Available online at www.sciencedirect.com

ScienceDirect

ICT Express 11 (2025) 34–40

www.elsevier.com/locate/icte

DDS-P: Stochastic models based performance of IoT disaster detection systems across multiple geographic areas☆

Israel Araújo[a], Luis Guilherme Silva[a], Carlos Brito[a], Dugki Min[b,*], Jae-Woo Lee[c], Tuan Anh Nguyen[b,d,*], Erico Leão[a], Francisco A. Silva[a]

[a] *Laboratory of Applied Research to Distributed Systems (PASID), Federal University of Piaui (UFPI), Picos, Piaui 64607-670, Brazil*
[b] *Department of Artificial Intelligence, Graduate School, Konkuk University, Seoul 05029, South Korea*
[c] *Department of Mechanical and Aerospace Engineering, Konkuk University, Seoul 05029, South Korea*
[d] *Konkuk Aerospace Design-Airworthiness Research Institute (KADA), Konkuk University, Seoul 05029, South Korea*

Received 26 April 2024; received in revised form 13 June 2024; accepted 8 September 2024
Available online 12 September 2024

Abstract

Effective management of catastrophic events in high-risk zones necessitates a holistic technological approach to protect ecosystems, biodiversity, and native populations. Limitations in sensor range and connectivity hamper real-time data gathering in secluded areas, while financial and technical hurdles hinder the creation of cost-effective, automated systems. This study presents stochastic models, the LoRaW protocol, and cloud technology to enhance sensor deployment simulations. Wireless Sensor Networks and LoRa technology are crucial for extensive monitoring and communication infrastructures. Stochastic Petri Net models optimize system components by assessing crucial performance indicators, such as average response time and system utilization, thus improving disaster response and supporting research hypotheses.

Keywords: Disaster monitoring; Lora; Performance; Stochastic petri nets; Sensitivity analysis

1. Introduction

The urgent need to detect and manage disasters in hazard-prone areas necessitates integrating advanced technologies like Long Range Wide Area Network (LoRaWAN) within IoT frameworks. LoRaWAN is a low-power, wide-area network protocol that connects battery-operated devices to the internet. It uses Chirp Spread Spectrum (CSS) technology for long-range communication with low power consumption. LoRaWAN networks, consisting of end devices, gateways, and servers, are arranged in a star topology. End devices communicate with gateways via single-hop wireless links, and the gateways relay the data to the network server, enabling significant power savings and extended battery life. Operating in unlicensed ISM bands, LoRaWAN is cost-effective and globally deployable, with robust security features like AES-128 encryption [1]. This protocol is ideal for monitoring vast forests, addressing the increasing frequency of forest fires, projected to rise by 50% by 2100 [2]. Stochastic Petri Nets (SPNs) play a crucial role in modeling and analyzing the dynamic behaviors of these networks to ensure operational correctness [3]. By employing SPNs in conjunction with LoRaWAN, Wireless Sensor Networks (WSNs) can enable early and precise disaster detection through long-distance, low-power data transmission and simulation of probabilistic system behaviors.

This research promotes the use of stochastic modeling to simulate sensor operations within a LoRaWAN-enhanced

☆ This research was partially supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, South Korea (No. 2020R1A6A1A03046811).; This research was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (2021R1A2C2094943), South Korea. This work was carried out with financial support from the National Council for Scientific and Technological Development (CNPq) under process PDPG-POSDOC-AUXPE No. 88881.830176/2023-01, and the Brazilian National Council for Scientific and Technological Development – CNPq, Universal Project 420365/2023-0, Brazil;.

* Corresponding author at: Department of Artificial Intelligence, Graduate School, Konkuk University, Seoul 05029, South Korea.

E-mail addresses: israel.araujo@ufpi.edu.br (I. Araújo), luis.e@ufpi.edu.br (L.G. Silva), carlosvictor@ufpi.edu.br (C. Brito), dkmin@konkuk.ac.kr (D. Min), jwlee@konkuk.ac.kr (J.-W. Lee), anhnt2407@konkuk.ac.kr (T.A. Nguyen), ericoleao@ufpi.edu.br (E. Leão), faps@ufpi.edu.br (F.A. Silva).

