Scientific Reports

https://doi.org/10.1038/s41598-026-41205-y

Article in Press

A cost-optimized medical digital twin framework for secure and efficient patient data management in smart healthcare

Received: 8 October 2025

Accepted: 18 February 2026

Published online: 28 February 2026

Cite this article as: Alotaibi F.M., Ahmad S., Akram T. *et al.* A cost-optimized medical digital twin framework for secure and efficient patient data management in smart healthcare. *Sci Rep* (2026). https://doi.org/10.1038/s41598-026-41205-y

Faisal Mohammed Alotaibi, Sadiq Ahmad, Tallha Akram, Sultan Alanazi, Moteeb Almoteri & Abdullah M. Alotaibi

A Cost-Optimized Medical Digital Twin Framework for Secure and Efficient Patient Data Management in Smart Healthcare

Faisal Mohammed Alotaibi[1*], Sadiq Ahmad[2], Tallha Akram[1], Sultan Alanazi[3], Moteeb Almoteri[4], and Abdullah M. Alotaibi[5]

[1]Department of Information Systems, College of Computer Engineering and Sciences, Prince Sattam bin Abdulaziz University, Al-Kharj, 11942, Saudi Arabia.
[2]Department of Electrical Engineering, COMSATS University Islamabad, Wah Campus, Pakistan.
[3]Department of Computer Science, College of Computer Engineering and Sciences, Prince Sattam bin Abdulaziz University, Al-Kharj, 11942, Saudi Arabia.
[4]Department of Management Information Systems, Business Administration College, King Saud University, Riyadh 11451, Saudi Arabia
[5]Research Center, King Fahad Medical City (KFMC), Riyadh, Saudi Arabia.
[*]Corresponding Author: Faisal Mohammed Alotaibi; faisal.alotaibi@psau.edu.sa

ABSTRACT

The increasing demand for personalized, real-time healthcare necessitates efficient, secure patient data management. Digital Twins (DTs) enable AI-powered monitoring and decision support but also introduce challenges related to latency, computational cost, and security. This paper proposes a cost-optimized, AI-driven Medical Digital Twin (MDT) framework that manages task allocation across heterogeneous edge, fog, and cloud infrastructures. The system is formulated as a tri-objective optimization model that jointly minimizes latency and operational cost while maximizing security, subject to resource and clinical-priority constraints. To solve this problem, three complementary approaches are developed: (i) an exact Integer Linear Programming (ILP) model for optimal benchmarking, (ii) a Patient-Aware Task Intelligence Greedy (PATI-Greedy) heuristic algorithm for low-latency decision-making, and (iii) a Hybrid Q-Learning Enhanced Genetic Algorithm (HybridQeGA) for scalable, near-optimal performance in complex environments. Extensive simulations in a smart ICU scenario with 4, 8, and 12 patients demonstrate that ILP consistently achieves the best objective values but is computationally impractical for large instances. PATI-Greedy executes rapidly with polynomial complexity, achieving results within 5–8% of ILP for small- to medium-scale workloads. HybridQeGA offers the closest match to ILP in larger problem sizes, with less than 3% deviation in overall objective value while maintaining scalability. Security-sensitive scenarios highlight HybridQeGA's adaptability, improving security scores by an average of 12% compared to PATI-Greedy. These findings establish a balanced trade-off between accuracy and computational efficiency, positioning the proposed framework as a robust and deployable solution for intelligent and trustworthy digital health ecosystems.

Table 1. Nomenclature

Symbol	Meaning
Sets and Indices	
P	Set of patients; index $p \in P$
T	Set of tasks in the MDT pipeline; index $t \in T$
N	Set of heterogeneous computing nodes (edge, fog, cloud); index $n \in N$
Decision Variable	
$x_{t,n}^{p} \in \{0,1\}$	Assignment variable: 1 if task t for patient p is executed on node n, 0 otherwise
Task/Node Parameters	
$C_{t,n}$	Cost of executing task t on node n
$L_{t,n}$	Latency for executing task t on node n
$S_{t,n}$	Security score when executing task t on node n (higher is better)

Continued on next page

Nomenclature (continued)

Symbol	Meaning
R_t	Computational resource demand of task t (e.g., MIPS-equivalent)
R_n	Available computational capacity of node n
Task Type and Priority	
$\delta_t \in \{1,2,3\}$	Lifecycle stage of task t: 1 =Sync, 2 =Inference, 3 =Update
ω_{δ_t}	Priority weight for the lifecycle stage of task t
Objective Weights and Thresholds	
$\alpha,\ \beta,\ \gamma$	Weights for cost, latency, and security terms in the composite objective
$S_{\min}$	Minimum acceptable security threshold
M	Big-M constant for conditional security enforcement (large positive scalar)
Normalized Metrics (used by heuristics)	
$C_{t,n}^{\text{norm}}$	Normalized cost for task t on node n
$L_{t,n}^{\text{norm}}$	Normalized latency for task t on node n
$S_{t,n}^{\text{norm}}$	Normalized security score for task t on node n
Auxiliary / Algorithmic Symbols	
R_n^{rem}	Remaining capacity of node n during assignment
$\text{Impact}(\delta_t)$	Data-driven impact score used to compute ω_{δ_t}
$Q(s,a)$	Q-learning value function for state s and action a in HybridQeGA
$M,\ G$	Population size (M) and generations (G) in HybridQeGA's genetic operators
Abbreviations and Acronyms	
DT	Digital Twin
MDT	Medical Digital Twin
EHR	Electronic Health Record
ILP	Integer Linear Programming (optimal solver)
PATI-Greedy	Patient-Aware Task Intelligence Greedy heuristic
HybridQeGA	Hybrid Q-enhanced Genetic Algorithm
ICU	Intensive Care Unit
HIPAA	Health Insurance Portability and Accountability Act
GDPR	General Data Protection Regulation
RL	Reinforcement Learning

1 Introduction

Modern healthcare systems are increasingly strained by the exponential growth of patient data, the need for real-time clinical decisions, and the demand for personalized treatment strategies. Healthcare data volumes are expected to increase at a rate of 35% annually and are expected to reach multiple zettabytes by 2030[1]. More sensors in wearables, the rapid growth of Internet of Things (IoT) devices, and AI-driven diagnostics have the potential to create multimodal data streams in the form of images, ECG signals, and biochemical markers at an unprecedented rate[2]. Although the appropriate data can help to improve patient care, there are multiple data-related issues, such as data storage costs, processing, and analysis. Although patient health data can essentially be maintained using traditional electronic health record (EHR) systems and cloud-based medical record databases, real-time changes in EHRs and cloud-based records are difficult to maintain[3]. Digital twins (DT) are considered among the most advanced technologies for addressing this problem. DT can track, retrieve, and analyze medical records in real time. Moreover, DT is a promising disruptive innovation in healthcare, enabling the development of customized clinical actions using artificial intelligence (AI)[4]. For example, it is possible to track a patient's cardiac rhythm and preemptively trigger therapeutic actions (cures) using DT of the heart before clinical interventions. Therefore, DT implementation in the health care system presents both opportunities and challenges. Although DT can process large amounts of data, the system must analyze, process, and store it securely after collection. Conversely, if the data is not handled accurately, it may threaten patients' privacy.

As sensitive patient data is stored across heterogeneous computing levels (edge, fog, and cloud), it is challenging to ensure data security while maintaining low latency and low computing costs[5]. These systems must simultaneously negotiate over limited resource allocation, cybersecurity threats, and processing latency, especially in environments that require higher reliability, such as intensive care units (ICUs) or remote patient monitoring systems. Medical Digital Twins (MDTs) are dependent on a physical patient. The MDTs must continuously build and update predictive models and perform decision-support

interventions. All of these activities are very resource-intensive and burden the available computational and communication networks[6,7]. On top of that, the compliance with system security and privacy, the unauthorized access, data leaks, and data breaches, which are detrimental in any setting, are required to add even more complexity to the health system[8]. Hence, a suitable approach requires a combination of access control, encryption, and adherence to the Health Insurance Portability and Accountability Act (HIPAA) and the General Data Protection Regulation (GDPR).

MDTs in clinical settings do not operate independently; they are most useful as decision-support tools, meaning that clinical judgment must always be exercised. The optimization framework to be discussed is aimed at improving explainability by employing understandable objectives that are cost, time, and security balanced, which allows for decision-making to be audited and understood by clinicians and administrators regarding the allocation of tasks. Transparency of objectives coupled with optimization techniques focused on the constraints of the system will minimize the occurrence of bias, thereby potentially balancing the disparate impacts on certain groups of patients, tasks, or the overall system. Furthermore, the design that directs audits and secures the resources in ways that are compliant with the data protection principles of the healthcare regulations, including HIPAA and GDPR, that are relevant to this case, such as data minimization, access restrictions, and accountability, will support the design principles aimed at integrating MDTs that are ethically unambiguous.

Since healthcare providers require scalable deployment options at minimal cost, making cost optimization a fundamental objective. The system implementation cost includes data transmission and storage, computational load, and security features, such as encryption and authentication.

AI plays a key role in analyzing large volumes of patient data to identify trends, predict health conditions, and customize treatments within the modern MDT system. However, the use of AI in clinical settings raises significant concerns about information privacy, algorithmic bias, and the need for explainable AI (XAI) models. As a result, there is a pressing need to develop AI-based MDT models that are both accurate and effective, safe, readable, explainable, and ethically sound. These concerns highlight the fact that, despite the unprecedented ability of AI to equip MDTs with a thorough evaluation of the system-level trade-offs, especially related to latency, cost, and security, are required to achieve successful practical implementation of AI in practice[9]. The first and the most significant issue is to balance the aforementioned three conflicting objectives: (i) an MDTs framework should have minimal system latency in order to make MDTs responsive in real time; (ii) operational expenses must be minimal to make MDTs affordable to institutions with limited budgets; and (iii) the level of data protection must be robust to ensure that patients remain confident and that regulatory requirements are met. The current literature tends to discuss these elements separately, for example, the reduction of latency through edge computing or the enhancement of security through cryptographic protocols. Nonetheless, the application in the clinical setting requires an integrative approach that simultaneously optimizes the latency, cost, and security.

Building on the discussion above, this paper proposes a new AI-based MDT system to manage related data safely, with low latency and cost efficiency, in the context of personalized health care. Our system is defined as a multi-objective optimization problem, which has the following objectives: (i) the minimization of the total cost of operation, (ii) the minimization of the latency of data processing and communication, and (iii) the maximization of data security. AI is also used to schedule, predict, and manage DT tasks on distributed computing resources in the most efficient way, while cryptographic protections and privacy-sensitive models protect the confidentiality of patient information. The suggested architecture combines edge computing to process data in real time, fog computing to analyze it at an intermediate level, and cloud computing to store data and train complex AI models over the long term. It uses sophisticated encryption, access control, and anomaly detection algorithms to protect patient data against unauthorized access and intrusion. In addition, it implements dynamic resource-allocation techniques in order to reduce operational costs and to make the computing resources efficiently used. Unlike current methods, which tend to focus on one aspect of MDT systems (e.g., data security or cost optimization), our framework is a comprehensive solution that addresses all three key issues.

The rest of this manuscript is structured in the following way. The literature review section examines related work in MDTs, edge-cloud computing, and healthcare data security. The section of the proposed model and problem formulation provides the AI-based MDT framework, the multi-objective optimization formulation, and three complementary solution strategies, including an exact integer linear programming (ILP) solver, the patient-aware task intelligence Greedy (PATI-Greedy) heuristic, and the hybrid Q-learning enhanced Genetic algorithm (HybridQeGA). The section on the solution approach defines the application scenario for the smart hospital ICU and describes the simulation setup to be used to test the framework. The experimental results are presented in the results and discussion section, where the findings are reported across a variety of objective-weighting scenarios, including convergence analysis and inter-scenario comparisons. Lastly, the conclusion summarizes the manuscript and outlines promising future research directions.

2 Literature Review

The increasing use of DTs in the medical field is transforming the way patients are cared for, enabling real-time monitoring and predictive decisions. Several studies have explored DTs from multiple perspectives. The article by Chen et al.[10] is a survey of

the current literature on the use of generative-AI-based human DTs in IoT healthcare systems, whereas Li et al. in [11] focus on network-based DTs enabled by generative AI and their importance in dynamically changing medical environments. Zhang et al. [12] discussed the conceptual basis and clinical effects of DTs in personalized healthcare, and Gourraud et al. in [13] described a patient-centric data model of DTs that promotes real-time decision-making. Similarly, Walton et al.[14] conducted a scoping review of health DTs, presenting key research themes and areas of application.

In addition to general frameworks, domain-specific implementations are studied, such as in cardiology and oncology. Rodrigues et al.[15] introduced AI-based algorithms and extended reality into cardiology DTs to enhance visualization of the patient's treatment process. Trayanova et al.[16] have shown that a heart DT system can be used to simulate treatment outcomes in pre-cardiac surgery. The authors in [17] suggested a DT ecosystem to optimize clinical operations in oncology. Apart from it, the generative AI plays an important role in the health DTs implementation. Ibrahim et al.[18] surveyed generative AI for generating synthetic medical data, which are indispensable to DT training. Similarly, the authors in[19] have highlighted the use of generative AI to create adaptive healthcare DTs.

Regarding system architecture, there has been increased research into MDT frameworks that use edge and fog computing. Jain and Patel[20] provided a design of a cloud-edge MDT system for personal healthcare services. Feng and Wang[21] focused on the real-time delivery of healthcare using fog-enabled DTs, and Xu and Li[22] proposed edge-intelligent MDTs for use in dynamic hospital settings to monitor patients in real time.

Related Work on Security, Optimization, and Deployment Strategies

Security, scalability, and deployment are the ongoing issues in multidisciplinary telemedicine research. Several methodologies have been proposed to improve the security and effectiveness of scheduling tasks. Mohammadi and Hosseinzadeh used federated reinforcement learning to obtain task scheduling in MDT systems[23]. He and Zhang developed a security-optimal deployment of DTs in edge AI environments[24]. To enable tamper-free record-keeping, Liu and Gao proposed a blockchain-based MDT architecture to address breaches of the integrity, confidentiality, and authenticity of data, records, and other information sources[25]. Nguyen and Pham used federated privacy-preserving learning to manage confidential health data[26], and Zhang and Yuan tested differential privacy schemes to ensure confidentiality in MDT settings[27].

Several surveys have provided detailed summaries of the findings from the vast MDT literature. The summary provided in[28] focused on diagnostic and interoperability issues, whereas the Frontiers review[29] considered the system-level implementation of MDTs and real-world limitations. Drummond et al.[30] described standardized definitions and functional attributes of patient DTs, deployment taxonomies, and lifecycle models were emphasized in the SAGE Digital Health survey[31]. Similarly, the authors in[32], who focused on AI-integrated healthcare twin systems, highlighted practice and ethical issues related to these systems. Lastly, specific case studies have been analyzed, demonstrating the relevance of MDTs to particular medical issues. Hu et al.[33] proposed an ECG-based DT that detects heart diseases in a personalized way, whereas Wang et al.[34] proposed dynamic patient-specific DTs to aid in the training of coronary intervention. Kuang et al.[35] introduced an atypical approach to simulating non-invasive DTs, called Med- Real2Sim, which is a physics-informed framework[36], simulator, and benchmark, and requires no training of any neural network. Taken together, these papers emphasize the various uses of DT and point to the need for combined frameworks that consider cost, latency, and security in practical implementations.

2.1 Research Gap and Comparative Insights

All these studies highlight the current trend in MDT research, with a focus on recent progress in personalization, real-time decision-making, and secure system integration. However, most of the current methods only maximize one of the aspects, i.e., latency, cost, or security, and show poor integration of these aspects in a single framework. In addition, few studies provide realistic validation through ICU or smart-hospital simulations, which limits their generalizability to operational deployments. This work addresses these gaps by proposing a tri-objective optimization model that balances the three factors: latency, cost, and security.

2.1.1 Research Gap

In spite of significant progress, there are still a number of unresolved issues:

- Single-objective focus: The current methodologies focus on only one of the three objectives, latency, cost, or security, and do not allow the combination of these goals in one framework.
- Not considered cryptographic overhead: The currently implemented AI-based scheduling algorithms rarely take into account the extra cost of computation of encryption and privacy-preserving operations, which are paramount in healthcare settings.
- Lack of validation in the real world: Most studies implement MDT frameworks in abstract or simulated situations and not in an actual ICU or smart hospital environment, which diminishes their perceived practical usability.

Ref.	Focus Area	Latency Handling	Security Approach	Cost/Resource Optimization
[10]	Survey on generative AI-driven human DTs	Limited (conceptual)	Not primary focus	Not addressed
[11]	Network DTs with generative AI	Yes (network-level)	Limited	Not addressed
[12]	Conceptual DT foundations in healthcare	Theoretical only	Not addressed	Not addressed
[15]	AI + XR for cardiology DTs	Yes	Limited	Not addressed
[17]	Oncology clinical operations DT	Moderate	Basic privacy	Limited
[20]	Cloud–edge MDT framework	Yes (edge-assisted)	Not addressed	Limited
[24]	Secure DT deployment in edge AI	Limited	Strong (cryptographic)	Not addressed
[25]	Blockchain-enabled MDT	Limited	Strong (tamper-proof)	High overhead
[26]	Federated privacy-preserving MDTs	Limited	Strong (federated learning)	Limited
Proposed model	**AI-driven MDT with tri-objective optimization**	**Yes (ILP + PATI-Greedy + HybridQeGA)**	**Yes (constraints + security-aware allocation)**	**Yes (cost-aware task placement)**

Table 2. Comparison of Related Work in Medical Digital Twins (MDTs)

- Lack of adaptive optimization: Dynamically adjusting resource placement, task-level security, and decision latency at heterogeneous computing tiers is a relatively unexplored territory of adaptive AI agents.

Table 2 shows that the available literature is usually focused on either system architecture, security, or latency, but seldom on all three goals simultaneously. This framework stands out because it simultaneously minimizes latency, reduces costs, and ensures data security within a single AI-based system, and it demonstrates its contributions through a smart ICU simulation. By directly addressing these gaps, the proposed tri-objective optimization framework will enable MDT research to advance to deployable, scalable, and secure digital healthcare systems.

2.2 Contribution

This work makes the following contributions to advancing AI-driven MDTs for secure, efficient, and scalable patient data management:

1. The placement problem of tasks across the edge, fog, and cloud layers is formulated as a unified integer linear programming model and optimizes the minimization of operational cost, end-to-end latency reduction, and the maximization of data security. Unlike prior work, which treats these goals separately, the proposed model allows systematic exploration of trade-offs.

2. A strict security criterion is presented through a minimum admissible security level, denoted S_{min}, that is used as a protection standard. Also, the formulation includes lifecycle-aware task priorities, represented as lifecycle-specific task priorities ω_{δ_t} (Sync, Inference, Update), which align with optimization, patient safety, and clinical workflows.

3. There are three complementary solution strategies that provide tradeoffs between optimality and scalability: (i) an exact ILP solver, which should be used to benchmark an offline problem; (ii) a lightweight PATI-Greedy heuristic, which should be used to make solutions during real-time scheduling; and (iii) an adaptive HybridQeGA algorithm that should be used to find solutions that are near-optimal when workloads change.

4. A systematic normalization and weighting model: A systematic normalization and weighting scheme $(C_{t,n}, L_{t,n}, S_{t,n}; \alpha, \beta, \gamma)$ is proposed, which offers clarity in the management of cost-latency-security trade-offs and allows sensitivity analysis in respect of different clinical priorities.

5. The framework is tested on a smart ICU case, and the results are reported for latency, breach probability, energy/cost, and task drop rates. The findings demonstrate that the ILP provides a theoretical maximum, but the PATI-Greedy and HybridQeGA methods offer scalable, practical trade-offs for real-world health care.

3 Proposed Model and Problem Formulation

This paper examines a smart hospital space equipped with IoT-enabled medical devices, including a wearable ECG monitor, an insulin pump, a smart inhaler, and AI-enabled imaging systems. Each patient is linked to a virtual counterpart, MDT, which is a real-time image of the patient's physiological and clinical condition. This synchronization is performed using multimodal data processed by the devices described above.

