Quantifying Cybersecurity–QoS Trade-Offs in Smart Hospitals: A Comparative Study Using CSPNs and Markovian Agent Models

Enrico Barbierato[1,*,†], Alice Gatti[1], Marco Gribaudo[2,†] and Mauro Iacono[3,†]

[1]Dipartimento di Matematica e Fisica, Università Cattolica del Sacro Cuore, via della Garzetta 48, 25133 Brescia, Italy
[2]Dip. di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, via Ponzio 34/5, 20133 Milano, Italy
[3]Dip. di Matematica e Fisica, Università degli Studi della Campania, "L. Vanvitelli", Viale Lincoln 5, 81100 Caserta, Italy

Abstract

In smart hospitals, achieving a balance between cybersecurity and quality of service (QoS) is a critical yet underexplored challenge. Cyberattacks can disrupt medical services, while overly aggressive countermeasures may degrade performance or availability, thus violating Service Level Agreements (SLAs). To quantify this trade-off, we model a representative smart healthcare system using two formal approaches: Colored Stochastic Petri Nets (CSPNs) and Markovian Agent Models (MAMs). The CSPN captures fine-grained, concurrent behaviors and stochastic delays at the token level, while the MAM abstracts global system dynamics via differential equations. Through extensive simulations, we evaluate mitigation latency, resource saturation, and system responsiveness under cyberattack scenarios. Confidence intervals, computed from repeated CSPN runs, provide statistically grounded insight into SLA compliance variability, highlighting that a significant portion of mitigations exceed the defined threshold. Despite the potential for rapid mitigation, stochastic delays and concurrency often result in critical SLA violations. This dual-model approach enables a complementary analysis: CSPNs reveal short-term congestion and resource contention, whereas MAMs uncover long-term systemic trends. The study offers a reproducible framework for evaluating cyber-resilience in safety-critical environments.

Keywords

smart hospital, sla, cyberattack

1. Introduction

The compromise between Service Level Agreements (SLAs), which ensure high performance and availability, and the negative impact that cyberattack countermeasures may have, remains a persistent challenge in contemporary digital infrastructures. In sectors such as healthcare, manufacturing, finance, and transportation, the continuity and quality of services are vital. Yet, the very security mechanisms designed to safeguard these systems can inadvertently degrade performance. For example, in healthcare, isolating a potentially compromised medical device may delay the transmission of critical physiological data, putting patients at risk. In manufacturing, containment of a cyber incident might interrupt production lines, while in finance or transportation, network reconfiguration or mitigation can impair real-time responsiveness. As these systems become increasingly interconnected and edge-driven, their exposure to cyber threats increases in proportion. Cyberattacks targeting distributed nodes can introduce service delays, compromise safety-critical workflows, and erode trust in automated mitigation. Conversely, actions like node quarantine or load redistribution can degrade Quality of Service (QoS), highlighting the trade-off between security and operational continuity. Understanding this trade-off requires formal, quantitative tools capable of capturing complex system behavior under both nominal and adversarial conditions. This paper presents a comparative study based on two

QualITA 2025: The Fourth Conference on System and Service Quality, June 25 and 27, 2025, Catania, Italy
*Corresponding author.
†These authors contributed equally.
enrico.barbierato@unicatt.it (E. Barbierato); alice.gatti@unicatt.it (A. Gatti); marco.gribaudo@polimi.it (M. Gribaudo); mauro.iacono@unicampania.it (M. Iacono)
0000-0002-0877-7063 (E. Barbierato); 0009-0008-8422-8024 (A. Gatti); 00000-0002-1415-5287 (M. Gribaudo); 0000-0002-2089-975X (M. Iacono)

complementary modeling formalisms: Colored Stochastic Petri Nets (CSPNs, [1]) and Markovian Agent Models (MAMs, [2]), applied to a shared smart hospital scenario. CSPNs provide fine-grained, token-level semantics with support for concurrency and resource contention, making them particularly suitable for modeling transient bottlenecks and asynchronous mitigation dynamics. MAMs, in contrast, offer a population-level view grounded in coupled differential equations, enabling tractable evaluation of emergent properties such as cascading failures and systemic resilience. This approach contrasts other techiques, such as multiformalism modeling ([3, 4, 5]. While prior work has analyzed security–QoS interactions using either micro-level or macro-level formalisms, a direct comparison between CSPNs and MAMs under identical assumptions has not been systematically explored. This study addresses that gap and frames the following research questions:

RQ1. How do CSPNs and MAMs differ in quantifying the trade-offs between cybersecurity measures and Quality of Service in smart healthcare systems?

RQ2. Can macroscopic approximations (MAMs) faithfully capture the same degradation and recovery patterns as token-level simulations (CSPNs)?

Our comparative evaluation reveals that while both formalisms detect the impact of cybersecurity interventions on availability and latency, they differ in sensitivity and representational clarity. CSPNs capture transient oscillations and local saturation more precisely, whereas MAMs offer scalable insight into systemic behavior and enable closed-form trend analysis. These results provide operational guidance for model selection in SLA-sensitive domains, depending on the desired granularity, computational budget, and analytic goals. Following this introduction, Section 2 presents the healthcare case study. Section 3 details the CSPN and MAM models developed. Section 4 comments on the comparative results. Related work is reviewed in Section 5, and final reflections are offered in Section 6.

2. Case study

This study focuses on a smart hospital scenario where cybersecurity measures may conflict with QoS requirements. Core QoS metrics include latency in transmitting patient data, system availability, data integrity, and service continuity. The hospital infrastructure is composed of interconnected medical devices (e.g., ventilators, ECGs), edge nodes, and backend servers, all of which are exposed to cyber threats such as malware and adaptive attacks.