Peer review under responsibility of The Korean Institute of Communications and Information Sciences (KICS).

https://doi.org/10.1016/j.icte.2024.09.005

environment, supported by fog computing for improved system integration. The objective is to develop cost-effective and sustainable systems for disaster prevention, aimed at enhancing response capabilities to protect environments, communities, and natural resources. The study assesses the effectiveness of integrated sensors and LoRaWAN for disaster detection, evaluates system performance including sensor processing and data transmission, and proposes the application of SPN for system simulation and optimization. It hypothesizes that the use of LoRaWAN and sensors for remote disaster detection is feasible and efficient, focusing on server capacity planning.

The key **contributions** of this study include: *(i.) Implementation of Stochastic Models for IoT Sensor Deployment in LoRaWAN Protocols*, which simulate the deployment and operation of IoT sensors; *(ii.) Application of SPNs for Detailed Performance Optimization of Disaster Detection Systems*, enhancing monitoring accuracy and response effectiveness; *(iii.) Comprehensive Performance Metric Analysis for IoT-Based Disaster Detection Systems*, identifying operational efficiencies and effectiveness; *(iv.) Strategic Integration of Cloud Computing with LoRa Technology for Enhanced Data Processing*, enabling robust data management and real-time processing; *(v.) Economic and Scalable System Design for Wide-Area IoT-Based Disaster Detection*, addressing financial and geographic challenges in extensive disaster-prone regions.

The **findings** reveal that SPN models enhance operational efficiency and extend sensor communication range while maintaining low power consumption, crucial for large, remote areas. The simulation results confirm the system's capability for swift disaster response. The **implications** include scalability and cost-effectiveness of the model, potential influence on policy-making, and technological advancement in IoT applications for disaster management.

The paper is structured as follows: Section 2 reviews related works. Section 3 details the system architecture. Section 4 presents models with/without absorbing states and numerical analyses. Section 5 concludes the paper and suggests future research directions.

2. Related work

The previous studies using methodologies akin to ours, aimed at disaster prevention and detection in high-risk areas are categorized into two groups based on their evaluation techniques, emphasizing the influence of performance measurement choices on assessment accuracy and cost-effectiveness. *Disaster Prevention:* Research such as [4] employs SPNs for disaster prevention, involving static and dynamic analysis based on expert opinions. Later studies, like [5], assess risks in subway systems using cloud-based Petri net models combined with fault tree analysis. Another significant contribution is [6], which optimizes disaster response strategies by suggesting targeted resource allocation and evaluating emergency response effectiveness [7]. *Disaster Detection:* Recent advancements in disaster detection include deploying unmanned aerial vehicles (UAVs) [8], integrating IoT solutions with LoRaWAN [9], and developing preemptive systems tested in real scenarios [1]. Additionally, [2] focuses on enhancing data transmission efficiency and improving responder tracking. *Performance optimization:* The use of Petri net-based modeling for performance metrics' optimization has been highlighted in other works. A SPN model was developed by Bhadra et al. [3] to optimize CI/CD pipelines, using sensitivity analysis to enhance timely delivery. Ni et al. [10] introduced a Priced Timed Petri Net (PTPN) model for fog computing resource allocation, predicting task completion time and cost, and employing a dynamic strategy to reduce response time. *Analogy of this work:* Our research integrates an SPN model for disaster monitoring, contrasting with traditional approaches by using predictive modeling to understand system behaviors across different scenarios. We evaluate the model configurations based on metrics such as MRT and Detection Probability (DP), enrich our model with an absorbing state to examine metrics like Mean Time To Absorb (MTTA), and conduct a sensitivity analysis using Design of Experiments (DoE).

