

Universidade Federal do Piauí
Centro de Ciências da Natureza
Programa de Pós-Graduação em Ciência da Computação

Avaliação de Desempenho em Redes LoRaWAN: Uma Abordagem com Redes de Petri Estocásticas

Israel Cardoso Araújo

Picos-PI, Março de 2025

Israel Cardoso Araújo

Avaliação de Desempenho em Redes LoRaWAN: Uma Abordagem com Redes de Petri Estocásticas

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Ciência da Computação da UFPI (área de concentração: Sistemas de Computação), como parte dos requisitos necessários para a obtenção do Título de Mestre em Ciência da Computação.

Universidade Federal do Piauí – UFPI

Centro de Ciências da Natureza

Programa de Pós-Graduação em Ciência da Computação

Orientador: Prof. Dr. Erico Meneses Leão

Coorientador: Prof. Dr. Francisco Airton Pereira da Silva

Picos-PI

Março de 2025

FICHA CATALOGRÁFICA
Universidade Federal do Piauí
Biblioteca Setorial do Centro de Ciências da Natureza
Serviço de Processos Técnicos

A663a Araújo, Israel Cardoso.
Avaliação de desempenho em redes LoRaWAN : uma abordagem com redes de Petri estocásticas / Israel Cardoso Araújo. – 2025.
89 f. : il. color.

Dissertação (Mestrado) – Universidade Federal do Piauí, Centro de Ciências da Natureza, Programa de Pós-Graduação em Ciência da Computação, Picos, 2025.
"Orientador: Prof. Dr. Erico Meneses Leão."
"Coorientador: Prof. Dr. Francisco Airton Pereira da Silva."

1. Servidores de rede. 2. Protocolos. 3. Rede de Petri . 4. Modelos estocásticos. 5. RSSFs. 6. LPWANs. 7. LoRaWAN. I. Leão, Erico Meneses. II. Silva, Francisco Airton Pereira da. III. Título.

CDD 044.678

Bibliotecário: Gésio dos Santos Barros - CRB3/1469

Israel Cardoso Araújo

Avaliação de Desempenho em Redes LoRaWAN: Uma Abordagem com Redes de Petri Estocásticas

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Ciência da Computação da UFPI (área de concentração: Sistemas de Computação), como parte dos requisitos necessários para a obtenção do Título de Mestre em Ciência da Computação.

gov.br Documento assinado digitalmente
ERICO MENESES LEAO
Data: 22/04/2025 16:58:46-0300
Verifique em https://validar.iti.gov.br

Prof. Dr. Erico Meneses Leão
Orientador

gov.br Documento assinado digitalmente
FRANCISCO AIRTON PEREIRA DA SILVA
Data: 22/04/2025 17:07:06-0300
Verifique em https://validar.iti.gov.br

Prof. Dr. Francisco Airton Pereira da Silva
Co-orientador

gov.br Documento assinado digitalmente
ELVIS MIGUEL GALEAS STANCANELLI
Data: 22/04/2025 11:37:16-0300
Verifique em https://validar.iti.gov.br

Prof. Dr. Elvis Miguel Galeas Stancanelli
Membro da Banca

ICP-Edu Documento assinado digitalmente
Carlos Barros Montez
Data: 22/04/2025 15:54:24-0300
CPF: ***.035.027-**
Verifique as assinaturas em https://v.ufsc.br

Prof. Dr. Carlos Barros Montez
Membro da Banca

gov.br Documento assinado digitalmente
JOSE VALDEMIR DOS REIS JUNIOR
Data: 22/04/2025 10:56:06-0300
Verifique em https://validar.iti.gov.br

Prof. Dr. Jose Valdemir Dos Reis Junior
Membro da Banca

Picos-PI
Março de 2025

Agradecimentos

Em primeiro lugar, agradeço a Deus, fonte de força, sabedoria e esperança, que iluminou meu caminho e me guiou em cada etapa desta jornada. Agradeço à minha esposa, pelo amor, paciência e apoio incondicional ao longo desta caminhada. Ao meu filho, cuja alegria e pureza iluminaram meus dias e me motivaram a seguir em frente, dedico este momento com todo meu coração. Sou grato aos meus pais, exemplos de dedicação e força, que com amor e sabedoria me ensinaram o valor do esforço e da persistência. O apoio deles foi essencial para que eu chegasse até aqui. Esta defesa simboliza não apenas meu esforço, mas o apoio de todos que caminharam ao meu lado. Muito obrigado!

"Sonhe, acredite, realize"
(Autor Desconhecido)

Resumo

As Redes de Sensores Sem Fio (RSSFs), uma categoria das Low Power Wide Area Networks (LPWANs), são tecnologias que ganharam interesse na pesquisa e comunidade industrial assim como a LoRa Wide Area Network (LoRaWAN), um protocolo que permite a implementação de aplicações de redes de longo alcance e baixa potência para a Internet das Coisas (IoT). No entanto, desafios técnicos e operacionais, a cobertura limitada dos sensores e a falta de conectividade em áreas remotas dificultam a obtenção das informações. Além disso, existe uma lacuna de um sistema automatizado e de baixo custo, já que experimentos no mundo real demandam o custo de aquisição de equipamentos. Esta pesquisa propõe a utilização de modelos estocásticos para avaliar o desempenho das redes LoRaWAN, proporcionando uma análise detalhada sobre a eficiência e a capacidade dessas redes em diferentes cenários operacionais. A primeira parte da pesquisa foca no planejamento da capacidade dos servidores de rede, de aplicação e de monitoramento, abrangendo a análise do sistema, incluindo o processamento dos dados dos sensores e sua transmissão. Utilizando um gateway único, são analisadas múltiplas métricas com foco no monitoramento contínuo da taxa de geração de pacotes de sensores agrupados por área. A segunda parte concentra-se no desempenho da camada de sensores ao nível individual, avaliando o impacto da estratégia de tolerância a falhas como mecanismo de retransmissão, utilizando múltiplos gateways e a métrica de vazão. O objetivo principal é fornecer suporte aos administradores de infraestruturas computacionais na concepção e avaliação de sistemas que utilizam o protocolo LoRaWAN, considerando aspectos de desempenho e robustez frente a falhas.

Palavras-chaves: RSSFs, LPWANs, LoRaWAN, desempenho, rede de Petri;

Abstract

Wireless Sensor Networks (WSNs), a category of Low Power Wide Area Networks (LPWANs), are technologies that are gaining interest in the research and industrial community, as is the LoRa Wide Area Network (LoRaWAN), a protocol that enables the implementation of long-range, low-power network applications for the Internet of Things (IoT). However, technical and operational challenges, limited sensor coverage, and lack of connectivity in remote areas make it difficult to provide information. In addition, there is a gap in an automated and low-cost system, since real-world experiments require the cost of acquiring equipment. This research proposes the use of stochastic models to evaluate the performance of LoRaWAN networks, providing a detailed analysis of the efficiency and capacity of these networks in different operational scenarios. The first part of the research focuses on the capacity planning of network, application, and monitoring servers, covering system analysis, including sensor data processing and transmission. Using a single gateway, several analyses are presented focusing on the continuous monitoring of the generation rate of sensor packets grouped by area. The second part focuses on the performance of the sensor layer at the individual level, evaluating the impact of the fault tolerance strategy as a retransmission mechanism, using multiple gateways and the throughput metric. The main objective is to provide support to administrators of computing infrastructures in the design and evaluation of systems that use the LoRaWAN protocol, considering aspects of performance and robustness against failures.

Keywords: WSNs, LPWANs, LoRaWAN, performance, Petri net;

Lista de ilustrações

Figura 1 – Componentes principais de uma SPN. 12
Figura 2 – Exemplo simples de SPN. 12
Figura 3 – Exemplo de SPN com fila de processamento. 13
Figura 4 – Exemplo de SPN com fila de chegada e fila de processamento. 13
Figura 5 – Exemplo genérico de gráfico de efeito dos fatores. 15
Figura 6 – Gráficos de interação genéricos. 16
Figura 7 – Metodologia de desenvolvimento do modelo analítico. 17
Figura 8 – Modelo SPN para monitoramento de desastres. 23
Figura 9 – Modelo SPN para monitoramento de desastres. 26
Figura 10 – Tempo Médio de Resposta. 32
Figura 11 – Utilização. 33
Figura 12 – Probabilidade de Descarte. 34
Figura 13 – Vazão. 34
Figura 14 – Tempo Médio de Resposta. 36
Figura 15 – Utilização. 37
Figura 16 – Probabilidade de Descarte. 37
Figura 17 – Vazão. 38
Figura 18 – Adaptação do modelo para estabelecer um estado absorvente. 39
Figura 19 – Probabilidade de absorção baseada em k elementos processados. 40
Figura 20 – Probabilidade de absorção baseada em um intervalo de tempo. 41
Figura 21 – Análise do MTTA variando a taxa de chegada. 42
Figura 22 – Análise do MTTA variando o número de núcleos. 43
Figura 23 – Visualização do impacto de vários fatores do modelo na métrica MRT. 46
Figura 24 – Interação entre fatores quanto ao seu impacto na métrica MRT. 47
Figura 25 – Arquitetura LoRaWAN proposta com tolerância a falhas. 54
Figura 26 – Modelo SPN desenvolvido a partir da arquitetura LoRaWAN. 55
Figura 27 – Visualização do impacto de vários fatores do modelo na métrica TP. . . 60
Figura 28 – Interação entre fatores quanto ao seu impacto na métrica TP. 61
Figura 29 – Tempo médio entre as gerações. 62
Figura 30 – Tempo de transmissão do nó para o gateway e do gateway para o servidor da rede. 63
Figura 31 – Peso de variação da falha. 65
Figura 32 – Impacto da variação da taxa de retransmissão na taxa de transferência do sistema. 65
Figura 33 – Variação simultânea do tempo de falha e do tempo de retransmissão. . 66
Figura 34 – Impacto do tempo médio de resposta no sistema. 67