The defensive architecture includes intrusion detection systems and human operators capable of quarantining compromised devices. However, such countermeasures may delay data transmission or disrupt care delivery. We examine how attacks on a limited set of devices can propagate system-wide degradation, impact alert-mitigation latency, and challenge SLAs.

To operationalize this, we define two SLA thresholds: (i) a maximum average mitigation latency of 10 steps ($\approx$ 2 minutes), and (ii) a minimum throughput of 0.5 mitigations per step. These benchmarks guide the comparative evaluation of system responsiveness and resilience across the two formal models.

3. Models and experiments

The case study is modelled using a CSPN and an MAM[1]. Latency was measured as the average delay between alert detection and successful mitigation, while throughput refers to the number of mitigations per hour. Each transition is identified by a semantic label, a probability of occurrence p, and its corresponding stochastic rate $\lambda = p \cdot \nu$, where $\nu = 0.1$ transitions/hour is the base velocity. This separation of p and ν allows clearer interpretation and flexibility across CSPN and MAM models. While not derived from first principles, this semi-empirical formulation aligns with common practice in mean-field modeling and is compatible with exponential timing assumptions in stochastic simulations. The base velocity $\nu = 0.1$ was selected to reflect typical response rates observed in healthcare infrastructures,

[1]The Python code used to perform the experiments is available at https://github.com/EBarbierato/qualITA2025

though we acknowledge its heuristic nature. All values for probabilities and transition rates have been derived from empirical evidence reported in recent literature on IoT security, medical device vulnerability, human-in-the-loop mitigation, and edge computing infrastructures for healthcare systems ([6]).

CSPN

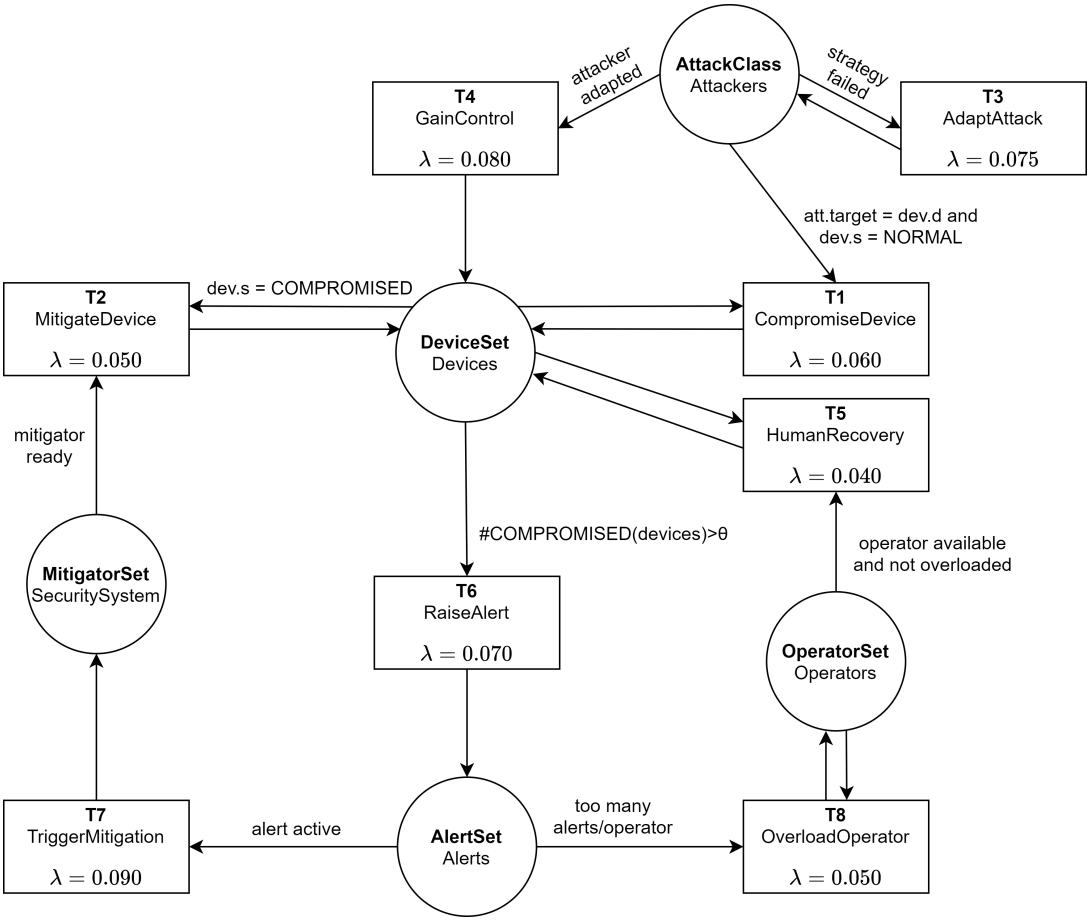


Figure 1: CSPN architecture.

In the CSPN depicted in Figure 1, four color sets represent the heterogeneity in system components: DeviceType = {ECG, VENT, PUMP}, Department = {ICU, EMERGENCY, SURGERY}, Status = {NORMAL, COMPROMISED, DEGRADED, QUARANTINED}, and finally, AttackClass = {STATIC, ADAPTIVE, STEALTH}. Each token in the `Devices` place is a tuple, `(d : DeviceType, dept : Department, s : Status)`. Similarly, each token in the `Attackers` place is: `(a : AttackClass, target : DeviceType)`. The places are the following: i) Devices: All active medical devices with their current type, department, and status; ii) Attackers: Cyber threats characterized by strategy and preferred device targets; iii) SecuritySystem: Holds mitigation agents, configured per department; iv) Operators: Tracks human intervention capability per department, and finally, v) Alerts: Queue of active alerts, identified by department and severity.