3. System architecture

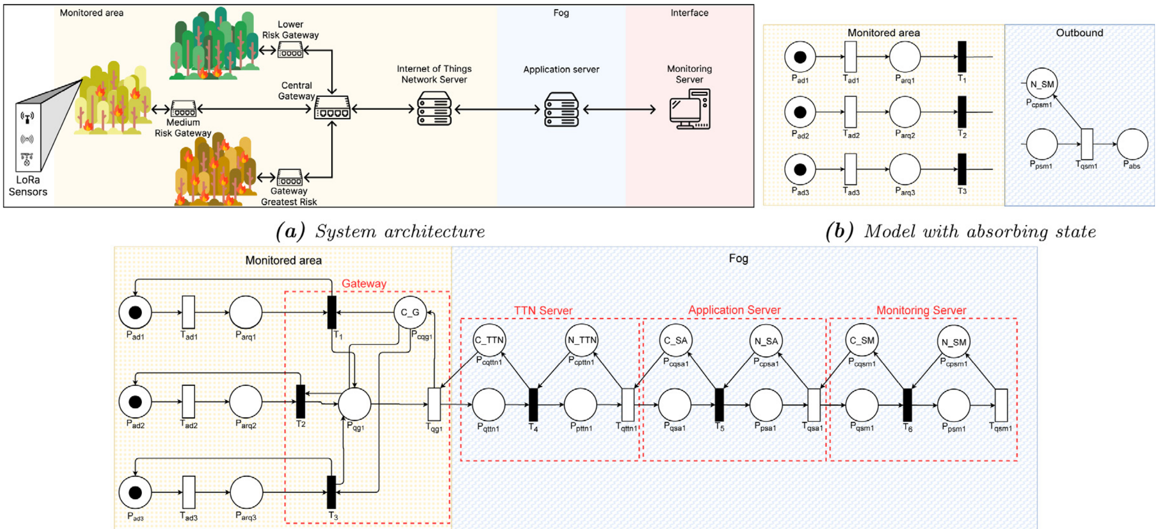
`Overall Description:` Fig. 1(a) depicts the network architecture designed for disaster monitoring in forested areas, connecting sensors to The Things Network (TTN), an Application Server, and a Monitoring Server to ensure robust surveillance [11]. Covering regions with varying disaster risk levels — high, medium, and low — the system uses LoRaWAN technology via central access points that facilitate data communication over the Internet. Class A LoRaWAN protocol is employed for optimal energy use, enhancing sustainability by reducing battery replacements. Sensor data are managed on a network storage server, with TTN's infrastructure ensuring efficient data handling and integrity. The Application Server processes and converts this data for analysis and visualization, providing customized monitoring solutions.

`Challenges:` Key challenges include maintaining coverage over extensive areas, optimizing sensor placement against natural elements, and ensuring sensor durability and accuracy amid environmental challenges. These issues are tackled through strategic sensor placement, protective measures, technology redundancy, and routine maintenance. *LoRa Nodes:* LoRa nodes are essential for collecting environmental data, adhering to the 30-30-30 rule for disaster evaluation, which facilitates timely disaster alerts. Their effectiveness is influenced by regional variables like rainfall, demonstrating the adaptability of the system. *Sensors:* The architecture integrates various sensors such as DHT22 and MQ135 to monitor temperature, humidity, and CO2 levels, alongside a modified DC motor for measuring wind speed, calibrated against standard anemometers. This diverse sensor setup enables comprehensive environmental monitoring, vital for accurate disaster risk assessment.

4. Stochastic models

4.1. Model without absorbing state

`Model Description:` Fig. 1(c) outlines the SPN model, consisting of five key layers: Data Arrival, Gateway, The



(a) System architecture

(b) Model with absorbing state

(c) SPN system model

Fig. 1. System architecture and stochastic models.

Table 1
Values for transitions and markings of the SPN model.

Time Transitions	Tad1, Tad12, Tad3	50.0 (ms)
	Tqg1	5.0 (ms)
	Tqttn1	25.0 (ms)
	Tqsa1	20.0 (ms)
	Tqsm1	1000.0 (ms)
Places	Pad1, Pad2, Pad3	1.0
	Parq1, Parq2, Parq3	0
	Pqg1	0
	Pcqg1	20
	Pcqttn1, Pcqsa1	50.0 (ms)
	Pcqsm1	200.0 (ms)
	Pcpttn1	50
	Pcpsa1	42
	Pcpsm1	2
	Pqttn1, Pqsa1, Pqsm1	0

Things Network (TTN) Server, Application Server, and Monitoring Server. The Data Arrival layer receives system requests from sensor anomalies and forwards them to the Gateway, which routes these requests to their destinations. The TTN Server processes and stores the sensor data, subsequently sending it to the Application Server for integration and visualization. The Monitoring Server provides a user interface for ongoing system surveillance. The SPN model incorporates operational elements like Pad1, Pad2, and Pad3 for initiating sensor requests, and queues such as Parq1, Parq2, Parq3, and Pqg1 for managing data flow. Server capacities are controlled through queues Pcqttn1, Pcqsa1, and Pcqsm1, while processing abilities are defined by Pcpttn1, Pcpsa1, and Pcpsm1. Timed transitions like Tad1, Tad2, and Tad3 manage the intervals between sensor requests. Transitions Tqg1, Tqttn1, Tqsa1, and Tqsm1 facilitate data progression through the system. Marking places indicate the maximum capacities (C_G, C_TTN, C_SA, C_SM) and the available processing resources (N_TTN, N_SA, N_SM) at each server, essential for accommodating variable loads and ensuring efficient data management.