Figura 35 – Fluxo de Validação. 71
Figura 36 – Modelo SPN da Validação. 72
Figura 37 – Ambiente de execução da validação. 74
Figura 38 – Resultado da Validação. 76

Lista de tabelas

Tabela 1 – Trabalhos Relacionados. 21
Tabela 2 – Parâmetros utilizados no modelo SPN. 27
Tabela 3 – Valores atribuídos às transições e marcações do modelo SPN. 30
Tabela 4 – Design de Experimentos 44
Tabela 5 – Combinação de fatores considerando a métrica MRT(ms). 45
Tabela 6 – Trabalhos Relacionados 52
Tabela 7 – Descrição dos principais componentes do modelo. 56
Tabela 8 – Design dos Experimentos. 59
Tabela 9 – Combinação de fatores considerando a métrica de Vazão 60
Tabela 10 – Parâmetros usados no modelo. 62
Tabela 11 – Principais transições temporizadas utilizadas no modelo SPN da Figura 36 73
Tabela 12 – Configuração do experimento 73
Tabela 13 – Parâmetros para validação 75
Tabela 14 – Resultados do Teste-T para diferentes tempos de serviço, incluindo médias e desvios padrão do modelo e do experimento. 77

Lista de abreviaturas e siglas

DoE	*Design os Experiments*
SPN	Rede de Petri Estocástica
PN	Redes de Petri
IoT	Internet das Coisas
CGROUPs	Grupos de Controle
MTTA	Tempo Total Para Absorção
MRT	Tempo Médio de Resposta
AR	Taxa de chegada
AD	Atraso de chegada
UN	Utilização
DP	Probabilidade de Descarte
TP	Vazão
RSSF	Rede de Sensor Sem Fio
LPWAN	Rede de Longo alcance e Baixo Consumo de Energia
SF	Fator de Espalhamento
CF	Canal de Frequência
BW	Largura de Banda
GSPN	Rede de Petri Estocástica Generalizada
CTMC	Cadeias de Markov em Tempo Contínuo
TIEs	Tempos Intermediários Externos
TIIs	Tempos Intermediários Internos
TCP	Protocolo de Controle de Transmissão
IP	Protocolo de Internet
MAC	Controle de acesso médio

Sumário

1 INTRODUÇÃO . . . 3
1.1 Problema . . . 4
1.2 Objetivos . . . 5
1.3 Estrutura do Trabalho . . . 6

2 REFERENCIAL TEÓRICO . . . 7
2.1 Internet das Coisas . . . 7
2.1.1 Computação em Edge . . . 8
2.1.2 Computação em Fog . . . 9
2.2 LoRa e LoRaWAN . . . 10
2.3 Redes de Petri Estotásticas . . . 11
2.4 Análise de Sensibilidade . . . 14

3 METODOLOGIA . . . 17

4 DESEMPENHO BASEADO EM MODELOS ESTOCÁSTICOS DE SISTEMAS DE DETECÇÃO DE DESASTRES IOT EM MÚLTIPLAS ÁREAS GEOGRÁFICAS . . . 19
4.1 Contextualização . . . 19
4.2 Trabalhos Relacionados . . . 20
4.3 Arquitetura do Sistema . . . 23
4.4 Modelo sem estado absorvente . . . 24
4.4.1 Resultados . . . 30
4.5 Modelo com estado absorvente . . . 38
4.5.1 Resultados do Modelo com Estado Absorvente . . . 40
4.6 Análise da Absorção de Elementos e Tempo de Processamento no Modelo Estocástico de Rede de Petri. . . . 41
4.7 Projeto de Experimentos em Análise Estocástica de Redes de Petri 43
4.7.1 Resultados: Análise dos Fatores que Afetam o Tempo Médio de Resposta. . . . 44
4.7.2 Resultados: Análise de Sensibilidade de Fatores Influentes no Desempenho do Sistema. . . . 46

5 ARQUITETURA LORAWAN COM TOLERÂNCIA A FALHAS: UMA AVALIAÇÃO DE DESEMPENHO COM FOCO NA CAMADA DE SENSORES . . . 49
5.1 Contextualização . . . 49

5.2 Trabalhos Relacionados 50
5.3 Arquitetura de Rede 53
5.4 Modelo 54
5.5 Design of Experiments 58
5.5.1 Análise de resultados 59
5.6 Análise de Impacto de Diferentes Métricas 61

6 VALIDAÇÃO 69
6.1 Processo de Validação 69
6.2 Modelo Simplificado 71
6.3 Ambiente do Experimento 72
6.4 Ferramenta de Validação 74
6.5 Cenário de Validação 75
6.6 Resultados 75

7 CONCLUSÕES 79
7.1 Fora de Escopo 80

REFERÊNCIAS 83

1 Introdução

Com o avanço acelerado da tecnologia e a crescente demanda por soluções que integrem dispositivos em larga escala, a Internet das Coisas (IoT) emergiu como um paradigma transformador, capaz de revolucionar diversos setores da economia. A IoT possibilita a interconexão de objetos físicos equipados com sensores e atuadores, que se comunicam e compartilham dados por meio da internet, promovendo automação, monitoramento e controle eficientes (MOUHA et al., 2021). Entre as diversas aplicações da IoT, o monitoramento de desastres naturais se destaca pela necessidade de detecção precoce e resposta eficiente, contribuindo diretamente para a minimização de danos ambientais e salvamento de vidas. No entanto, para que a IoT atinja seu pleno potencial, é essencial contar com tecnologias de comunicação que suportem a transmissão de dados de maneira eficiente, segura e econômica, especialmente em cenários que envolvem grandes distâncias e áreas remotas.

As Redes de Sensores Sem Fio (RSSFs) se tornaram importantes pela sua capacidade de adaptação em aplicações em larga escala (PASTÓRIO; RODRIGUES; CAMARGO, 2020). Estas redes, que são uma categoria específica das Redes de Longo Alcance e Baixo Consumo de Energia (LPWANs), destacam-se pelos dispositivos de baixa potência, custo acessível e baixas taxas de transmissão. Esses dispositivos têm a habilidade de monitorar, processar e transmitir dados sobre variáveis ambientais nos locais em que são instalados (CHAUDHARI; ZENNARO; BORKAR, 2020). Devido a essas características, as RSSFs são altamente valorizadas em uma vasta gama de aplicações IoT, como monitoramento ambiental, controle de animais e processos industriais, agricultura de precisão e sistemas veiculares. Além disso, são particularmente eficazes em ambientes hostis ou inadequados para a presença humana ou para sistemas cabeados (PETÄJÄJÄRVI et al., 2017).

A tecnologia de comunicação LoRa (*Long Range*) desempenha um papel fundamental em sistemas IoT, principalmente na obtenção de informações críticas para diversas aplicações de monitoramento. Nós de rede LoRa são projetados para operar em redes de baixa potência e de longo alcance, o que os torna ideais para aplicações em áreas remotas e de difícil acesso (GONZáLEZ-PALACIO et al., 2023). Os nós LoRa podem ser equipados com sensores que monitoram uma variedade de variáveis físicas, tais como temperatura, umidade, concentração de dióxido de carbono, velocidade do vento, entre outras. Esses dados são essenciais para diversas aplicações, incluindo monitoramento climático, controle de qualidade do ar, gestão de recursos hídricos e agricultura de precisão. Esses dispositivos podem ser configurados para operar em conjunto com outros sistemas de monitoramento, formando uma rede abrangente que cobre grandes áreas geográficas. Isso é particularmente útil em regiões de difícil acesso ou onde é necessária a cobertura de áreas extensas. A

capacidade de transmissão de dados de longo alcance com baixo consumo de energia faz de LoRa uma solução robusta e eficiente para a coleta contínua de dados ambientais (MAUDET et al., 2021).

