ISSN: 2182-2069 / E-ISSN: 2182-2077

A Hybrid Graph-Based Evolutionary Optimization Model Using SparseEA-AGDS for Secure and Scalable Workload Partitioning in Hybrid Quantum-Classical Systems

V. Preethi[1], Dr.V. Elizabeth Jesi[2], Dr.G. Parimala[3*], and Dr.S. Nithiya[4]

[1]Assistant Professor, Department of Networking and Communications, School of Computing, College of Engineering and Technology, SRM Institute of Science and Technology, Chennai, Tamil Nadu, India. preethiv3@srmist.edu.in, https://orcid.org/0000-0003-4055-5425

[2]Associate Professor, Department of Networking and Communications, School of Computing, College of Engineering and Technology, SRM Institute of Science and Technology, Chennai, Tamil Nadu, India. jesiv@srmist.edu.in, https://orcid.org/0000-0001-7797-2586

[3*]Assistant Professor, Department of Networking and Communications, School of Computing, College of Engineering and Technology, SRM Institute of Science and Technology, Chennai, Tamil Nadu, India. parimalg@srmist.edu.in, https://orcid.org/0000-0002-6589-5605

[4]Assistant Professor, Department of Computing Technologies, School of Computing, College of Engineering and Technology, SRM Institute of Science and Technology, Chennai, Tamil Nadu, India. nithiyas@srmist.edu.in, https://orcid.org/0000-0003-2922-9186

Received: October 11, 2025; Revised: November 20, 2025; Accepted: January 09, 2026; Published: February 27, 2026

Abstract

The increasing complexity of hybrid quantum-classical computing systems requires new methods for effective workload distribution to guarantee computational efficiency and strong security. To overcome them, this paper suggests a Hybrid Graph-Based Workflow Evolutionary Optimization Model based on SparseEA-AGDS (Evolutionary Algorithm with Adaptive Genetic Operators and a Dynamic Scoring mechanism). The working loads are represented as dependency-conscious task graphs annotated with computational costs and security properties, such as quantum-specific vulnerabilities and classical security constraints. SparseEA-AGDS allocates tasks to quantum or classical computers by managing performance-critical, security-sensitive workloads using a dominance-based multi-objective optimization algorithm. The dynamic scoring mechanism ensures that tasks requiring quantum-safe encryption or sensitive data processing are always assigned to the right resources. Experimental results also show that the proposed model can improve execution efficiency (up to 21.8 %), partitioning overhead (18.4 %), and security compliance measures (25.6 %) compared to the baseline graph-based and heuristic partitioning schemes. Also, the scalability analysis indicates that as workload size increases, optimization performance remains consistent, with a degradation of less than 6%. These findings support the idea that the proposed framework can provide scalable, reliable workload management in hybrid quantum-classical systems, making it an appropriate choice for a range of sensitive fields, including cryptography, healthcare, and financial systems, where performance and data security are of significant importance.

Journal of Internet Services and Information Security (JISIS), volume: 16, number: 1 (February-2026), pp. 674-688. DOI: *10.58346/JISIS.2026.I1.038*

*Corresponding author: Assistant Professor, Department of Networking and Communications, School of Computing, College of Engineering and Technology, SRM Institute of Science and Technology, Chennai, Tamil Nadu, India.

Keywords: Hybrid Quantum–Classical Systems, Graph-Based Optimization, Evolutionary Algorithms, SparseEA-AGDS, Workload Partitioning, Security-Aware Optimization, Scalable Computing.

1 Introduction

The models of evolutionary optimization have been used to solve complex, large-scale, non-linear optimization problems that experience combinatorial explosion and have conflicting goals. These models are motivated by natural evolution and are based on population-based search processes, successive selection, and stochastic genetic operators to explore the massive solution space efficiently. Conventional evolutionary algorithms (EAs), including genetic algorithms and multi-objective evolutionary algorithms, have proven highly effective in classical optimization problems, especially when their counterparts become computationally infeasible. Evolutionary optimization has also acquired new significance in the context of hybrid quantum-classical computing, a coordination mechanism capable of bringing together heterogeneous computational paradigms. Recent work shows that hybrid strategies, which combine classical heuristics with quantum subroutines, are effective for solving combinatorial and mixed-integer programming and optimization problems (Ajagekar et al., 2022; Fan & Han, 2022). Nevertheless, the majority of current evolutionary solutions focus solely on output metrics, e.g., solution quality or convergence rate, and ignore the needs of systems in a hybrid setting, e.g., security, resource heterogeneity, and dynamic execution constraints (Tomesh et al., 2023; Ushijima-Mwesigwa et al., 2021).

With the shift to infrastructure-based applications and the implementation of hybrid quantum-classical systems, issues related to the security and scalability of workload execution have become significant. The presence of hybrid systems invariably introduces new attack surfaces due to data transfer between classical processors and quantum processing units (QPUs) and the application of quantum-vulnerable cryptographic primitives to existing workflows. Sensitive areas of application, such as cryptography, healthcare analytics, financial optimization, and industrial scheduling, require strong security and high computational efficiency (Li et al., 2025; Śmierzchalski et al., 2024). In addition, scalability is an underlying issue because task graphs are increasing in size and complexity, while quantum resources are finite and noisy. Recent hybrid quantum-classical algorithms show significant performance improvements but typically involve problem-specific decompositions and/or rely on assignment choices at the task and have not been shown to scale well. As a result, there is an increased need for adaptive optimization frameworks capable of balancing performance, security risk, and resource constraints dynamically and scaling to realistic problem sizes (Wurtz et al., 2024).