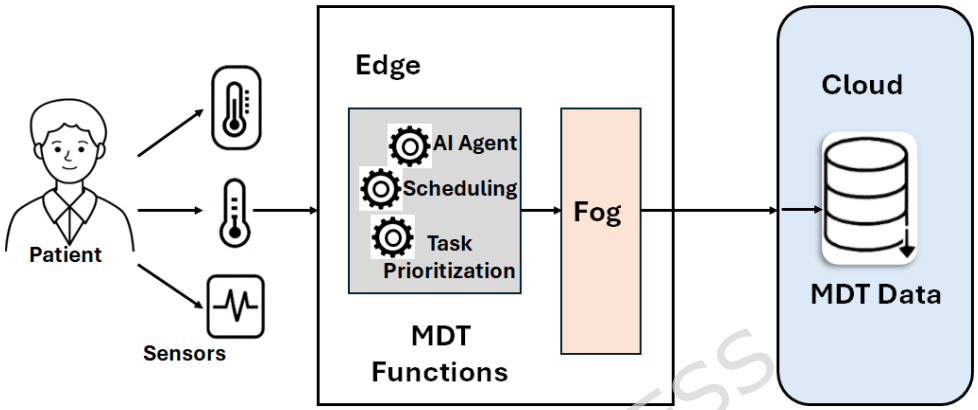

Figure 1. Architecture of the proposed AI-driven Medical Digital Twin (MDT) system showing patient data flow across edge, fog, and cloud layers. Latency-sensitive tasks are processed at the edge, while fog and cloud layers support analytics, storage, and model training with security-aware task allocation.

The MDT uses AI technologies to examine incoming data and perform the following functions:

- Forecast possible developments (for example, cardiac arrhythmia or quick changes in glucose levels, etc.),
- Suggest methods of treatment and appropriate medication dosages, and provide real-time notifications to medical practitioners about trending issues in health abnormalities.

The edge layer in the proposed framework, shown in Figure 1, provides a quick response to events requiring timely detection, such as anomaly detection, while the cloud-based server handles long-term activities, such as model retraining, which requires lengthy computation. In the proposed architectural framework, sensitive data is protected using confidentiality and integrity encryption. The framework shown in Figure 1 describes the MDT flexible distribution of compute and storage resources across the edge, fog, and cloud layers, with the goal of achieving optimal performance in terms of latency, operational cost, and security for end users. The performance parameters are also used to adapt the AI agents to changes in data distribution, patient states, resource availability, and data placement strategies. The MDT task allocation problem described in this manuscript uses a multi-objective optimization model that focuses on three parameters: data security, cost, and latency. A patient, their data, the processing activities, and the system's geographically distributed nodes are all designed to represent the system's components. The proposed framework also maintains anonymity, data security, and resource prioritization.

3.1 Sets and Indices

The model shows three primary components that engage the actors and the infrastructure woven into the healthcare system's complexity. Consider $\mathscr{P} = 1, 2, \ldots, P$, the set of patients that require computational assistance with their medical DTs. The *workings* set, denoted by $(\mathscr{T} = 1, 2, 3, 4, T)$, contains key healthcare tasks, such as sensor synchronization, predictive inference, and model updating. Finally, the set $\mathscr{N} = 1, 2, \ldots, N$ contains diverse system-architectural computing nodes, including edge, fog, and cloud computing devices. All the variable, symbols, abbreviation are summarized in Table 1.

3.2 Decision Variable

The task-to-node assignment model involves defining a binary decision variable, i.e., $x_{t,n}^{p}$, to represent task assignment. This variable is 1 when a task t related to the patient p is being run on a computing node n, and 0 otherwise. The optimization

process identifies the optimal values of these variables within the set of constraints, thereby ensuring that task assignments meet system requirements and target performance goals.

$$x_{t,n}^{p} = \begin{cases} 1, & \text{if task } t \text{ for patient } p \text{ is assigned to node } n \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

Parameters

Several parameters are used to measure the system's cost, latency, security, and resource requirements. The cost and latency of executing a task t in a node n are denoted by $C_{t,n}$ and $L_{t,n}$, respectively. Security score, $S_{t,n}$, is an inverse portrayal of the risk of data breach of a node n in a given task. The computational resources of each task t are represented by R_t, and the capacity of each node n is represented by R_n. In addition, tasks have a lifecycle stage, denoted by $\delta_t \in \{1,2,3\}$, which indicates the position of the task in the DT pipeline. The clinical importance of these task categories is distinguished by priority weights, represented by the omega weights of the time delta. Other scalar parameters are:

- α, β, γ: Weights for cost, latency, and security objectives
- $S_{\min}$: Minimum acceptable security level
- M: A large constant used to relax conditional constraints

3.2.1 Security Metric Modeling

Security score $S_{t,n}$ measures the reliability of performing task t on node n and is developed as a composite measure encompassing security dimensions relevant to healthcare data protection. Specifically, we define:

$$S_{t,n} = \sum_{k=1}^{K} \lambda_k \hat{S}_n^{(k)}, \tag{2}$$

In which the normalized score of node n on security attribute k is denoted by $\hat{S}_n^{(k)} \in [0,1]$ and its relative weight is denoted by λ_k, where the weight of each security attribute, i.e., k, must sum to one, i.e., $\sum_k \lambda_k = 1$.

The attributes to be taken into consideration are: (i) trust level in hardware (e.g., secure enclave or trust platform module (TPM) support); (ii) the strength of the data encryption at rest and at the transmission level; (iii) the access control and authentication mechanisms; and (iv) the ability to comply with healthcare regulations (e.g., HIPAA or GDPR). Such a formulation allows dynamic instantiation based on the deployment environment and provides a similar level of consistent, reproducible, security-conscious optimization.

3.3 Objective Function

The objective of the optimization problem is to devise a task-node allocation strategy that minimizes operational cost and latency and maximizes data security. These goals are combined into a single weighted function, with each term weighted by the task's clinical importance, ω_{δ_t}, to give higher priority to higher-priority tasks.

$$\min_{x_{t,n}^{p}} \underbrace{\alpha \sum_{p\in\mathcal{P}} \sum_{t\in\mathcal{T}} \sum_{n\in\mathcal{N}} \omega_{\delta_t} C_{t,n} x_{t,n}^{p}}_{\textbf{(1) Cost minimization}} + \underbrace{\beta \sum_{p\in\mathcal{P}} \sum_{t\in\mathcal{T}} \sum_{n\in\mathcal{N}} \omega_{\delta_t} L_{t,n} x_{t,n}^{p}}_{\textbf{(2) Latency minimization}} - \underbrace{\gamma \sum_{p\in\mathcal{P}} \sum_{t\in\mathcal{T}} \sum_{n\in\mathcal{N}} \omega_{\delta_t} S_{t,n} x_{t,n}^{p}}_{\textbf{(3) Security maximization}}. \tag{3}$$

Equation (3) combines three clinically motivated objectives into a single scalarized function, each modulated by the task priority ω_{δ_t} and by trade-off weights (α, β, γ). Cost, latency, and security are the three objectives considered in the tri-objective optimization model, with weighting coefficients of α, β, and γ. These parameters are used to characterize different clinical priorities in actual healthcare settings rather than prescribing a fixed configuration. The general-purpose operation is associated with balanced weightings; the privacy-critical conditions are characterized by security-dominant weightings; and the combined weightings can represent the optimization requirements as holistically as possible. This expression promotes a broad customization of the MDT model to a variety of deployment contexts.

(1) Cost term. The first one will impose a cost to the financial and operational overhead involved in assigning a task t to node n; $C_{t,n}$ may include compute time, storage, bandwidth, and energy costs. A larger value α increases the distrust of expensive allocations, and thus the tasks are redirected to low-cost nodes wherever possible.

(2) Latency term. The second term penalizes the end-to-end delay, $L_{t,n}$ (communication and processing). By increasing the parameter, the optimization will favor edge/fog nodes or underutilized resources, thereby increasing real-time responsiveness, a factor of particular importance for the tasks of *Sync* and time-critical *Inference*.
(3) Security term. The third component *rewards* higher security scores $S_{t,n}$ (hence the subtraction inside a minimization). A larger γ prioritizes nodes offering stronger protection (e.g., hardened enclaves, stricter access control), complementing the hard feasibility constraint $S_{t,n} \geq S_{\min}$.

The factor ω_{δ_t} weights each term by clinical importance. For example, if *Sync* tasks use $\omega_{\delta_t} = 1.0$ and *Update* tasks use $\omega_{\delta_t} = 0.4$, then cost and latency penalties (and the security reward) for *Sync* dominate the optimization, reflecting their urgency. This mechanism ensures that clinically critical operations are systematically favored in the decision-making process.

The relative priority of tasks is determined by task priority settings, but the absolute effects of each measure are also determined by their natural scales. In particular, the magnitudes of the terms in $C_{t,n}$, $L_{t,n}$, and $S_{t,n}$ can vary significantly, allowing one of the dimensions to be overrepresented by the optimization. To overcome this, the normalized forms of normalization: $C_{t,n}^{\text{norm}}$, $L_{t,n}^{\text{norm}}$, and $S_{t,n}^{\text{norm}}$, have been introduced in Eq. (3), or alternatively, the weighting coefficients (α, β, γ) can be adjusted to counter this. To ensure numerical stability and equalization of trade-offs in terms of objectives, the normalized metrics are clearly used in Algorithm 2. Lifecycle-conscious priorities and normalization together facilitate the optimization framework: the former addresses the urgency of clinical work, whereas the latter prevents the unfairness in the heterogeneous performance measures. This two-fold enabling mechanism gives the deployment specific tuning a principled basis in which the parameter options are required to mirror the facts of the target healthcare setting.

While task priorities govern the relative emphasis across lifecycle stages, the absolute impact of each metric also depends on its inherent scales. Specifically, $C_{t,n}$, $L_{t,n}$, and $S_{t,n}$ may differ significantly in magnitude, which can bias the optimization toward one dimension. To address this, normalized forms $C_{t,n}^{\text{norm}}$, $L_{t,n}^{\text{norm}}$, and $S_{t,n}^{\text{norm}}$ are introduced in Eq. (3), the weights (α, β, γ) can be tuned accordingly. In Algorithm 2, normalized metrics are explicitly applied to guarantee numerical stability and balanced trade-offs across objectives. Together, lifecycle-aware priorities and normalization enable flexible tailoring of the optimization framework. The lifecycle-aware priorities ensure clinical urgency, while the fairness across heterogeneous performance metrics is ensured by the latter. This dual mechanism provides a principled foundation for deployment-specific tuning, where parameter choices must reflect the realities of the target healthcare environment.

Finally, depending on deployment priorities, the weights (α, β, γ) and task priorities ω_{δ_t} may be tuned accordingly. For latency-critical systems, higher values of β and ω_{δ_t} for *Sync*/*Inference* tasks are appropriate. In budget-constrained settings α may be emphasized to avoid expensive placements, while in security-sensitive contexts, larger γ and stricter $S_{\min}$ thresholds should be applied.

3.4 Constraints

To ensure valid and practical task assignments, the following constraints are imposed:

1. **Unique Task Assignment:** Each task must be assigned to exactly one computing node. This constraint guarantees that no task is duplicated or omitted.

$$\sum_{n \in \mathcal{N}} x_{t,n}^{p} = 1, \quad \forall p \in \mathcal{P}, t \in \mathcal{T} \tag{4}$$

2. **Node Resource Capacity:** The total resource demand from assigned tasks must not exceed the available capacity of each node. This prevents node overload and ensures feasible deployment.

$$\sum_{p \in \mathcal{P}} \sum_{t \in \mathcal{T}} R_t x_{t,n}^{p} \leq R_n, \quad \forall n \in \mathcal{N} \tag{5}$$

3. **Security Constraint:** A task may only be assigned to a node if its security level meets the predefined minimum threshold $S_{\min}$. The big-M technique is used to enforce this condition only when a task is assigned[37].

$$S_{t,n} \geq S_{\min} - M(1 - x_{t,n}^{p}), \quad \forall p, t, n \tag{6}$$

3.5 Priority Weight (ω_{δ_t}) Assignment

To differentiate the urgency and importance of tasks, the model applies task-specific priority weights ω_{δ_t}. These can be determined through several strategies:

Task Type δ_t	Description	Example	ω_{δ_t}
1 (Sync)	Real-time sensing	ECG, vitals monitoring	1.0
2 (Inference)	AI-based prediction	Heart failure prediction	0.7
3 (Update)	DT model updating	Retraining, backup	0.4

Table 3. Task Prioritization Weights (ω_{δ_t}) Assigned to Different Stages of the DT Lifecycle

Algorithm 1 ILP Algorithm

Require: Number of patients P, tasks T, nodes N, weights α, β, γ, matrices C_{tn}, L_{tn}, S_{tn}, resource demands R_t, node capacities R_n, minimum security $S_{\min}$, priority weights ω_{δ_t}

Ensure: Optimal assignment of tasks to nodes

1: Initialize binary decision variable $x_{t,n}^p \in \{0,1\}$

2: Objective function:

$$\min_{x_{t,n}^p} \quad \alpha \sum_{p\in\mathscr{P}} \sum_{t\in\mathscr{T}} \sum_{n\in\mathscr{N}} \omega_{\delta_t} C_{t,n} x_{t,n}^p + \beta \sum_{p\in\mathscr{P}} \sum_{t\in\mathscr{T}} \sum_{n\in\mathscr{N}} \omega_{\delta_t} L_{t,n} x_{t,n}^p - \gamma \sum_{p\in\mathscr{P}} \sum_{t\in\mathscr{T}} \sum_{n\in\mathscr{N}} \omega_{\delta_t} S_{t,n} x_{t,n}^p.$$

3: Subject to:

- $\sum_{n=1}^{N} x_{t,n}^p = 1 \quad \forall t$
- $\sum_{t=1}^{T} x_{t,n}^p R_t \le R_n \quad \forall n$
- $x_{t,n}^p S_{tn} \ge x_{t,n}^p S_{\min} \quad \forall t, n$

4: Solve ILP using an exact solver

5: **Return:** Optimal task-to-node assignment matrix x

3.5.1 Expert-Driven Heuristic

This approach assigns fixed priority values based on expert knowledge of clinical relevance[38]. Synchronization tasks receive the highest priority, while non-urgent tasks like model updates receive lower values.

In this approach, fixed priority weights are manually assigned to each task category based on expert knowledge of clinical importance. As shown in Table 3, real-time sensing tasks are assigned the highest priority, followed by inference and update tasks. Higher weights indicate more urgent or critical processing requirements

3.5.2 Data-Driven Method

A dynamic weight assignment can be computed by normalizing task impact scores obtained from empirical performance data.

$$\omega_{\delta_t} = \frac{\text{Impact}(\delta_t)}{\max_{\delta} \text{Impact}(\delta)} \tag{7}$$

4 Solution Approach

The task assignment problem is formulated as a multi-objective optimization that balances cost, latency, and security under heterogeneous computing resources. Three strategies are presented to solve the problem: an exact ILP model to provide an optimal baseline, the lightweight PATI-Greedy heuristic, and HybridQeGA, which combines reinforcement learning with evolutionary operators for scalable near-optimal solutions.

4.1 Optimal Solution: ILP

The ILP formulation serves as the baseline for evaluating solution quality. It minimizes a weighted sum of cost and latency while maximizing security, adjusted by task-specific priority weights ω_{δ_t}.

ILP guarantees global optimality but exhibits exponential complexity, making it impractical for large-scale or real-time systems.

4.2 Heuristic Solution: PATI-Greedy Algorithm

The PATI-Greedy algorithm provides a lightweight approximation by greedily assigning each task to the node that minimizes a simplified objective. It respects resource and security constraints at every step.

Algorithm 2 PATI-Greedy Algorithm

Require: $P, T, N, \alpha, \beta, \gamma, C_{tn}, L_{tn}, S_{tn}, R_t, R_n, S_{\min}, \omega_{\delta_t}$
Ensure: Greedy assignment of tasks to nodes
Initialize residual capacities $R_n^{\text{rem}} \leftarrow R_n$
Initialize assignment matrix $x \leftarrow 0$
for each task $t = 1$ to T **do**
  bestScore $\leftarrow \infty$, bestNode $\leftarrow -1$
  for each node $n = 1$ to N **do**
    if $R_n^{\text{rem}} \geq R_t$ **and** $S_{tn} \geq S_{\min}$ **then**
      Compute local score:

$$s = \omega_{\delta_t}\left(\alpha C_{tn}^{\text{norm}} + \beta L_{tn}^{\text{norm}} - \gamma S_{tn}^{\text{norm}}\right)$$

      if $s <$ bestScore **then**
        bestScore $\leftarrow s$, bestNode $\leftarrow n$
      end if
    end if
  end for
  if bestNode $\neq -1$ **then**
    Assign $x_{t,bestNode} \leftarrow 1$
    Update $R_{bestNode}^{\text{rem}} \leftarrow R_{bestNode}^{\text{rem}} - R_t$
  end if
end for
Return: Greedy assignment x

The PATI-Greedy algorithm has polynomial complexity $O(|\mathscr{P}||\mathscr{T}||\mathscr{N}|)$, as each task checks all nodes before assignment. This efficiency makes it suitable for real-time or medium-scale deployments, though it may sacrifice global optimality.

4.3 Hybrid Q-Learning-based Enhanced Genetic (HybridQeGA) Algorithm

The HybridQeGA uses adaptive decision-making from Q-learning and integrates with genetic operators to enhance the search process in vast solution spaces. For example, in reward-driven learning, Q-learning directs task-to-node mapping, while GA facilitates movement from local minima.

The intricacy of HybridQeGA is a result of Q-learning repetitions and GA population changes. Considering a generation, for Q-update, it is $(O(|\mathscr{S}||\mathscr{A}|))$ whereas for GA on population of size M and across G generations, it is $(O(GM|\mathscr{P}||\mathscr{T}|))$. Thus, the overall complexity is polynomial and can be expressed as:

$$O\big(GM(|\mathscr{P}||\mathscr{T}||\mathscr{N}| + |\mathscr{S}||\mathscr{A}|)\big). \tag{8}$$

Even though it requires more computations than PATI-Greedy, HybridQeGA is still manageable for large-scale systems and often provides optimal solutions. Its scalable nature especially fits dynamic, high-load conditions like multi-patient healthcare systems.

To clarify the complementary roles of the three solution strategies, ILP serves as an exact optimization benchmark that provides the theoretical performance upper bound but suffers from exponential complexity. PATI-Greedy offers a lightweight, deterministic heuristic with polynomial complexity, making it suitable for real-time clinical decision-making. HybridQeGA bridges the gap by combining reinforcement learning–based adaptation with evolutionary search, enabling near-optimal performance under large-scale and dynamic workloads.

4.4 Comparative Summary

The differences among the three algorithms being addressed are threefold in terms of their computational characteristics and their suitability for deployment in various scenarios; the summary table of the tradeoffs of the algorithms being addressed is in Table 4. To summarize,

- ILP is an exact benchmark; however, it is not scalable to real-time or large environments with patients, tasks, and nodes.