Para que a comunicação sem fio ocorra, é utilizada a tecnologia LoRaWAN (Rede de Longa Distância e Área Ampla) para transmitir dados a longas distâncias com baixo consumo de energia. O protocolo LoRaWAN opera na camada de rede do protocolo LoRa, que utiliza a modulação de espalhamento espectral (*spread spectrum*). A arquitetura é composta por dispositivos finais (*end devices*), gateways, servidores de rede e servidores de aplicação (ULLAH; MIKHAYLOV; ALVES, 2022). Os dispositivos finais se comunicam diretamente com os gateways através de enlances sem fio, que, por sua vez, encaminham os dados para os servidores de rede. Estes servidores gerenciam o tráfego de rede, aplicam políticas de segurança e transmitem os dados para os servidores de aplicação, onde a informação é processada e utilizada conforme necessário. Além disso, diferentes frequências podem ser utilizadas, como a faixa de 868 MHz na Europa e 915 MHz nas Américas, que são padrões do LoRaWAN para comunicação de longa distância (MASEK et al., 2022). No entanto, o desempenho de redes LoRaWAN é desafiado por gargalos em servidores e congestionamento de pacotes, tornando fundamental a avaliação eficiente da arquitetura antes da implementação. Esse cenário é crítico em aplicações de monitoramento de desastres, onde a latência e a confiabilidade são determinantes para a resposta emergencial. Como experimentos em larga escala são custosos, o uso de modelos analíticos, como Redes de Petri Estocásticas (SPNs), oferece uma abordagem econômica e flexível para simular e analisar o desempenho da rede.

1.1 Problema

Um dos desafios centrais abordados nesta dissertação é a detecção precoce de desastres, com ênfase especial na identificação de incidentes críticos em áreas de alto risco. Ressalta-se que essa abordagem constitui uma contribuição particular do trabalho, que também explora outras estratégias para a monitorização e resposta a emergências. Métodos tradicionais, frequentemente baseados em relatórios manuais, enfrentam atrasos significativos na detecção e mobilização das respostas necessárias (SENDRA, 2020). Além disso, a gama limitada de sensores e os problemas de conectividade em locais remotos dificultam a aquisição em tempo real de dados ambientais. Consequentemente, existe uma necessidade urgente de um sistema automatizado e econômico, especialmente considerando os desafios logísticos e financeiros envolvidos na implantação de equipamentos e sensores para monitorização contínua e abrangente de áreas em risco (SABZALI A., 2017). A implementação de tal sistema promete não só uma detecção mais atempada de catástrofes, mas também uma resposta de emergência mais rápida e eficaz. Esta dissertação analisa a eficácia da integração de sensores e do protocolo LoRaWAN na detecção de desastres,

avaliando o desempenho do sistema em termos de processamento de dados e transmissão. Além disso, propõe a modelagem do sistema por meio de SPN para simulação e análise de desempenho. A hipótese é que o uso de LoRaWAN e sensores para detecção remota de desastres é viável e eficiente, com foco no planejamento da capacidade do servidor.

Portanto, em uma arquitetura LoRaWAN, muitos fatores podem influenciar a vazão dos dados gerados na camada de sensores até chegar na camada de aplicação. O servidor de rede e o servidor de aplicação facilmente podem se tornar gargalos no processamento dos dados. Os sensores podem operar com diferentes configurações de parâmetros, como taxa de amostragem, potência de transmissão, fator de espalhamento e periodicidade do envio de dados. Essa variação pode afetar diretamente o desempenho da rede LoRaWAN, influenciando métricas como consumo de energia, alcance da comunicação e taxa de colisão de pacotes. Ademais, a escolha inadequada desses parâmetros pode comprometer a confiabilidade da transmissão e aumentar a latência do sistema. Embora os sensores possam apresentar falhas ocasionais, os principais desafios em redes LoRaWAN estão relacionados à confiabilidade da transmissão, congestionamento de pacotes e limitações nos servidores de rede.

Para lidar com falhas pontuais nos sensores, estratégias como redundância de nós, reconfiguração dinâmica de parâmetros e retransmissão adaptativa podem ser aplicadas, garantindo maior robustez ao sistema. Além disso, falhas em outras camadas da arquitetura, como perdas no canal de comunicação e sobrecarga no processamento dos gateways, também são consideradas neste estudo (PASTÓRIO; CAMARGO, 2021). A implementação de mecanismos de retransmissão de dados em caso de falha é crucial para garantir a integridade e a confiabilidade das comunicações. A tolerância a falhas desempenha um papel crucial na garantia da robustez e confiabilidade de sistemas IoT, onde a retransmissão de dados emerge como uma solução essencial (MAGRO et al., 2022). Entretanto, a realização de experimentos em ambientes reais, com múltiplos gateways e servidores, revela-se custosa e complexa.

Dessa forma, a presente pesquisa tem como questão principal: **Como avaliar o desempenho de uma arquitetura LoRaWAN com foco no planejamento de capacidade, garantindo ampla flexibilidade de configurações?**

1.2 Objetivos

O objetivo geral desta dissertação é auxiliar administradores de infraestruturas computacionais a planejar e avaliar sistemas que utilizam o protocolo LoRaWAN, considerando aspectos de desempenho e tolerância a falhas.

Os objetivos específicos deste trabalho sem utilização de equipamentos reais são:

- Desenvolver um modelo para a análise de desempenho de sistemas de monitoramento de desastres, com foco no planejamento da capacidade dos servidores de rede, aplicação e monitoramento. O modelo deverá considerar a taxa de geração de pacotes por sensores agrupados por área, utilizando um gateway único e múltiplas métricas.
  - Um artigo foi publicado com base neste objetivo específico:
    * Araújo, Israel, et al. "DDS-P: Stochastic models based performance of *IoT* disaster detection systems across multiple geographic areas". ICT Express, Scienc e Directed, 08 set. 2024.
- Conceber um modelo que permita a análise de tolerância a falhas como mecanismo de retransmissão para sensores ao nível individual, com foco na métrica de vazão suportando múltiplos gateways.
  - Um artigo foi publicado com base neste objetivo específico:
    * Araújo, Israel, et al. "Assessing the Performance of a Fault Tolerant LoRaWAN Architecture with a Focus on the Sensor Layer and Data Retransmission Strategy". Cluster Computing, 08 dez. 2024.
- Realizar um experimento prático para validar a arquitetura proposta em um modelo SPN simplificado, utilizando a ferramenta *PASID-Validator* [1]. A validação mostra que o modelo é preciso e fiel à realidade. Por meio de técnicas estatísticas, uma análise prática pode determinar a validade do modelo proposto. A métrica de tempo médio de resposta foi aplicada no *PASID-Validator* para avaliar o desempenho do modelo.

1.3 Estrutura do Trabalho

O restante desta dissertação está organizado da seguinte forma: O Capítulo 2 apresenta os conceitos necessários para entender esta dissertação; Capítulo 3 mostra o fluxo de trabalho usado para escrever este trabalho; Capítulo 4 apresenta a análise de desempenho de um sistema de monitoramento de desastres, incluindo: trabalhos relacionados, arquitetura, visão geral dos modelos sem e com estado absorvente e análise de sensibilidade; Capítulo 5 ilustra a parte da dissertação que foca no desempenho individual dos sensores com tolerância a falhas, incluindo: trabalhos relacionados, arquitetura, visão geral do modelo, análise de sensibilidade e estudos de casos; Capítulo 6 apresenta o processo de validação do modelo SPN; Capítulo 7 apresenta as principais conclusões obtidas acerca dos estudos realizados.

[1] https://gitlab.com/pasid3/pasid-validator.git

2 Referencial Teórico

Este capítulo apresenta conceitos essenciais para a compreensão da proposta e dos trabalhos relacionados. Inicialmente, são introduzidos os conceitos de IoT e da tecnologia LoRa, destacando suas principais características. Em seguida, discute-se o funcionamento do **Design os Experiments** (DoE) e a interpretação de seus resultados. Além disso, são abordados os conceitos de Redes de Petri Estocásticas, juntamente com algumas equações utilizadas para a análise do modelo proposto.

2.1 Internet das Coisas

O conceito de Internet das Coisas surgiu da evolução de diversas áreas como microeletrônica, sensoriamento, comunicação e sistemas distribuídos. Porém, em poucas palavras, *IoT* nada mais é do que uma extensão da Internet atual, que permite controlar e monitorar a rotina diária em objetos inteligentes, mas com habilidades computacionais e de comunicação, para se conectar à Internet (LEA, 2018). A *IoT* é a interligação de redes de dispositivos físicos incorporados com dispositivos eletrônicos, software, sensores e troca de dados (também chamados de "dispositivos conectados" e "dispositivos inteligentes") (BRIEF, 2019). Sendo assim, a *IoT* transforma objetos físicos (sensores, dispositivos de comunicação com/sem fio) em dispositivos inteligentes para coletar, comunicar, monitorar e interpretar informações de seus arredores em tempo real, sobre a infraestrutura de Internet existente (PAUL; JEYARAJ, 2019).

