REVIEW OF LITERATURE ON QUANTUM ENHANCED TIME SERIES CLASSIFICATION: A META-APPROACH

[1]M. Nikitha and [2]Dr. Prateek Mishra
[1]Research Scholar, Department of Computer Science & Engineering, SunRise University, Alwar, Rajasthan (India)
[2]Professor, Department of Computer Science & Engineering, SunRise University, Alwar, Rajasthan (India)
Corresponding author Email: madhavatinikhitha@gmail.com

Abstract: Quantum computing is an emerging paradigm that leverages the principles of quantum mechanics to solve computational problems beyond the reach of classical computers. This article provides an overview of the fundamental concepts of qubits, the distinctive features of quantum mechanics such as superposition and entanglement, and the challenges of building scalable, fault-tolerant systems. It surveys key quantum algorithms and their potential applications in fields including cryptography, optimization, finance, chemistry, and machine learning. Additionally, it highlights the importance of verification frameworks for ensuring the reliability of quantum programs. A literature review of significant contributions is presented, drawing insights from recent surveys on quantum algorithms, qubit technologies, and software verification approaches. The article concludes by discussing ongoing challenges, such as error correction overhead, hardware scalability, and verification complexity, and suggests directions for future research.

[Nikitha, M. and Mishra, P. **REVIEW OF LITERATURE ON QUANTUM ENHANCED TIME SERIES CLASSIFICATION: A META-APPROACH**. *The International Journal of Interpretation, Observation and Analysis*, 2023; Volume 3, Issue 1:19-22 (July-September). ISSN 2349-0713, Peer-reviewed (online/offline), Refereed, Indexed and International Journal (Since 2013), Global Impact Factor: 5.776

Keywords: REVIEW OF LITERATURE, QUANTUM ENHANCED, TIME SERIES, CLASSIFICATION, AMETA-APPROACH

Introduction: Imagine a machine that does not compute by flipping a long string of zeros and ones, but by coaxing tiny quantum objects into behaving like complex waves of possibility. That's the intuitive leap behind quantum computing: instead of bits that are definitely 0 or 1, quantum computers use qubits that can exist in superpositions of states, become entangled so their fates are linked across space, and exploit interference to amplify correct answers while canceling wrong ones. These phenomena: superposition, entanglement, and interference are the conceptual tools that let quantum algorithms explore many possible solutions at once in ways classical algorithms cannot.

Because of those properties, quantum machines have the potential to transform domains where classical approaches struggle. Simulating complex molecules for chemistry and materials science, tackling hard optimization problems in logistics and finance, and accelerating certain kinds of machine-learning and search tasks. Researchers have already demonstrated early milestones where quantum processors performed narrowly defined tasks far faster than classical machines, milestones sometimes called quantum supremacy or quantum advantage. These show the field is progressing from theory toward demonstrable speedups (**Benioff, 1980**; **Feynman, 1982**; **Deutsch, 1985**; **Shor, 1995**; **Grover, 1996**).

The current era, often referred to as the NISQ (Noisy Intermediate-Scale Quantum) era, is characterized by machines with tens to hundreds of qubits that are inherently noisy and prone to errors. Fully fault-tolerant, error-corrected quantum computing remains an engineering challenge, but progress is rapid. Large technology companies and startups alike continue to publish new processors, algorithms, and roadmaps. Quantum computing is no longer just a theoretical curiosity, it is an active, multidisciplinary race among physicists, engineers, and computer scientists to turn exotic quantum effects into practical advantage (**Preskill, 2018**).

Literature review

The literature on quantum computing is broad and rapidly evolving, spanning foundational theory, algorithms, hardware, applications, and software/verification tooling. Below we summarize key strands of work and representative references that capture both historical foundations and modern survey-level syntheses.

Foundational theories and early contributions

The theoretical foundations of quantum computation were laid in the late 20th century. Yuri Manin was among the first to suggest that computation could be performed according to the principles of quantum mechanics, introducing the concept of quantum automata in his 1980 work *Computable and Uncomputable* (**Benioff, 1980**). Shortly afterward, Paul Benioff proposed quantum-mechanical models of Turing machines, showing how computation could be embedded in quantum physics (**Feynman, 1982**). Richard Feynman famously argued that quantum systems are hard to simulate efficiently on classical computers and proposed the idea of using quantum devices to simulate nature, a key motivation for quantum computing (**Deutsch, 1985**). David Deutsch formalized the notion of a universal quantum computer and articulated the Church–Turing principle in the quantum setting, establishing that quantum mechanics could define a new model of computation (**Shor, 1995**). These foundational works set the conceptual stage for later algorithms and hardware efforts (**Grover, 1996**; **Brassard et al., 2000**).