MAM

The MAM in Figure 2 includes both degradation and recovery transitions to represent realistic operational behaviours. For instance, while devices may become compromised ($P_0 \to P_2$), they may later degrade ($P_2 \to P_1$) and eventually recover ($P_1 \to P_0$). Similarly, operator overload is represented not only

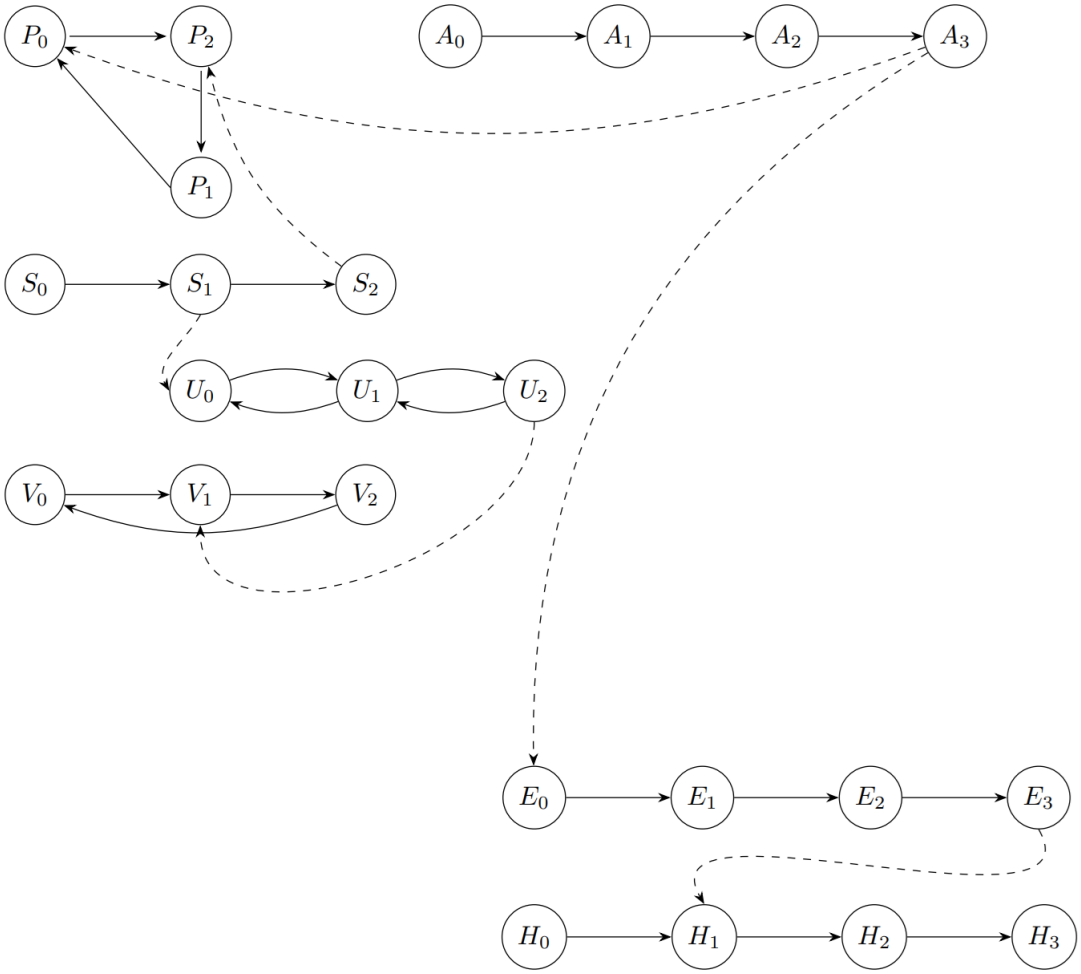


Figure 1: Markovian Agent Model (MAM) with external influences. Dashed arrows represent cross-class effects.

Figure 2: MAM architecture.

as a state (U_2) but also influences the mitigation capacity and responsiveness delay through its feedback on S_2 activity. Recovery of operators ($U_2 \to U_1 \to U_0$) is essential to restoring system responsiveness. The MAM comprises several interacting agent classes. *Patient Devices* (P), which include equipment such as ECGs, ventilators, and infusion pumps, can be in one of three states: normal (P_0), degraded (P_1), or compromised (P_2). *Ventilators* (V), due to their critical function, are modelled separately and transition through operational (V_0), delayed (V_1), and quarantined (V_2) states. *Edge Nodes* (E), acting as local data processors, evolve from healthy (E_0) to overloaded (E_1) and isolated (E_2). The *Security Module* (S) performs anomaly detection and response, progressing through monitoring (S_0), alerted (S_1), and mitigating (S_2). The *Hospital Backend* (H), responsible for long-term records and analytics, may become delayed or fail, moving from responsive (H_0) to delayed (H_1) and ultimately unavailable (H_2). *Operators* (U) represent human personnel who can be idle (U_0), engaged in mitigation (U_1), or overloaded (U_2). Finally, the *Attacker* (A) is a dynamic threat agent advancing through static (A_0), injecting (A_1), adapting (A_2), and fully controlling (A_3) stages.