De acordo com (PAUL; JEYARAJ, 2019), a arquitetura *IoT* consiste em cinco fases: coleta de dados, transmissão de dados, protocolo usado para transportar dados, plataforma para armazenar e processar e, finalmente, a tomada de decisão. A fase de coleta de dados pode ser realizada, por exemplo, com o uso de sensores, atuadores, transceptores e GPS. Na fase de transmissão de dados, diversos tipos de rede podem ser utilizados: redes de longa distância (*Wide Area Networks* - WAN), redes locais (*Local Area Networks* - LAN), e redes de área pessoal (*Personal Area Networks* - PAN). Na fase de transportar dados, o protocolo usado pode ser IP ou *Leach* (FERREIRA et al., 2018). Na fase de armazenamento e processamento, algumas plataformas que podem ser utilizadas são: computação em *cloud*, computação em *fog* e computação em *edge*. Por fim, a fase de tomada de decisão se refere ao contexto de aplicação, que pode ser referente à cuidados de saúde, segurança, conforto climático, entre outros.

O surgimento de várias novas aplicações de computação e dispositivos inteligentes, tornou-se possível devido à disponibilidade de um grande pool de recursos. O ambiente *IoT* requer uma infraestrutura para armazenar e processar os dados para a tomada de decisões,

no entanto os Aplicações sensíveis a atraso apresentam alto nível de exigência para as infraestruturas computacionais convencionais. Dessa forma, o uso de serviços de computação em *cloud*, *fog* e *edge* tornaram-se partes importantes para aplicações *IoT*. A computação em *cloud* tem como principal objetivo fazer melhor uso dos recursos distribuídos e resolver problemas de computação em grande escala. Por exemplo, a computação em *cloud* pode concentrar o poder de milhares de computadores em um problema, permitindo que os pesquisadores façam seu trabalho de forma mais rápido (SADIKU; MUSA; MOMOH, 2014). No entanto, o paradigma da computação em *cloud* não consegue atender aos requisitos de baixa latência e reconhecimento de local. Nesse contexto, pode-se introduzir a computação em *edge* e *fog*.

2.1.1 Computação em Edge

Com a crescente quantidade de dados gerados na *edge* da rede, a velocidade do transporte de dados está se tornando o gargalo para o paradigma da computação baseada em *cloud* (SHI et al., 2016). Nesse caso, os dados precisam ser processados na *edge* para um tempo de resposta mais curto, processamento mais eficiente e menor uso da infraestrutura de rede. A computação em *edge* se refere às tecnologias que permitem que a computação seja realizada na borda da rede, para que ela aconteça perto das fontes de dados. Dispositivos como servidores de rede podem atuar como nós de borda, processando dados localmente e reduzindo a dependência da *cloud* para tomadas de decisão em tempo real. Dessa forma, os provedores de conteúdo e desenvolvedores de aplicações podem usar os sistemas de computação em *edge*, oferecendo aos usuários serviços mais próximos a eles. Os casos de uso da computação em *edge* são serviços de localização, realidade aumentada, análise de vídeo e armazenamento em cache de dados (KHAN et al., 2019). Em (SHI et al., 2016), os autores citam algumas formas de melhorar a qualidade de serviços interativos usando a *edge*, de acordo com o tipo de aplicação:

- As aplicações de navegação podem mover os serviços de navegação ou pesquisa para a *edge* de uma área local, nos caso em que apenas alguns blocos de mapa estão envolvidos.
- A filtragem/agregação de conteúdo pode ser feita nos nós de extremidade para reduzir o volume de dados a ser transferido.
- Aplicações em tempo real, como jogos de entretenimento de auxílio à visão, realidade aumentada e saúde conectada, podem dar respostas rápidas usando nós de extremidade.

Na computação em *edge*, os dispositivos podem ser qualquer recurso de computação ou rede que resida entre fontes de dados e data centers baseados em *cloud*. Por exemplo,

um dispositivo de *edge* pode ser um *smartphone* que se encontra entre os sensores e a *cloud*. Na computação em *edge*, o dispositivo final não apenas consome dados, mas também produz dados. Na extremidade da rede, os dispositivos não apenas solicitam serviços e informações da *cloud*, mas também lidam com tarefas de computação - incluindo processamento, armazenamento, armazenamento em cache e balanceamento de carga - em dados enviados de e para a *cloud*. A *edge* deve ser bem projetada o suficiente para lidar com essas tarefas de forma eficiente, confiável, segura e com privacidade em mente. Portanto, deve oferecer suporte a requisitos como diferenciação, extensibilidade, isolamento e confiabilidade. Segundo (KHAN et al., 2019), os requisitos para utilizar os sistemas de computação em *edge* podem ser classificados em mecanismo de faturamento inteligente, suporte a Aplicações em tempo real, modelo de negócios conjunto para gerenciamento e implantação, gerenciamento de recursos, arquitetura escalonável, recursos de redundância e segurança.

2.1.2 Computação em Fog

Computação em *fog* refere-se a um paradigma de computação distribuída que move o armazenamento e a computação geralmente perto dos nós finais da rede com o objetivo de reduzir a sobrecarga da rede e computar as informações coletadas o mais rápido possível (BELLAVISTA et al., 2019). Semelhante a *cloud*, a *fog* fornece serviços de dados, computação, armazenamento e aplicações para usuários finais. Em contraste com a *cloud*, a *fog* fornece serviços com resposta mais rápida e maior qualidade. Na computação em *fog*, os serviços podem ser hospedados em dispositivos finais, como decodificadores ou pontos de acesso. A infraestrutura desta computação distribuída permite que as aplicações sejam executados o mais próximo possível de dados vindos de pessoas, processos e coisas.

A computação em *fog* oferece vantagens importantes para as aplicações atuais, como processamento em tempo real, escalonamento rápido e acessível, conteúdo local e *pool* de recursos. Assim, a computação em *fog* atraiu rapidamente muita atenção da indústria e da academia. Naturalmente, ela conecta a IoT com a infraestrutura de computação da internet. Em (CHEN; ZHANG; SHI, 2017), os autores citam algumas aplicações que exigem computação com *fog*, são eles: redes de sensores e atuadores sem fio, redes inteligentes, casas inteligentes, cidades inteligentes, manufatura conectada, sistemas de óleo e gás conectados e sistemas móveis de saúde.

Os ambientes de computação em *fog* e *cloud* devem cumprir alguns requisitos para operar corretamente e alcançar a satisfação do usuário. Em (BELLAVISTA et al., 2019) tais requisitos são definidos como escalabilidade, interoperabilidade, capacidade de resposta em tempo real, qualidade de dados, segurança, percepção de localização, mobilidade e confiabilidade.

2.2 LoRa e LoRaWAN

A tecnologia LoRa, desenvolvida pela *Semtech Corporation* e promovida pela LoRa Alliance (MAYER et al., 2019), é uma especificação das camadas físicas projetadas para proporcionar uma comunicação sem fio de longo alcance com baixo consumo de energia. Ela permite conectividade em distâncias de quilômetros, garantindo robustez na comunicação sem fio. A LoRa Alliance foi criada por alianças industriais de áreas como tecnologia da informação, automação industrial e telecomunicações, com o objetivo de padronizar redes de comunicação de grandes áreas e baixa potência, conhecidas como Low Power Wide Area Networks (LPWAN) (IQBAL; ABDULLAH; SHABNAM, 2020).

LoRaWAN é o nome dado ao protocolo que define a arquitetura do sistema e os parâmetros de comunicação, enquanto o LoRa permite o link de comunicação de longo alcance. LoRaWAN é responsável pelas camadas superiores do modelo OSI, a partir da camada de enlace e *Medium Access Control* (MAC). Enquanto LoRa define a camada física e enlace, onde LoRaWAN estabelece as regras de comunicação, segurança e gerenciamento da rede, atuando no controle do acesso ao meio e na entrega dos dados. O protocolo LoRaWAN implementa detalhes de funcionamento, segurança dos dados e qualidade do serviço, além de servir como padrão para as comunicações com os gateways, permitindo a interoperabilidade entre diferentes fabricantes de dispositivos finais e gateways (NOURA et al., 2020). Sendo um protocolo LPWAN sem fio da camada MAC, o LoRaWAN é focado em aplicações IoT que requerem grande área de cobertura e baixo custo energético, sem a necessidade de alta taxa de transmissão de dados. A cobertura pode alcançar até 4 km em áreas urbanas e mais de 12 km em áreas rurais (RAMSON et al., 2021), com taxas de transmissão variando entre 300 bps e 50 kbps, dependendo da distância entre o transmissor e o receptor (BHATIA et al., 2020).

A topologia de rede adotada pelo LoRaWAN segue o modelo estrela de estrelas, no qual os dispositivos finais (EDs – End Devices) se comunicam diretamente com gateways, que por sua vez encaminham os dados para um servidor de rede central(CHEN et al., 2020). Essa arquitetura permite que os dispositivos finais operem de forma simplificada, com baixo consumo de energia, enquanto os gateways, que possuem maior capacidade de processamento e energia, realizam o papel de ponte entre os dispositivos e a internet ou sistemas corporativos(LEE et al., 2019). Essa separação de responsabilidades contribui para a escalabilidade da rede, sendo possível conectar milhares de dispositivos a um único gateway, dependendo das configurações de rede e da densidade de tráfego (DUISEBEKOVA et al., 2019).