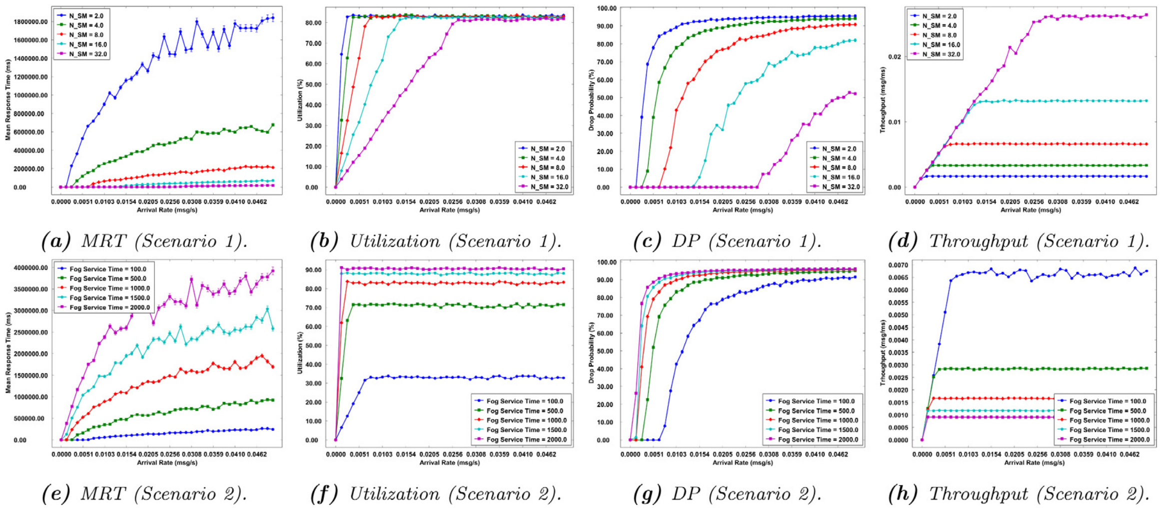
In the SPN model depicted in Fig. 1(c), specific values are assigned to transitions and markings to simulate realistic system operations. Table 1 details these values, which are crucial for understanding the model's performance under various scenarios. These values are derived from literature's empirical data and preliminary simulations, ensuring the model's accuracy and relevance.

`Model Metrics:` This subsection discusses the SPN model's metrics to evaluate system performance, focusing on MRT, fog processing utilization, data communication integrity, and overall throughput. MRT, detailed in Eq. (1a), measures the average time to process requests, indicative of the system's response efficiency. Utilization, defined in Eq. (1b), calculates the resource usage efficiency in the fog layer by assessing the processing queue's average token load relative to total capacity. Discard Probability (DP), explained in Eq. (1c), assesses the likelihood of data loss due to overloads, crucial for maintaining data integrity during critical operations. Throughput (TP), shown in Eq. (1d), evaluates the system's capacity to handle and transfer data effectively. These metrics are instrumental for understanding and enhancing operational efficiency in disaster management scenarios, providing a foundation for future advancements in disaster response technologies. The SPN model was validated through stationary simulations with a 2% error margin using Mercury Tool version 5.0.1.[1]

$$\begin{aligned}\textbf{MRT} = \frac{1}{1-DP} \times \Big(& Esp(Pqd1) + Esp(Pd1) \\ & + Esp(Pqd2) + Esp(Pd2) \\ & + Esp(Pqtc) + Esp(Pqgat) \\ & + Esp(Pqe1) + Esp(Pe1) + Esp(Pqe2) \\ & + Esp(Pe2) + Esp(Pqb) \Big) \times AD \end{aligned} \tag{1a}$$

$$\textbf{UN} = \frac{Esp(Pn1)}{CoreE} \times 100 \tag{1b}$$

[1] https://www.modcs.org/



(a) MRT (Scenario 1). (b) Utilization (Scenario 1). (c) DP (Scenario 1). (d) Throughput (Scenario 1).