Transitions between states are governed by rates $\lambda_{i\to j} = p_{i\to j} \cdot \nu$, where $p_{i\to j}$ is the empirical transition probability and $\nu = 0.1$ transitions/hour is a fixed base velocity. The system dynamics are governed by a set of probabilistic transitions across agent classes. Patient devices may be compromised with probability 0.60 when the attacker reaches state A_3 ($P_0 \to P_2$), while mitigation, triggered by the security system

in S_2, reduces severity ($P_2 \to P_1$, 0.50). Devices can autonomously recover to normal operation with probability 0.30 ($P_1 \to P_0$). The attacker progresses through increasingly dangerous phases, starting with activation ($A_0 \to A_1$, 0.50), adaptation ($A_1 \to A_2$, 0.75), and finally gaining control ($A_2 \to A_3$, 0.80). The security system responds by detecting anomalies ($S_0 \to S_1$, 0.70) and confirming attacks ($S_1 \to S_2$, 0.65). Human operators react to alerts generated in S_1 ($U_0 \to U_1$, 0.65), may become overwhelmed ($U_1 \to U_2$, 0.50), and can partially or fully recover ($U_2 \to U_1$, 0.25; $U_1 \to U_0$, 0.20). Ventilators experience delays ($V_0 \to V_1$, 0.40), may be quarantined due to operator stress ($V_1 \to V_2$, 0.50), and later resume normal function ($V_2 \to V_0$, 0.20). Edge nodes deteriorate from healthy to overloaded ($E_0 \to E_1$, 0.40), then isolated ($E_1 \to E_2$, 0.45), and finally enter an alarm state ($E_2 \to E_3$, 0.35). The hospital backend is similarly affected, with latency initiation ($H_0 \to H_1$, 0.50), degradation due to edge alarm ($H_1 \to H_2$, 0.45), and full unavailability ($H_2 \to H_3$, 0.30).

The time evolution of the MAM system is described by the following representative equations, respectively, for the patient devices (equation 1), the attacker (equation 2), the security system (equation 3), the operators (equation 4), the ventilators (equation 5), the edge nodes (equation 6), and the hospital backend (equation 7).

$$\begin{aligned}
\dot{p}_0 &= -\lambda_{P_0 \to P_2}\, a_3\, p_0 + \lambda_{P_1 \to P_0}\, p_1, \\
\dot{p}_2 &= \lambda_{P_0 \to P_2}\, a_3\, p_0 - \lambda_{P_2 \to P_1}\, s_2\, p_2, \\
\dot{p}_1 &= \lambda_{P_2 \to P_1}\, s_2\, p_2 - \lambda_{P_1 \to P_0}\, p_1
\end{aligned} \tag{1}$$

$$\begin{aligned}
\dot{a}_0 &= -\lambda_{A_0 \to A_1}\, a_0, \\
\dot{a}_1 &= \lambda_{A_0 \to A_1}\, a_0 - \lambda_{A_1 \to A_2}\, a_1, \\
\dot{a}_2 &= \lambda_{A_1 \to A_2}\, a_1 - \lambda_{A_2 \to A_3}\, a_2, \\
\dot{a}_3 &= \lambda_{A_2 \to A_3}\, a_2
\end{aligned} \tag{2}$$

$$\begin{aligned}
\dot{s}_0 &= -\lambda_{S_0 \to S_1}\, s_0, \\
\dot{s}_1 &= \lambda_{S_0 \to S_1}\, s_0 - \lambda_{S_1 \to S_2}\, s_1, \\
\dot{s}_2 &= \lambda_{S_1 \to S_2}\, s_1
\end{aligned} \tag{3}$$

$$\begin{aligned}
\dot{u}_0 &= -\lambda_{U_0 \to U_1}\, s_1\, u_0 + \lambda_{U_1 \to U_0}\, u_1, \\
\dot{u}_1 &= \lambda_{U_0 \to U_1}\, s_1\, u_0 - (\lambda_{U_1 \to U_2} + \lambda_{U_1 \to U_0})\, u_1 + \lambda_{U_2 \to U_1}\, u_2, \\
\dot{u}_2 &= \lambda_{U_1 \to U_2}\, u_1 - \lambda_{U_2 \to U_1}\, u_2
\end{aligned} \tag{4}$$

$$\begin{aligned}
\dot{v}_0 &= -\lambda_{V_0 \to V_1}\, v_0 + \lambda_{V_2 \to V_0}\, v_2, \\
\dot{v}_1 &= \lambda_{V_0 \to V_1}\, v_0 - \lambda_{V_1 \to V_2}\, u_2\, v_1, \\
\dot{v}_2 &= \lambda_{V_1 \to V_2}\, u_2\, v_1 - \lambda_{V_2 \to V_0}\, v_2
\end{aligned} \tag{5}$$

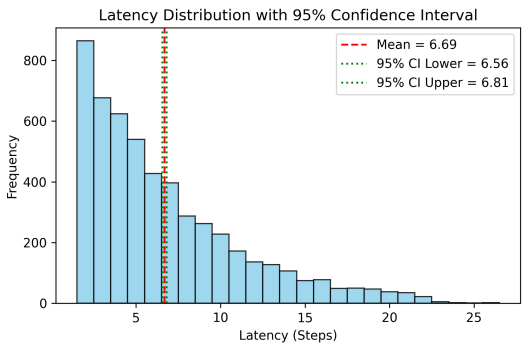
$$\begin{aligned}
\dot{e}_0 &= -\lambda_{E_0 \to E_1}\, e_0, \\
\dot{e}_1 &= \lambda_{E_0 \to E_1}\, e_0 - \lambda_{E_1 \to E_2}\, e_1, \\
\dot{e}_2 &= \lambda_{E_1 \to E_2}\, e_1 - \lambda_{E_2 \to E_3}\, e_2, \\
\dot{e}_3 &= \lambda_{E_2 \to E_3}\, e_2
\end{aligned} \tag{6}$$

$$\begin{aligned}
\dot{h}_0 &= -\lambda_{H_0 \to H_1}\, h_0, \\
\dot{h}_1 &= \lambda_{H_0 \to H_1}\, h_0 - \lambda_{H_1 \to H_2}\, e_3\, h_1, \\
\dot{h}_2 &= \lambda_{H_1 \to H_2}\, e_3\, h_1 - \lambda_{H_2 \to H_3}\, h_2, \\
\dot{h}_3 &= \lambda_{H_2 \to H_3}\, h_2
\end{aligned} \tag{7}$$

