Journal of Electronics and Informatics (ISSN: 2582-3825)
www.irojournals.com/iroei/

Quantum Speedup for Linear Systems: An Analysis of the HHL Algorithm Using IBM Qiskit

Shwetha V.[1], Abinaya Selvarajan[2], Aarthi A.[3], Sneka R.[4]

[1-4]Department of Electronics Engineering, Madras Institute of Technology, Anna University, Chennai, India

E-mail: [1]vshwetha999@gmail.com, [2]abinayaselvarajan2002@gmail.com, [3]aarthitamil28@gmail.com, [4]sneka429@gmail.com

Abstract

One of the most significant developments in quantum computing is the Harrow-Hassidim-Lloyd (HHL) method, which can solve linear equation systems at exponential speedup. Because linear systems are essential to many scientific fields, including physics, engineering, and machine learning, this approach has great potential to revolutionize computational paradigms. The HHL algorithm is thoroughly examined in this work, with particular attention paid to its theoretical framework, real-world application utilizing IBM's Qiskit platform, and the difficulties in simulating quantum algorithms on noisy intermediate-scale quantum (NISQ) devices. Using parameters like fidelity, time complexity, and scalability, the research further evaluates the HHL algorithm's performance in comparison to traditional methods. According to the findings, quantum simulations work well for small-scale matrices like 2x2 and 4x4, but expanding the approach to bigger systems is still difficult because of hardware and software constraints. Finally, the research emphasizes the key directions for advancing quantum hardware and algorithms to overcome current scalability challenges, enabling broader applicability of the HHL algorithm in solving complex linear systems.

Keywords: HHL Algorithm, Quantum Computing, Linear Systems of Equations, Quantum Phase Estimation, IBM Qiskit, Noisy Intermediate-Scale Quantum Devices, Fidelity Analysis.

DOI: https://doi.org/10.36548/jei.2024.4.003
Received: 17.11.2024, received in revised form: 08.12.2024, accepted: 29.12.2024, published: 09.01.2025
Extended version received from International Conference on Engineering and Technology, 2024: ***MACET, Kanyakumari.***

1. Introduction

With the potential to transform problem-solving in domains where classical computing is constrained by exponential time complexity, the science of quantum computing has attracted enormous attention in recent years. A common challenge in many scientific, technical, and financial applications is the solving of linear systems of equations. Effectively resolving these systems can result in advances in the analysis of big datasets, process optimization, and modeling of intricate phenomena. Conventionally, linear systems of equations are solved using classical algorithms that do not scale well with the size of the issue. Under some circumstances, iterative techniques like conjugate gradient have time complexities of $O(N^2)$, but direct techniques like Gaussian elimination have time difficulties of $O(N^3)$. When working with big datasets or high-dimensional systems, this exponential scaling becomes unaffordable.

1.1 Quantum Computing:A Revolutionary Approach to Linear Systems

Quantum computing provides a radically different method. The ability of quantum algorithms to manipulate superpositions of states enables them to analyze enormous volumes of data at once. The Harrow-Hassidim-Lloyd (HHL) algorithm was one of the earliest quantum algorithms created to solve linear equation systems. It was initially presented in 2009. The HHL algorithm claims to lower the time complexity of solving linear systems to $O(\log N)$ by utilizing quantum phase estimates and controlled rotations. Under ideal circumstances, this would result in an exponential speedup. HHL is especially appealing for applications in domains where big equation systems are frequently encountered, like differential equation solvers, quantum machine learning, and cryptography, because of this speedup. Due to limitations in current quantum hardware, the HHL algorithm is still difficult to implement in practice, despite its theoretical promise. The algorithm is hard to scale beyond small matrices because of the noise in quantum systems, the complexity of quantum gates, and the number of qubits needed. This study examines the HHL algorithm's theoretical foundations, provides an implementation utilizing IBM's Qiskit platform, and evaluates the algorithm's effectiveness on small systems. The difficulties that arise during simulation, such as software incompatibilities, import errors, and fidelity problems, are discussed, and solutions are suggested.