Além disso, o protocolo LoRaWAN define três classes de dispositivos finais — Classe A, B e C — que diferem em termos de comportamento de recepção de mensagens(CHEN et al., 2020). A Classe A, obrigatória para todos os dispositivos, é a mais eficiente em termos energéticos, pois permite a recepção apenas após o envio de uma mensagem. A Classe B

introduz janelas de recepção adicionais baseadas em sincronização com o gateway, enquanto a Classe C mantém a recepção ativa quase continuamente, à custa de maior consumo de energia(HASEGAWA; SUZUKI, 2019). Essa flexibilidade possibilita o uso do LoRaWAN em diversos cenários da Internet das Coisas, desde aplicações críticas que exigem baixa latência, até casos onde a eficiência energética é prioritária, como monitoramento ambiental e medição remota.

2.3 Redes de Petri Estotásticas

Rede de Petri (PN) é uma ferramenta de modelagem gráfica e matemática, que permite descrever e estudar sistemas de processamento de informações caracterizados como sendo concorrentes, assíncronos, distribuídos e paralelos (PETERSON, 1977; LIMA; CALLOU; ANDRADE, 2021). Como ferramenta gráfica, PN podem ser usadas como auxílio à comunicação visual, semelhante aos gráficos de fluxo, diagramas de blocos e diagramas de redes. Modelos diversas variações de PN foram propostos visando desenvolver uma ferramenta que permitisse integrar descrição formal, prova de correção e avaliação de desempenho. Dentre as variações de PN, destacam-se as Redes de Petri Estocásticas Generalizadas (GSPNs), que incorporam elementos estocásticos para representar sistemas de forma ainda mais precisa (MOLLOY, 1982; GERMAN, 2000a; MARSAN et al., 1994; MACIEL, 2023b; MACIEL, 2023a). Entretanto, por simplicidade, chamaremos essas redes apenas de SPN. Como ferramenta matemática, é possível configurar equações de estado, equações algébricas e outros modelos matemáticos que regem o comportamento de sistemas. Desde o trabalho seminal de Petri, muitas representações e extensões foram propostas, permitindo descrições mais concisas e a representação de características de sistemas não observadas nos modelos iniciais (MURATA, 1989).

Uma *spn* é representada graficamente por arcos, que podem ter pesos, transições temporizadas, transições imediatas, lugares e *tokens* (LIMA; CALLOU; ANDRADE, 2021). A Figura 1 exibe os componentes usados para modelar uma SPN. Os lugares são representados por círculos, que significam possíveis estados de um determinado sistema. Os lugares podem conter *tokens* (pontos pretos), que representam o estado da SPN (PETERSON, 1977). As transições são representadas como retângulos e correspondem à execução de uma ação. As transições temporizadas possuem um atraso associado, sendo a principal característica de uma SPN (MURATA, 1989). As transições determinísticas têm um tempo de disparo fixo, enquanto as transições imediatas não têm tempo associado. As transições imediatas possuem tempo de disparo equivalente a zero, assim têm precedência em relação às transições temporizadas e determinísticas. Arcos conectam lugares a transições e vice-versa. Um arco inibidor é representado por um pequeno círculo branco em uma borda, em vez de uma seta. Arcos inibidores são usados para desativar as transições se houver *tokens* presentes em um lugar.

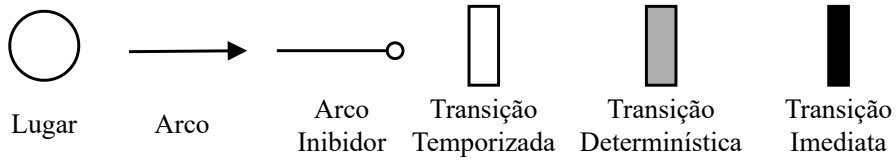


Figura 1 – Componentes principais de uma SPN.

A Figura 2 mostra uma SPN que representa um procedimento simplificado de embarque. O *token* no lugar *Fila de check-in* indica que há um passageiro pronto para iniciar o procedimento. Neste momento a transição imediata fica habilitada para disparar e remove o *token* para o lugar *Passageiro em check-in*. O tempo de atendimento varia de passageiro para passageiro, pois depende de questões como assentos disponíveis, quantidade de bagagem para despachar, documentação correta, dentre outros fatores. Quando decorrer o tempo necessário para atendimento, a transição *Finalizar check-in* é habilitada e dispara. Assim o *token* migra para o *Check-in realizado*. Como não existem mais *tokens* nos lugares anteriores, a SPN finaliza.

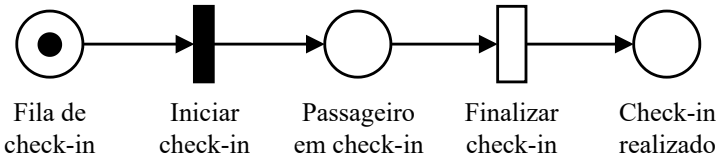


Figura 2 – Exemplo simples de SPN.

A Figura 3 mostra uma SPN com capacidade de processamento paralelo. Nesta versão do procedimento de check-in, a sala conta com quatro guichês de atendimento, onde o lugar *Guichês disponíveis* mostra quantos atendentes podem chamar um passageiro para iniciar o procedimento. Na figura temos três passageiros (*tokens*) em atendimento, três passageiros aguardando e um guichê com atendente disponível. Neste caso a transição imediata *Iniciar check-in* fica habilitada para disparar. A transição realiza uma junção, onde remove um token que representa um atendente, juntamente com o token que representa um passageiro, migrando para o lugar *Passageiro em check-in*. Quando *Finalizar check-in* dispara, ela remove um *token* e divide, onde o passageiro agora está com *Check-in realizado* e um atendente fica com o guichê disponível para novo atendimento. Considerando que neste terminal o embarque é realizado com intervalo de tempo fixo, a transição determinística *Embarcar* representa o momento de embarque. Todavia, o arco que chega nessa transição possui peso 10, ou seja, além do tempo fixo é necessário que existam 10 passageiros em *Check-in realizado*, para que a transição *Embarcar*.

A SPN da Figura 4 detalha o procedimento de *check-in*. Neste modelo, a sala de embarque possui vagas limitadas para o *check-in*. Os passageiros que se dirigem ao terminal chegam no tempo determinado pela transição exponencial *Tempo de chegada*. Quando essa transição dispara, um *token* é depositado em *Passageiros chegando* representando

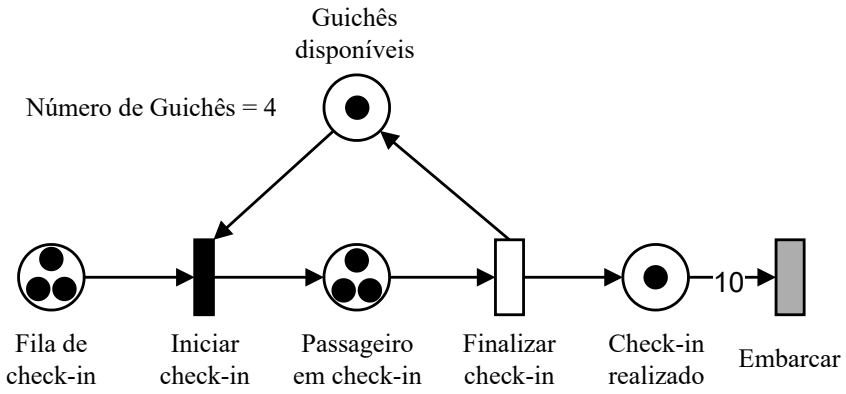


Figura 3 – Exemplo de SPN com fila de processamento.

os passageiros que querem entrar na sala de embarque. Se houver *token* em *Passageiros chegando* e um *token* em *Vagas na sala de embarque*, então a transição *Acessar a fila* é habilitada para disparo. Assim, a transição junta os *tokens* e os deposita na *Fila de check-in*. Quando houver atendente disponível, o passageiro inicia o *check-in* e uma vaga na fila é liberada.

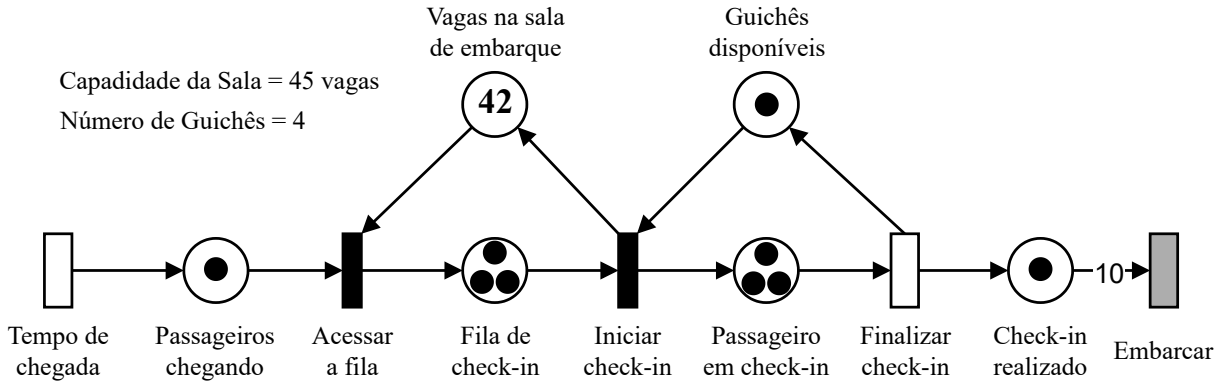


Figura 4 – Exemplo de SPN com fila de chegada e fila de processamento.