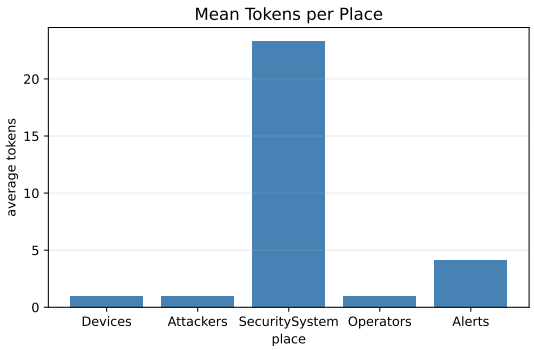
4. Discussion

CSPN

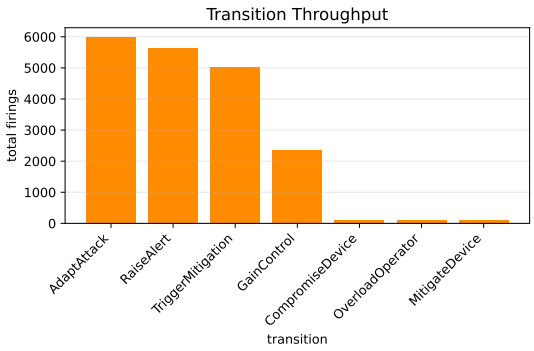
Regarding the CSPN, 100 independent simulation runs were executed. The latency distribution (Figure 3a) reflects the number of simulation steps elapsed between the firing of the `RaiseAlert` transition and the corresponding `TriggerMitigation` event. While most mitigations occur within 2–4 steps, a significant portion exceeds the SLA threshold of 10 steps, risking delays in critical responses. The empirical mean latency is approximately 6.69 steps, with a 95% confidence interval spanning $[6.56, 6.81]$. This statistical range highlights that, although fast mitigation is possible under favorable conditions, the system fails to guarantee SLA compliance in a consistent manner. The fact that the upper bound of the confidence interval remains well below the SLA threshold suggests that the system performs



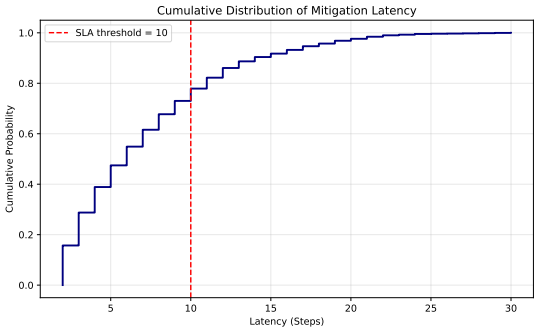
(a) Latency Distribution with Confidence Interval.



(b) Mean Tokens per Place.



(c) Transition Throughput.



(d) Cumulative Distribution of Latency.

Figure 3: Results of the CSPN simulations with confidence-based analysis.

adequately on average; however, the long tail of the distribution indicates that significant variability exists, and a non-negligible portion of alert responses breach acceptable delay boundaries. This discrepancy is further confirmed in the cumulative distribution function of latency (Figure 3d), where only about 65% of the mitigation actions are completed within 10 steps. The exponential decay of the latency distribution is consistent with the stochastic firing logic and fixed delays embedded in the model: once an alert is raised, the probability of fast mitigation is initially high, but probabilistic variability, resource contention, and delayed token availability can induce substantially longer wait times. Figure 3b reports the average number of tokens observed in each place across the simulations. Notably, the `SecuritySystem` place accumulates the highest token count, reflecting the retention of mitigated alerts post-intervention. The `Alerts` place also holds a non-trivial average token count, indicating that alerts often remain unresolved for multiple steps, reinforcing the observation of SLA violations. The low token counts in places such as `Devices`, `Attackers`, and `Operators` suggest rapid token turnover or relatively infrequent production, emphasizing the selectivity and responsiveness of the model's defensive routines. The transition throughput, shown in Figure 3c, reveals that the most active transitions are `AdaptAttack`, `RaiseAlert`, and `TriggerMitigation`. These transitions are crucial for reactive behavior in adversarial settings, suggesting a robust attempt by the system to detect and mitigate threats. However, their frequency alone is insufficient to ensure SLA compliance, as shown by the latency statistics. In contrast, transitions like `CompromiseDevice`, `MitigateDevice`, and `OverloadOperator` exhibit significantly lower activity, hinting at strict guard conditions or rare triggering scenarios. These could be considered focal points for model refinement to better align system behavior with SLA targets. Lastly, Figure 3d provides a compact visualization of SLA compliance across the 100 runs. The CDF confirms that while the majority of alerts are mitigated within 10 steps, approximately 35% of cases fall outside the target window. This tail behavior, although gradually tapering off, reflects the inherent randomness in transition delays and the saturation of mitigation mechanisms under concurrent demands. From a system design perspective, this suggests a need for enhanced resilience

features—either by optimizing token dispatch or by provisioning additional mitigation resources during high-load periods.