(e) MRT (Scenario 2). (f) Utilization (Scenario 2). (g) DP (Scenario 2). (h) Throughput (Scenario 2).

Fig. 2. Analysis results varying processing cores and service time across two scenarios.

$$\textbf{DP} = P(Pcqtc = 0) \times 100 \tag{1c}$$

$$\textbf{TP} = \frac{Esp(Pqc1)}{ServiceTime} \tag{1d}$$

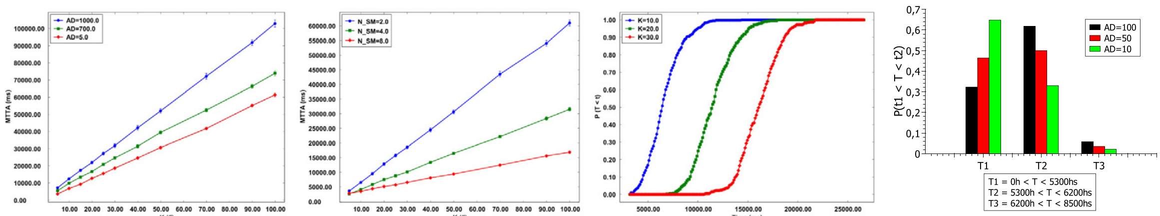
`Case-studies:` This section evaluates two scenarios within a disaster management model, focusing on the number of processing cores and service times in the fog layer. The objective is to determine the optimal configurations that improve system efficiency. The primary goal is to optimize the number of processing cores at fog nodes to enhance disaster monitoring and response. By analyzing these setups, we aim to identify configurations that maximize system performance, guiding architectural improvements. System performance was assessed using metrics such as MRT, Drop Probability (DP), Throughput (TP), and Utilization of the fog computing layer. These metrics provide insights into the system's response time, processing capacity, and resource usage.

Scenario 1 — Variation in the Number of Fog Cores: In the first scenario, the number of fog layer cores (2, 4, 8, 16, 32) was increased, and the impact on system metrics was analyzed. Results, as shown in Figs. 2(a)–2(d), indicate that performance is generally enhanced by more cores, especially for larger data volumes and real-time responses. It was revealed that 8 cores suffice for high demands, with minimal gains beyond this point. In the second scenario, 8 cores were kept constant while service time (100 ms to 2000 ms) was varied to assess system efficiency. As data volume increased, the fog layer's capacity was reached at high arrival rates, demonstrating its load management ability. Request discards were reduced by higher core counts, improving capacity, but throughput was not always enhanced, indicating an optimal performance threshold. The selection of 8 cores in the second scenario was informed by a comprehensive analysis of performance metrics including MRT, Utilization, DP, and Throughput. Our findings indicated that while increasing the number of cores to 32 slightly enhanced performance, the improvement beyond 8 cores was marginal. Specifically, 8 cores provided an optimal balance between performance and resource utilization, effectively meeting high demands characterized by high data arrival rates, low discard probabilities, and optimal response times. This choice ensures the system's efficiency and cost-effectiveness by avoiding unnecessary hardware and energy costs.

Scenario 2 — Variation in Fog Service Time: The impact of varying service times (100 ms to 2000 ms) on fog devices was analyzed. Figs. 2(e)–2(h) revealed that longer service times elevated MRTs, notably at 2000 ms, where increased arrival rates correlated with higher response times. Utilization trends (Fig. 2(f)) showed a rise in utilization in the fog layer with arrival rates, reaching 90% in scenarios with extended service times. Shorter service times facilitated efficient processing, alleviating stress on the fog layer. Elevated service times heightened discard probability, notably at lower arrival rates during prolonged service durations, impacting system reliability. Throughput analysis (Fig. 2(h)) indicated optimal throughput at 100 ms service time, decreasing with extended service times and higher arrival rates. These findings are vital for optimizing fog computing configurations in disaster management systems.

4.2. Model with absorbing state

`Model Structure:` Fig. 1(b) displays the model modifications necessary for integrating an efficient absorbing state. Changes were primarily made to the Admission and Outbound Processing layers to enhance the system's capability to efficiently absorb elements. These adjustments help evaluate the time it takes for elements to be absorbed and the probability of absorbing a specific number of elements within a given timeframe, key metrics for system evaluation. In the Admission layer, adjustments to places Pad1, Pad2, and Pad3 control the entry of requests, with a variable KEL introduced to cap entry from these locations. For instance, if KEL is set to 10, no more than 10 elements can enter from each place, managing the flow into the system and preventing overloads. The Outbound Processing layer includes place Pabs, an absorbing place indicating whether transactions or requests have been



(a) MTTA vs. arrival rate *(b) Absorption probability* *(c) Prob. based on k elements* *(d) Prob. based on intervals*

Fig. 3. Analysis results and absorption probability.