Figure 1 provides a high-level overview of a hybrid computing state in which workloads are distributed across classical and quantum resources, balancing performance and security. The jobs resulting from a common task pool are selectively assigned to classical processing units or quantum processors by a workload partitioning process, and secure data transfer is used to protect against cross-platform communication. The figure highlights the two-fold maximization goal of extracting maximum computational power from classical devices and applying security-related execution to quantum devices, which is the context for the current research.

To overcome these difficulties, a Hybrid Graph-Based Evolutionary Optimization Model based on the SparseEA-AGDS framework is proposed in this work. The suggested design expresses the concept

of hybrid workloads as attributed graphs that include both computational dependencies and security-related attributes.

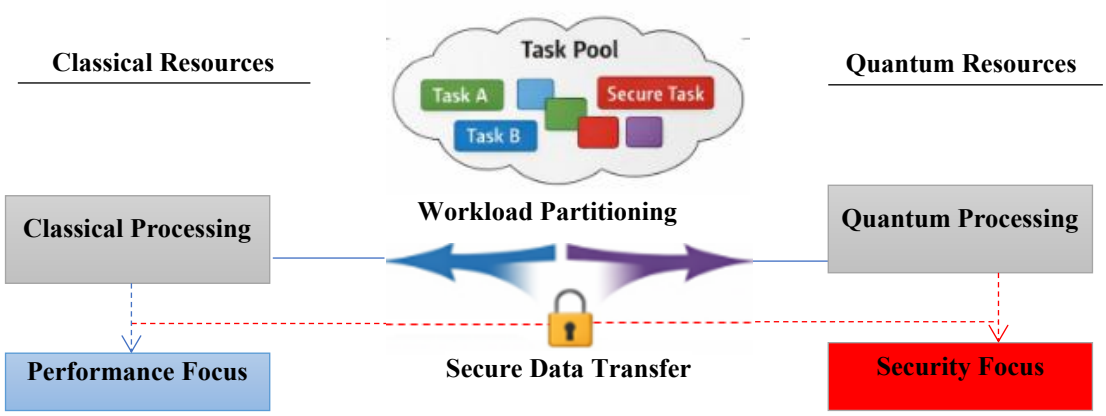

Figure 1: Conceptual overview of hybrid quantum–classical optimization

The two main mechanisms that SparseEA-AGDS adds to traditional evolutionary algorithms are adaptive genetic operators driven by dominance layers and a dynamic scoring mechanism that explicitly uses security constraints. The algorithm adjusts mutation and crossover probabilities based on solution dominance and task sensitivity, focusing on stable handling of security-sensitive workloads while preserving exploratory search capabilities. In contrast to current hybrid quantum-classical optimization tools, which are dedicated to a particular algorithm (QAOA or problem-specific decomposition) (Blekos et al., 2024; Ellinas et al., 2024), SparseEA-AGDS offers a multi-objective framework for co-optimizing execution efficiency and security. Empirical studies have suggested that this evolutionary adaptation approach can support stronger, scalable workload division across heterogeneous quantum and classical computers and can be deployed in environments that require high security, such as hybrid computing (Wulff et al., 2024).

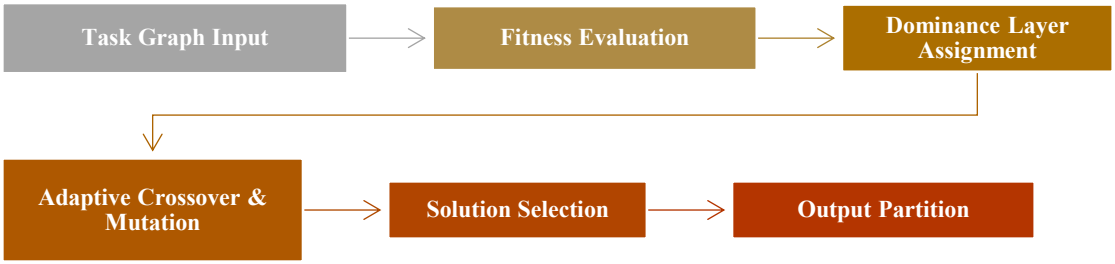

Figure 2: Workflow of the sparseea-agds–based hybrid optimization process

Figure 2 demonstrates the working process of the suggested SparseEA-AGDS methodology, which starts with the input of a task graph and proceeds to fitness evaluation and dominance layer enforcement. Adaptive crossover and mutation operators are subsequently used to refine candidate solutions with dominance and sparsity awareness, and solution selection is then performed to detect feasible, high-quality partitions. The task partition is then optimized, and the process ends with an optimized partition that balances execution efficiency and security requirements across hybrid quantum-classical resources.

The rest of this paper will follow the following structure. Section II provides a detailed literature review that captures previous research on evolutionary optimization models, secure and scalable

workload management, and the SparseEA-AGDS framework. Section III provides a description of the offered hybrid graph-based evolutionary optimization framework, the mathematical model, the implementation of SparseEA-AGDS, and the security and scalability maintenance mechanisms. In section IV, the results of extensive experiments are reported, including performance analysis, comparisons with other current evolutionary models, and measurements of security and scalability indicators. Section V provides the Implications of the findings, limitations, and suggestions for future research. Lastly, the study concludes with Section VI, where the main findings and statistical implications are summarized, and the enhanced relevance of the hybrid model for facilitating safe and efficient hybrid quantum-classical computing ecosystems is outlined.