MAM

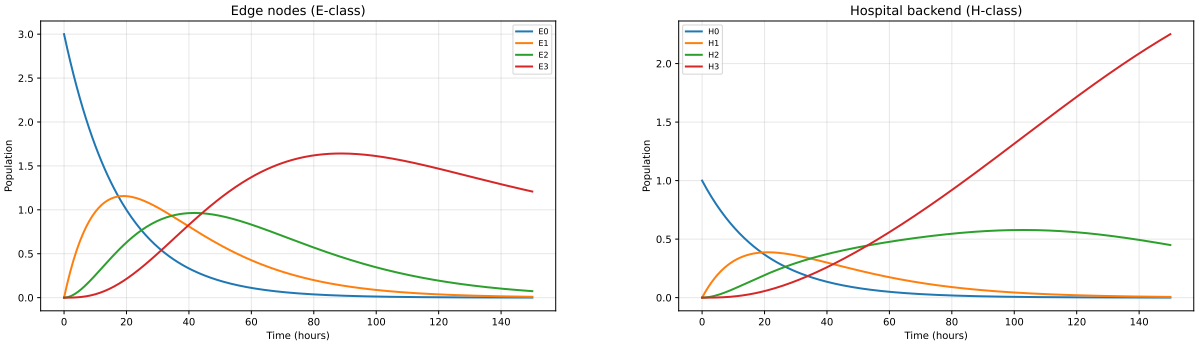
Figure 4 illustrates the temporal dynamics of six distinct agent classes under the MAM. Each subplot shows the population trajectories over time across the different internal states of each class. Panel 4a shows the dynamics of the edge computing nodes (E-class). Initially concentrated in the healthy state E_0, the population progressively shifts toward the degraded E_1 and disconnected E_2 states, with a small but non-negligible fraction ultimately reaching the isolated state E_3. The timing and magnitude of this transition suggest that edge nodes are highly exposed to cumulative stress, possibly from increasing traffic or compromised dependencies, and that the system lacks sufficient mitigation to prevent propagation toward isolation. In Panel 4b, the hospital backend servers (H-class) exhibit a smoother degradation pathway. While there is a steady migration from the operational state H_0 to the quarantined state H_3, the intermediate states H_1 and H_2 act as temporary buffers. This indicates that the backend infrastructure has layered fault-tolerance mechanisms in place but is still susceptible to long-term cascading effects under sustained adversarial pressure. Panel 4c focuses on patient devices (P-class), which experience a rapid transition from the normal state P_0 to the compromised state P_2, followed by a slower recovery via the degraded state P_1. The presence of oscillations between P_2 and P_1 highlights the system's attempt to restore service while still facing ongoing attacks, evidencing a conflict between resilience and persistent vulnerability.

Panel 4d captures the security module (S-class), which exhibits a clear pipeline behavior: tokens flow sequentially from S_0 (inactive) to S_1 (active) and finally to S_2 (fully deployed). The fast depletion of S_0 and concurrent rise of S_2 suggest a prompt but irreversible activation of the mitigation response, reflecting a design that favors rapid response over rollback capability. The operator dynamics in Panel 4e reveal a concerning overload pattern. Agents initially in the available state U_0 rapidly transition to the overloaded state U_2, bypassing the intermediate U_1 state in many cases. This abrupt shift may indicate that operators are quickly overwhelmed by alert surges or manual intervention requests, thus reducing the system's adaptive capacity. Finally, Panel 4f shows the dynamics of the ventilator subsystem (V-class). The curves show a significant and early drop in the operational state (V_0) and a corresponding rise in the delayed (V_1) and quarantined (V_2) states. The stagnation in recovery suggests that ventilators are both high-priority and high-risk nodes: once compromised or delayed, they do not return quickly to operational status, emphasizing their criticality and limited redundancy.

Figure 5 illustrates the temporal evolution of the attacker population, subdivided into four distinct states: A_0, A_1, A_2, and A_3. The simulation begins with all attackers concentrated in the initial state A_1, which denotes a baseline active configuration. Over time, a significant portion of the attacker population transitions toward more advanced stages, notably A_2 and A_3, reflecting adaptive and stealthy behaviors. The transition from A_1 to A_2 occurs rapidly due to the high transition rate assigned to the corresponding process, and is followed by a delayed but steady accumulation in A_3, representing attackers that have successfully gained control or evaded detection. The state A_0, potentially associated with inactive or defeated attackers, remains unpopulated throughout the simulation, suggesting that the current model does not include recovery or neutralization mechanisms for attackers. This lack of reversion results in a monotonic escalation of the attacker's capability, posing increased challenges for mitigation systems.

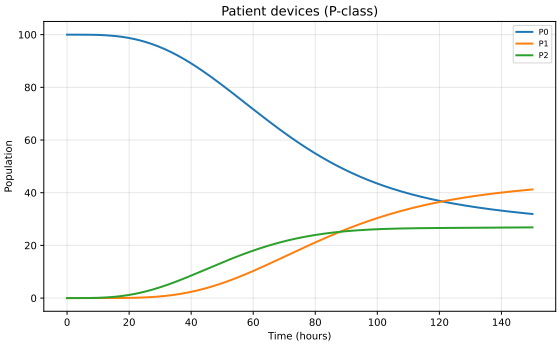
5. Related work

The interplay between cybersecurity enforcement and QoS in healthcare cyber-physical systems has been addressed from multiple modeling and architectural perspectives in recent literature. Shmeleva *et al.* [7] apply Infinite Petri Nets to the analysis of cybersecurity in intelligent systems, showcasing how scalable formal models can capture the complexity of modern infrastructures under adversarial conditions. Their work, while not specific to healthcare, is directly relevant to our use of CSPNs for

