbitrary quantum state from given classical data. We presented full and optimized synthesis circuits and provided its spatial and temporal complexity analysis. Compared to existing methods, the circuit gate count and circuit depth are improved by at least a factor of 2. We experimentally evaluated the proposed methods using simulation on classical platforms as well as implementation on a real quantum device.

The experimental results show that the proposed methodology correctly and efficiently encodes the classical data into synthesized quantum states at a high degree of state fidelity.

Our future work will include further analysis of the physical constraints and limitations posed by currently available NISQ devices. Quantum Error Correction (QEC) techniques will be investigated and added to our methodology to reduce the noise effects of NISQ devices. The applicability of our proposed method for arbitrary state synthesis could be demonstrated by integrating it with real-world quantum algorithms such as quantum search, quantum wavelet transform, and quantum machine learning.

REFERENCES

- [1] Lov K Grover. Quantum mechanics helps in searching for a needle in a haystack. *Physical review letters*, 79(2):325, 1997.
- [2] Yaakov S Weinstein, MA Pravia, EM Fortunato, Seth Lloyd, and David G Cory. Implementation of the quantum fourier transform. *Physical review letters*, 86(9):1889, 2001.
- [3] Peter W Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM review*, 41(2):303–332, 1999.
- [4] Colin P Williams. *Explorations in quantum computing*. Springer Science & Business Media, second edition, 2011.
- [5] Nikitas Stamatopoulos, Daniel J Egger, Yue Sun, Christa Zoufal, Raban Iten, Ning Shen, and Stefan Woerner. Option pricing using quantum computers. *Quantum*, 4:291, 2020.
- [6] Google AI Quantum et al. Hartree-fock on a superconducting qubit quantum computer. *Science*, 369(6507):1084–1089, 2020.
- [7] Carmen G Almudever, Lingling Lao, Robert Wille, and Gian G Guerreschi. Realizing quantum algorithms on real quantum computing devices. In *2020 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, pages 864–872. IEEE, 2020.
- [8] IonQ. Scaling ionq’s quantum computers: The roadmap.
- [9] Microsoft Quantum Team. Developing a topological qubit.
- [10] John Watrous. Quantum computational complexity. *arXiv preprint arXiv:0804.3401*, 2008.
- [11] Armin Uhlmann. The “transition probability” in the state space of a-algebra. *Reports on Mathematical Physics*, 9(2):273–279, 1976.
- [12] Richard Jozsa. Fidelity for mixed quantum states. *Journal of modern optics*, 41(12):2315–2323, 1994.
- [13] Daniel Koch, Brett Martin, Saahil Patel, Laura Wessing, and Paul M Alsing. Demonstrating nisq era challenges in algorithm design on ibm’s 20 qubit quantum computer. *AIP Advances*, 10(9):095101, 2020.
- [14] Xi-Wei Yao, Hengyan Wang, Zeyang Liao, Ming-Cheng Chen, Jian Pan, Jun Li, Kechao Zhang, Xingcheng Lin, Zhehui Wang, Zhihuang Luo, et al. Quantum image processing and its application to edge detection: Theory and experiment. *Physical Review X*, 7(3):031041, 2017.
- [15] Jacob Biamonte, Peter Wittek, Nicola Pancotti, Patrick Rebentrost, Nathan Wiebe, and Seth Lloyd. Quantum machine learning. *Nature*, 549(7671):195–202, 2017.
- [16] Lin Song and Colin P Williams. Computational synthesis of any n-qubit pure or mixed state. In *Quantum Information and Computation*, volume 5105, pages 195–203. International Society for Optics and Photonics, 2003.
- [17] Mikko Möttönen, Juha J Vartiainen, Ville Bergholm, and Martti M Salomaa. Transformation of quantum states using uniformly controlled rotations. *Quant. Inf. Comp.*, 5(467), 2004.
- [18] Vivek V Shende, Stephen S Bullock, and Igor L Markov. Synthesis of quantum-logic circuits. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 25(6):1000–1010, 2006.
- [19] Philipp Niemann, Rhitam Datta, and Robert Wille. Logic synthesis for quantum state generation. In *2016 IEEE 46th International Symposium on Multiple-Valued Logic (ISMVL)*, pages 247–252. IEEE, 2016.
- [20] IBM. Get started with ibm quantum experience. <https://quantum-computing.ibm.com/docs/>. Accessed: 2021-07-30.
- [21] Maximilian Schlosshauer. Quantum decoherence. *Physics Reports*, 831:1–57, 2019.
- [22] Mikko Möttönen and Juha J Vartiainen. Decompositions of general quantum gates. *Trends in Quantum Computing Research*, 2006.