Table 2
Design of experiments.

Factor name	Low setting	High setting
C_G	20.0	40.0
C_TTN	50.0	100.0
C_SA	50.0	100.0
C_SM	600.0	1200.0
N_SM	2.0	4.0

Table 3
Combination of factors considering the MRT(ms) metric.

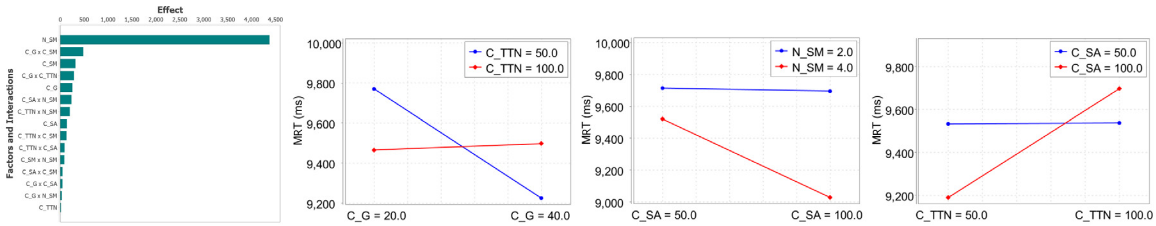
C_G	C_TTN	C_SA	C_SM	N_SM	MRT(ms)
20.00	50.00	50.00	600.00	2.00	11 789.14
20.00	50.00	50.00	600.00	4.00	8247.83
20.00	50.00	50.00	1200.00	2.00	12 503.13
20.00	50.00	50.00	1200.00	4.00	6409.96
20.00	50.00	100.00	600.00	2.00	12 054.15
20.00	50.00	100.00	600.00	4.00	9286.06
20.00	50.00	100.00	1200.00	2.00	11 401.64
20.00	50.00	100.00	1200.00	4.00	6475.20
20.00	100.00	50.00	600.00	2.00	12 560.43
20.00	100.00	50.00	600.00	4.00	7951.83
20.00	100.00	50.00	1200.00	2.00	10 872.14
20.00	100.00	50.00	1200.00	4.00	6984.48
20.00	100.00	100.00	600.00	2.00	11 816.65
20.00	100.00	100.00	600.00	4.00	6451.89
20.00	100.00	100.00	1200.00	2.00	11 308.01
20.00	100.00	100.00	1200.00	4.00	7781.86
40.00	50.00	50.00	600.00	2.00	10 984.09
40.00	50.00	50.00	600.00	4.00	7369.21
40.00	50.00	50.00	1200.00	2.00	11 467.05
40.00	50.00	50.00	1200.00	4.00	8150.59
40.00	50.00	100.00	600.00	2.00	11 622.22
40.00	50.00	100.00	600.00	4.00	5383.04
40.00	50.00	100.00	1200.00	2.00	12 468.02
40.00	50.00	100.00	1200.00	4.00	6362.54
40.00	100.00	50.00	600.00	2.00	11 385.80
40.00	100.00	50.00	600.00	4.00	7681.47
40.00	100.00	50.00	1200.00	2.00	11 477.62
40.00	100.00	50.00	1200.00	4.00	7144.52
40.00	100.00	100.00	600.00	2.00	12 135.27
40.00	100.00	100.00	600.00	4.00	7690.10
40.00	100.00	100.00	1200.00	2.00	10 991.31
40.00	100.00	100.00	1200.00	4.00	7487.34

fully processed. This modification allows precise measurement of the absorbed requests, offering insights into the system's processing efficiency and capacity. The correlation of incoming elements (KEL) with absorption at Pabs provides valuable data on the system's capacity to manage and process incoming demands effectively.