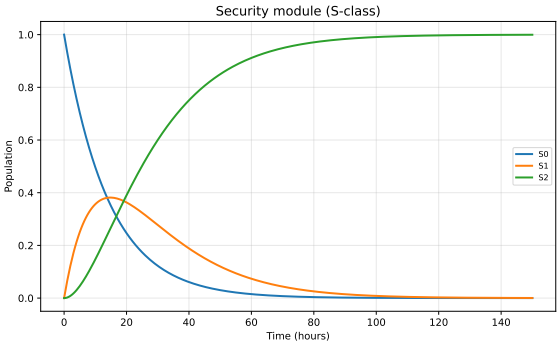


(a) Edge nodes.

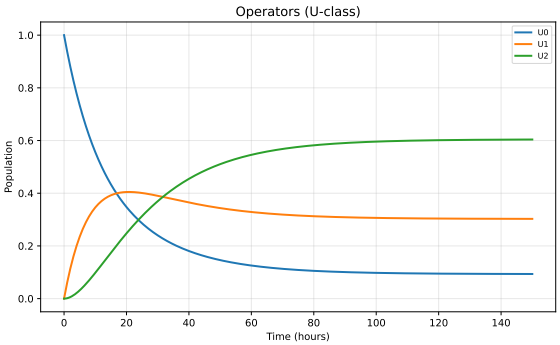
(b) Hospital backend.



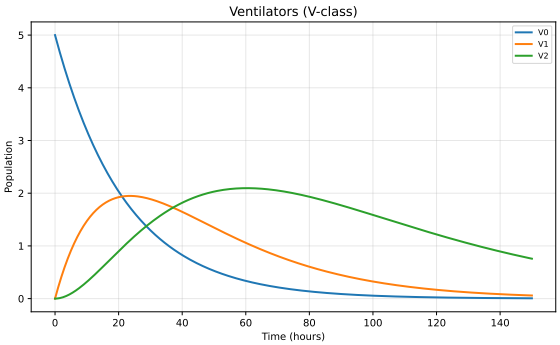
(c) Patient devices.



(d) Security module.



(e) Operators.



(f) Ventilators.

Figure 4: Temporal Evolution of Agent Classes in the Markovian Agent Model (MAM).

representing concurrent device interactions and transition delays in smart hospitals. Wu *et al.* [8] propose a personalized cyber-physical system for smart healthcare based on Petri net modeling, explicitly addressing both data privacy and system responsiveness. Their framework reinforces the suitability of Petri nets for modeling real-time mitigation strategies and structural dependencies across distributed medical devices, validating the core assumptions behind the CSPN portion of our study. A broader review of trust, security, and privacy in high-speed smart city infrastructures is proposed by Iftikhar *et al.* [9], who highlight the challenges faced by edge-enabled architectures, including smart hospitals, when enforcing mitigation policies that may inadvertently degrade QoS. Their findings confirm the lack of formal tools for quantifying SLA violations under adversarial pressure, motivating our comparative modeling strategy. Buyya *et al.* [10] outline a comprehensive vision for QoS-driven edge computing in smart hospital systems, identifying mitigation-induced latencies and node isolation as critical design concerns. Although their work does not involve formal modeling, it provides valuable context for the architectural scenarios we simulate, particularly in terms of mitigation throughput and alert propagation

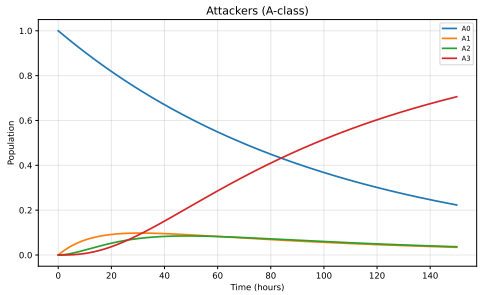


Figure 5: Temporal Dynamics of the Attacker (A-class).

delays. Carramiñana et al. [11] employ agent-based simulations to evaluate healthcare infrastructure resilience under cyber and pandemic stress. Unlike our hybrid CSPN–MAM approach, their model focuses on strategic policy evaluation rather than formal stochastic analysis. Nonetheless, both studies share the goal of assessing dynamic QoS degradation and recovery. Bobbio *et al.* [12] provide a systemic risk analysis of Health IoT and contact tracing infrastructures under cyber warfare scenarios. Their modeling strategy integrates CSPNs and MAMs to analyze the cascading effects of cyberattacks in complex healthcare ecosystems, with a focus on data breaches and functional degradations. The current article positions itself at the intersection of formal modeling and system-level resilience analysis in smart healthcare infrastructures. Unlike prior works that rely solely on Petri nets [7, 8] or agent-based simulations [11], this study combines CSPNs with MAMs to capture both short-term, discrete-event dynamics and long-term, population-level trends. While architectural overviews and systematic reviews have highlighted the tension between cybersecurity and QoS [9, 10], they lack executable models to quantify SLA violations or simulate mitigation strategies under adversarial conditions. Bobbio *et al.* emphasize system-wide resilience metrics and systemic failure propagation, with a primary concern for data privacy violations and contact-tracing vulnerabilities. In contrast, our work focuses on Quality of Service (QoS) guarantees under targeted attacks, with a specific emphasis on mitigating latency, SLA violations, and ensuring operational continuity in smart hospitals. Moreover, our CSPN implementation incorporates dynamic alerting and human operator overload, while Bobbio *et al.* adopt a more static modeling of networked IoT nodes. Finally, we extend the analysis through confidence intervals over multiple simulation runs, offering statistical robustness that complements their more deterministic performance profiles.