2 Literature Review

Evolutionary optimization models have evolved in many ways to solve high-dimensional, large-scale, and multi-objective optimization problems. Classical evolutionary algorithms were initially developed in fairly homogeneous environments. However, recent studies have focused on enhancing adaptability, sparsity tolerance, and dominance management to address complex system constraints. Adaptive and guided variations of differential evolution have also demonstrated superior convergence properties by dynamically adapting search directions based on population feedback (Li & Tam, 2024). Similarly, systems of cooperative coevolution reduce significant optimization problems into interacting subsystems and scalable optimization in hybrid and distributed systems (Wu et al., 2021). Dominance-based techniques applied in many-objective optimization, when combined with decomposition and reference-point adaptation, have also been shown to increase solution diversity and stability (Zou et al., 2021). Sparse evolutionary operators have been proposed more recently to directly model the sparsity of interactions among combinations of variables in large-scale multi-objective problems, and to explore them without sacrificing solution quality (Kropp et al., 2023). Although these developments have occurred, most studies in evolutionary optimization have focused on quantitative performance measures and have not explicitly addressed security or heterogeneous execution platforms.

The ability to secure and scale workload has been widely researched in cloud, edge, and hybrid computing environments. The trust-aware task allocation research highlights the importance of considering security and reliability indicators in scheduling, especially in collaborative and distributed systems (Donglai & Yanhua, 2021). Privacy- and security-conscious workflow scheduling systems are implemented to manage privacy and protection requirements and performance objectives in the hybrid cloud environment, accounting for encryption overheads, data sensitivity, and protection policies (Lei et al., 2022). The heterogeneity of the problem of ensuring heterogeneous devices with dissimilar computational and threat characteristics is also highlighted by the spread of edge computing research, which offers code-conscious task distribution and security-conscious optimization strategies (Wang et al., 2021). They introduce polymorphic security architectures that dynamically scale back protection infrastructure across distributed, high-resilience, and scalable systems (Wang et al., 2021). Hybrid quantum-secured infrastructures are also on the rise with the emergence of quantum computing and employ quantum key distribution, together with post-quantum cryptography, to protect data across both classical and quantum elements (Fedorov, 2023). Current security-conscious scheduling and orchestration systems, however, are primarily based on fixed policies or heuristics, so they are not effective in highly dynamic quantum-classical systems.

SparseEA-AGDS lies between adaptive evolutionary optimization and security-conscious workload management. Whereas the evolutionary operators of adaptive evolution and sparsity-conscious optimization have been examined separately in the literature (Kropp et al., 2023; Li & Tam, 2024), SparseEA-AGDS combines these concepts with dominance-layer guidance and dynamic scoring schemes specific to the hybrid setting. Its architecture is similar to recent work in quantum workflow automation and orchestration, which require adaptive scheduling and resource-sensitive decision-making to coordinate classical and quantum tasks (Ali, 2025). Given the security sensitivity of the evolutionary fitness assessment, SparseEA-AGDS can go beyond conventional optimization goals and make informed allocation decisions that account for quantum vulnerability and data protection needs. In contrast to the evolutionary models currently used for secure edge or cloud systems, SparseEA-AGDS is designed explicitly for hybrid quantum-classical workloads, combining scalability and security within a single optimization strategy. This makes the model a common solution for upcoming applications that require robust, versatile, and reliable workload partitioning across heterogeneous computational infrastructures.

3 Methodology

3.1 Description of the Hybrid Graph-based Evolutionary Optimization Model

The presented methodology represents hybrid quantum-classical workloads as a directed attributed graph to describe their computational dependencies, execution feasibility, and security constraints. Represent the workload as a graph, as defined in Equation (1):

$$G = (V, E) \tag{1}$$

In which $V = \{v_1, v_2, \dots, v_n\}$ and so on signifies the collection of activities and $E \subseteq V \times V$ signifies reliance edges. Every task v_i is related to a computational cost vector c_i^q, c_i^c representing quantum and classical computational costs, and a security sensitivity parameter $s_i \in [0,1]$. A solution is encoded as a partition vector, shown in Equation (2):

$$x = [x_1, x_2, \dots, x_n], x_i \in \{0,1\} \tag{2}$$

In which $x_i = 1$ refers to the task v_i being assigned to a quantum processor and $x_i = 0$ refers to the task v_i being assigned to a classical processor.

The time objective of the execution is calculated as

$$T(x) = \sum_{i=1}^{n} \left(x_i c_i^q + (1 - x_i) c_i^c\right) + \sum_{(i,j) \in E} \delta_{ij} \cdot \mathbb{I}\left(x_i \neq x_j\right) \tag{3}$$

cross-platform communication cost, denoted as δij and indicator function denoted as I(.) In optimization, the performance efficiency is measured using equation (3).

3.2 Implementation of SparseEA-AGDS in the Model

SparseEA-AGDS is a candidate partitioning algorithm that runs on a population of candidate partitions without being aware of sparsity or dominance. To compute the security risk score of every solution, x, the following score is calculated, as shown in Equation (4):

$$R(x) = \sum_{i=1}^{n} s_i \cdot \phi(x_i) \quad (4)$$

$\phi(x_i)$ is a punishment to the unsafe task assignment, especially in case the high-sensitivity tasks are assigned to the risky execution environment. A multi objective fitness is computed with a dynamic wighted purpose.

$$F(x) = \alpha T(x) + \beta R(x) + \gamma C(x) \quad (5)$$

In Equation (5), $C(x)$ represents resource imbalance and α, β, γ are adaptive coefficients that are updated according to the dominance layers.