2. Related Work

The solution of linear systems of equations, a basic issue in many scientific fields, could be completely transformed by quantum computing. Under favourable circumstances, iterative techniques like conjugate gradient reduce the time complexity of solving these systems to $O(N^2)$, although classical algorithms like Gaussian elimination usually solve them with time complexity scaling as $O(N^3)$. On the other hand, the Harrow-Hassidim-Lloyd (HHL) algorithm, which was first presented by Harrow et al. in 2009 [1], offers an exponential speedup and, in the best-case scenario, solves linear systems in O(logN) time. Because of its potential use in computationally demanding domains like quantum machine learning and large-scale data analysis, the HHL algorithm has now grown to be one of the most studied quantum algorithms. The HHL algorithm solves the linear system by effectively computing a matrix's eigenvalues through quantum phase estimation (QPE). This method simultaneously explores huge solution areas by taking advantage of the intrinsic qualities of quantum systems, such as entanglement and superposition. HHL provides significant speedups over traditional solvers as a result, especially for sparse matrices where traditional algorithms have trouble keeping up with the computing demands.

HHL has been shown to have promise in both theoretical and experimental settings. The foundation for HHL's quantum advantage was established by Harrow et al. [1], who demonstrated its exponential speedup for large, sparse matrices. Using a photonic quantum processor, Cai et al. [2] achieved the first experimental realization of the HHL algorithm, solving a 2x2 system with great fidelity. Their research has shown that while scaling to bigger systems remained difficult because of hardware constraints and quantum noise, quantum computing could provide workable solutions for small-scale linear systems.

Pan et al. [3] extended this research by implementing the HHL algorithm on a superconducting quantum processor. They explored both 2x2 and 4x4 matrices, achieving reasonable fidelity for small systems but encountering significant errors as the matrix size and complexity increased. Their work highlighted the importance of matrix sparsity and condition number, with sparse, well-conditioned matrices yielding the best results in quantum experiments.

The practical implementation of the HHL algorithm, however, faces considerable challenges, primarily due to the limitations of current quantum hardware. Ji and Meng [4] conducted a detailed study of HHL on IBM’s Qiskit platform, demonstrating that while small matrices (e.g., 2x2) could be solved with high fidelity, the fidelity rapidly degraded as matrix size increased to 4x4. They identified quantum gate errors and noise in NISQ (Noisy Intermediate-Scale Quantum) devices as the primary obstacles to scaling the algorithm. Their findings align with earlier works by Cai et al. [2] and Pan et al. [3], both of which emphasize the difficulty of implementing HHL on larger matrices due to noise and gate fidelity issues.

In addition to hardware challenges, software limitations also hinder the widespread application of the HHL algorithm. Qiskit, the primary development platform for quantum algorithms, has undergone rapid development, leading to frequent updates that sometimes introduce incompatibilities between versions. Ji and Meng [4] noted issues with importing necessary modules, resulting in errors during implementation. This has been a common issue for quantum developers, who often face compatibility problems as quantum software platforms evolve.

Comparative studies between quantum and classical solvers provide further insights into the current capabilities of quantum algorithms. Adedoyin et al. [5] examined the performance of HHL in comparison with classical solvers such as Gaussian elimination and conjugate gradient methods. Their study concluded that while HHL holds theoretical advantages, classical methods often outperform quantum solvers in practical scenarios, particularly for dense matrices where quantum noise significantly impacts fidelity. However, for well-conditioned, sparse matrices, HHL offers a promising alternative that could outperform classical solvers as quantum hardware continues to improve.

Research by Barz et al. [6] investigated the use of HHL in solving machine learning problems, particularly in the context of linear regression. Their work demonstrated that HHL could provide a quantum advantage in regression analysis, but only under certain conditions, such as when dealing with high-dimensional, sparse datasets. For poorly conditioned matrices or dense systems, classical regression methods remain more reliable, given the current state of quantum hardware.

In 2024, Morgan et al. [7] proposed a hybrid modification to the HHL algorithm, integrating enhanced quantum eigenvalue estimations with refined classical processing techniques. This combination led to improved accuracy when solving linear systems, particularly in cases of sparse and well-conditioned matrices. Their approach highlights potential strategies for enhancing the performance of the HHL algorithm on current quantum hardware, which is especially relevant to the goals of this research.