`Analyses:` The analysis in Fig. 3(c) evaluates system performance under varying request volumes (K = 10, 20, 30) using the Cumulative Distribution Function (CDF) for completion probability within specific timeframes and the Mean Total Absorption Time (MTTA) for efficiency. Lower K values yield higher efficiency, with completion times up to 10 000 units, while higher K values extend completion times beyond 20 000 units, reducing completion probability. Scenarios (T1, T2, T3) examine absorption rates over intervals of 0–5300, 5300–6200, and 6200–8500 h, considering arrival rates (10, 50, 100 units). T1 shows higher absorption at lower arrival rates, decreasing with increased rates. T2 and T3 indicate that longer periods might enhance absorption despite higher rates. Fig. 3(a) shows that higher arrival rates increase MTTA due to longer intervals between inputs. Conversely, Fig. 3(b) illustrates that more processing cores in the fog layer significantly reduce MTTA, highlighting core count's importance in optimizing fog layer efficiency.

4.3. Design of experiments

This study employs SPNs to assess MRT and its impact on system performance, focusing on optimization opportunities. Using the Design of Experiments (DoE) methodology, we systematically analyze variable effects. By merging SPN modeling with DoE, we perform a sensitivity analysis to identify critical factors influencing system outputs. Our experiments manipulate variables and measure the MRT variations, utilizing factor interaction and effects graphs to highlight significant variables. Sensitivity analysis prioritizes factors with substantial MRT impact. Table 2 lists these factors and their settings, the values result from a factorial design, varying factors at defined levels to analyze their impact on MRT, ensuring robust, practical optimization insights. While, Table 3 presents their experimental combinations, aiding in understanding how these factors alter system responsiveness. `Analysis of Factors Affecting MRT:` Effects graphs (Fig. 4(a)) rank factors by their MRT influence. Key findings indicate the Monitoring Server's capacity as a primary determinant of performance, suggesting that enhancements here can significantly improve efficiency. Other notable factors include the number of Monitoring Server cores and the interaction between the gateway and Monitoring Server, which are crucial for optimizing response time. The TTN Server capacity has a



(a) (b) (c) (d)

Fig. 4. Impact and interaction of model factors on MRT; (a) Impact of model factors, (b) Gateway vs. TTN Server, (c) App Server vs. Monitoring Cores, (d) TTN vs. App Server.

lesser impact, directing optimization efforts towards more critical factors to enhance performance. `Sensitivity Analysis of Impacting Factors in System Performance`: Further analysis (Fig. 4(b)) demonstrates that increasing both Application Server capacity and Monitoring Server cores significantly reduces MRT. Fig. 4(c) explores interactions between TTN Server and Application Server capacities, uncovering complex patterns that necessitate careful adjustments for optimal performance. Fig. 4(d) shows the effect of TTN and App Server capacities on MRT, indicating constant MRT for $C_{SA} = 50.0$ and increasing MRT for $C_{SA} = 100.0$, suggesting inefficiencies at higher capacities and emphasizing balanced capacity planning in IoT disaster detection systems.

4.4. Discussions

(i.) Machine learning and real-world testing measures: This study uses SPNs to optimize IoT-based disaster detection by analyzing MRT and DP. SPNs were chosen for their precise modeling of asynchronous activities over machine learning. Future work will integrate machine learning for better predictive capabilities and conduct real-world testing to validate the model's effectiveness in dynamic environments. *(ii.) Cost-effectiveness analysis measures*: While the study demonstrates cost-effectiveness through the use of low-power LoRaWAN, optimized server capacity, and fog computing, a detailed cost evaluation will be included in future work. This will encompass comprehensive cost–benefit analyses, economic impact studies, and financial simulations to ensure robust economic assessments.

5. Conclusion

The study validates SPN models for optimizing disaster prevention systems, emphasizing key metrics like MRT and system utilization. Future research will integrate real-time data, develop adaptive algorithms, and explore machine learning for better disaster prediction and mitigation. Additional metrics, IoT-based real-world testing, simulation validation, and comparative analysis in disaster-prone areas will enhance robustness and effectiveness, highlighting potential and limitations.