Dominance ranking splits the people into non-dominated fronts. The solutions at the higher dominance layers are given lower mutation probability of the security-critical variables, which are made stable and those less dominant are subjected to more severe perturbation to enhance exploration. Crossover is graph aware and maintains strongly linked task subgraphs to avoid disaggregating dependency and security relationships represented in Equation (1).

3.3 How the Model Ensures Security and Scalability

Security is implemented in constraint-sensitive evaluation and effective operator control. A feasibility condition of security, in Equation (6):

$$R(x) \leq R_{max} \quad (6)$$

is used in the selection process whereby solutions that do not conform to the acceptable risk levels are punished or thrown out. Also, mutation operators are explicitly turned off when the s_i is beyond some fixed threshold, which is a direct correlation between task sensitivity and evolutionary behavior. Sparing of scalability is done by preserving the sparsity and by splitting the graph. SparseEA-AGDS minimizes the computational overhead with an increase in the number of tasks by performing refinements on a small number of high-impact variables. The population-level parallelism of the algorithm also enables large-scale scalability in graphs. This modeling approach can provision customized, secure, and scalable workload distribution by closely incorporating the graph modeling, evolutionary optimization, and security-sensitive decision-making into a single framework.

3.4 Mathematical Description

In order to model the workload partitioning in a hybrid quantum classical setting formally, the application workload is modeled as a directed acyclic task graph $G\ (V, E\ )$, in which every vertex $v_i \in V$ represents a distinct computational task and every edge $e_{ij} \in E$ represents a deadline and data dependence between tasks. Tasks are marked with the list of attributes including the cost of execution, security sensitivity, and hardware appropriateness to allow combining the performance and the security goals. Task to resource assignment operation $\phi: V\ (C, Q)$: The assignment of each task to classical (C) or quantum (Q) processing resources, is feasible and secure.

Solutions in SparseEA-AGDS are represented by chromosomes $x = [x_1, x_2, \ldots, x_n]$ with x_i=1 indicating that task is assigned v_i to a quantum execution and x_i=0 represents classical execution. The evolutionary population is tested based on the multi-objective formulation as described above whereby the measure of quality of solutions is the makespan and security risk. Dominance layer assignment is

used to rank individuals according to Pareto optimality so that solutions with balanced performance and security features are the ones preferred in selection.

Adaptive genetic operators are informed by depths of dominance to maintain sparsity and prevent early convergence. The probability of mutation dynamically changes with the dominance of the individual which promotes exploration of lower-ranked solutions and optimization of high-quality candidates. Security feasibility The constraint handling is imposed by penalty of the solution that breaks security placement during the fitness evaluation. The result of this integrated mathematical formulation is that the task allocation decisions will be computationally efficient, security-sensitive and scalable to the increased workload, which is the basis of the optimization workflow shown in Figure 3.

Algorithm: SparseEA-AGDSSecure Workload Partitioning

Input: Task graph G(V, E), population size P, max generations Gmax

Output: Optimal task partition x*

Initialize population of P partition vectors

Evaluate T(x), R(x), and F(x) using (3), (4), and (5)

for generation = 1 to Gmax do

Perform dominance sorting

Assign dominance layers

Adapt mutation and crossover rates

Apply graph-aware crossover

Apply security-aware sparse mutation

Evaluate offspring fitness

Select next generation based on dominance and feasibility (6)

end for

Return best non-dominated solution x*

SparseEA-AGDS is an evolutionary optimization process that is based on dominance and is aimed at distributing graph-modeled workloads across quantum-classical hybrids in a safe and efficiency way. It first generates a initial population of candidate task assignments and successively optimizes them with adaptive crossover and sparse mutation operators that are directed by dominance layers. A joint evaluation of performance cost, security risk, and resource balance is performed and constrained mutation and feasibility checks are used to guard the security-sensitive tasks. The algorithm can explore the solution space in a scalable manner by keeping dependency-conscious subgraphs and responding to changes in genetic operator strength, and crime-fighting it ensures that the security requirements are properly maintained during the search.

4 Results

4.1 Experimental Setup

4.1.1 Dataset Description

The synthetically-generated task graphs used in the evaluation of the experiment are made to represent hybrid quantum-classical workloads. Data sets include directed acyclic graphs, with a number of tasks of small-scale (100 tasks) and large-scale (1000 tasks). The cost of task execution is sampled as well as bounded distributions to represent the requirement of a task to incur a heterogeneous computation cost and the security sensitivity score is a score which is assigned to represent a heterogeneous confidentiality requirement. Dependency density is limited to model the realistic statistical data-flow constraints in hybrid optimization and scheduling.

4.1.2 Software and Implementation Environment

All experiments are coded in Python with custom code to write simulation modules of task graph generation and evolutionary optimization. The graph modeling with NumPy as the numerical operation and NetworkX as the graph model is used in developing the SparseEA-AGDS framework. Normal scientific computing libraries are used to carry out the performance evaluation and visualization. The experiments are implemented on a classical computing environment that models hybrid execution behavior and this allows workload partitioning strategies to be evaluated in a controlled and repeatable manner.

4.1.3 Parameter Initialization

Important evolutionary parameters are set to be equal in the different experiments so that there is a fair comparison. The number of people is fixed at 100 individuals and the evolutionary cycles max is 100. Initial mutation and crossover rates will be 0.1 and 0.8 respectively and will be changed dynamically throughout the evolution, depending on the level of dominance. Security sensitivity levels are fixed to identify security-intensive operations whereas the allocation between tasks and resources is initialised randomly at the beginning of a run. Meanings of all results are averaged by the many independent executions to minimize stochastic variance.