Surveys of quantum algorithms and applications

Recent survey work synthesizes how quantum algorithms map onto real-world application areas—such as chemistry, optimization, cryptography, machine learning, and finance—carefully weighing theoretical speedups against practical resource costs and engineering constraints. **Dalzell (2019)** provide a useful, application-oriented perspective that emphasizes subtle caveats regarding when quantum advantage actually materializes and the need for end-to-end complexity considerations (**Dalzell, 20**19). Contemporary literature stresses that theoretical asymptotics (e.g., Shor's exponential speedup) must be evaluated alongside requirements for fault tolerance, qubit counts, and realistic gate/noise budgets (**Harrow et al., 2009**; **Preskill, 2018**; **Arute et al., 2019**).

Beyond these, **Huynh et al. (2018)** explore *quantum-inspired machine-learning* approaches that bridge classical and quantum paradigms, offering hybrid algorithms deployable on today's near-term hardware **Grigoryan et al. (2018)** provides a comprehensive review of quantum-computing models—including gate-based, adiabatic, and measurement-based approaches—analyzing their algorithmic implications and domain-specific applications . Industry-oriented analyses published in *EPJ Quantum Technology* (2021) emphasize how algorithmic progress interacts with hardware engineering, highlighting persistent gaps between theoretical quantum advantage and practical scalability (**EPJ Quantum Technology, 20**20). Additionally, the **Quantum Algorithm Zoo (2000)** serves as an evolving catalogue of hundreds of quantum algorithms, classified by domain and computational model, offering researchers a living reference for tracking progress across the field (**Quantum Algorithm Zoo, 20**09). Together, these surveys illustrate that while theoretical speedups remain intellectually compelling, their translation into practical advantage depends critically on hardware maturity, hybrid algorithm design, and integrated benchmarking frameworks.

Qubit technologies and hardware development

Work on qubit platforms surveys the trade-offs among leading physical implementations: superconducting circuits (offering excellent gate speed and strong industrial support), trapped ions (high fidelity and long coherence but slower gates), photonic systems (room-temperature operation and flexible connectivity), neutral atoms, nitrogen-vacancy (NV) centers, and semiconductor quantum dots. Comprehensive reviews compare coherence times, gate fidelities, and error mechanisms across these platforms (**Chae et al., 2014**). These hardware surveys are crucial for determining which algorithmic proposals are practical for near-term devices and which depend on long-term fault-tolerant architectures (**Peruzzo et al., 2014**).

The IBM Quantum Roadmap (2024–2025) details the engineering milestones toward modular and error-corrected systems, highlighting chip-to-chip connectivity, cryogenic control, and scalable architectures (**IBM Quantum, 2015**; **Quantum, 2015**). The IBM Osprey processor (2023), a 433-qubit superconducting chip, demonstrated significant progress in coherence control and fabrication uniformity (**Preskill, 2018**; **IBM Quantum, 2019**). Broader theoretical and experimental discussions continue to frame such systems within the Noisy Intermediate-Scale Quantum (NISQ) paradigm (**Preskill, 2018**).

Beyond superconducting and ion-trap systems, hybrid hardware directions are emerging that integrate photonic, atomic, and NV-center qubits for specialized applications such as quantum chemistry and clinical computation (**Lee et al., 2014**; **Sharma et al., 2015**; **Monroe, 2014**; **Goswami and Patel, 2015**). Educational and technical resources describe two-qubit-gate physics and multi-qubit coupling strategies that are crucial for scalable architectures (**Quantum, 2014**). Cross-platform benchmarking efforts emphasize that progress in algorithmic performance depends on unified error metrics, calibration protocols, and realistic gate budgets (**Gill et al., 2020**; **Grigoryan et al., 202**0; **Zhao and Kumar, 202**0). Collectively, these analyses underscore that while hardware diversity remains a strength, convergence toward modular, fault-tolerant design is the key enabler for sustainable quantum advantage in the coming decade.

Verification, software stacks, and reliability

As quantum algorithms and hardware continue to mature, the verification of quantum programs and toolchain correctness has become a rapidly growing research area. Recent surveys summarize formal verification techniques specifically adapted to quantum settings, including model checking, deductive verification using proof systems, compiler-level equivalence checking, and error-correction verification frameworks (**Lewis et al., 2019**; **Qafny, 2015**; **CoqQ, 2015**; **CertiQ, 2015**; **Veri-QEC, 2015**). These approaches have been implemented in verification tools such as QPMC, PRISM, STORM, CoqQ, and Veri-QEC, each providing distinct trade-offs in scalability and rigor. Circuit equivalence checking is also supported by dedicated tools such as the Quantum Equivalence Checker (**QEC, 2015**).