Algorithm 3 HybridQeGA Algorithm

Require: $P, T, N, \alpha, \beta, \gamma, C_{tn}, L_{tn}, S_{tn}, R_t, R_n, S_{\min}, \omega_{\delta_t}$
Ensure: Near-optimal assignment via hybrid learning-evolution approach
1: Initialize Q-values $Q(s,a) \leftarrow 0$, GA population of candidate assignments
2: Objective function (reward):

$$r_t = -\Big(\alpha C_{p,t,n} + \beta L_{p,t,n} - \gamma S_{p,t,n}\Big)$$

3: **for** each generation **do**
4: **for** each candidate assignment in population **do**
5: Evaluate fitness using weighted objective
6: Update Q-values via:

$$Q(s_t,a_t) \leftarrow Q(s_t,a_t) + \alpha_{RL}\big[r_t + \gamma_{RL} \max_{a'} Q(s_{t+1},a') - Q(s_t,a_t)\big]$$

7: **end for**
8: Apply GA operators: selection, crossover, mutation
9: Replace population with offspring
10: **end for**
11: **Return:** Best assignment found

Algorithm	Computational Complexity	Optimality	Scalability
ILP (Optimal)	Exponential (NP-hard)	Global Optimum	Poor (only feasible for small-scale instances)
PATI-Greedy	$O(\lvert\mathscr{P}\rvert\lvert\mathscr{T}\rvert\lvert\mathscr{N}\rvert)$	Near-optimal (small gap)	High (suitable for real-time systems)
HybridQeGA	$O\big(GM(\lvert\mathscr{P}\rvert\lvert\mathscr{T}\rvert\lvert\mathscr{N}\rvert + \lvert\mathscr{S}\rvert\lvert\mathscr{A}\rvert)\big)$	Near-optimal (close to ILP)	High (scales well to large/dynamic systems)

Table 4. Comparison of ILP, PATI-Greedy, and HybridQeGA in Terms of Complexity, Optimality, and Scalability.

- PATI-Greedy has the best runtime and is most suitable for real-time, low-weight decision-making, where some error relative to the optimal solution is permissible.
- Of the three algorithms, HybridQeGA is the best in balancing runtime and accuracy in large, dynamic, multi-patient scenarios in attaining scalable, near-optimal solutions.

5 Results and Discussion

The simulation study was conducted to assess the performance of the proposed algorithms under varying objective-weight configurations and different patient-load conditions in a smart ICU. The simulation parameters and task settings were selected to reflect realistic MDT operations within an ICU, incorporating heterogeneous device capabilities, security constraints, and dynamic computational workloads. ECG signal synchronization, real-time anomaly detection, AI-based inference of patient deterioration, and periodic updates to DT models are the MDT tasks used to simulate ICU workload. These activities involve the clinical processes we use to track patients, make judgments, and maintain models current in the ICU.

To evaluate the performance of these three algorithms, we use ILP formulation (used as an optimal benchmark), the PATI-Greedy heuristic, the HybridQeGA algorithm, and a random baseline. Three experimental conditions are established to investigate the trade-offs between these algorithms. The parametric values of the three experimental scenarios vary in terms of the α, β, and γ parameters which are usually known as objective-weight configurations. A fine variety of such configurations provides a sensitive analysis in a structured manner. By adjusting the relative significance of the cost, latency and security as part of these weightings, it is possible to explore the resilience and flexibility of the suggested methods in a range of clinical priority. Similarly, it is possible to study the behavior of the algorithm and associated trade-offs through the variation of the weights of the objective functions. In the former two cases individual performance measures are being considered and the cost, latency

and security measures are being reported separately. Cost and latency were given equal weights while lowering the security importance ($\alpha = 0.4, \beta = 0.4, \gamma = 0.2$). Conversely, security is emphasized in Scenario 2 using ($\alpha = 0.1, \beta = 0.1, \gamma = 0.8$), resulting in more pronounced trade-offs among the three objectives. However, Scenario 3 is designed to accomplish the overall performance of the system through a composite objective function which combines cost, latency and security measurements at once. The simulation experiments were carried out with a patient count of P=4, 8 and 12, thus, indicating the small, medium and large-scale workload of the work of MDT. These workload levels were chosen deliberately to reflect the increasing complexity in the operations, hence making it possible to conduct a systematic assessment of scalability and robustness in different clinical settings.The following subsections present the detailed results and comparative analyses for each scenario.

5.1 Scenario 1: Balanced Cost-Latency with Light Security Weighting ($\alpha = 0.4,\ \beta = 0.4,\ \gamma = 0.2$)

This scenario is designed to give equal importance to cost and latency, while placing less importance on security. The latency metric implicitly presents the distributed MDT task associated computation and communication delays during execution. Explicit modeling of bandwidth variation, packet loss, and network congestion is beyond the scope of this study; however, in future work, it will be incorporated as a network-aware extension of the framework.

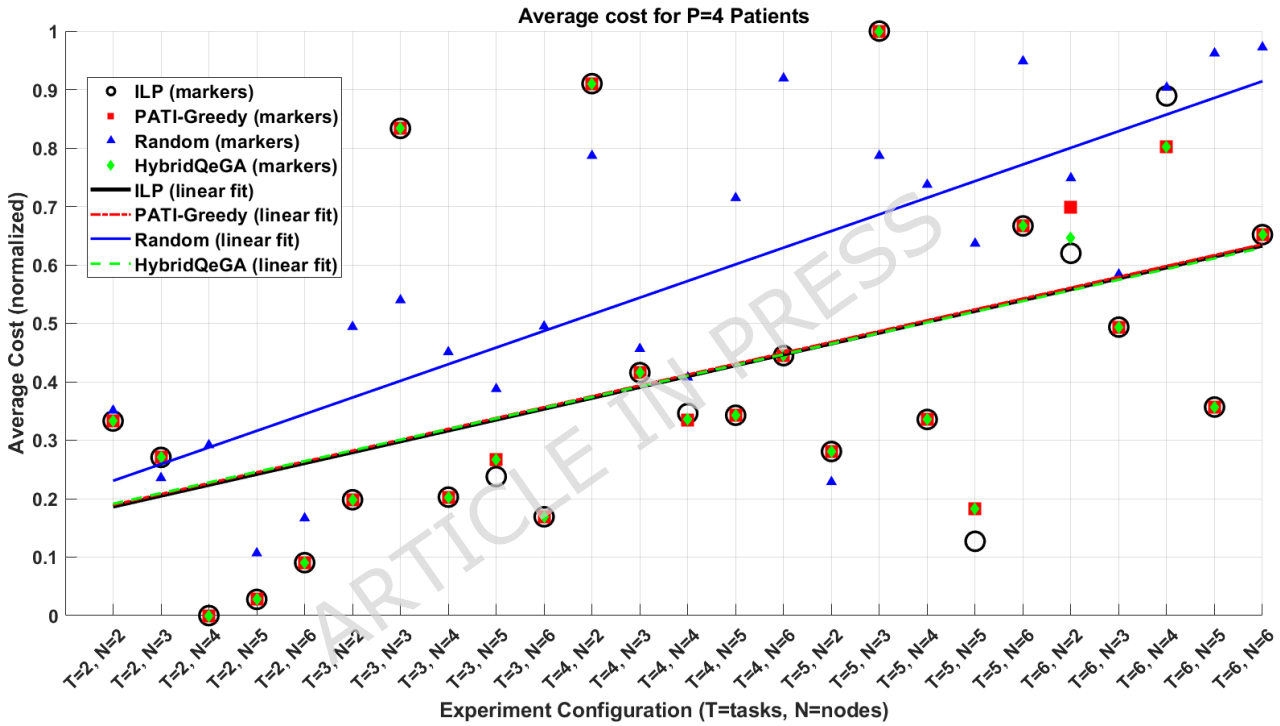

Figure 2. Average cost under Scenario 1 for $P = 4$ patients. The figure compares ILP (black circles), PATI-Greedy (red squares), HybridQeGA (green diamonds), and Random assignment (blue triangles)

Figure 2 shows the average cost output for $P = 4$ patients. The x-axis represents the number of tasks (T) and available nodes (N), while the y-axis represents the average execution cost. The complete figure shows the cost behavior among three algorithms, ILP (black circles), PATI-Greedy (red squares), and HybridQeGA (green diamonds). A random (blue triangles) assignment strategy was also added, in addition to proposed algorithms, as a lower-bound baseline, providing a sanity-check reference that underscores the importance of optimization-aware task placement in MDT environments. The ILP solution is the primary performance benchmark, providing the global optimum for small-scale problem instances. The three approaches, ILP, PATI-Greedy, and HybridQeGA, represent a complementary set of approaches spanning exact optimization, heuristic decision-making, and learning-based metaheuristics. This combination enables a meaningful evaluation of the trade-offs between solution optimality and computational scalability.

The random strategy is always the most expensive, a fact that becomes worse as the number of tasks increases. The ILP formulation, on the other hand, has the lowest cost of all configurations and can therefore be used as a benchmark for performance. In small examples (between $T = 2, N = 2$ and $T = 3, N = 3$), PATI-Greedy is a close approximation to ILP, but as problem complexity increases, HybridQeGA is highly adaptive and achieves near-optimal performance, with inconsistencies with ILP usually less than 5%. These results support the idea that the suggested algorithms are efficient at an ILP level and

computationally feasible. To explore how these cost dynamics change with increased workloads, we then discuss the results when the patient load is doubled.

Figure 3 shows a graph of the average performance of eight patients ($P = 8$). The doubling effect increases the patient's mathematical load, thereby increasing total costs. The most apparent escalation is always presented by the Random strategy, as it does not align with the other methodologies. The ILP methodology is highly optimal, but HybridQeGA is the closest approximation to ILP, and the variation across most settings does not exceed 5%. PATI-Greedy is not as scalable as HybridQeGA, but it is more competitive at medium-scale configurations based on task-node, and it exhibits increasingly large deviations at high loads. In this trend, the analysis was further extended to a larger workload ($P = 12$) to once again confirm the effect of scalability on the cost efficiency of the studied algorithms.

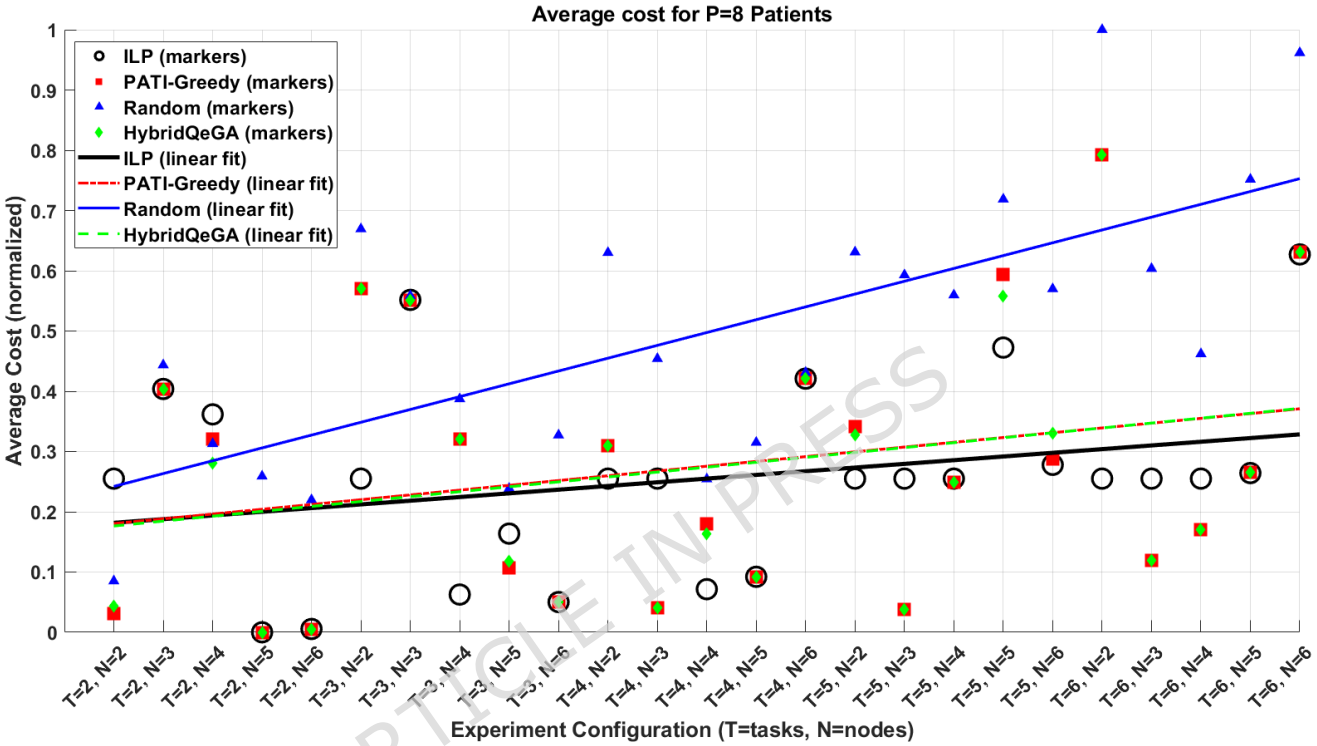

Figure 3. Average cost under Scenario 1 for $P = 8$ patients. The figure compares ILP (black circles), PATI-Greedy (red squares), HybridQeGA (green diamonds), and Random assignment (blue triangles).

The results of a cohort of ($P = 12$) patients, or a setting that is associated with a significant workload, are provided in Figure 4. The gap in the performance of the Random strategy and the structured algorithms is quite clear in this context. The ILP solution remains optimal, but its impracticality at this scale makes it unsuitable for real-time use due to its high computational cost. Instead, HybridQeGA has performance similar to that of ILP, with differences within a 3 percent range even at the largest task-node ratios. The PATI-Greedy methodology still works well, but it lags behind HybridQeGA, with gaps ranging from 7 to 10 percent compared with ILP in large-scale conditions. In turn, these results underscore the scalability and robustness of HybridQeGA to make production deployments of MDT.

Cost-minimization analysis in 4, 8 and 12 patient studies has a clear rank order, with ILP finding the least costly solution; HybridQeGA consistently approximates ILP at large workloads; PATI-Greedy is effective at smaller and medium-scale systems but becomes prohibitively expensive at large scale, and Random is competitive at smaller scale. This then leads to HybridQeGA as the most viable near-optimal method of cost-sensitive MDT operations. The next crucial key performance indicator in MDT framework is latency. Latency is vital, as it helps health administrators make real-time MDT operational decisions. When we look into latency results with the same balanced weighting, we better understand how different algorithms manage the trade-off between promptness and cost. The metric of latency that is used in this study also implicitly captures the computational and communicational delays of distributed MDT task execution across the edge, fog, and cloud levels. Although the current study does not provide a concrete network-level model that integrates bandwidth variability, packet loss, and congestion, these phenomena are still reflected indirectly in the updated latency aggregation figures. The integration of a network-sensitive, fine-grained modeling technique is thus a promising avenue for future improvements to the framework.

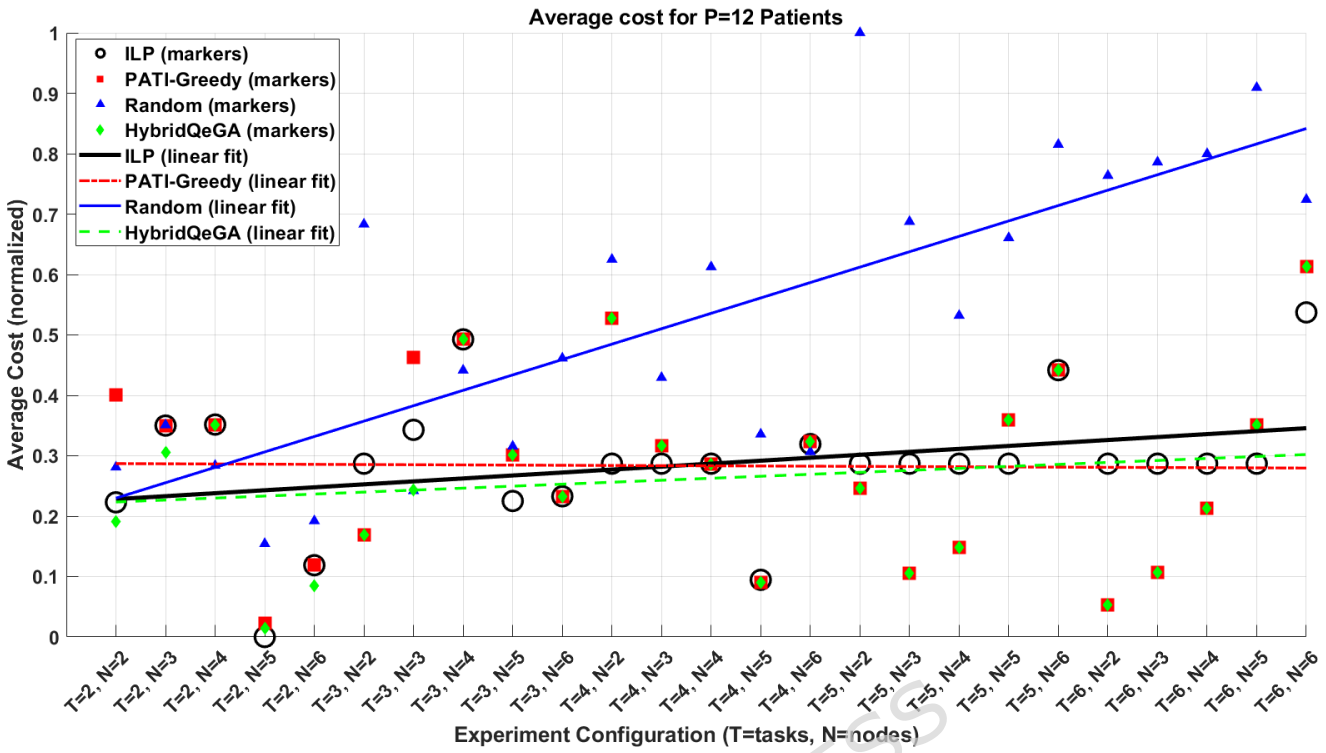

Figure 4. Average cost under Scenario 1 for $P = 12$ patients. The figure compares ILP (black circles), PATI-Greedy (red squares), HybridQeGA (green diamonds), and Random assignment (blue triangles).

The results in Figure 5 show the mean latency for $P = 4$ patients. The horizontal axis shows the task-node settings, which are similar to those shown in the figures above, and the vertical axis shows the normalized end-to-end task latency. Achieving the lowest possible latency, ILP (black circles), is always positioned as the benchmark. PATI-Greedy (red squares) also has near-optimal latency, especially in small- to mid-scale systems, because it has an explicit latency-minimization policy. HybridQeGA (green diamonds) is also competitive, often outperforming PATI-Greedy at large sizes due to its ability to avoid local minima. Random (blue triangles), on the other hand, generates the worst latencies in every situation. To analyze the development of latency characteristics with increasing workload intensity, the analysis presented below considers a medium-scale setup with 8 patients. The results of $P = 8$ patients are shown in Figure 6. An increase in patient load results in a monotonic increase across all methods considered. The ILP approach will always minimize delay to the greatest extent possible, and HybridQeGA will always approximate the ILP solution, which is better than PATI-Greedy when there are more task-node associations. PATI-Greedy is close to the ILP benchmark on small-scale architectures, but the performance gap between them increases with system size, a characteristic of its limited flexibility. The Random strategy degrades exponentially with problem size, underscoring its inefficiency. To fully analyze these scalability patterns under high system demand, the analysis is extended to a large-scale system of 12 patients as shown in Figure 7. At this scale, random assignment strategies cause too large a latency, making them impractical. The ILP method is optimal but comes at a high computational cost. HybridQeGA provides the most consistently near-optimal performance, with latency values close to those of ILP even under high-load conditions. PATI-Greedy shows a clear performance lapse in larger cases, with an 8-12% increase in latency compared with ILP, highlighting its disadvantages in large-scale operations. These results demonstrate the scalability benefit of HybridQeGA to latency-intensive MDT settings.

Across the 4 to 12 patient cohorts, the latency measures support the claim that the ILP approach performs best, whereas the HybridQeGA algorithm is the most efficient in scaling, keeping latency close to optimal levels. The PATI-Greedy algorithm shows good performance for low-load settings but fails as load intensity increases. This trend shows the importance of the HybridQeGA in MDT operations, where time is a critical parameter. Since information integrity and privacy are among the most critical issues in MDT cases, it is necessary to verify these algorithms for security. Subsequent outcomes outline the average security performance realized on the same balanced weighting regime.