Ginzburg et al. [8] conducted a study in 2024 focusing on the convergence of error in quantum linear system solvers. Their research addressed the behavior of the HHL algorithm under the influence of quantum noise, and identified the key factors that affect the scalability and precision of the method. Their findings are significant to this research, as they provide insights into overcoming challenges with fidelity in the implementation of HHL on IBM Qiskit.

Tsemo et al. [9] in 2024 introduced a modified version of the HHL algorithm known as Psi-HHL, designed to tackle the challenges posed by large condition numbers in linear systems. Their work showed that this improved algorithm works more efficiently on well-conditioned quantum systems, especially in situations where traditional HHL methods face difficulties. This research is relevant to the research's focus on evaluating the efficiency of HHL for sparse matrices using Qiskit.

Kumar et al. [10] explored the quantum resources necessary for implementing the HHL algorithm in a study published in 2024. Their research looked into the requirements for entanglement and coherence, offering suggestions for optimizing quantum resource utilization and overcoming practical limitations. This is closely aligned with the objectives of the research, which aims to assess the resources required to implement HHL effectively.

Duan et al. [11] re-evaluated the performance of the HHL algorithm in the context of quantum machine learning. They proposed modifications to improve its efficiency under real-world conditions, suggesting that while HHL shows promise, practical limitations must be addressed. This work supports the research's goal of optimizing HHL for real-world applications.

Qinghai et al. [12], explored the use of the HHL algorithm in solving linear systems in finance, particularly for portfolio optimization. Their work demonstrated the potential of HHL to solve complex financial equations efficiently. This highlights the versatility of the algorithm and its relevance to the research, which aims to explore its use in diverse real-world scenarios.

Xiaonan et al. [13] conducted a thorough review of the HHL algorithm's implementation and effectiveness. They focused on emerging research directions and the advancements needed to overcome current hardware limitations. This aligns with the research's focus on identifying challenges in scaling up HHL and implementing it on quantum platforms like IBM Qiskit.

Angara et al. [14] proposed a hybrid quantum-classical approach for the implementation of the HHL algorithm on NISQ devices. Their study showed promising results when using small-scale systems, bridging the gap between theoretical research and practical hardware constraints. This is directly relevant to the research's goal of evaluating HHL's performance on existing quantum hardware.

Zhang et al [15] demonstrated a improved circuit implementation of the HHL algorithm and its simulations on QISKIT for 2x2 and 4x4 matrix linear system and showed that the improved circuit implementation of the HHL algorithm can effectively reduce quantum resources without losing the fidelity of the results.

In summary, while the HHL algorithm holds significant promise for solving large-scale linear systems, its practical implementation is currently limited by quantum hardware and software challenges. As quantum technology advances, particularly with improvements in qubit coherence and gate fidelity, the HHL algorithm may become a more viable option for solving complex linear systems in scientific and engineering applications.

3. Proposed Work

The Harrow-Hassidim-Lloyd (HHL) algorithm is a quantum algorithm designed to solve systems of linear equations, offering exponential speedup over classical algorithms under specific conditions. The classical problem can be expressed as $A\vec{x} = \vec{b}$ where A is a $N \times N$ matrix, $\vec{x}$ is the vector of unknowns, and $\vec{b}$ is the known vector of constants. In quantum computing, the equivalent formulation is $A \mid x\rangle = \mid b\rangle$, $\mid x\rangle$ and $\mid b\rangle$, are quantum states corresponding to the vectors $\vec{x}$ and $\vec{b}$, respectively.

The HHL algorithm consists of the following key steps:

A. Quantum Phase Estimation

At the heart of the HHL algorithm is quantum phase estimation (QPE), a procedure that estimates the eigenvalues of a unitary operator. Given a matrix A, quantum phase estimation decomposes it into its eigenvalues λi and eigen vectors $\mid ui\rangle$ such that $A \mid ui\rangle = \lambda i \mid ui\rangle$. This decomposition is crucial for efficiently solving the linear system since quantum computers can process all eigenvalues and eigenvectors in superposition. The output of the QPE step is a quantum state encoding the eigenvalues of Ā, which are stored in an ancillary qubit register.