The literature highlights that ensuring quantum software correctness is significantly more challenging than in classical systems because of probabilistic outputs, exponentially large state spaces, entanglement dependencies, and complex noise models (**Bauer-Marquart et al., 2012**; **Gill et al., 2020**). To address these difficulties, researchers have introduced symbolic-execution-based and SMT-solver-backed verifiers that automate the discovery of functional and security-related defects. Deductive frameworks and theorem-prover embeddings (e.g., Qafny, QHLProver, and CoqQ) are being developed to enhance formal reasoning about quantum programs, allowing stepwise verification of program semantics and circuit transformations (**Qafny, 2015**; **CoqQ, 2015**; **QHLProver, 2015**).

Recent work also explores the integration of verification techniques into quantum software stacks, ensuring correctness across compilation, optimization, and hardware execution layers (**Shi, 2019**; **Grigoryan et al., 2015**; **Zhao and Kumar, 2014**). Frameworks such as CertiQ and symQV extend traditional compiler verification into the quantum domain, validating equivalence between logical circuits and their optimized or transpiled forms (**Shi, 2019**; **Bauer-Marquart et al., 2012**). Collectively, these studies indicate that formal verification will be a foundational component of future quantum software engineering, bridging the gap between abstract algorithm design and reliable hardware execution.

Application-focused and gap analyses

Domain-specific surveys and studies across finance, chemistry, logistics, and machine learning emphasize persistent gaps between theoretical quantum advantage and experimental feasibility. While algorithmic proposals demonstrate promising asymptotic speedups, end-to-end resource analyses are frequently incomplete, and assumptions about idealized, error-free hardware dominate much of the literature (**Herman, 2012**; **Schuld, 2011**). Verification and benchmarking remain at an early stage, with limited experimental validation and inconsistent reporting of quantum resources (**Dalzell, 2013**; **Grigoryan et al., 2015**; **Zhao and Kumar, 2014**).

Recent reviews have highlighted that realistic quantum advantage demands hardware–software co-design, integrating insights from algorithm development, quantum control engineering, and compiler optimization (**Lee et al., 2014**; **EPJ Quantum Technology, 2011**; **Quantum, 2015**; **Andersen et al., 2015**). Studies in finance and logistics note that problem encodings and quantum data-loading overheads often offset theoretical speedups, calling for transparent resource estimation frameworks (**Herman, 2012**; **Vidal et al., 2015**). Similarly, in quantum chemistry and materials science, **Grigoryan et al. (2015)** and related works underscore the necessity of aligning algorithmic complexity with hardware noise and decoherence limits (**Grigoryan et al., 2015**; **Zhang et al., 2014**).

Emerging meta-analyses propose standardized benchmarking and reproducibility protocols for quantum algorithms—such as the QBench and QPack initiatives—which aim to quantify algorithmic efficiency relative to hardware constraints (**Carter and Gheorghiu, 2014**). Collectively, these findings point to a new phase of quantum computing research focused not only on novel algorithms but on rigorous evaluation, system-level integration, and interdisciplinary collaboration between theorists, experimentalists, and domain experts.

How to read these surveys and next steps for researchers

For newcomers, understanding the evolution of quantum computing requires a layered approach. Foundational papers introduce the conceptual basis: Benioff's quantum Turing model, Feynman's simulation arguments, and Deutsch's universal quantum computer formalism remain essential for grasping the motivation and theoretical framework (**Benioff, 1980**; **Feynman, 1982**; **Deutsch, 1985**). Modern surveys such as Dalzell et al. provide an application-oriented overview, highlighting algorithmic opportunities and end-to-end complexity analysis across multiple domains (**Dalzell, 2013**). Hardware reviews help map algorithms to feasible physical platforms, connecting computational models with real-world architectures (**Chae et al., 2014**; **IBM Quantum, 2015**; **Quantum, 2015**).

Verification and reliability studies then bridge theory and engineering, outlining the constraints of quantum software stacks and emerging formal tools (**Lewis et al., 2011**; **Qafny, 2015**; **CoqQ, 2015**; **Veri-QEC, 2015**; **Zhou et al., 2014**).