Clinically, there is a direct relationship between the minimization of latency and the speed of response and identification of critical patient events in the intensive care unit. Lower end-to-end latency makes it easier to detect anomalies in time and

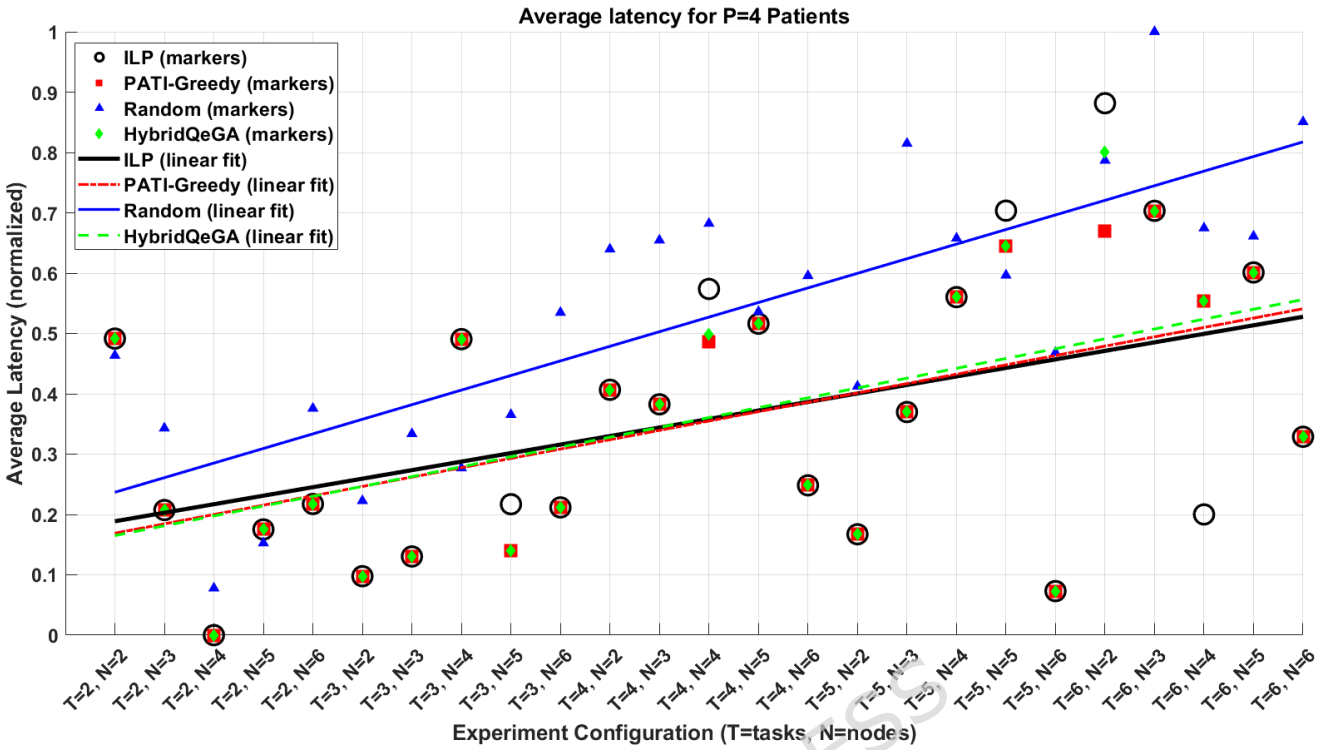

Figure 5. Average latency under Scenario 1 for $P = 4$ patients. The figure compares ILP (black circles), PATI-Greedy (red squares), HybridQeGA (green diamonds), and Random assignment (blue triangles).

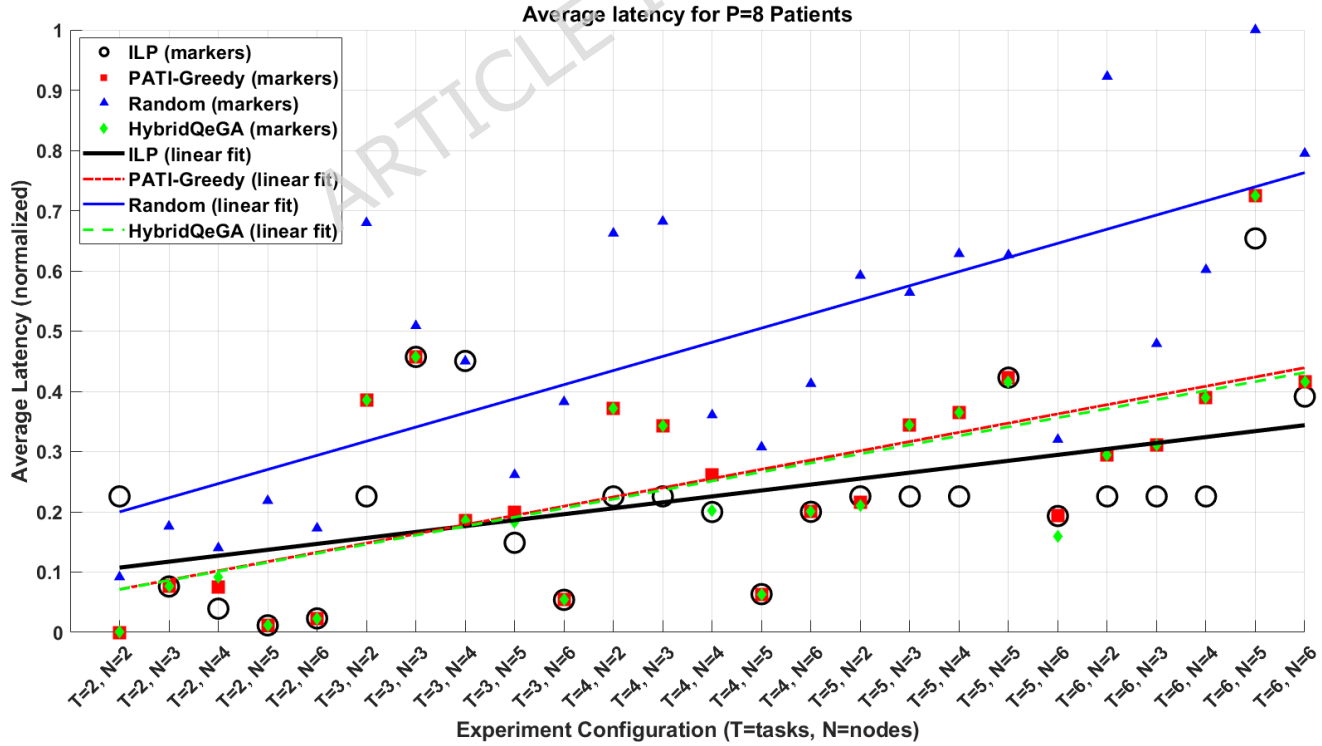

Figure 6. Average latency under Scenario 1 for $P = 8$ patients. The figure compares ILP (black circles), PATI-Greedy (red squares), HybridQeGA (green diamonds), and Random assignment (blue triangles)..

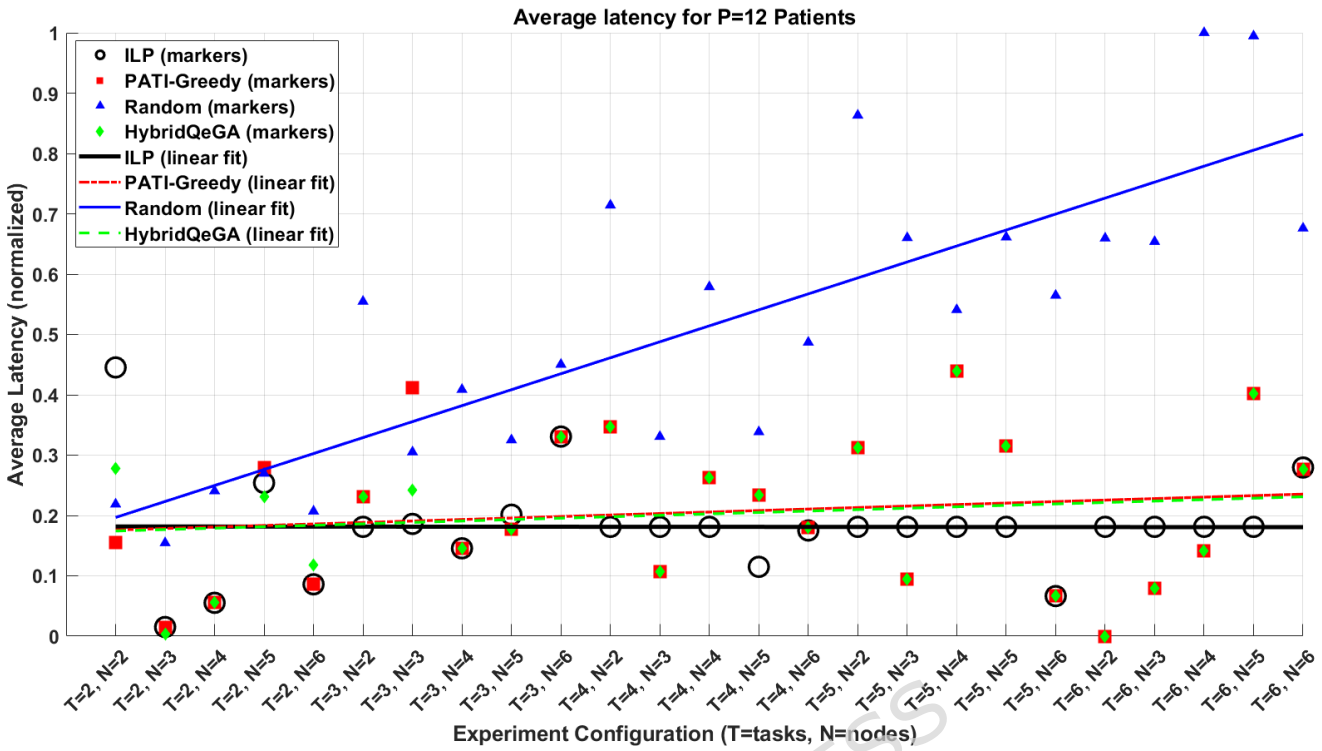

Figure 7. Average latency under Scenario 1 for $P = 12$ patients. The figure compares ILP (black circles), PATI-Greedy (red squares), HybridQeGA (green diamonds), and Random assignment (blue triangles)..

strengthen decision-support systems, which are crucial for aligning and inferring processes, including vital-sign monitoring in real-time and early-warning systems. As a result, the low latency achieved by HybridQeGA methods is optimal and thus does not improve patient safety by increasing response times in emergency situations. Figure 8 is a representation of the normalized security scores of $P = 4$ patients. The normalized security value is reflected on the y-axis in this graph. ILP has the best security scores because it allocates tasks to the safest nodes in an optimal manner. HybridQeGA and PATI-Greedy algorithms outperform Random, which offers the lowest security due to its non-discriminative assignments. HybridQeGA is similar to ILP, demonstrating its ability to adapt to stricter security requirements.

To examine the development of security performance with increasing workload complexity, the future recombination of outcomes is directed toward a medium-sized configuration with patients $(P = 8)$. The results for this eight-patient case are shown in Figure 9. The ILP methodology is the best approach for the increase in workload. The HybridQeGA method still estimates ILP with high fidelity and often results within 5% of optimal. The PATI-Greedy algorithm has moderate effectiveness, but of course, it is not as scalable as HybridQeGA to the number of tasks and nodes. On the other hand, the Random strategy continues to be underperforming, with less-than-optimal security guarantees.

Building on the results above, the current research is conducted with a considerably larger cohort (12 patients), allowing a more rigorous test of the algorithm's robustness. The results of this setting, indicated in Figure 10, are the most challenging case, i.e., $P = 12$. The ILP scheme still provides the best security levels, though HybridQeGA will always approach the optimal threshold, which is practically indistinguishable, hence providing near-perfect security. PATI-Greedy, by contrast, has a larger performance gap, a symptom of its inefficient behaviour in a security-dominated scenario. On the other hand, the random approach performs worst, underscoring the urgency of an ethically informed structured-optimization paradigm in the MDT setting. The above observations thus support the notion that HybridQeGA is the most reliable near-optimal solver when security constraints are imposed heavily.

The statistics for the security measures across the 4, 8, and 12-patient samples are organized in a similar manner, and they clearly indicate that the ILP model is used as the standard case. The HybridQeGA technique achieves near-optimal security performance for increasing system scale and maintaining cost-efficiency. Although the PATI-Greedy algorithm shows a lower cost than the random assignment strategy, its performance is lower than HybridQeGA under high-demand conditions. Conversely, consistent poor performance of the random baseline highlights the requirements of optimization-driven strategies for secure MDT deployment. In addition to comparative performance evaluation, the convergence behavior of each algorithm

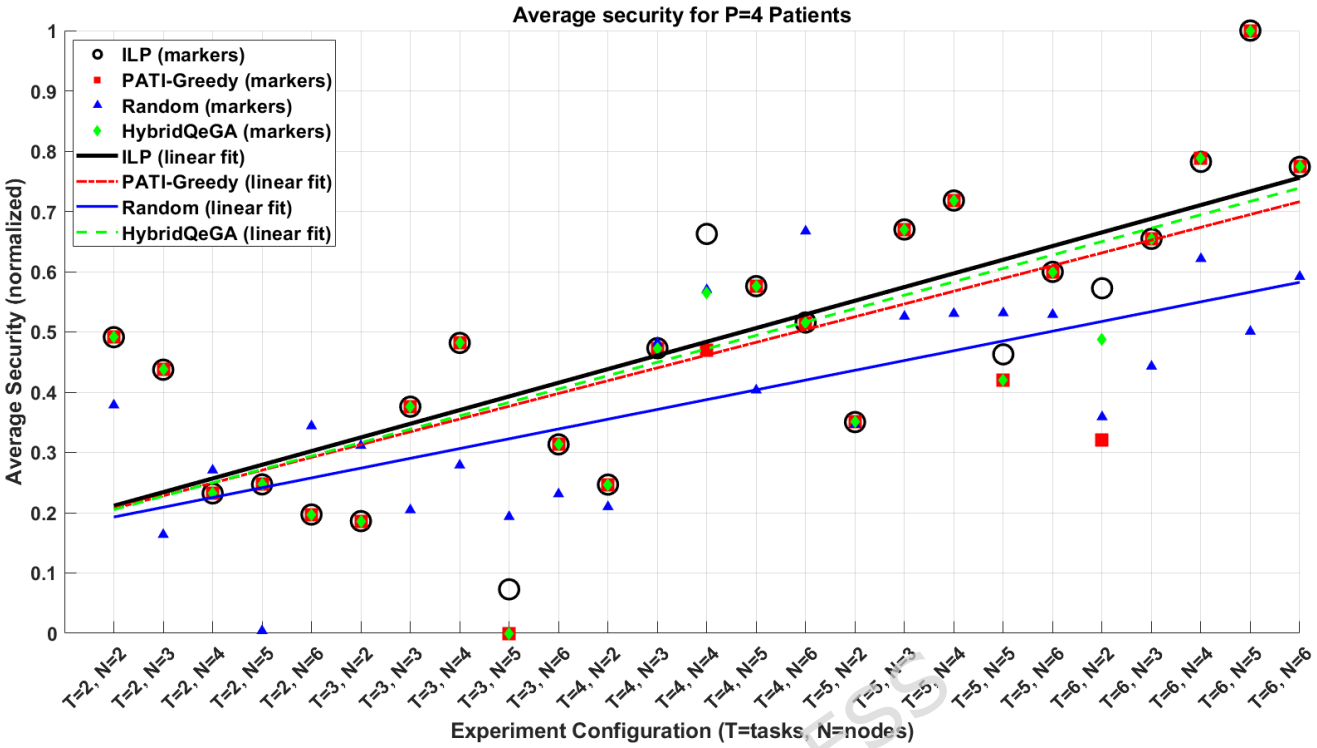

Figure 8. Average security under Scenario 1 for $P = 4$ patients. The figure compares ILP (black circles), PATI-Greedy (red squares), HybridQeGA (green diamonds), and Random assignment (blue triangles).

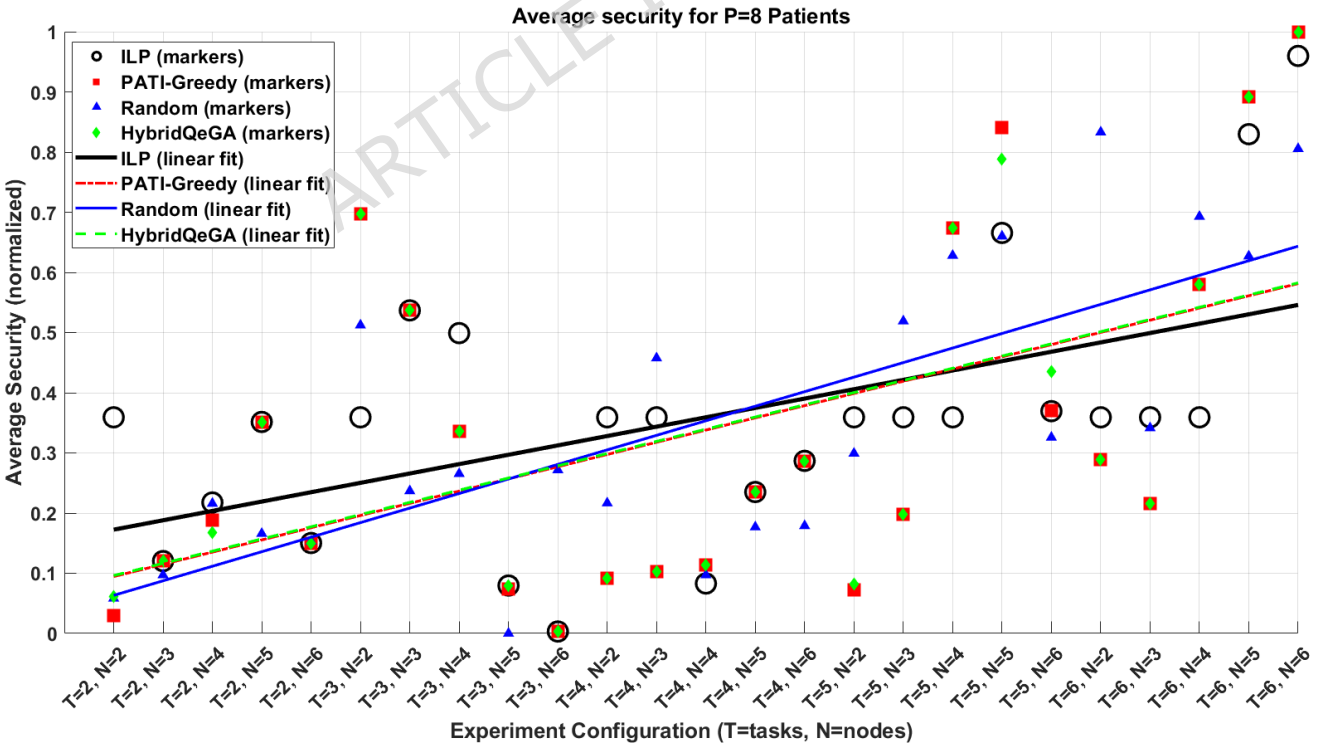

Figure 9. Average security under Scenario 1 for $P = 8$ patients. The figure compares ILP (black circles), PATI-Greedy (red squares), HybridQeGA (green diamonds), and Random assignment (blue triangles).