B. Controlled Rotation

Once the eigenvalues are extracted, a controlled rotation is applied based on these values. The goal of this step is to invert the eigenvalues to compute A^{-1} . However, a direct inversion would cause issues if any eigenvalues are close to zero. Therefore, controlled rotations are performed on the ancillary qubits to approximate $\frac{1}{\lambda i}$, thus allowing the system to invert A. This step encodes the solution to the system of equations into the quantum state.

C. Inverse Quantum Phase Estimation

After the controlled rotation, inverse quantum phase estimation is applied to return the system to its original basis, yielding a quantum state $|x\rangle$, which is the solution to the linear system. The quantum state now represents the vector x such that $A\vec{x} = \vec{b}$. This state can be measured to obtain an approximation of the solution.

D. Conditions for Optimal Performance

For the HHL algorithm to achieve its theoretical exponential speedup, the matrix A must satisfy specific conditions. First, A must be Hermitian, i.e., $A=A^\dagger$, where $A^\dagger$ is the conjugate transpose of A [4]. Non-Hermitian matrices can be transformed into Hermitian ones by augmenting the matrix with an ancillary system, but this increases the overall computational complexity. Moreover, the matrix should have a well-conditioned spectrum, meaning that the ratio of the largest to smallest eigenvalue should not be too large. Poorly conditioned matrices introduce significant errors during phase estimation, leading to degraded accuracy in the final solution.

Lastly, sparsity is yet another crucial element. Because the technique simplifies quantum operations, it is especially effective when A is sparse. Dense matrices increase the probability of mistakes due to noise on quantum hardware since the method needs more quantum gates [5].

4. Results and Discussion

This section presents the results obtained from running the HHL algorithm, followed by an analysis of how well the quantum solution approximates the exact classical solution.

A. Exact versus Experimental Solution

The system of linear equations $A\vec{x} = \vec{b}$ solved using both classical and quantum methods. The classical solution was calculated using conventional numerical methods, providing the following exact solution:

$$\text{Exact Solution} = \begin{bmatrix} 4 \\ -1 \end{bmatrix}$$

The quantum solution obtained from the HHL algorithm, after running the quantum circuit and applying post-selection, was:

$$\text{Experimental solution}=\begin{bmatrix} 3.99999935 \ - \ 1.59 \ \times \ 10 - 14j \\ -1.00000261 \ + \ 5.38 \ \times \ 10 \ - 15j \end{bmatrix}$$

The error between the classical and quantum solutions was computed to be:

$$Error\ in\ solution : 2.69 \times 10^{-6}$$

This minimal error demonstrates that the HHL algorithm successfully approximated the solution with high accuracy.

B. State vector Real Amplitude Visualization

To further analyze the quantum state, the study visualized the real part of the state vector after the HHL algorithm was applied. Figure 1 shows the bar chart of the real amplitudes of the quantum state components, illustrating the superposition of states generated by the algorithm

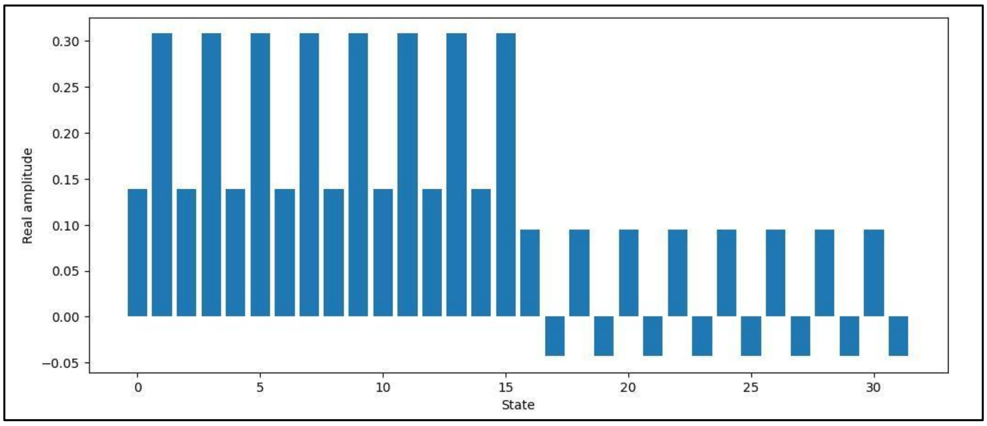

Figure 1. Real Amplitude of Quantum State after HHL Algorithm

This visualization highlights the contribution of each quantum state to the final solution. The dominant amplitudes correspond to the quantum state components that contribute most to the solution, confirming that the algorithm has encoded the correct solution vector.