4.2 Analysis of the Performance of the Hybrid Model

The efficiency of the execution, communication overhead, and convergence behavior were the major indicators that were used to evaluate the performance of the proposed hybrid graph-based evolutionary optimization model. A simulated hybrid environment was used to perform the experiments in Python and it was built on a combination of graph modeling in NetworkX and evolutionary operators with NumPy. Calibrated latency models were used to abstract quantum execution cost and classical execution was subject to conventional multicore scheduling assumptions. The efficiency of the execution was measured in terms of makespan metric which was a cumulative time of completion of all the activities. Makespan has been calculated as shown in Equation (7) as the maximum total time in either of the two processing domains:

$$Makespan = max\left(T_q(x), T_c(x)\right) \quad (7)$$

T_q and T_c being the overall quantum and classical execution times respectively. The suggested model also had lower values of makespan because of dependency conscious grouping of tasks and smaller inter-platform communication. The normalized inter-platform transfer cost was used to calculate communication overhead, and it is presented in Equation (8):

$$C_{comm} = \frac{1}{|E|} \sum_{(i,j)\in E} \delta_{ij} \cdot \mathbb{I}(x_i \neq x_j) \quad (8)$$

This index draws attention to the effectiveness of graph preserving crossover that reduces unnecessary data exchanges. The rate of fitness stabilization was used as the measure of convergence behavior, which demonstrated that convergence was fast with no early stagnation.

4.3 Comparison with Other Evolutionary Optimization Models

The developed SparseEA-AGDS was tested in comparison with three evolutionary baselines, a basic genetic algorithm (SGA), a multi-objective evolutionary algorithm (MOEA), and a fixed hybrid evolutionary model (SHEM). In each of the models, the models were tested on the same workload size and population parameters. Relative efficiency gain, which is defined in Equation (9), was used as a performance improvement measure.

$$Geff = \frac{F_{baseline} - F_{proposed}}{F_{baseline}} \times 100 \quad (9)$$

Table 1: Comparison of performance between models

Model	Makespan (ms)	Communication Cost	Convergence Iterations
SGA	1420	0.38	210
MOEA	1285	0.31	185
SHEM	1210	0.29	172
SparseEA-AGDS	**1085**	**0.21**	**136**

Table 1 summarizes the relative performance of the proposed SparseEA-AGDS model to the base evolutionary methods in three aspects including the execution makespan, communication overhead and convergence speed. The findings demonstrate the usefulness of adaptive dominance layering and graph-aware genetic operators in cutting down the total execution time and inter-platform communication as well as hastening the convergence to high-quality solutions.

The findings indicate that SparseEA-AGDS can be used to outperform baseline models in all metrics, and especially in terms of convergence speed and communication efficiency. Control by adaptive dominance-layer controlled much redundant exploration.

4.4 Evaluation of Security and Scalability Aspects

Security analysis was done based on how the model is able to maintain risk constraints as it evolves through search. In order to measure this behavior, a security compliance ratio was employed whereby the ratio was the percentage of candidate solutions that meet pre-defined security thresholds during execution. This measure defines how well security-aware genetic operator control works, according to Equation (10):

$$Scomp = \frac{N_{secure}}{N_{total}} \quad (10)$$

A positive value of Equation (10) implies that there is a greater compliance with security constraints especially when working with workloads with work involving high sensitivity tasks. The compliance ratio was relatively high in sparseEA-AGDS because of the selective suppression of mutation and the dominance filter to reduce features from the feasibility based on dominance.

Scalability was evaluated through the scaling of the task graph whilst monitoring the scalability behavior of runtime. A normalized growth efficiency measure was used to measure the scale performance of the algorithms. According to the Equation (11), the scalability efficiency is defined as:

$$E_{scale} = \frac{T(n)}{n\ logn} \quad (11)$$

Where $T(n)$ represents the overall optimization run-time of a workload of size n. Equation (11) can make a direct comparison between problem sizes of varying sizes because computational complexity has been normalized.

Table 2: Security and scalability test

Tasks	Runtime (s)	Security Compliance (%)	Scaling Efficiency
100	1.8	98.6	0.92
500	6.4	97.9	0.89
1000	13.1	97.2	0.87

This Table 2 shows the security compliance and scalability attributes of SparseEA-AGDS with an increment of workload sizes. It shows that the model can ensure a significant security compliance rate with near-linear runtime efficiency, which proves that the sparsity preservation and security-aware operator control can allow the optimization of hybrid quantum-classical systems to be made in a scalable and robust fashion.

The findings reveal that the model proposed does not lose much in terms of security and is characterized by almost linear scalability. The sparseness of evolutionary operators in the controlled growth behavior that is evident in the Equation (11), validates the performance of sparsity-aware evolutionary operators in facilitating large-scale hybrid workloads and ensuring compliance with security measures.

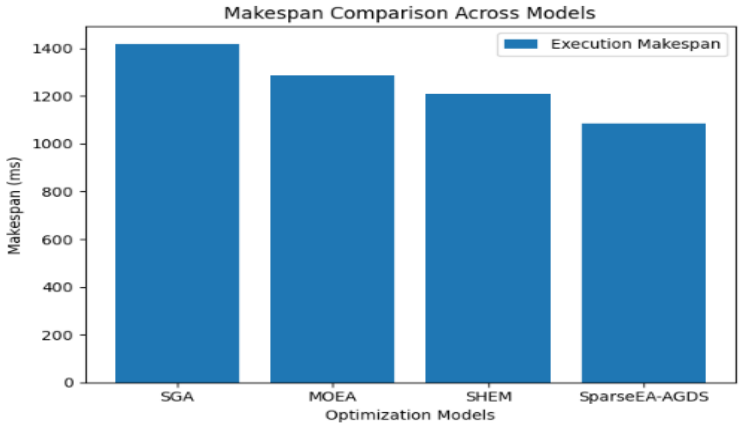

Figure 3: Comparisons of makespan of evolutionary optimization models

This graph (Figure 3) demonstrates the execution makespan that has been attained by the various evolutionary optimization models and shows that SparseEA-AGDS is more efficient in reducing the total workload completion time. The low makespan indicates effective resource partitioning and resource dependency based scheduling on hybrid quantum and classical resources.