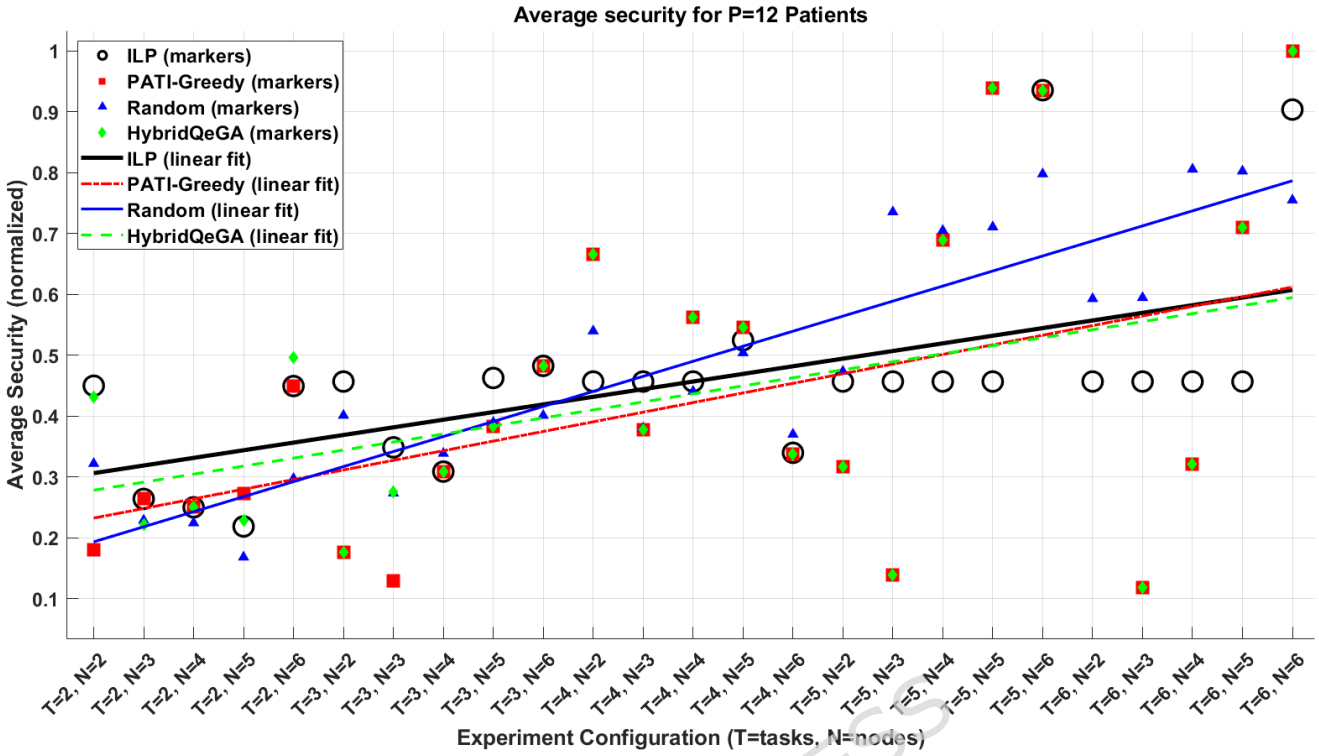

Figure 10. Average security under Scenario 1 for $P = 12$ patients. The figure compares ILP (black circles), PATI-Greedy (red squares), HybridQeGA (green diamonds), and Random assignment (blue triangles).

was analyzed to assess stability and learning dynamics. The convergence curves shown in Figure 11 show that HybridQeGA approaches the ILP benchmark, while PATI-Greedy converges rapidly at the cost of reduced accuracy. In comparative analysis, the random strategy shows unstable convergence behavior throughout the learning process.

5.2 Scenario 2: Security-Dominant Weighting ($\alpha = 0.1$, $\beta = 0.1$, $\gamma = 0.8$)

In this case, security is prioritized over cost and latency, as in the healthcare environment with advanced diagnostic and monitoring applications, where safeguarding patient data is of utmost importance. Consequently, jobs are often processed on more secure computing nodes, which in turn raises the system's operational cost and increases delays. To study the effects of such security-biased weighting on cost performance, we begin with different patient-load scenarios and use the smallest scale, $P = 4$. In Figure 12, we see that regardless of the security restrictions, the ILP model gets to the lowest possible cost. HybridQeGA yields results within 5% of the lowest possible cost. PATI-Greedy is less optimal than HybridQeGA, producing more costly solutions, but it still finds feasible solutions. The random approach is the least optimal, as it incurs the highest cost, and that cost is highest in the most secure mode.

In the security-dominant scenario, the cost increases with the number of patients. To perform a moderate-demand-based analysis, the number of patients is increased to a medium-scale setting with $P = 8$. Figure 13 presents an increase in overall costs because more costly security-related tasks are assigned. The performance of HybridQeGA remains close to that of ILP in scheduling, whereas the complexity of PATI-Greedy increases. Random increases rapidly due to the absence of systematic allocation.

To verify the heavy workload behavior, another analysis is performed with $P = 12$ patients. This is the upper limit of system complexity examined in this study. In Figure 14 with 12 patients, it is evident that the cost penalty is the most pronounced. HybridQeGA continues to outperform PATI-Greedy, being within 6% of ILP, while PATI-Greedy deviates by 10% or more. Random reaches unacceptable levels of costs.

Across all patient scales, ILP establishes the benchmark; HybridQeGA consistently tracks ILP closely, PATI-Greedy remains functional but less scalable, and Random is the worst. Having examined cost behavior under security-dominant weighting, we next explore latency performance to understand how prioritizing protection affects system responsiveness. Figure 15 shows that latency rises compared to Scenario 1 because tasks are routed to secure nodes with slower processing. ILP results in the lowest latency values. HybridQeGA also performed comparably well, and PATI-Greedy begins to show lag. However, Random

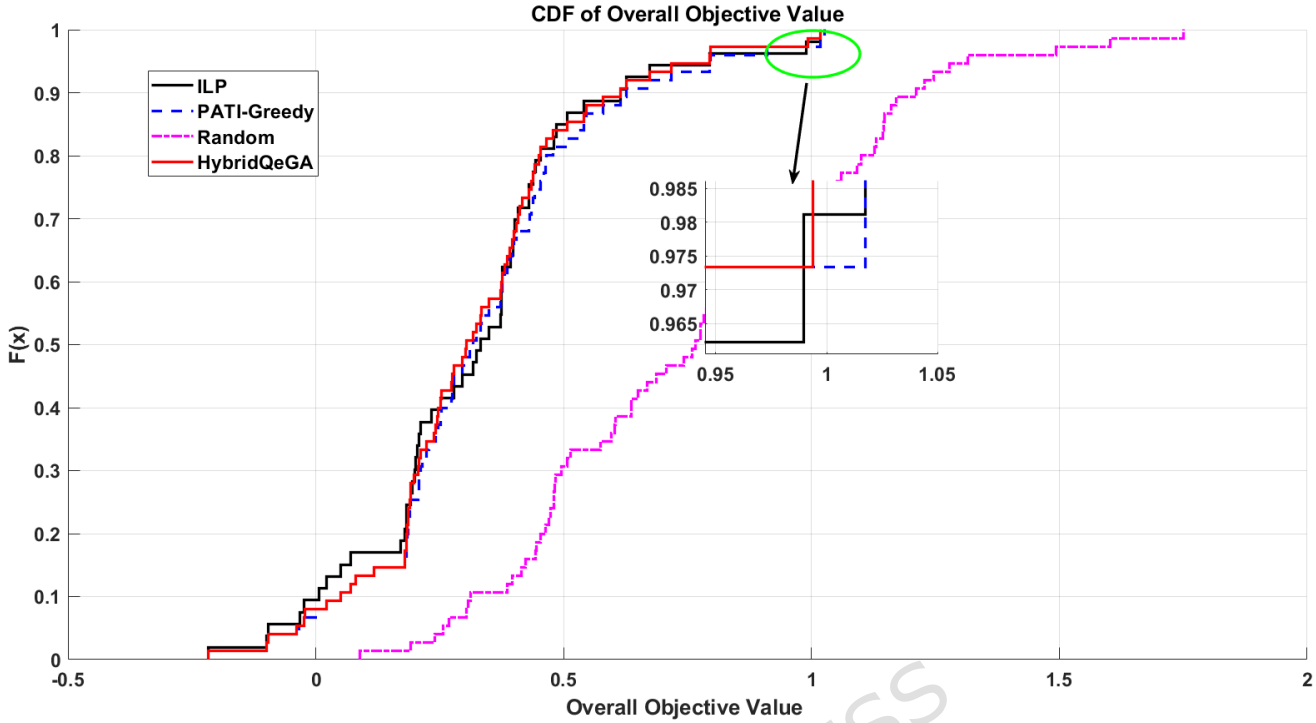

Figure 11. Convergence of objective values under Scenario 1 (balanced cost–latency weighting). HybridQeGA converges close to the optimal ILP solution, PATI-Greedy converges faster but suboptimally, while random assignment shows unstable behavior..

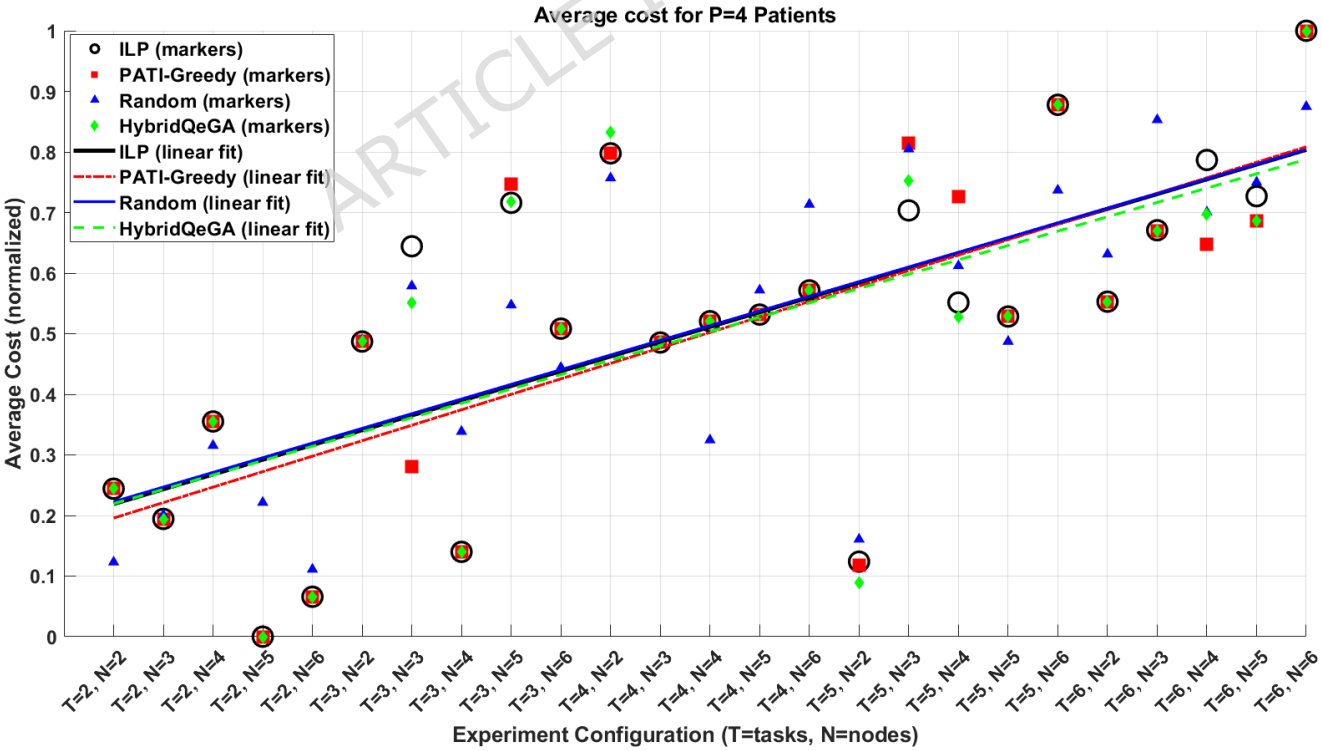

Figure 12. Average cost under Scenario 2 for $P = 4$ patients. The figure compares ILP (black circles), PATI-Greedy (red squares), HybridQeGA (green diamonds), and Random assignment (blue triangles).

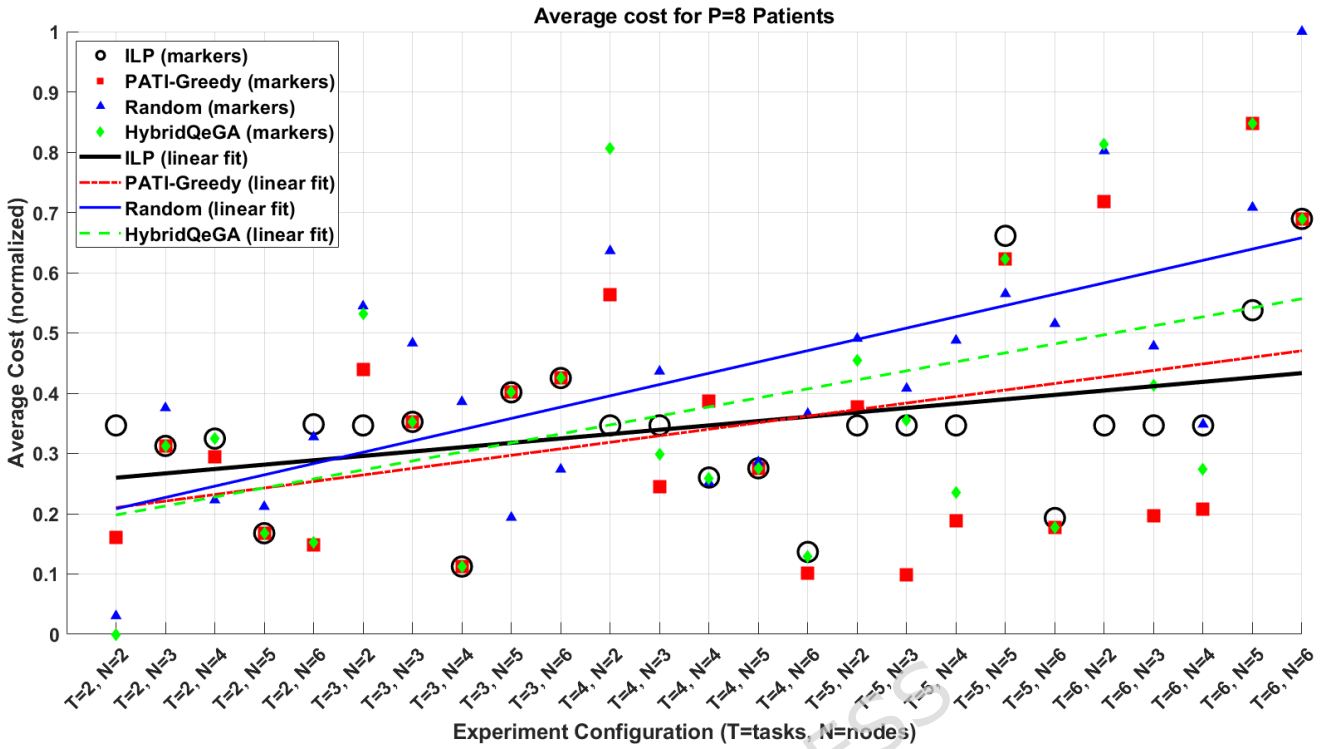

Figure 13. Average cost under Scenario 2 for $P = 8$ patients. The figure compares ILP (black circles), PATI-Greedy (red squares), HybridQeGA (green diamonds), and Random assignment (blue triangles).

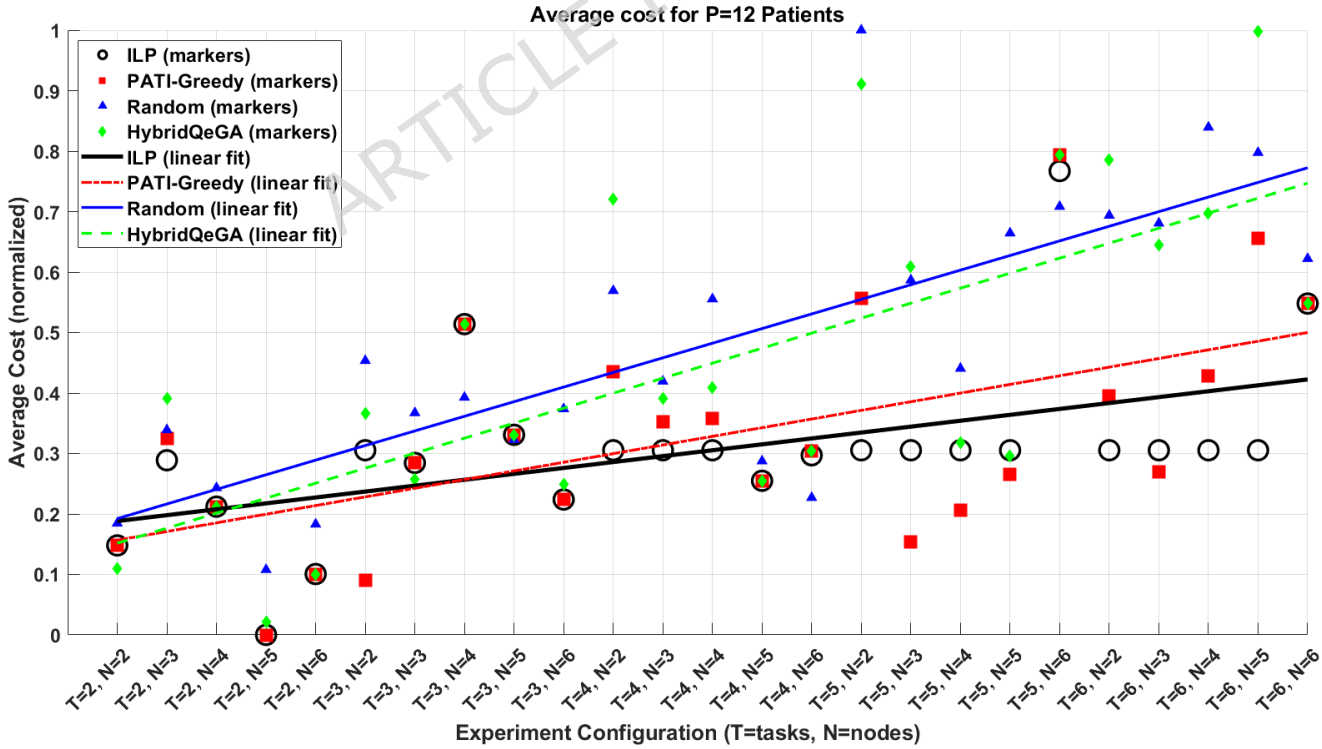

Figure 14. Average cost under Scenario 2 for $P = 12$ patients. The figure compares ILP (black circles), PATI-Greedy (red squares), HybridQeGA (green diamonds), and Random assignment (blue triangles).

produces the highest delays.