A Figura 2 apresenta um modelo SPN com estado absorvente, ou seja, o *token* segue o fluxo determinado e quando todos os *tokens* chegam ao fim do modelo, nenhuma transição pode ser habilitada novamente. A Figura 3 não possui estado absorvente, entretanto, ela finaliza porque os *tokens* disponíveis transitam até o último estado, mas os novos *tokens* não são gerados após a transição determinística disparar. A Figura 4 também não possui estado absorvente, entretanto, neste caso a movimentação dos *tokens* nunca termina, tende ao infinito, pois novos *tokens* sempre são gerados e a capacidade é sempre reposta.

O comportamento da SPN é definido em termos de um fluxo de *tokens*. Os *tokens* são criados e destruídos conforme os disparos de transição (GERMAN, 2000b). As transições imediatas representam atividades instantâneas e têm maior prioridade de disparo do que transições temporizadas. Tais transições também podem conter uma condição de guarda

e o usuário pode especificar uma prioridade de disparo diferente entre outras transições imediatas. Há também funções de guarda em SPN As funções de guarda são expressões booleanas que controlam o disparo de uma transição, declarando alguma condição em relação à marcação da rede. Se a função de guarda de uma transição produz um valor verdadeiro, ela é habilitada; caso contrário, a transição é desabilitada (MARSAN et al., 1998).

2.4 Análise de Sensibilidade

A análise de sensibilidade examina o impacto de dados de entrada específicos nos resultados, identificando possíveis falhas em sistemas computacionais. Posteriormente, são aplicadas técnicas para otimizar esses sistemas em diferentes cenários (CAMPOLONGO; TARANTOLA; SALTELLI, 1999). Em geral, projetistas de sistemas utilizam análises de sensibilidade para avaliar o quanto uma métrica é afetada por alterações no modelo (SANTOS et al., 2021). Existem várias formas de realizar essa análise, incluindo análise de regressão, análise de perturbação, simulação de Monte Carlo, análise diferencial paramétrica, análise de correlação e diferença percentual.

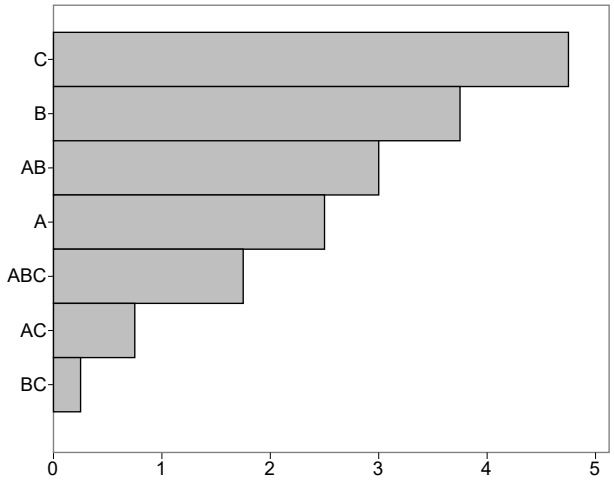
Escolher o método adequado pode ser desafiador, sendo necessário considerar os recursos computacionais disponíveis e as características dos problemas abordados. A análise de sensibilidade pode fornecer a segurança necessária e direcionar os resultados conforme a perspectiva pré-estabelecida pelos administradores do sistema. Neste trabalho, aplicamos técnicas de análise de sensibilidade baseadas no *Design of Experiments* (DoE) e na Diferença Percentual. A Diferença Percentual foi aplicada para quantificar a variação de métricas-chave ao modificar parâmetros específicos do modelo. Essa técnica calcula a alteração percentual em um indicador de desempenho quando um fator é ajustado dentro de um determinado intervalo, auxiliando na identificação de variáveis mais sensíveis dentro do sistema modelado.

Design of Experiments

O DoE é um conjunto de técnicas estatísticas que aperfeiçoam a compreensão do produto ou processo em questão. O DoE possibilita a identificação de parâmetros críticos e a melhoria da eficiência do sistema, proporcionando otimizações relevantes para aplicações práticas. Na literatura especializada (FEITOSA et al., 2021; COSTA et al., 2016; SANTOS et al., 2021), são encontradas categorias de gráficos que são frequentemente usadas em experimentos com a abordagem DoE. O gráfico de efeito dos fatores, representado por barras dispostas em ordem crescente, enfatiza o impacto relativo de cada fator. Quanto maior a barra, maior o seu impacto, permitindo uma compreensão clara das influências de cada fator. A Figura 5 apresenta um gráfico do efeito dos fatores. O gráfico apresenta três

fatores: A, B e C. O fator C é o fator de maior impacto.

Figura 5 – Exemplo genérico de gráfico de efeito dos fatores.

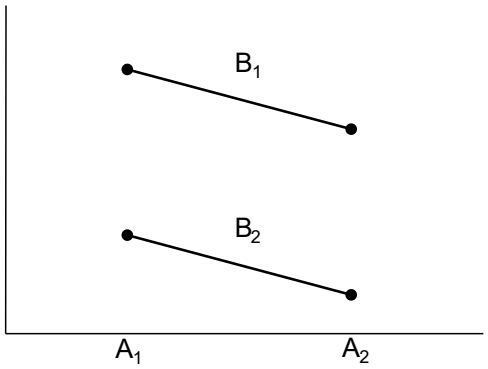


O gráfico de efeito dos fatores nos permite identificar qual interação de fatores tem um impacto mais relevante no processo de otimização ou projeto do estudo, indicando onde a atenção deve ser direcionada. A compreensão das interações entre os fatores permite selecionar a melhor combinação de medidas, identificando padrões de efeitos cumulativos ou degradantes entre os fatores. A interação entre os fatores A e B pode ser estimada usando a Equação (2.1). O gráfico {A,B(+1)} representa o efeito do fator A no nível mais elevado do fator B, enquanto o {A,B(-1)} representa o efeito do fator A no nível mais baixo do fator B.

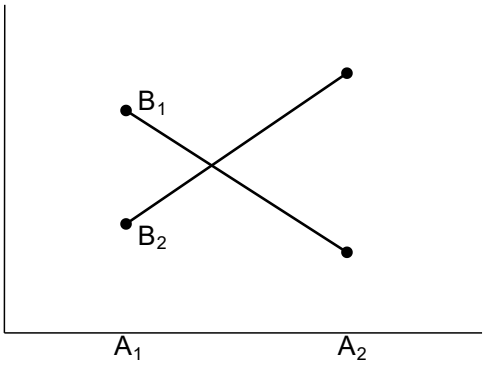
$$I_{A,B} = \frac{1}{2}(E_{A,B(+1)} - E_{A,B(-1)}) \tag{2.1}$$

O objetivo dos gráficos de interação é estabelecer relações entre fatores. Uma interação ocorre quando a influência de um fator no resultado é ampliada (ou reduzida) pela variação nos níveis de outro fator. Se as linhas no gráfico forem paralelas, isso significa que os fatores não estão relacionados. Se as linhas não forem paralelas, isso indica uma interação significativa entre os fatores em questão. A Figura 6 (a) ilustra um exemplo em que não há interação entre os fatores, uma vez que as linhas são paralelas. A Figura 6 (b) ilustra um exemplo de interação entre fatores, pois as linhas se intersectam. A variação de uma métrica para o fator A no nível A1 é maior do que a variação no nível A2. As alterações nos níveis do fator A em uma métrica dada indicam uma dependência entre os níveis do fator A e os níveis do fator B.

Figura 6 – Gráficos de interação genéricos.



(a) Gráficos de interação com baixa interação entre fatores.