C. Measurement Results and Histogram

After applying the HHL algorithm, the quantum state was measured using the QASM simulator, which simulates quantum measurements with a large number of shots. The measurement results were aggregated into a histogram, shown in Figure 2, representing the frequency of each quantum state after measurement

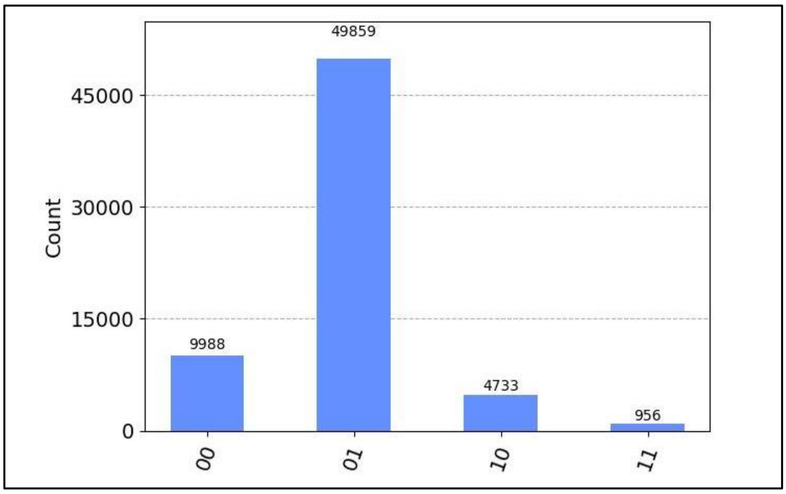

Figure 2. Histogram of Measurement Results

The histogram demonstrates the likelihood of different quantum states being observed, indicating the probabilities of specific outcomes in the quantum circuit. The majority of measurements correspond to the correct quantum states, reinforcing the fidelity of the HHL algorithm in solving the linear system.

The HHL algorithm has been implemented using the IBM Qiskit framework to solve both 2 × 2 and 4 × 4 systems of linear equations. In this section, we analyze the performance of the algorithm on these systems, evaluating key metrics such as fidelity, computational efficiency, and the algorithm’s robustness on current quantum hardware.

D. 2 × 2 Matrix Case Study

The 2 × 2 system chosen for simulation is simple and well-conditioned, allowing the HHL algorithm to achieve near-ideal results. The matrix A and vector $\vec{b}$ for this case are as follows:

$$A = \begin{bmatrix} 1 & 0.5 \\ 0.5 & 1 \end{bmatrix}, \vec{b} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

This system was solved on the IBM Qiskit state vector simulator, a quantum simulator designed to run quantum circuits in a noise-free environment, allowing us to focus solely on the algorithm's performance. Figure 1 presents the quantum circuit generated for this implementation, which utilizes quantum phase estimation and controlled rotation to compute the solution.

The classical solution to this system, obtained using conventional numerical methods, is:

$$\vec{x}_{\text{classical}} = \begin{bmatrix} 0.7866 \\ 0.6175 \end{bmatrix}$$

The quantum solution generated using the HHL algorithm on Qiskit is

$$\vec{x}_{\text{quantum}} = \begin{bmatrix} 0.7866 \\ 0.6175 \end{bmatrix}$$

The fidelity, a measure of how closely the quantum solution matches the classical solution, was computed using the inner product of the normalized quantum and classical states. For this 2 × 2matrix, the fidelity was approximately 0.9999, indicating near-perfect accuracy in solving the system. The runtime for this implementation was minimal, demonstrating that the HHL algorithm is well-suited to small, well-conditioned systems of linear equations on quantum simulators [1], [4].

The high fidelity achieved in the 2 × 2 case highlights the potential of the HHL algorithm to deliver accurate solutions when the matrix is small, sparse, and well-conditioned. However, this result was obtained on a simulator free from the noise and errors inherent in real quantum hardware. Thus, while the result is promising, the performance may differ when the same experiment is run on actual quantum devices.