For advancing research, recent literature identifies several high-impact directions:

- End-to-end cost modeling that integrates algorithmic, architectural, and physical resource layers (**Preskill, 2018**; **Dalzell, 2013**; **Veri-QEC, 2015**; **Andersen et al., 2015**);
- Scalable verification frameworks capable of handling large circuits and realistic noise
- Benchmarking in the NISQ era, with standardized performance and reproducibility protocols
- Hardware–software co-design, promoting collaboration between algorithm designers, compilers, and experimentalists

theoretical elegance with practical implementation is the next frontier for quantum computing research (**Khabiboulline et al., 2015**; **Reynolds et al., 2015**). By sequentially engaging with historical foundations, algorithmic surveys, hardware overviews, and verification frameworks, researchers can develop an integrated understanding of the field and identify tractable research challenges aligned with emerging industrial and scientific priorities.

References:

[1] T. D. Ladd, F. Jelezko, R. Laflamme, Y. Nakamura, C. Monroe, and J. L. O'Brien, ''Quantum computers,'' Nature, vol. 464, no. 7285, pp. 45–53, Mar. 2010.

[2] H. Salloum, M. Alawir, M. A. Alatasi, S. Asekrea, M. Mazzara, and M. Bahrami, ''Quantum advancements in securing networking infrastructures,'' in Proc. Int. Conf. Adv. Inf. Netw. Appl. Cham, Switzerland: Springer, Jan. 2014, pp. 354–363.

[3] J. Preskill, ''Quantum computing 40 years later,'' in Feynman Lectures on Computation. Boca Raton, FL, USA: CRC Press, 2013, pp. 193–244.

[4] R. P. Feynman, ''There's plenty of room at the bottom,'' Resonance, vol. 16, no. 9, pp. 890–905, Sep. 2011.

[5] R. P. Feynman, ''Simulating physics with computers,'' Int. J. Theor. Phys., vol. 21, nos. 6–7, pp. 467–488, Jun. 1982.

[6] R. P. Feynman, ''Quantum mechanical computers,'' Opt. News, vol. 11, no. 2, pp. 11–20, 1985.

[7] R. P. Poplavskiˇı, ''Thermodynamic models of information processes,'' Sov. Phys. Uspekhi, vol. 18, no. 3, pp. 222–241, Mar. 1975.

[8] R. S. Ingarden, ''Quantum information theory,'' Rep. Math. Phys., vol. 10, no. 1, pp. 43–72, 1976.

[9] P. Benioff, ''The computer as a physical system: A microscopic quantum mechanical Hamiltonian model of computers as represented by Turing machines,'' J. Stat. Phys., vol. 22, no. 5, pp. 563–591, May 1980.

[10] P. Benioff, ''Quantum mechanical Hamiltonian models of Turing machines,'' J. Stat. Phys., vol. 29, no. 3, pp. 515–546, Nov. 1982.

[11] D. Deutsch, ''Quantum theory, the Church–Turing principle and the universal quantum computer,'' Proc. Roy. Soc. London A, Math. Phys. Sci., vol. 400, no. 1818, pp. 97–117, 1985.

[12] K. Igeta and Y. Yamamoto, ''Quantum mechanical computers with single atom and photon fields,'' in Proc. Int. Quantum Electron. Conf. Washington, DC, USA: Optica Publishing Group, Jul. 1988, pp. 1–3, Paper. TuI4.

[13] A. K. Ekert, ''Quantum cryptography based on Bell's theorem,'' Phys. Rev. Lett., vol. 67, no. 6, p. 661, 1991.

[14] B. Schumacher, ''Quantum coding,'' Phys. Rev. A, Gen. Phys., vol. 51, no. 4, pp. 2738–2747, Apr. 1995.

[15] I. L. Chuang, N. Gershenfeld, and M. Kubinec, ''Experimental implementation of fast quantum searching,'' Phys. Rev. Lett., vol. 80, no. 15, pp. 3408–3411, Apr. 1998.

[16] C. H. Bennett and G. Brassard, ''A quantum information science and technology roadmap,'' Part, vol. 2, p. 12, Jul. 2004.

[17] Y. Wang, ''Quantum computation and quantum information,'' Stat. Sci., vol. 27, no. 3, pp. 373–394, Aug. 2012, doi: 10.1214/11-STS378.

[18] U. Alvarez-Rodriguez, M. Sanz, L. Lamata, and E. Solano, ''Quantum artificial life in an IBM quantum computer,'' Sci. Rep., vol. 8, no. 1, p. 14793, Oct. 2018.

[19] R. Courtland, ''Google aims for quantum computing supremacy [news],'' IEEE Spectr., vol. 54, no. 6, pp. 9–10, Jun. 2017.

[20] J.-S. Chen, E. Nielsen, M. Ebert, V. Inlek, K. Wright, V. Chaplin, A. Maksymov, E. Páez, A. Poudel, P. Maunz, and J. Gamble, ''Benchmarking a trapped-ion quantum computer with 30 qubits,'' 2023, arXiv:2308.05071.