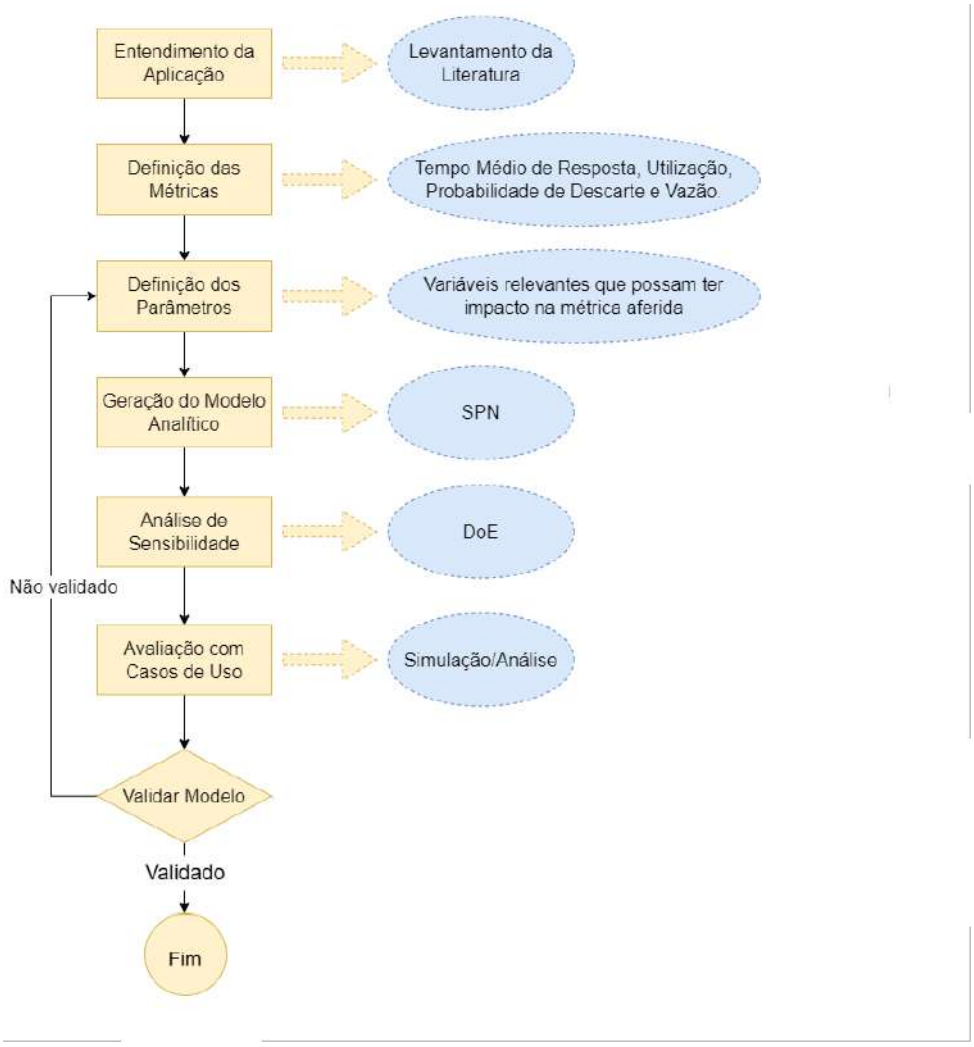


(b) Gráficos de interação com alta interação entre fatores.

3 Metodologia

Nesta seção, detalhamos a metodologia empregada ao longo do mestrado para atingir os objetivos definidos neste trabalho, com foco no desenvolvimento dos modelos para avaliar o desempenho de redes LoRaWAN. Ao longo deste período, as atividades metodológicas foram estruturadas em múltiplas etapas, conforme representado na Figura 7, que apresenta um fluxograma resumindo a estratégia adotada.

Figura 7 – Metodologia de desenvolvimento do modelo analítico.



- **Fundamentação Teórica:** é importante a realização da fundamentação teórica. Dessa forma, foi realizado um levantamento da literatura para analisar o funcionamento das redes LoRaWANs.
- **Definição das Métricas:** as métricas de interesse devem ser identificadas, levando em consideração as informações que o modelo pode fornecer para diagnosticar

o desempenho do sistema. Neste trabalho, as métricas selecionadas podem ser importantes na percepção do usuário final e úteis para administradores de sistema, são elas: *Mean Time Response* (MRT), Utilização, *Drop Probability* (DP), Vazão, *Mean Time To Absorption* (MTTA) e *Cumulative Distribution Function* (CDF).

- **Definição dos Parâmetros:** os parâmetros que serão inseridos no modelo definem o comportamento e capacidade de recursos para os componentes. Neste trabalho, os parâmetros adicionados foram a quantidade de requisições e os tempos de transmissão e execução dos sensores, servidor de rede, servidor de aplicação e servidor de monitoramento.

- **Geração do Modelo Analítico:** um modelo de desempenho usando Rede de Petri Estocásticas é desenvolvido. Nesta parte, o modelo é construído considerando as métricas e parâmetros definidos, e os resultados esperados. A Rede de Petri Estocásticas foi escolhida porque oferece o suporte necessário e suficiente para descrever os diversos comportamentos e especificações do sistema avaliado.

- **Análise de Sensibilidade:** a utilização de técnicas de análise de sensibilidade permite ao projetista do sistema identificar quais componentes são mais críticos. A metodologia utilizada neste trabalho foi o DoE.

- **Avaliação com Casos de Uso:** casos de uso foram criados para verificar a capacidade de geração de resultados do modelo. Os casos de uso foram planejados para comparar o desempenho das redes LoRaWAN.

- **Validação do Modelo:** o experimento de validação utilizou o software *PASID Validator*, permitindo configurar parâmetros de execução e de comunicação em rede.

4 Desempenho baseado em modelos estocásticos de sistemas de detecção de desastres IoT em múltiplas áreas geográficas

Este capítulo apresenta o uso de modelagem estocástica para simular operações de sensores IoT dentro de um ambiente em diferentes áreas geográficas, apoiado por computação em névoa para melhorar a integração do sistema.

4.1 Contextualização

A necessidade urgente de detectar e gerir desastres em áreas propensas a perigos exige a integração de tecnologias avançadas, como o protocolo LoRaWAN, nas estrutura IoT. Este protocolo é particularmente eficaz para monitorar vastas áreas florestais devido aos seus baixos requisitos de energia e capacidades de longo alcance. A frequência crescente de incêndios florestais, que deverá aumentar 50% até 2100, de acordo com o Programa das Nações Unidas para o Ambiente, realça as inadequações das estratégias atuais e a necessidade de soluções inovadoras (DEBDAS et al., 2022).

O principal desafio na detecção de catástrofes em áreas de alto risco reside no estabelecimento de um sistema que seja simultaneamente eficiente e fiável para a detecção precoce. Os métodos tradicionais muitas vezes dependentes de relatórios manuais, sofrem atrasos significativos na detecção e resposta. Além disso, a gama limitada de sensores e os problemas de conectividade em locais remotos dificultam a aquisição de dados ambientais em tempo real. Consequentemente, existe uma necessidade urgente de um sistema automatizado e econômico, especialmente considerando os desafios logísticos e financeiros envolvidos na implantação de equipamentos e sensores para monitorização contínua e abrangente de áreas em risco. A implementação de tal sistema promete não só uma detecção mais atempada de catástrofes, mas também uma resposta de emergência mais rápida e eficaz. As principais contribuições dentro deste contexto:

- Implementação de modelos estocásticos para implantação de sensores IoT em protocolos LoRaWAN, que simulam a implantação e operação de sensores IoT.
- Aplicação de Redes de Petri Estocásticas para Otimização Detalhada do Desempenho de Sistemas de Detecção de Desastres, melhorando a precisão do monitoramento e a eficácia da resposta.

- Análise abrangente de métricas de desempenho para sistemas de detecção de desastres baseados em *IoT*, identificando eficiências e eficácia operacionais.

- Integração estratégica de computação em nuvem com tecnologia LoRa para processamento aprimorado de dados, permitindo gerenciamento robusto de dados e processamento em tempo real.

- Projeto de sistema econômico e escalável para detecção de desastres baseada em *IoT* em áreas amplas, abordando desafios financeiros e geográficos em extensas regiões propensas a desastres.

4.2 Trabalhos Relacionados

Esta seção apresenta esforços de pesquisa que demonstram metodologias semelhantes às empregadas no presente estudo, com foco particular na prevenção e detecção de desastres em áreas de alto risco. Os artigos de pesquisa selecionados para revisão foram escolhidos com base em sua relevância para o contexto de aplicação do gerenciamento de desastres, os tipos de modelos utilizados, as métricas avaliadas e a natureza de sua arquitetura computacional, especificamente se empregavam estruturas computacionais Local, Borda, Névoa ou Nuvem. Esses estudos foram então sistematicamente categorizados em dois grupos distintos de acordo com os métodos de avaliação utilizados. A seleção do método de medição de desempenho é extremamente importante, pois impacta diretamente a qualidade, a neutralidade, a relação custo-benefício e a viabilidade prática da avaliação. É imperativo que a análise seja adaptada aos objetivos específicos do estudo, considerando as limitações dos recursos disponíveis e as características únicas do sistema sob investigação. A Tabela 1 oferece uma visão comparativa dos trabalhos selecionados, destacando as principais diferenças entre eles e os critérios específicos usados para suas avaliações. Esta análise comparativa é crucial para a compreensão das diversas metodologias e estratégias aplicadas no domínio da prevenção e detecção de desastres, especialmente em áreas propensas a riscos elevados.

Para as pesquisas baseadas na aplicação de Redes de Petri Estocásticas (SPN) para prevenção e detecção de desastres no trabalho de (TAO; XIA; ZHU, 2019), foi realizada uma análise estática e dinâmica abrangente do modelo, enfatizando a contribuição significativa da opinião de especialistas e das informações do plano de descarte. No entanto, o processo foi marcado por consideráveis desafios de carga de trabalho e eficiência, afetando a oportunidade e a precisão das previsões da resposta inicial. Com base nisso, (ZHANG et al., 2022) explorou a avaliação de risco de desastres em ambientes metroviários, introduzindo uma abordagem de rede de Petri baseada em nuvem. Os resultados da simulação demonstraram a eficácia deste método, especialmente quando integrado com a análise de árvore de falhas existente ou operadores de agregação para uma avaliação de

Tabela 1 – Trabalhos Relacionados.