E. 4 × 4 Matrix Case Study

Next, the study applied the HHL algorithm to a more complex 4 × 4 system to evaluate its scalability and performance on larger matrices. The matrix A and vector $\vec{\vec{b}}$ for this case are:

$$A = \begin{bmatrix} 2 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 3 & 0 \\ 1 & 0 & 0 & 2 \end{bmatrix}, \vec{b} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

This system was solved using the same Qiskit simulator with the HHL algorithm. The classical solution for this system was computed as:

$$\vec{x}_{\text{classical}} = \begin{bmatrix} 0.3333 \\ -0.3333 \\ 0 \\ 0.3333 \end{bmatrix}$$

The quantum solution produced by the HHL algorithm for this 4x4 matrix was:

$$\vec{x}_{\text{quantum}} = \begin{bmatrix} 0.349 \\ -0.088 \\ 0 \\ 0.306 \end{bmatrix}$$

The fidelity for this system was calculated to be approximately 0.823, significantly lower than in the 2 × 2 case. The reduction in fidelity is largely due to the increased size of the matrix and the challenges of simulating larger quantum circuits. As more qubits and gates are used, the likelihood of errors increases, even in simulated environments [4], [5].

The lower fidelity in the 4 × 4 case demonstrates the challenges of scaling the HHL algorithm to larger matrices. While the algorithm still provides an approximation of the solution, the accuracy diminishes as matrix size grows. This issue is further compounded in real quantum devices, where noise and qubit decoherence can drastically affect performance. These results suggest that while the HHL algorithm performs well on small systems, its scalability remains limited by current hardware and algorithmic constraints.

5. Conclusion

Using the concepts of quantum computing, the Harrow-Hassidim-Lloyd (HHL) algorithm offers a novel method for solving systems of linear equations with the potential for exponential speedups over traditional algorithms. The HHL algorithm's theoretical framework, IBM Qiskit implementation, and performance on small-scale linear systems have all been examined in this work. The findings show that although the HHL algorithm works very well for small, well-conditioned matrices, the present limits of quantum hardware and the inherent noise in quantum calculations make it difficult to scale. As shown in the analysis, the HHL algorithm achieved high fidelity in solving a 2x2 system but faced difficulties when applied to a larger 4x4 system, where the fidelity dropped significantly. This highlights the need for ongoing research to enhance the robustness of quantum algorithms, particularly in dealing with larger and more complex matrices. The potential applications of the HHL algorithm span various fields, including quantum machine learning, financial modeling, and quantum chemistry, making it a topic of substantial interest for researchers and practitioners alike. The HHL method represents a promising frontier in quantum computing, despite its present drawbacks. Unlocking the full potential of the HHL algorithm for real-world applications requires ongoing developments in quantum hardware, error correction methods, and algorithmic improvements. For quantum algorithms to become a usual computational tool, future studies should concentrate on experimental implementations on actual quantum devices, tackling both hardware and software issues.

References

[1] W. Harrow, A. Hassidim, and S. Lloyd. "Quantum Algorithm for Linear Systems of Equations." Physical Review Letters, 103, no. 15 (2009): 150502.

[2] X.-D. Cai, C. Weedbrook, Z.-E. Su, M.-C. Chen, M. Gu, et al. "Experimental Quantum Computing to Solve Systems of Linear Equations." Physical Review Letters, 110, no. 23 (2013): 230501.

[3] J. Pan, Y. Cao, X. Yao, et al. "Experimental Realization of Quantum Algorithm for Solving Linear Systems of Equations." Physical Review A, 89, no. 2 (2014): 022313.

[4] W. Ji and X. Meng. "Demonstration of Quantum Linear Equation Solver on the IBM Qiskit Platform." 19th International Symposium on Distributed Computing and Applications for Business Engineering and Science (DCABES), 2020: 308-311.

[5] Adedoyin, Adetokunbo, John Ambrosiano, Petr Anisimov, William Casper, Gopinath Chennupati, Carleton Coffrin, Hristo Djidjev et al. "Quantum algorithm implementations for beginners." arXiv preprint arXiv:1804.03719 (2018).