6. Conclusions

This study investigated the interplay between cybersecurity enforcement and QoS preservation in smart hospital environments through a comparative modeling approach. By applying both a CSPN and a MAM to a shared healthcare scenario, we addressed two key research questions.

In response to **RQ1**, our results confirm that CSPNs and MAMs offer complementary insights into the cybersecurity–QoS trade-off. The CSPN enables fine-grained tracking of transient behaviors such as operator overload, mitigation latency, and SLA violations at the transition level. The incorporation of confidence intervals in CSPN simulation results has further enhanced the model's diagnostic power by exposing the range and likelihood of SLA breaches under adversarial variability. These confidence intervals clarify system stability and confirm that a notable share of mitigations still exceed SLA limits. The MAM, in contrast, abstracts the system into agent-level dynamics and reveals macroscopic degradation trends, making it more suitable for studying long-term resilience and equilibrium under persistent threats.

Addressing **RQ2**, we find that MAMs can approximate the overall evolution of degradation and recovery patterns reasonably well, especially over extended time horizons. However, they fail to reproduce sharp discontinuities and local saturation effects evident in the CSPN. These include bottlenecks in mitigation paths and operator overloads that critically impact SLA compliance. While MAMs offer

analytical tractability and scalability, their assumptions of homogeneity and continuous dynamics smooth out localized violations, limiting their utility in high-resolution SLA analysis.

Overall, the dual-modeling strategy underscores the value of combining discrete-event formalisms with differential equation-based abstractions to achieve a balanced view of cyber-physical system performance. CSPNs offer detailed temporal resolution and now—thanks to confidence intervals—quantitative reliability bounds, making them indispensable for short-term diagnostics and SLA validation. MAMs, in turn, scale effectively to larger systems and longer horizons, offering efficient evaluation of global resilience patterns. Each model has intrinsic limitations: CSPNs face scalability challenges, while MAMs omit heterogeneity and tail risks.

Declaration on Generative AI

The author(s) have not employed any Generative AI tools.

References

[1] K. Jensen, Coloured Petri Nets: Basic concepts, analysis methods and practical use, Monographs in Theoretical Computer Science 1 (1994). doi:10.1007/978-3-642-58069-7.

[2] A. Bobbio, D. Bruneo, D. Cerotti, M. Gribaudo, A new quantitative analytical framework for large-scale distributed interacting systems, in: Proceedings of the 2nd International Conference on Performance Evaluation Methodologies and Tools (ValueTools), ACM, 2008. doi:10.4108/ICST.VALUETOOLS2008.4341.

[3] E. Barbierato, A. Bobbio, M. Gribaudo, M. Iacono, Multiformalism to support software rejuvenation modeling, in: 2012 IEEE 23rd International Symposium on Software Reliability Engineering Workshops, IEEE, 2012, pp. 271–276.

[4] E. Gianniti, A. M. Rizzi, E. Barbierato, M. Gribaudo, D. Ardagna, Fluid petri nets for the performance evaluation of mapreduce and spark applications, ACM SIGMETRICS Performance Evaluation Review 44 (2017) 23–36.

[5] D. Ardagna, E. Barbierato, E. Gianniti, M. Gribaudo, T. B. Pinto, A. P. C. da Silva, J. M. Almeida, Predicting the performance of big data applications on the cloud: D. ardagna et al., The Journal of Supercomputing 77 (2021) 1321–1353.

[6] F. Meneghello, M. Calore, D. Zucchetto, M. Polese, A. Zanella, Iot: Internet of threats? a survey of practical security vulnerabilities in real iot devices, IEEE Internet of Things Journal 6 (2019) 8182–8201. doi:10.1109/JIOT.2019.2935189.

[7] T. Shmeleva, D. Zaitsev, I. Zaitsev, Applying Infinite Petri Nets to the Cybersecurity of Intelligent Systems, Applied Sciences 11 (2021) 11870. doi:10.3390/app112411870.

[8] Z. Wu, L. Tian, Y. Zhang, Y. Wang, Y. Du, A personalized eccentric cyber-physical system architecture for smart healthcare using Petri net, Security and Communication Networks 2023 (2023) 4005877. doi:10.1155/2023/4005877.

[9] A. Iftikhar, K. N. Qureshi, M. Shiraz, S. Albahli, Security, trust and privacy risks, responses, and solutions for high-speed smart cities networks: A systematic literature review, Journal of King Saud University-Computer and Information Sciences 35 (2023) 101788.

[10] R. Buyya, S. Srirama, R. Mahmud, M. Goudarzi, L. Ismail, V. Kostakos, Quality of Service (QoS)-driven Edge Computing and Smart Hospitals: A Vision, Architectural Elements, and Future Directions, arXiv preprint (2023). Discusses QoS challenges and architectures in smart-hospitals.

[11] D. Carramiñana, A. M. Bernardos, J. A. Besada, J. R. Casar, Enhancing healthcare infrastructure resilience through agent-based simulation methods, Computer Communications (2025). doi:10.1016/j.comcom.2025.108070, open Access, Scopus-indexed.

[12] A. Bobbio, L. Campanile, M. Gribaudo, M. Iacono, F. Marulli, M. Mastroianni, A cyber warfare perspective on risks related to health IoT devices and contact tracing, Neural Computing and Applications 35 (2023) 13823 – 13837. doi:10.1007/s00521-021-06720-1.