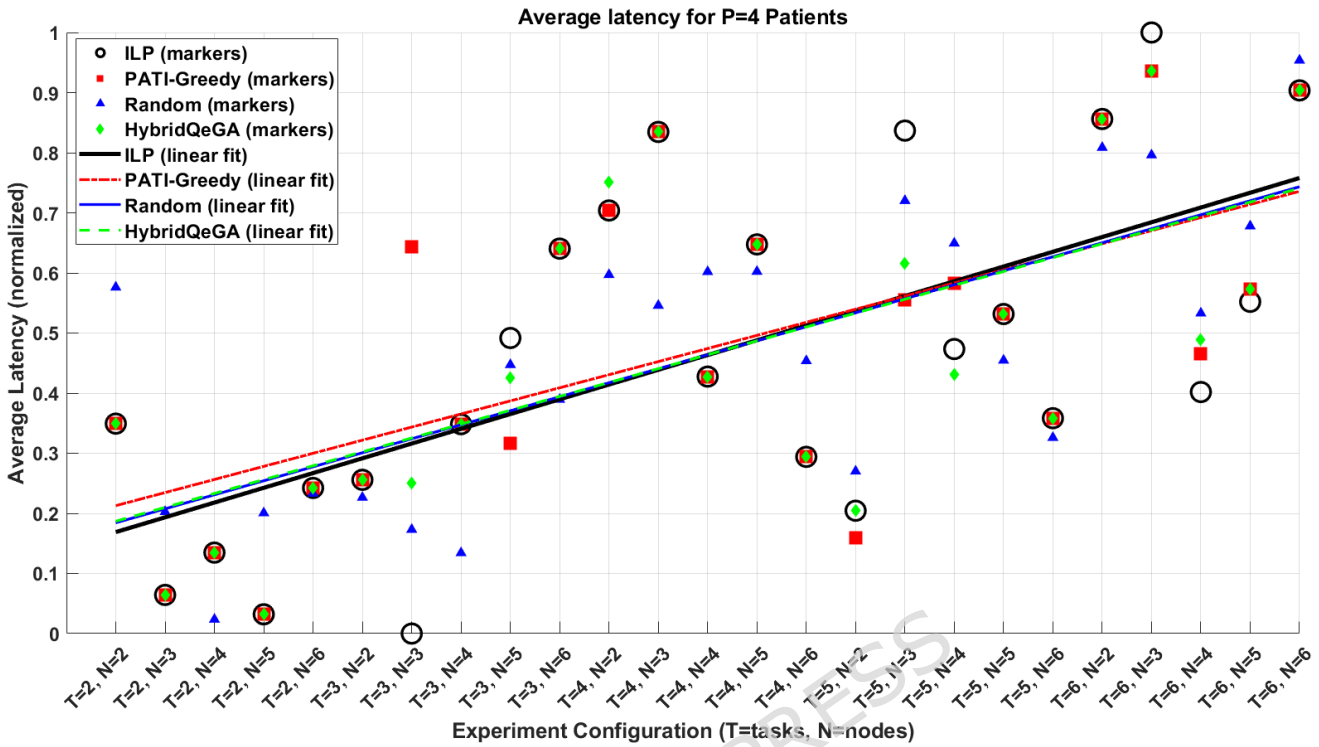

Figure 15. Average latency under Scenario 2 for $P = 4$ patients. The figure compares ILP (black circles), PATI-Greedy (red squares), HybridQeGA (green diamonds), and Random assignment (blue triangles).

As with cost, latency rises rapidly as we increase the workload. The following results for $P = 8$ patients represent these emerging trends. Figure 16 illustrates that with 8 patients, latency is enhanced. However, HybridQeGA still maintains near-optimal performance, closely following ILP. PATI-Greedy deviates more significantly, while Random remains highly inefficient. To comprehensively analyze the scalability under maximum load, latency results are analyzed for $P = 12$ patients, providing deeper insight into algorithm performance in large-scale, security-prioritized MDT settings. Figure 17 confirms that at 12 patients, HybridQeGA maintains scalability advantages, with latencies only 7-8% above ILP. PATI-Greedy falls further behind, while Random deteriorates drastically. While the latency results balance the trade-off between timeliness and security, it is important to analyze the different configurations and the corresponding levels of security achieved. The next results will analyze security performance at small, medium, and large scales, keeping the patient load constant and weighting factors dominant. For 4 patients, the security levels in this scenario are higher across all cases than in Scenario 1 in Figure 18. ILP scored the highest; HybridQeGA nearly matched ILP; PATI-Greedy showed slight improvements; and Random scored the least.

To understand how trends evolve with increasing system complexity, the forthcoming analysis examines a medium-scale setup with $P = 8$ patients. Figure 19 shows, for 8 patients, that HybridQeGA remains the closest to ILP while PATI-Greedy falls behind. Random does not demonstrate a better or a worse level of secure allocation.

Extending the analysis to a large-scale workload $P = 12$ sheds more light on the algorithms' ability to meet high standards under high resource pressure. Figure 20 shows, for 12 patients, that HybridQeGA is the only method that remains, for most of the time, within 3% of ILP, while PATI-Greedy stands out with a considerable distance below HybridQeGA, and Random is performing badly as always.

The MDT can be used with trust within the healthcare system, since security improvements can be made in security-dominant cases. This means the frameworks can be adjusted to create strong safeguards without negatively affecting the system's efficiency. Apart from the comparative performance, it is also important to focus on the learning behavior of the algorithms and how consistent such behaviors are in relation to security in order to be certain of the convergence reliability under tight security. Figure 21 shows the behavior of the convergence curves. HybridQeGA exhibits a consistent learning behavior and converges steadily to the ILP. PATI-Greedy also moves quickly toward convergence, but it stagnates in attaining suboptimal solutions. Random shows no stability and keeps changing patterns and performance, which is consistent.

The performance of Scenario 2 shows that the focus on security changes performance trade-offs, increasing protection

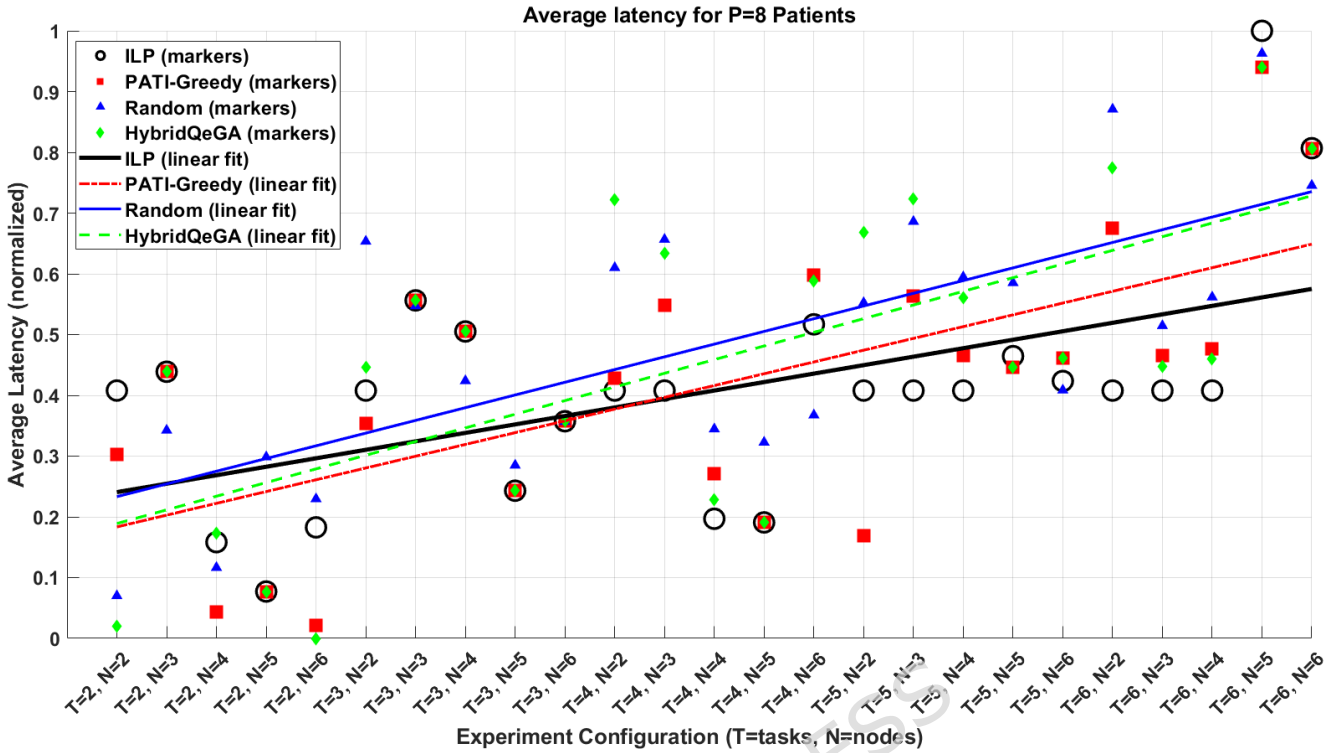

Figure 16. Average latency under Scenario 2 for $P = 8$ patients. The figure compares ILP (black circles), PATI-Greedy (red squares), HybridQeGA (green diamonds), and Random assignment (blue triangles).

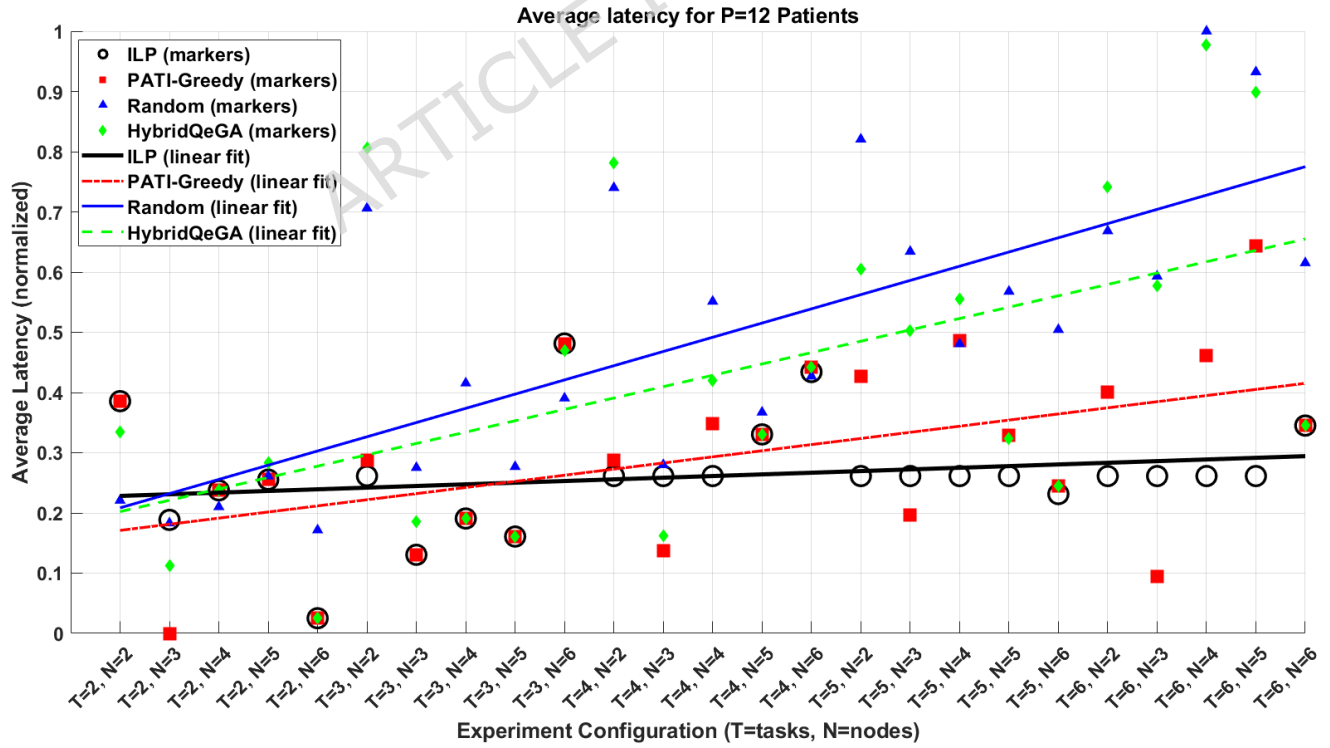

Figure 17. Average latency under Scenario 2 for $P = 12$ patients. The figure compares ILP (black circles), PATI-Greedy (red squares), HybridQeGA (green diamonds), and Random assignment (blue triangles).

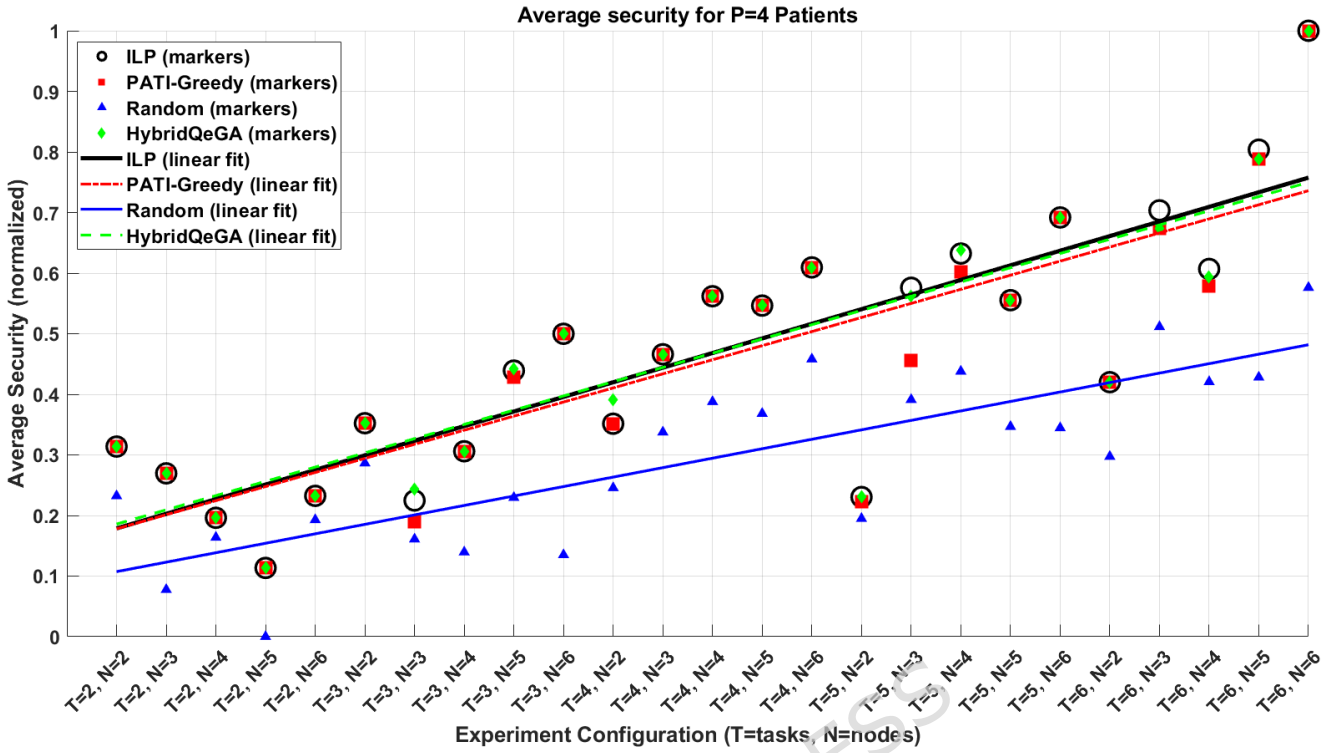

Figure 18. Average security under Scenario 2 for $P = 4$ patients. The figure compares ILP (black circles), PATI-Greedy (red squares), HybridQeGA (green diamonds), and Random assignment (blue triangles).

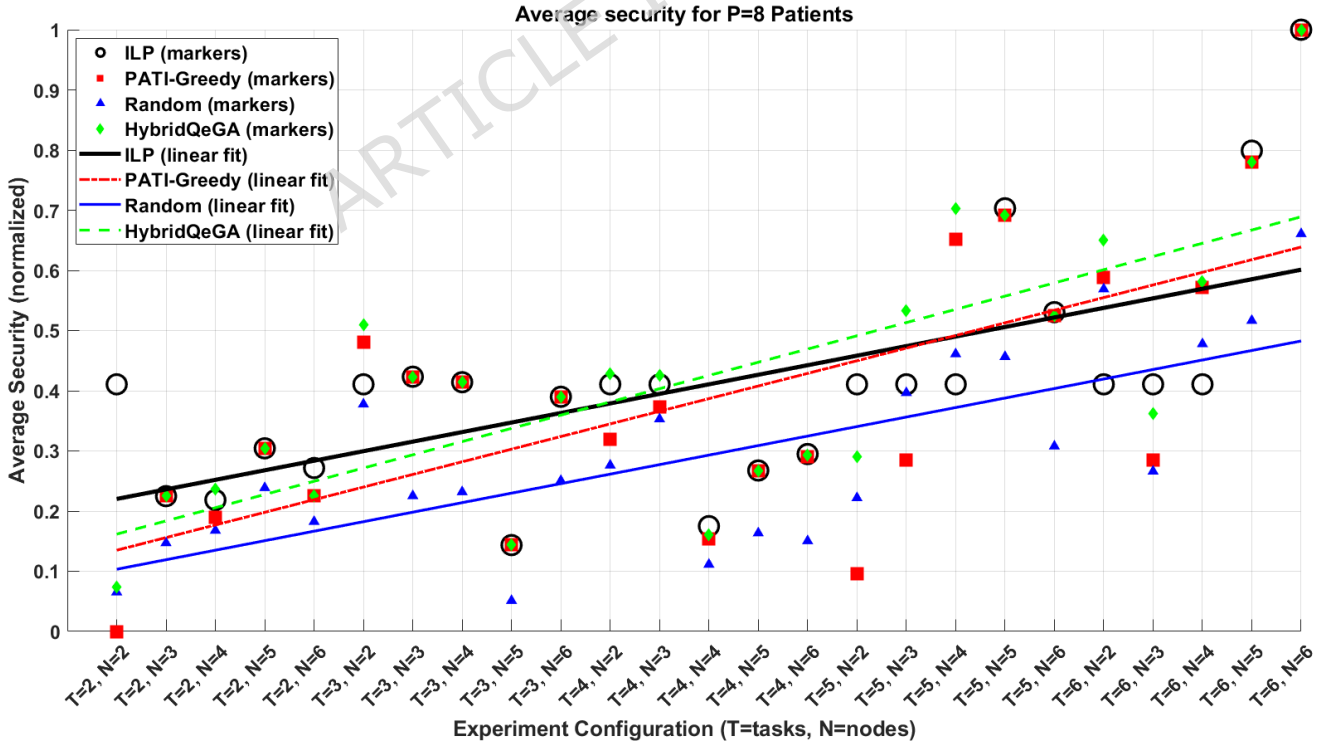

Figure 19. Average security under Scenario 2 for $P = 8$ patients. The figure compares ILP (black circles), PATI-Greedy (red squares), HybridQeGA (green diamonds), and Random assignment (blue triangles).

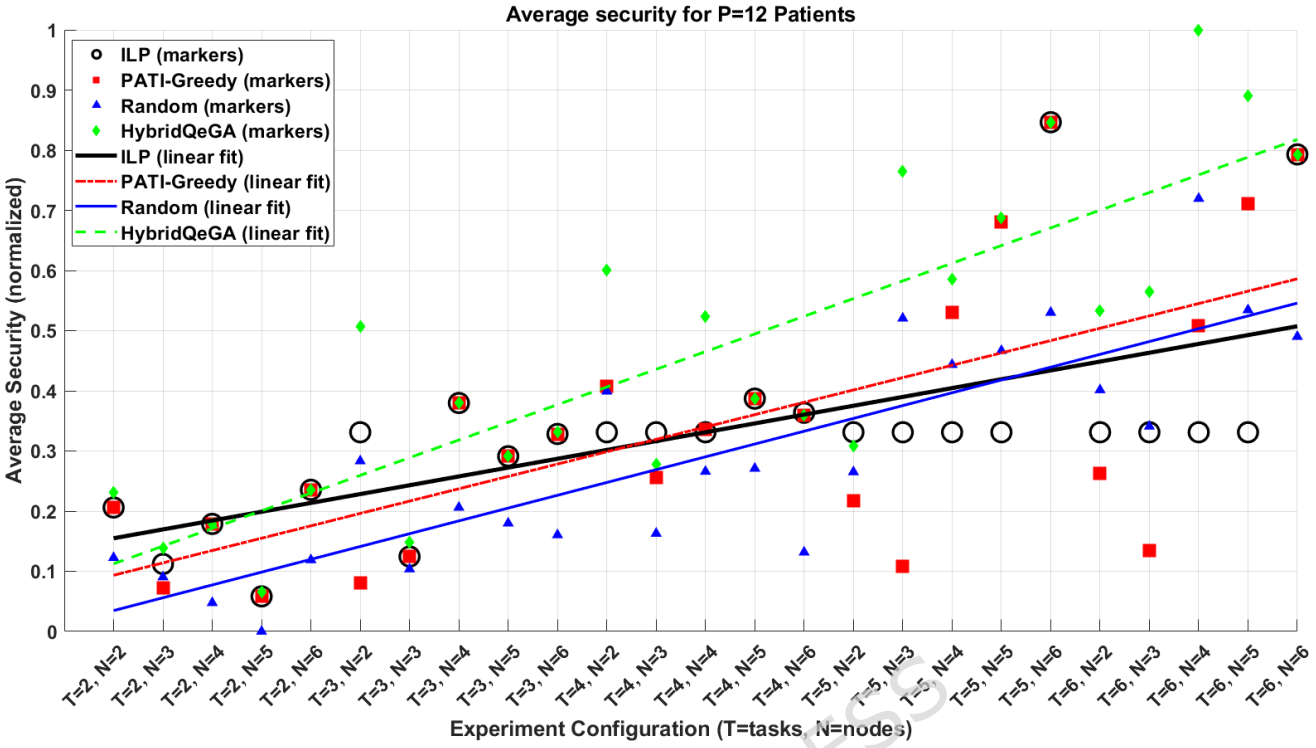

Figure 20. Average security under Scenario 2 for $P = 12$ patients. The figure compares ILP (black circles), PATI-Greedy (red squares), HybridQeGA (green diamonds), and Random assignment (blue triangles).