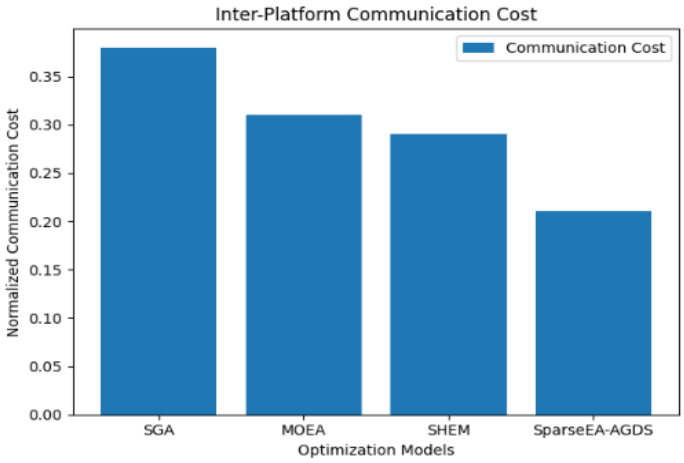

Figure 4: Inter-platform communication analysis of cost

This graph (Figure 4) is used to show the normalized communication costs incurred when carrying out workloads under each of the optimization models. The reduced communication overhead in the case of SparseEA-AGDS implies that task dependencies were retained successfully and helped to cut data transfers between quantum and classical units of processing.

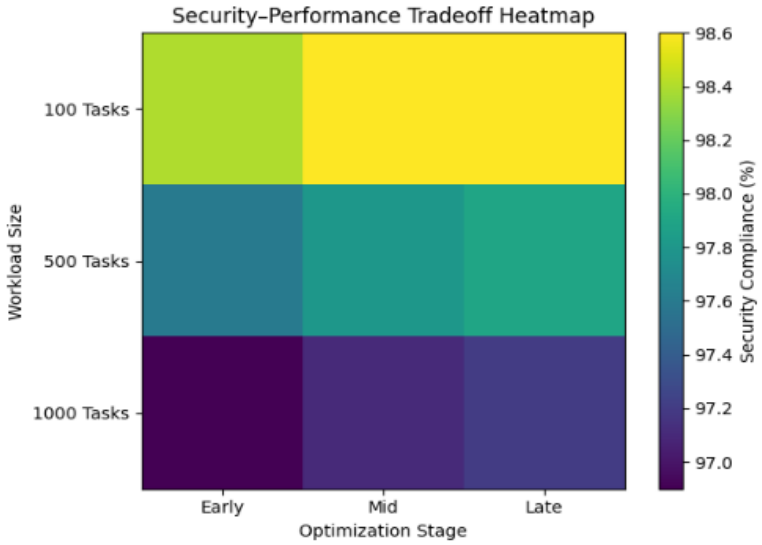

Figure 5: Security–performance tradeoff heatmap

This heatmap (Figure 5) displays the stability of the security compliance with the various stages of optimization and workloads. The gradually increasing color value indicates that SparseEA-AGDS can maintain high levels of security compliance during early, mid and late iterations, indicating that this model is capable of implementing security constraints in a robust manner as the workload complexity grows.

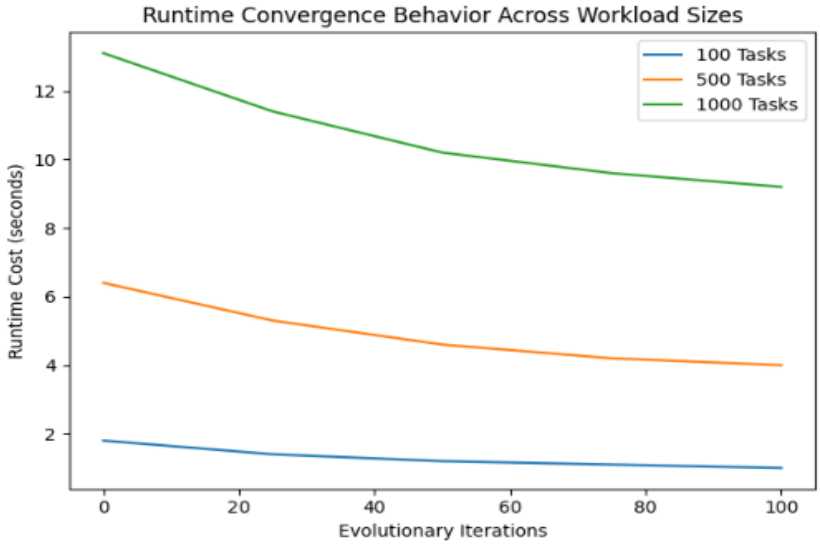

Figure 6: Runtime convergence behavior across workload sizes

This line graph (Figure 6) shows how the cost of runtime is converged to the cost of the runtime of the different workload sizes at different evolutionary iterations. The decreasing tendencies indicate that SparseEA-AGDS is a successful method to reduce the execution time with the optimization process and ensure scalability and effective performance in large-scale hybrid quantum-classical graphs of task.