CRediT authorship contribution statement

Israel Araújo: Writing – original draft, Visualization, Validation, Software, Formal analysis, Data curation. **Luis Guilherme Silva:** Writing – original draft, Visualization, Validation, Software, Formal analysis, Data curation. **Carlos Brito:** Writing – review & editing, Visualization, Validation, Software, Resources, Formal analysis. **Dugki Min:** Writing – review & editing, Validation, Supervision, Investigation, Funding acquisition. **Jae-Woo Lee:** Writing – review & editing, Validation, Supervision, Methodology, Investigation, Funding acquisition. **Tuan Anh Nguyen:** Writing – review & editing, Validation, Supervision, Resources, Project administration, Methodology, Investigation, Conceptualization. **Erico Leão:** Writing – review & editing, Validation, Project administration, Investigation, Formal analysis. **Francisco A. Silva:** Writing – original draft, Validation, Supervision, Project administration, Methodology, Investigation, Conceptualization.

Declaration of competing interest

The authors declare that there is no conflict of interest in this paper.

References

[1] M. Antunes, L.M. Ferreira, C. Viegas, A.P. Coimbra, A.T. de Almeida, Low-cost system for early detection and deployment of countermeasures against wild fires, in: 2019 IEEE 5th World Forum on Internet of Things (WF-IoT), 2019, pp. 418–423, http://dx.doi.org/10.1109/WF-IoT.2019.8767331.

[2] S. Debdas, A. Chakraborty, H.K. Verma, A. Kushwaha, D.V.P. Varma, A. Karmakar, Long range (LoRa) fire fighter in dense forest, in: 2022 International Conference on Computer, Power and Communications, ICCPC, 2022, pp. 286–293, http://dx.doi.org/10.1109/ICCPC55978.2022.10072080.

[3] S. Bhadra, A stochastic Petri net model of continuous integration and continuous delivery, in: 2022 IEEE International Symposium on Software Reliability Engineering Workshops, ISSREW, 2022, pp. 114–117, http://dx.doi.org/10.1109/ISSREW55968.2022.00050.

[4] Y. Tao, D. Xia, Y. Zhu, Research on emergency decision-making model for major disasters based on stochastic Petri nets, in: 2019 9th International Conference on Fire Science and Fire Protection Engineering, ICFSFPE, 2019, pp. 1–6, http://dx.doi.org/10.1109/ICFSFPE48751.2019.9055778.

[5] C. Zhang, G. Tian, A.M. Fathollahi-Fard, W. Wang, P. Wu, Z. Li, Interval-valued intuitionistic uncertain linguistic cloud Petri net and its application to risk assessment for subway fire accident, IEEE Trans. Autom. Sci. Eng. 19 (1) (2022) 163–177, http://dx.doi.org/10.1109/TASE.2020.3014907.

[6] J. Zhou, G. Reniers, Petri-net based simulation analysis for emergency response to multiple simultaneous large-scale fires, J. Loss Prev. Process Ind. 40 (2016) 554–562, http://dx.doi.org/10.1016/j.jlp.2016.01.026, URL: https://www.sciencedirect.com/science/article/pii/S0950423016300249.

[7] H. Ning, J. Longzhe, W. Zongzhi, X. Bo, Performance analysis for emergency response capability based on stochastic Petri net, in: 2010 IEEE International Conference on Emergency Management and Management Sciences, 2010, pp. 61–64, http://dx.doi.org/10.1109/ICEMMS.2010.5563503.

[8] G. Hristov, J. Raychev, D. Kinaneva, P. Zahariev, Emerging methods for early detection of forest fires using unmanned aerial vehicles and lorawan sensor networks, in: 2018 28th EAEEIE Annual Conference, EAEEIE, 2018, pp. 1–9, http://dx.doi.org/10.1109/EAEEIE.2018.8534245.

[9] P. Luna, S. Gutiérrez, R. Espinosa, Design and implementation of a node geolocation system for fire monitoring through LoRaWAN, in: 2020 IEEE International Autumn Meeting on Power, Electronics and Computing, ROPEC, 4, 2020, pp. 1–6, http://dx.doi.org/10.1109/ROPEC50909.2020.9258741.

[10] L. Ni, J. Zhang, C. Jiang, C. Yan, K. Yu, Resource allocation strategy in fog computing based on priced timed Petri nets, IEEE Internet Things J. 4 (5) (2017) 1216–1228, http://dx.doi.org/10.1109/JIOT.2017.2709814.

[11] I. Afanasyev, M. Mazzara, S. Chakraborty, N. Zhuchkov, A. Maksatbek, A. Yesildirek, M. Kassab, S. Distefano, Towards the internet of robotic things: Analysis, architecture, components and challenges, in: 2019 12th International Conference on Developments in ESystems Engineering (DeSE), IEEE, 2019, pp. 3–8.