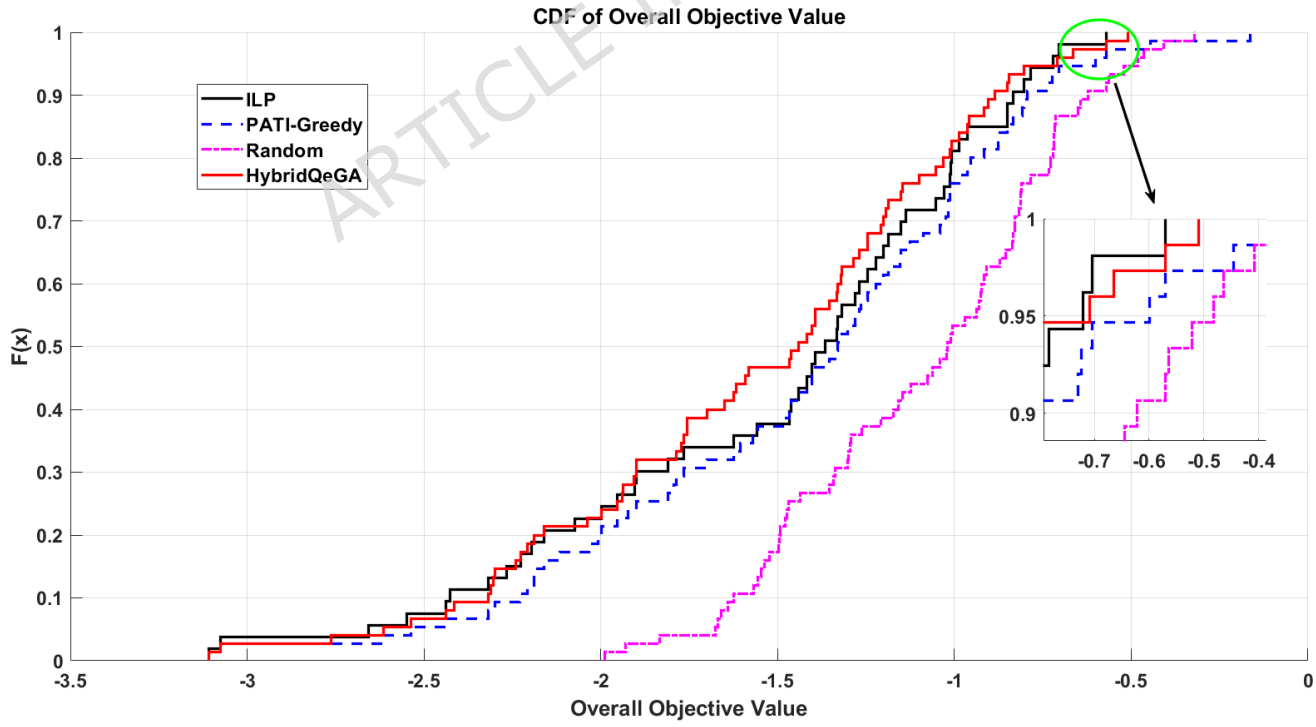

Figure 21. Convergence behavior under Scenario 2 with security-dominant weighting. HybridQeGA demonstrates stable convergence toward the ILP benchmark, whereas PATI-Greedy stagnates at suboptimal solutions and random assignment fails to converge.

while only moderately increasing costs and latency. Among the proposed methods, HybridQeGA offers the best balance, providing optimal security while preserving scalability and convergence stability, thereby narrowing the scope of security-critical situations. The following scenario applies these findings and goes beyond assessing a single metric by considering them in totality. More specifically, Scenario 3 integrates the metrics of cost, latency, and security into a single unified objective function, providing a comprehensive overview of the algorithmic efficiency across all operational conditions.

5.3 Scenario 3: Overall Objective Performance

In Scenario 3, the integrated objective function combines cost, latency, and security into a single unified metric that provides a detailed analysis of the algorithm's performance. The combination of these three criteria provides a comprehensive analysis of the algorithm trade-off across various objectives under varying workloads. The study begins with a small setup and then increases patient loads, thereby challenging the method's strength and reliability.

As shown in Figure 22, among all algorithms, the ILP with 4 patients yields the best integrated objective value. HybridQeGA comes close to ILP, being 4% worse. PATI-Greedy comes next, although there is a significant gap compared to HybridQeGA, and Random is the worst. To analyze the potential of this near-optimal behavior as the workload intensity increases, the next

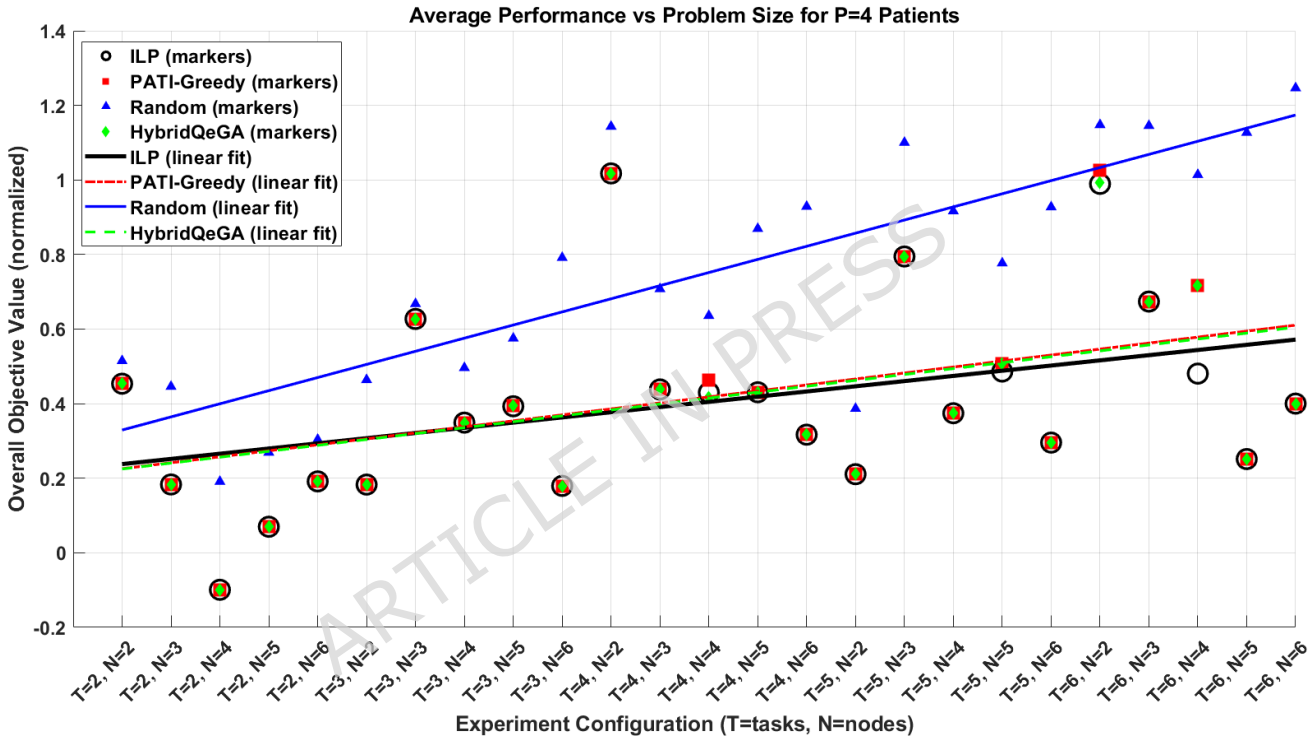

Figure 22. Overall objective values under Scenario 3 for $P = 4$. HybridQeGA closely approximates the ILP benchmark, outperforming PATI-Greedy and random assignment in small-scale MDT settings.

set of analyses focuses on a medium-scale configuration with $P = 8$ patients. In Figure 23, with 8 patients, HybridQeGA is still the best and remains the closest to ILP, especially in heavier workloads compared to PATI-Greedy. As the scale increases, PATI-Greedy diverges more, while Random diverges much more than the others. To further explore scalability under the most challenging conditions, we then examine the large-scale configuration at $P = 12$, capturing the algorithms' performance dynamics in problem-sensitive, high-load MDT environments. As shown in Figure 24, the HybridQeGA method nearly matches the performance of the ILP benchmark across 12 patients, demonstrating its high scalability. PATI-Greedy, on the other hand, shows a growing number of deviations as the problem size increases, whereas the Random strategy is pathetically inefficient and cannot be used in practice.

In order to further validate the comparative validation in respect of the diversity of the baseline, a succession of additional experiments was conducted using two well-established metaheuristic algorithms, Particle Swarm Optimization (PSO) and Ant Colony Optimization (ACO). Figures 25 and 26 show the objective performance of representative configurations with 4 tasks and 6 nodes and 6 tasks and 6 nodes, respectively. HybridQeGA is consistently better than PSO and ACO across all patient scales, and the solutions obtained by this method are much closer to the ILP benchmark. Even though PSO and ACO demonstrate significant improvement over Random assignment, their performance reduces with the increase in problem

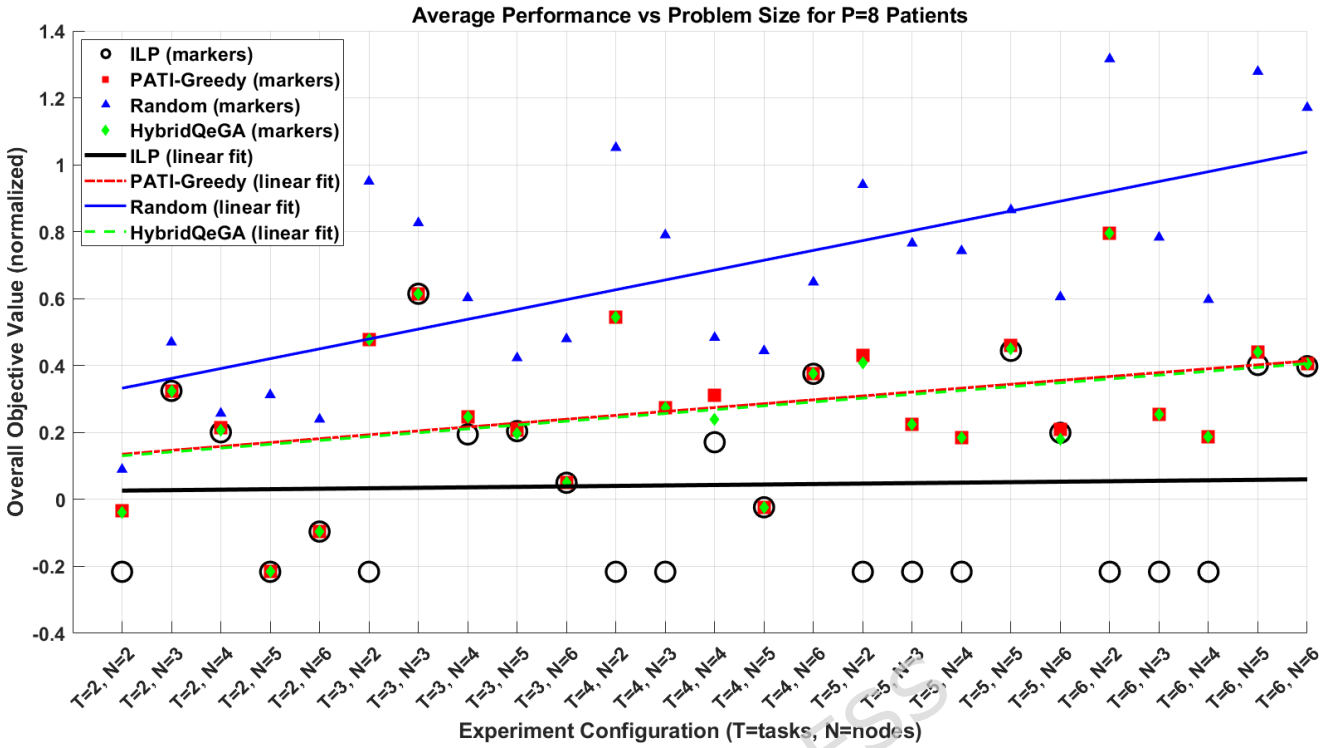

Figure 23. Overall objective performance under Scenario 3 for $P = 8$. HybridQeGA remains near-optimal compared to ILP, while PATI-Greedy and random assignment exhibit increasing performance degradation.

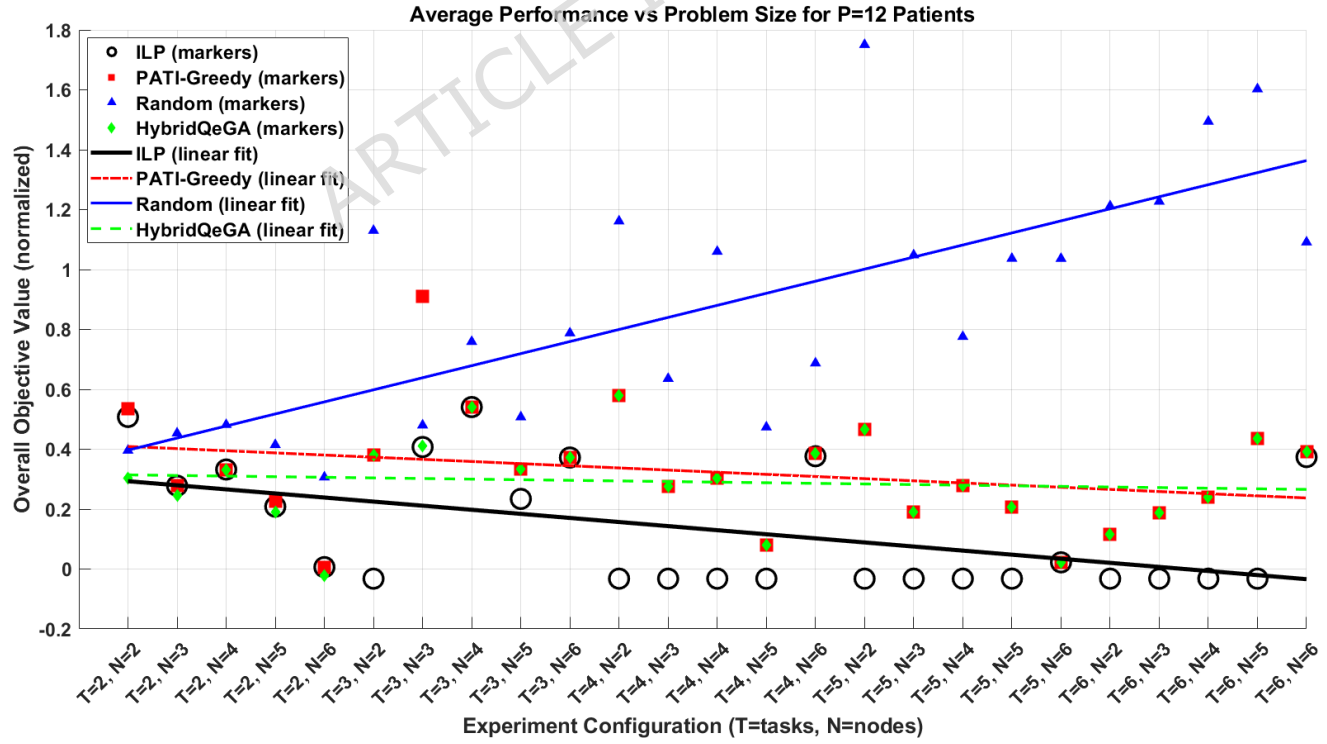

Figure 24. Overall objective performance under Scenario 3 for $P = 12$. HybridQeGA maintains near-optimal performance close to ILP under heavy workloads, whereas PATI-Greedy diverges and random assignment becomes inefficient.

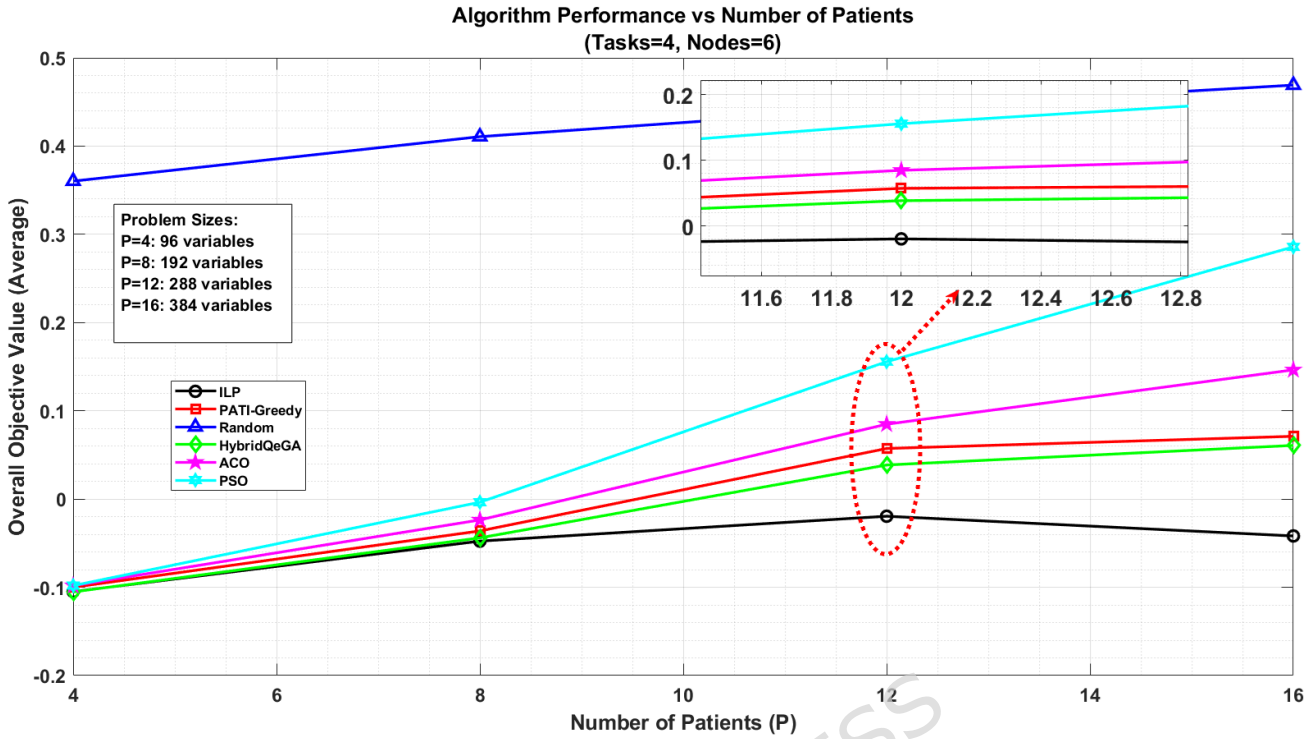

Figure 25. Overall objective performance versus number of patients for $T = 4$ and $N = 6$, comparing ILP, PATI-Greedy, HybridQeGA, Random, Particle Swarm Optimization (PSO), and Ant Colony Optimization (ACO). HybridQeGA consistently achieves near-optimal performance close to ILP, while outperforming PSO and ACO across all patient scales

.