[6] Barz, Stefanie, Ivan Kassal, Martin Ringbauer, Yannick Ole Lipp, Borivoje Dakic, Alán Aspuru-Guzik, and Philip Walther. "Solving systems of linear equations on a quantum computer." arXiv preprint arXiv:1302.1210 (2013).

[7] Morgan, Jack, Eric Ghysels, and Hamed Mohammadbagherpoor. "An Enhanced Hybrid HHL Algorithm." arXiv preprint arXiv:2404.10103 (2024).

[8] Ginzburg, Matias, and Ugo Marzolino. "Error convergence of quantum linear system solvers." arXiv preprint arXiv:2410.18736 (2024).

[9] Tsemo, Peniel Bertrand, Akshaya Jayashankar, K. Sugisaki, Nishanth Baskaran, Sayan Chakraborty, and V. S. Prasannaa. "Enhancing the Harrow-Hassidim-Lloyd (HHL) algorithm in systems with large condition numbers." arXiv preprint arXiv:2407.21641 (2024).

[10] Kumar, Pradeep, Tanoy Kanti Konar, Leela Ganesh Chandra Lakkaraju, and Aditi Sen De. "Quantum resources in Harrow-Hassidim-Lloyd algorithm." Physics Letters A 517 (2024): 129668.

[11] Duan, Bojia, Jiabin Yuan, Chao-Hua Yu, Jianbang Huang, and Chang-Yu Hsieh. "A survey on HHL algorithm: From theory to application in quantum machine learning." Physics Letters A 384, no. 24 (2020): 126595.

[12] Li, Qinghai, Hao Wu, Weizhong Qian, Xiaoyu Li, Qinsheng Zhu, and Shan Yang. "Portfolio Optimization Based on Quantum HHL Algorithm." In International Conference on Artificial Intelligence and Security, pp. 90-99. Cham: Springer International Publishing, 2022.

[13] Liu, Xiaonan, Haoshan Xie, Zhengyu Liu, and Chenyan Zhao. "Survey on the Improvement and Application of HHL Algorithm." In Journal of Physics: Conference Series, vol. 2333, no. 1, p. 012023. IOP Publishing, 2022.

[14] Angara, Prashanti Priya, Ulrike Stege, Hausi A. Müller, and Mehdi Bozzo-Rey. "Hybrid quantum-classical problem solving in the NISQ era." In Proceedings of the

30th Annual International Conference on Computer Science and Software Engineering, pp. 247-252. 2020.

[15] Zhang, Meng, Lihua Dong, Yong Zeng, and Ning Cao. "Improved circuit implementation of the HHL algorithm and its simulations on QISKIT." Scientific Reports 12, no. 1 (2022): 13287.

Author's biography

Shwetha V is a final-year undergraduate student in the Department of Electronics and Communication Engineering. Her interests lie in data analysis, quantum computing, and FPGA-based biometric systems. She has previously worked on researchs involving biometric authentication using FPGA hardware accelerators and quantum algorithms such as the HHL algorithm. Currently, she is pursuing expertise in data science and quantum computing, with a focus on practical applications of Python and Qiskit in research and industry.

Abinaya Selvarajan is a final-year undergraduate student in the Department of Electronics and Communication Engineering at Madras Institute of Technology, Anna University. Her research interests span image processing, cybersecurity, and artificial intelligence. She has worked on researchs including secure data transmission using Li-Fi, exploring both image processing and cybersecurity aspects, as well as handwritten text recognition using convolutional and recurrent neural networks.

Aarthi A is a final-year undergraduate student in the Department of Electronics and Communication Engineering at Madras Institute of Technology, Anna University. Her research interests include quantum computing, electronics, and IoT applications. She has previously worked on researchs involving thermoelectric cooling systems, noise meters, and air quality monitoring systems. She is currently exploring the realm of quantum computing and its applications, with a focus on quantum speedup for linear systems.

Sneka R is a final-year undergraduate student in the Department of Electronics and Communication Engineering at Madras Institute of Technology. Her research interests include embedded systems, sensor-based applications, and IoT technologies. She has worked on diverse researchs, including an OCR-based image-to-speech conversion system using Python, an MLX90614-based non-contact infrared temperature sensor, a condenser microphone preamplifier circuit, and an air quality monitoring system. Sneka is passionate about leveraging technology to develop innovative solutions for real-world challenges.