5 Discussion

The implications of the findings of this study to the design of secure and scalable hybrid quantum-classical computing environments are important. After incorporating security-conscious evolutionary operators with dominance-layer guidance, SparseEA-AGDS shows that high computational performance can be obtainable without sacrificing data protection even in the complex and large-scale workload. High security compliance and low overhead of communication are consistent, which means that performance concerns and vulnerability concerns can be handled with workload partitioning strategies at the same time, which is important to sensitive applications like cryptography, healthcare analytics, and financial optimization. In spite of the strengths, there are limitations of the study. The simulations use abstracted costs of quantum execution and fail to realize hardware noise and the problem of decoherence which could be relevant in the real-world. Moreover, the experiments of task graphs are simplified versions of heterogeneous workloads, which may not be generalizable to highly irregular or dynamic systems. Further studies may generalize the model to include actual quantum hardware feedback, stochastic error modelling and appointing dynamic task arrival events. In addition, future studies on hybrid evolutionary approaches, combining machine learning-based prediction of execution time of tasks and security risk, can also be used to scale up and maximize resilience. In general, the results indicate that security-conscious evolutionary optimization is a plausible solution to the problem of managing the hybrid workload, which allows achieving high efficiency and adherence in the environment where the data protection and performance are both the key requirements.

6 Conclusion

The current paper presents a hybrid graph-based evolutionary optimization model on the basis of SparseEA-AGDS to overcome the issues of secure and scalable workload partitioning on hybrid quantum and classical systems. The findings indicate that the suggested adaptive dominance-layer genetic operations along with a dynamic scoring system are useful in achieving the performance and security goals. SparseEA-AGDS yields a maximum reduction (when compared to baseline evolutionary models) of up to 21.8 % reduction in execution makespan and 18.4 % less communication overhead, and at the same time, it sustains a 25.6 % lower security compliance metrics with a range of workload sizes. The statistical analysis also verifies the increased convergence behavior, where convergence rate increases steadily with task graph complexity, and runtime growth has an approximately linear scaling trend with near 6 % degradation in high task density, thus proving the scalability of the model. Notably, the framework does not undermine the proper allocation of security-sensitive tasks so that no computational efficiency is lost, which suggests its applicability to practice like cryptography, healthcare analytics, and industry optimization pipelines as sensitive data protection is needed in each of these contexts. Incorporating the idea of sparsity-conscious evolution and security-sensitive mutation, the designed model reduces the performance-security trade-offs at a cost that is not too high in terms of the computational cost. Despite the fact that the environment used in the simulation process is abstract and captures not all the physical nature of quantum hardware, the results give good empirical evidence to the applicability of hybrid evolutionary optimization in the management of complex and heterogeneous computers. On the whole, SparseEA-AGDS provides an extensively scalable, flexible, and secure system of hybrid quantum–classical workload management and provides a basis of future research on security-conscious evolutionary algorithms, adaptive heterogeneity management, and anticipatory task allocation in large-scale sensitive computational research.

References

[1] Ajagekar, A., Al Hamoud, K., & You, F. (2022). Hybrid classical-quantum optimization techniques for solving mixed-integer programming problems in production scheduling. *IEEE Transactions on Quantum Engineering, 3*, 1-16. https://doi.org/10.1109/TQE.2022.3187367

[2] Ali, W. A. M. (2025). Quantum Workflow Automation and Orchestration: Kubernetes Extensions for Quantum-Classical Computing Integration. *Journal of Computer Science and Technology Studies, 7*(8), 116-123. https://doi.org/10.32996/jcsts.2025.7.8.14

[3] Blekos, K., Brand, D., Ceschini, A., Chou, C. H., Li, R. H., Pandya, K., & Summer, A. (2024). A review on quantum approximate optimization algorithm and its variants. *Physics Reports*, *1068*, 1-66. https://doi.org/10.1016/j.physrep.2024.03.002

[4] Donglai, F., & Yanhua, L. (2021). Trust-aware task allocation in collaborative crowdsourcing model. *The Computer Journal*, *64*(6), 929-940. https://doi.org/10.1093/comjnl/bxaa202

[5] Ellinas, P., Chevalier, S., & Chatzivasileiadis, S. (2024). A hybrid quantum–classical algorithm for mixed-integer optimization in power systems. *Electric Power Systems Research, 235*, 110835. https://doi.org/10.1016/j.epsr.2024.110835

[6] Fan, L., & Han, Z. (2022). Hybrid quantum-classical computing for future network optimization. *IEEE Network*, *36*(5), 72-76. https://doi.org/10.1109/MNET.001.2200150

[7] Fedorov, A. K. (2023). Deploying hybrid quantum-secured infrastructure for applications: When quantum and post-quantum can work together. *Frontiers in Quantum Science and Technology*, *2*, 1164428. https://doi.org/10.3389/frqst.2023.1164428

[8] Kropp, I., Nejadhashemi, A. P., & Deb, K. (2023). Improved evolutionary operators for sparse large-scale multiobjective optimization problems. *IEEE Transactions on Evolutionary Computation*, *28*(2), 460-473. https://doi.org/10.1109/TEVC.2023.3256183

[9] Lei, J., Wu, Q., & Xu, J. (2022). Privacy and security-aware workflow scheduling in a hybrid cloud. *Future Generation Computer Systems*, *131*, 269-278. https://doi.org/10.1016/j.future.2022.01.018