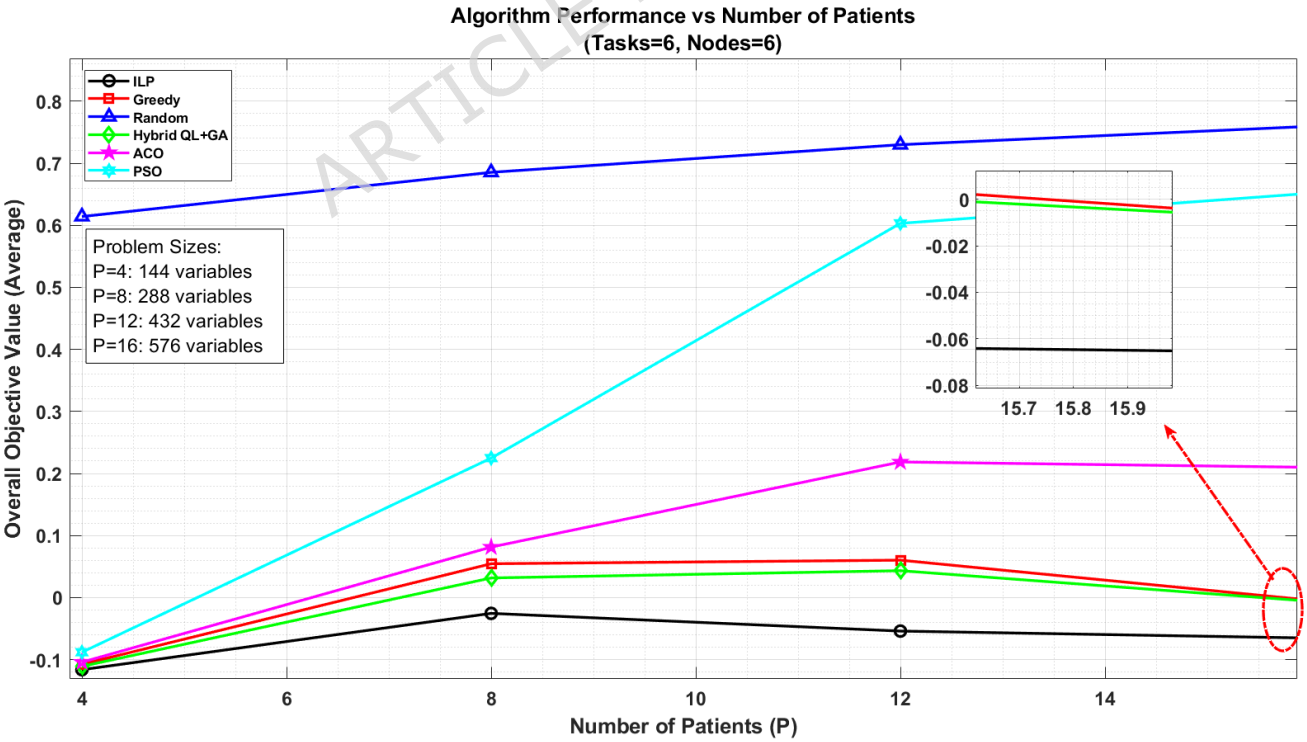

Figure 26. Overall objective performance versus number of patients for $T = 6$ and $N = 6$, comparing ILP, PATI-Greedy, HybridQeGA, Random, PSO, and ACO.

complexity, showing the limitations of handling multi-objective trade-offs and a large combinatorial search space. Conversely, combining Q-learning with HybridQeGA can provide adaptive control for the evolutionary search process, resulting in greater scalability and robustness. These results suggest that the proposed method is not only faster than baselines but also faster than classical bio-inspired optimization methods on real-life MDT workloads. Similar performance patterns are expected in standard genetic algorithms without a reinforcement learning mechanism and in deep Q-network-based schedulers, which should be systematically investigated in future studies. Taken together, these results form a coherent hierarchy of scales: the ILP model can achieve the theoretical optimum, HybridQeGA can achieve near-optimal results even when the load is large, PATI-Greedy can be competitive at smaller scales, and the Random method is ineffective. Lastly, to summarize these observations and enable a comparative analysis of trends across the different scenarios, the following tables present the main performance results.

Scenario	Metric	Best Performer	Near-Optimal Performer	Worst Performer
S1: Balanced	Cost	ILP, HybridQeGA	PATI-Greedy	Random
	Latency	ILP, PATI-Greedy	HybridQeGA	Random
	Security	ILP	HybridQeGA, PATI-Greedy	Random
S2: Security-Dominant	Cost	ILP	HybridQeGA	Random
	Latency	ILP	HybridQeGA, PATI-Greedy	Random
	Security	ILP	HybridQeGA	Random
S3: Overall Objective	Small Scale ($P = 4$)	ILP	PATI-Greedy, HybridQeGA	Random
	Large Scale ($P = 12$)	ILP	HybridQeGA	Random

Table 5. Best and Worst Performing Algorithms Across Scenarios

Algorithm	Mean Cost	Mean Latency	Mean Security	Overall Gap vs. ILP (%)
ILP (Optimal)	100.0	100.0	100.0	0.0
PATI-Greedy	108.6	109.2	91.8	9.4
HybridQeGA	102.3	103.1	97.6	2.7
Random	142.8	148.5	72.4	43.9

Table 6. Quantitative Performance Comparison of Algorithms under Integrated Objective (Scenario 3)

Tables 5 and 6 provide a brief but comprehensive summary of the performance of the algorithms in all experimental conditions, including solution and deployment trade-offs and real-world issues. Table 5 summarizes the performance of each algorithm in each scenario in terms of cost, latency, and security. ILP continues to provide the optimal benchmark, and HybridQeGA is the best alternative across most scenarios, particularly in the larger-scale scenarios, while the Random assignment is the poorest performer across all scenarios. Table 6 contributes to this analysis by including averages and deviations in percentages for the ILP benchmark.

The comparative analyses have reinforced multiple insights. First, ILP remains a reference point across all dimensions; however, its extreme exponential complexity severely limits its relevance to small/offline scenarios. Second, while exhibiting a moderate optimality gap of approximately 8-10% in large-scale, or security-dominant, scenarios, PATI-Greedy, despite being polynomial-time, remains fast and competitive in latency- and cost-sensitive scenarios (Scenario 1), with a slight degradation in optimality. Third, HybridQeGA demonstrates the best overall performance and the most balanced performance while remaining highly scalable. In the high-load and fully integrated objectives scenarios (Scenarios 2 and 3), it achieves near-optimal performance with less than 3% deviation from the ILP, showing the most balanced performance. Conversely, Random assignment consistently results in performance deficits of more than 40% which confirms the need for structured, intelligence-based optimization for MDT task allocation.

All these quantitative comparisons are consistent with the fact that HybridQeGA is the best balance between near-optimality and practical computability, and PATI-Greedy is a lightweight alternative that is applicable to fast decision-making in moderate-workload scenarios. Although the above analysis has focused on the quality of the solution, it is also necessary to understand each method's computational requirements to promote its use in the real world. In this regard, the following subsection examines runtime efficiency and scalability in a more rigorous manner.

Across all scenarios, consistent performance trends are observed, indicating that the proposed framework and HybridQeGA solution are not overly sensitive to specific parameter choices. This demonstrates the robustness of the optimization approach under varying objective weight configurations.

5.4 Computational Complexity and Runtime Considerations

The ILP algorithm proved to be superior; however, its computational time is also exponentially increasing, which makes it impractical to use in large-scale implementation. PATI-Greedy is a time-saving algorithm in comparison with the ILP technique, which is a key requirement of applying it in real-time in the hospital. HybridQeGA is moderately overhead because of Q-learning and genetic operators; nevertheless, its performance is found to be significantly faster than the PATI-Greedy, and the results are approached to optimum performance. As a result, HybridQeGA can serve as a very potential choice to apply in the real-life environment due to its accuracy and the good ratio of the execution speed to the accuracy. Comprehensively, the results of the three scenarios can be summarized into some salient insights that shed some light on the relative strengths and the contextual relevance of each of the algorithms in varying operating conditions.

- ILP sets the optimal benchmark but is computationally infeasible for large-scale use.
- PATI-Greedy is simple, fast, and highly effective in smaller instances.
- HybridQeGA consistently achieves near-optimal performance, particularly in large and complex scenarios.
- Random assignment scales poorly and is unsuitable for clinical MDT environments.

6 Conclusion

This study has developed an AI-driven optimization system of MDTs and patient information management in smart hospital settings. The framework considers cost, latency, and security and is modeled as a multi-objective optimization problem. The proposed framework is solved through an ILP model, the PATI-Greedy heuristic, and a HybridQeGA algorithm. Moreover, a series of simulation experiments was conducted in which objective weightings and patient workloads (P = 4, 8, 12) were varied, thereby simulating MDT deployments at small, medium, and large scales. The results demonstrated that ILP set the optimal benchmark across all metrics, but its exponential runtimes make it infeasible for real-time applications aside from small instances. In contrast, PATI-Greedy solved the problem to within a few percentiles of ILP cost and latency (5-8%), and HybridQeGA offered a more optimal solution for a greater number of patients. Among the three algorithms, HybridQeGA produced the best solutions for larger problem sets. For $P = 12$ patients, it deviated from ILP by $(lessthan3\%)$ and kept practical run times and polynomial scalability. Results across various scenarios show that PATI-Greedy performs best when low-cost, fast solutions are needed. The results appear to confirm that PATI-Greedy performs best in ICUs with a moderate patient volume. On the other hand, HybridQeGA showed better performance under high-load or security-dominant conditions, with an average improvement of 12% in security scores compared to PATI-Greedy at $\alpha = 0.1, \beta = 0.1, \gamma = 0.8$. The random baseline assignments showed poor performance, and in all cases, there was a greater than 40% deviation from ILP. The proposed MDT framework is meant to supplement clinician decision-making in a manner that is ethically sound, legally compliant, and aligned with the human-in-the-loop design principles of accountability in AI healthcare.

Overall, the empirical evidence shows that the proposed framework can provide a scalable and flexible solution for managing MDT resources. Although the ILP provides the theoretical performance limit, a combination of PATI-Greedy and HybridQeGA yields practical solvers that are sensitive to precision, performance, run time, and robustness across a range of clinical workloads. This framework shall be extended to various deployment-oriented axes in further research. To start, the MDT architecture will be interoperable with known healthcare standards, in particular Fast Healthcare Interoperability Resources (FHIR), and will thus be easily integrated with electronic health records and hospital informatics systems. Second, the framework will be extended to include energy sensitivity, federated data-privacy limitations, and cross-hospital MDT collaborations, thereby guiding the creation of advanced, reliable, and real-time digital-health systems. Though the current study is based on simulation, that is, the analysis under heterogeneous clinical workloads and security parameters is controlled and reproducible, future research will prove these results empirically based on semi-realistic or anonymized task traces of an intensive care unit, real-time sensor streams of wearable devices and bedside monitors. These stringent validations will further prove the real-world clinical viability of the proposed framework in an actual clinical setting. In addition, we will extend the optimization layer by incorporating bio-inspired optimization techniques, such as standard genetic algorithms without reinforcement learning and deep Q-networks, to broaden comparative validation and explore complementary optimization trade-offs in more complex, large-scale MDT deployment settings.

References

1. Kannampallil, T. G., Schauer, G. F., Cohen, T. & Patel, V. L. Considering complexity in healthcare systems. *J. biomedical informatics* **44**, 943–947 (2011).
2. Khaleel, M. I., Safran, M., Alfarhood, S. & Zhu, M. Workflow scheduling scheme for optimized reliability and end-to-end delay control in cloud computing using ai-based modeling. *Mathematics* **11**, 1–24 (2023).

3. Shrivastava, V. *et al.* Evolutionary patterns in modern-era cloud-based healthcare technologies. In *International Conference on Information and Communication Technology for Competitive Strategies*, 19–32 (Springer, 2023).
4. Balasubhramanyam, A., Ramesh, R., Sudheer, R. & Honnavalli, P. B. Revolutionizing healthcare: A review unveiling the transformative power of digital twins. *IEEE Access* **12**, 69652 – 69676 (2024).
5. Hartmann, M., Hashmi, U. S. & Imran, A. Edge computing in smart health care systems: Review, challenges, and research directions. *Transactions on Emerg. Telecommun. Technol.* **33**, e3710 (2022).
6. Rahim, M., Lalouani, W., Toubal, E. & Emokpae, L. A digital twin-based platform for medical cyber-physical systems. *IEEE Access* **12**, 174591 – 174607 (2024).
7. Chen, J., Yi, C., Okegbile, S. D., Cai, J. & Shen, X. Networking architecture and key supporting technologies for human digital twin in personalized healthcare: A comprehensive survey. *IEEE Commun. Surv. & Tutorials* **26**, 706–746 (2023).
8. Sirigu, G., Carminati, B. & Ferrari, E. Privacy and security issues for human digital twins. In *2022 IEEE 4th international conference on trust, privacy and security in intelligent systems, and applications (TPS-ISA)*, 1–9 (IEEE, 2022).
9. Khaleel, M. I., Safran, M., Alfarhood, S. & Zhu, M. Energy-latency trade-off analysis for scientific workflow in cloud environments: the role of processor utilization ratio and mean grey wolf optimizer. *Eng. Sci. Technol. an Int. J.* **50**, 101611 (2024).
10. Chen, J. *et al.* Generative ai-driven human digital twin in iot-healthcare: A comprehensive survey. *IEEE Internet Things J.* **11**, 34749 – 34773 (2024).
11. Li, T. *et al.* Generative ai empowered network digital twins: Architecture, technologies, and applications. *ACM Comput. Surv.* **57**, 1–43 (2025).
12. Zhang, K. *et al.* Concepts and applications of digital twins in healthcare and medicine. *Patterns* **5**, 1–15 (2024).
13. Gourraud, P. *et al.* Digital representation of patients as medical digital twins: Data-centric framework. *JMIR Med. Informatics* **13**, e53542 (2025).
14. Walton, N. *et al.* Digital twins for health: A scoping review. *npj Digit. Medicine* **7**, 1–11 (2024).
15. Rodrigues, L. *et al.* Health digital twins supported by ai-based algorithms and extended reality in cardiology. *arXiv preprint arXiv:2401.14208* (2024).
16. Trayanova, N. *et al.* A 'digital twin' of your heart lets doctors test treatments before surgery. *The Wall Str. J.* (2024).
17. Pandey, H. *et al.* Digital twin ecosystem for oncology clinical operations. *arXiv preprint arXiv:2409.17650* (2024).
18. Ibrahim, M. *et al.* Generative ai for synthetic data across multiple medical modalities: A systematic review. *arXiv preprint arXiv:2407.00116* (2024).
19. Vengathattil, S. Advancing healthcare systems with generative ai-driven digital twins. *Int. J. Innov. Sci. Res. Technol.* **10**, 1678–1688 (2025).
20. Almasan, P. *et al.* Network digital twin: Context, enabling technologies, and opportunities. *IEEE Commun. Mag.* **60**, 22–27 (2022).
21. Mahmood, K. *et al.* Adaptive resource aware and privacy preserving federated edge learning framework for real time internet of medical things applications. *Sci. Reports* **15**, 36468 (2025).
22. Stephanie, V., Khalil, I. & Atiquzzaman, M. Dsfl: A decentralized splitfed learning approach for healthcare consumers in the metaverse. *IEEE Transactions on Consumer Electron.* **70**, 2107–2115 (2024).
23. Jiang, L., Ming, X. & Zhang, X. Dt-dofl: Digital-twin-empowered decentralized online federated learning for user-centered smart healthcare service systems. *IEEE Transactions on Comput. Soc. Syst.* **12**, 4441 – 4455 (2025).
24. Jameil, A. K. & Al-Raweshidy, H. Enhancing offloading with cybersecurity in edge computing for digital twin-driven patient monitoring. *IET Wirel. Sens. Syst.* **14**, 363–380 (2024).
25. Shankhdhar, A. & Garg, H. Blockchain-enabled secure data transmission for personalized e-healthcare and digital twin well-being. *Clust. Comput.* **28**, 956 (2025).
26. Li, J. & Wang, D. Federated learning for digital twin applications: a privacy-preserving and low-latency approach. *PeerJ Comput. Sci.* **11**, e2877 (2025).
27. Kuštelega, M., Mekovec, R. & Shareef, A. Privacy and security challenges of the digital twin: systematic literature review. *J. Univers. Comput. Sci. (JUCS)* **30** (2024).

28. Diakakis, S. *et al.* A review of interoperability challenges and solutions towards a digital twin of the european electricity grid. In *2024 16th Electrical Engineering Faculty Conference (BulEF)*, 1–5 (IEEE, 2024).

29. Vallee, A. Digital twin for healthcare systems. *Front. Digit. Heal.* **5**, 1253050 (2023).

30. Drummond, D. & Gonsard, A. Definitions and characteristics of patient digital twins being developed for clinical use: scoping review. *J. Med. Internet Res.* **26**, e58504 (2024).

31. Nadeem, M., Kostic, S., Dornhöfer, M., Weber, C. & Fathi, M. A comprehensive review of digital twin in healthcare in the scope of simulative health-monitoring. *Digit. Heal.* **11**, 1–24 (2025).

32. Tortora, M. *et al.* Medical digital twin: A review on technical principles and clinical applications. *J. Clin. Medicine* **14**, 324 (2025).

33. Hu, Y. *et al.* Personalized heart disease detection via ecg digital twin generation. *arXiv preprint arXiv:2404.11171* (2024).

34. Wang, S., Ren, T., Cheng, N., Wang, R. & Zhang, L. Patient-specific dynamic digital-physical twin for coronary intervention training: An integrated mixed reality approach. *arXiv preprint arXiv:2505.10902* (2025).

35. Kuang, K., Ouyang, D. S. & Alaa, A. M. Med-real2sim: Non-invasive medical digital twins using physics-informed ssl. *arXiv preprint arXiv:2403.00177* (2024).

36. Khaleel, M. I., Safran, M., Alfarhood, S. & Zhu, M. A hybrid many-objective optimization algorithm for job scheduling in cloud computing based on merge-and-split theory. *Mathematics* **11**, 3563 (2023).

37. Kleinert, T. & Schmidt, M. Why there is no need to use a big-m in linear bilevel optimization: A computational study of two ready-to-use approaches. *Comput. Manag. Sci.* **20**, 1–20 (2023).

38. Sharif, Z., Jung, L. T., Ayaz, M., Yahya, M. & Pitafi, S. Priority-based task scheduling and resource allocation in edge computing for health monitoring system. *J. King Saud Univ. Inf. Sci.* **35**, 544–559 (2023).

Data Availability Statement

No external datasets were generated or analyzed during the current study, as the evaluation was conducted using simulation-based experiments. The implementation details, algorithmic configurations, and parameter settings used in this work are described in the manuscript. Source code and additional simulation materials can be made available upon reasonable request to the corresponding author.

Funding Statement

The authors extend their appreciation to Prince Sattam bin Abdulaziz University for funding this research work through the project number (PSAU/2025/31124).

Acknowledgements

The authors extend their appreciation to Prince Sattam bin Abdulaziz University for funding this research work through the project number PSAU/2025/31124.

Author contributions statement

F. M .A and S. A, created the paper's fundamental idea. T. A, and S. A, have developed the methodology. S. A, and M. A have analyzed the paper. Original draft was prepared by S. A, which was edited by A. M. A., F. M. A, and S. A . T. A, have supervised the overall work. All authors reviewed the manuscript.

AI-Assisted Editing Disclosure

The authors used an AI-assisted language editing tool to improve the manuscript's clarity, grammar, and readability; all scientific content, analysis, and conclusions remain the authors' sole responsibility.