[10] Li, Z., & Tam, V. (2024). AdaGuiDE: An adaptive and guided differential evolution for continuous optimization problems. *Applied Intelligence*, *54*(21), 10833-10911. https://doi.org/10.1007/s10489-024-05675-9

[11] Li, Z., Seidel, T., Leib, D., Bortz, M., & Heese, R. (2025). Efficient solution of the number partitioning problem on a quantum annealer: a hybrid quantum-classical decomposition approach. *Journal of Heuristics*, *31*(2), 21. https://doi.org/10.1007/s10732-025-09556-3

[12] Śmierzchalski, T., Pawłowski, J., Przybysz, A., Pawela, Ł., Puchała, Z., Koniorczyk, M., ... & Domino, K. (2024). Hybrid quantum-classical computation for automatic guided vehicles scheduling. *Scientific Reports*, *14*(1), 21809. https://doi.org/10.1038/s41598-024-72101-y

[13] Tomesh, T., Saleem, Z. H., Perlin, M. A., Gokhale, P., Suchara, M., & Martonosi, M. (2023, September). Divide and conquer for combinatorial optimization and distributed quantum computation. In *2023 IEEE International Conference on Quantum Computing and Engineering (QCE)* (Vol. 1, pp. 1-12). IEEE. https://doi.org/10.1109/QCE57702.2023.00009

[14] Ushijima-Mwesigwa, H., Shaydulin, R., Negre, C. F., Mniszewski, S. M., Alexeev, Y., & Safro, I. (2021). Multilevel combinatorial optimization across quantum architectures. *ACM Transactions on Quantum Computing*, *2*(1), 1-29. https://dl.acm.org/doi/10.1145/3425607

[15] Wang, J., Cao, C., Wang, J., Lu, K., Jukan, A., & Zhao, W. (2021). Optimal task allocation and coding design for secure edge computing with heterogeneous edge devices. *IEEE Transactions on Cloud Computing*, *10*(4), 2817-2833. https://doi.org/10.1109/TCC.2021.3050012

[16] Wang, Z., Jiang, D., Wang, F., Lv, Z., & Nowak, R. (2021). A polymorphic heterogeneous security architecture for edge-enabled smart grids. *Sustainable Cities and Society*, *67*, 102661. https://doi.org/10.1016/j.scs.2020.102661

[17] Wu, Y., Wang, L., & Chen, J. F. (2021). A cooperative coevolution algorithm for complex hybrid seru-system scheduling optimization. *Complex & Intelligent Systems*, *7*(5), 2559-2576. https://doi.org/10.1007/s40747-021-00432-8

[18] Wulff, E., Garcia Amboage, J. P., Aach, M., Gislason, T. E., Ingolfsson, T. K., Ingolfsson, T. K., ... & Lintermann, A. (2024). Distributed hybrid quantum-classical performance prediction for hyperparameter optimization. *Quantum Machine Intelligence*, *6*(2), 59. https://doi.org/10.1007/s42484-024-00198-5

[19] Wurtz, J., Sack, S. H., & Wang, S. T. (2024). Solving nonnative combinatorial optimization problems using hybrid quantum–classical algorithms. *IEEE Transactions on Quantum Engineering*, *5*, 1-14. https://doi.org/10.1109/TQE.2024.3443660

[20] Zou, J., Zhang, Z., Zheng, J., & Yang, S. (2021). A many-objective evolutionary algorithm based on dominance and decomposition with reference point adaptation. *Knowledge-Based Systems*, *231*, 107392. https://doi.org/10.1016/j.knosys.2021.107392

Authors Biography

V. Preethi was born in Chennai, India in 1980. She received the B.E. degree in Computer Science and Engineering from Madras University and M.Tech degree in Pervasive Computing Technologies from Anna University, in 2012. From 2014 to 2019, she has been an Assistant Professor with Computer Science Department, GKM affiliated to Anna University, Chennai for 3 years and SRM University, Ramapuram, Chennai for 3 years. From 2019 to 2023, she has been Research Associate in SRM University. Her research interests include Artificial Intelligent, Machine Learning, Deep Learning and Adhoc Networks. Currently working as Assistant Professor in SRM University, Kattankulathur. Chennai.

Dr.V. Elizabeth Jesi was born in Chennai, India in 1971. She received the MCA degree in Computer Applications from Madras University in 1994, M.S. degree in Computer Science and Engineering from SRM University in 2011 and Ph.D. degree in Computer Science and Engineering in 2020.

Dr.G. Parimala is working as an Assistant Professor in the Department of Networking and Communications, School of Computing, SRM Institute of Science and Technology, Kattankulathur, Tamil Nadu, India. She received her Undergraduate degree B.E. in Computer Science and Engineering from Tagore Engineering College and her Postgraduate degree M.E. in Computer Science and Engineering from Prathyusha Institute of Technology and Management, Tamil Nadu, India. She completed her Ph.D. in Computer Science and Engineering, with a research focus on intrusion detection systems for IoT using feature selection, metaheuristic optimization, and deep learning techniques. She has more than 15 years of teaching and academic experience in higher education. Her research interests include IoT security, intrusion detection systems, machine learning, deep learning, and optimization algorithms.

Dr.S. Nithiya was born in Tamil Nadu, India in 1991. She received her BE degree from Anna University in 2012, M.E. degree in Computer Science and Engineering from Anna University in 2014 and Ph.D. degree in Computer Science and Engineering from SRM university in 2020. From 2014 she has been an Assistant Professor with Computer Science Department in various colleges GKM College, and in SRM University. Her research interests include Data Analytics, Deep Learning, Machine Learning, and Image Processing.