Trabalhos	**Contexto de Aplicação**	**Usa SPN**	**Métricas**	**Arquitetura**
(TAO; XIA; ZHU, 2019)	Prevenção de desastres	✓	Taxa de utilização do sistema, probabilidade de estado estacionário	Local
(ZHANG et al., 2022)	Prevenção de desastres	✓	Probabilidade de fatores	Nuvem
(ZHOU; RENIERS, 2016)	Prevenção de desastres	✓	Probabilidade de escalonamento	Local
(ZHOU; RENIERS; COZZANI, 2023)	Prevenção de desastres	✓	Probabilidade de escalonamento	Local
(HAN; JIE; ZUYAN, 2000)	Prevenção de desastres	✓	Tempo e posição de ocorrência de desastres	Local
(HRISTOV et al., 2018)	Detecção de desastres	✗	Temperatura Média	Local
(LUNA; GUTIéRREZ; ESPINOSA, 2020)	Detecção de desastres	✗	Taxa de ruído e intensidade	Nuvem
(ANTUNES et al., 2019)	Detecção de desastres	✗	Temperatura Média	Local
(DEBDAS et al., 2022)	Detecção de desastres	✗	Temperatura Média	Local
(TSAKMAKIS et al., 2022)	Detecção de desastres	✓	Atraso médio	Local
(HAJOVSKY et al., 2023)	Detecção de desastres	✗	Variação de Temperatura	Nuvem
(ASTERIOU; PAPADIMITRIOU; NICOPOLITIDIS, 2021)	Detecção de desastres	✗	Tempo médio de resposta	Borda
(TAYEH et al., 2021)	Detecção de desastres	✗	Tempo médio de resposta, taxa de falso positivo	Nuvem
Este trabalho	Detecção de desastres	✓	Tempo médio de resposta, probabilidade de descarte, Vazão, utilização	Névoa

risco abrangente. Outras contribuições de (ZHOU; RENIERS, 2016) e (ZHOU; RENIERS; COZZANI, 2023) estabeleceram um modelo de combate a desastres, investigando a otimização de estratégias e alocação de recursos. As suas conclusões sugerem que uma distribuição de recursos baseada na gravidade do desastre supera uma abordagem de distribuição igualitária. O trabalho de (HAN; JIE; ZUYAN, 2000) ampliou esta estrutura

incorporando a dinâmica da resposta do serviço de desastres em cenários de desastres residenciais. Na mesma linha, (NING et al., 2010) apresentou uma metodologia para avaliar a eficácia da resposta a emergências, oferecendo informações significativas para os comandantes de desastres melhorarem as estratégias de resposta e otimizarem as operações de resgate.

A pesquisa em detecção precoce de eventos de desastre é exemplificada por (HRISTOV et al., 2018), que apresentou duas novas abordagens: uma empregando veículos aéreos não tripulados (UAVs) com tecnologia de imagem especializada, e a outra uma sinergia de UAVs de asa fixa e de asa rotativa. (LUNA; GUTIéRREZ; ESPINOSA, 2020) discutiu uma solução IoT ponta a ponta que emprega técnicas de geolocalização baseadas em LoRaWAN, visando a detecção oportuna de desastres para mitigar danos materiais, econômicos e ambientais. (ANTUNES et al., 2019) focou na proteção de áreas isoladas e vulneráveis por meio de um sistema proativo de detecção de desastres, validado por meio de experimentos em desastres reais.

(DEBDAS et al., 2022) introduziu um método para detectar eventos de desastre em ambientes densos, enfatizando a detecção de longo alcance e a resposta rápida. O estudo de (TSAKMAKIS et al., 2022) desenvolveu um sistema baseado em LoRaWAN, empregando um modelo MAC híbrido com autômatos de aprendizagem para minimizar atrasos de transmissão. (HAJOVSKY et al., 2023) combinou técnicas de medição de superfície e profundidade, confirmando a viabilidade de sua abordagem para intervenção oportuna em desastres subterrâneos e superficiais. (ASTERIOU; PAPADIMITRIOU; NICOPOLITIDIS, 2021) propôs protocolos adaptativos que fazem a transição entre os modos ALOHA e TDMA para transmissão eficiente de dados de zonas críticas de desastre. Por fim, (TAYEH et al., 2021) apresentou o sistema LPWAN PERSONAL (PERLY), que visa melhorar a confiabilidade e funcionalidade dos sistemas de localização de bombeiros e avaliação de estado.Diversos estudos avaliaram o desempenho de redes LoRaWAN para aplicações de monitoramento e detecção de eventos críticos. Como, por exemplo, a analise de cobertura e confiabilidade do protocolo e o impacto da retransmissão de pacotes em cenários urbanos. No entanto, essas abordagens frequentemente não consideram uma modelagem matemática robusta que permita analisar diferentes configurações da rede antes da implementação. Medições empíricas geralmente podem ser limitadas pelos recursos e pela infraestrutura disponível.

Assim, este trabalho apresenta um modelo SPN para avaliar o desempenho de sistemas de monitoramento de desastres, oferecendo uma abordagem preditiva e abrangente que contrasta com os métodos tradicionais. Isso permite uma análise flexível e escalável antes da implantação real. O objetivo é fornecer um método eficiente para prever gargalos, otimizar recursos e melhorar a confiabilidade da comunicação em redes LoRaWAN. Ao usar modelagem abstrata, esta abordagem fornece informações sobre o comportamento do

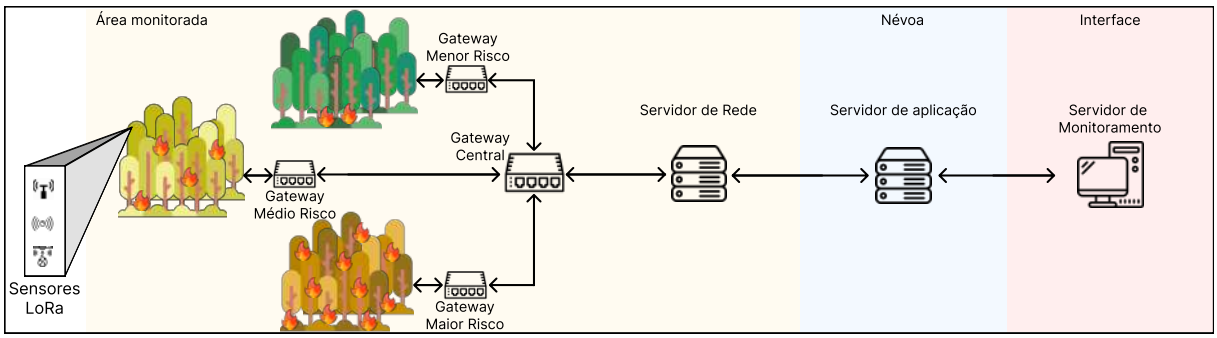
sistema em vários cenários, incluindo aqueles não encontrados anteriormente. A avaliação envolveu duas configurações distintas de modelo, analisadas por meio de métricas como Tempo Médio de Resposta (MRT), Verdadeiro Positivo (TP), Probabilidade de Detecção (DP) e utilização de recursos. O modelo foi expandido para incluir um componente de estado absorvente, permitindo análises de métricas adicionais como Tempo Médio de Absorção (MTTA) e Função de Distribuição Cumulativa (CDF). Esse aprimoramento facilitou a análise de sensibilidade por meio de Design of Experiments (DoE), esclarecendo o impacto de componentes individuais no desempenho geral do sistema. O design flexível do modelo permite ajustes de parâmetros para adaptar o sistema a requisitos específicos, reduzindo a dependência de estruturas físicas predefinidas.

4.3 Arquitetura do Sistema

Esta seção delineia a arquitetura considerada, melhorando a compreensão do projeto proposto. Ele aborda aspectos essenciais, como a estrutura abrangente do sistema, nós LoRa e sensores.

A Figura 8 retrata a arquitetura de rede prevista, revelando a estrutura integral do sistema de monitoramento de desastres em regiões florestadas. Este sistema compreende um conjunto de sensores, *The Things Network* (TTN), em conjunto com um Servidor de aplicação e um Servidor de Monitoramento (AFANASYEV et al., 2019). A arquitetura é projetada para abranger diversas áreas que apresentam níveis de risco variados. Inclui zonas de alto, médio e baixo risco de desastres, com o entendimento de que a estratégia para lidar com potenciais desastres varia correspondentemente. Esta segmentação estratégica permite uma gestão eficaz e adaptada às necessidades específicas de cada zona, aumentando assim a segurança (ROMEO et al., 2020).

Figura 8 – Modelo SPN para monitoramento de desastres.



Baseada na tecnologia LoRaWAN, a rede constitui uma arquitetura do tipo infraestruturada. Pontos de acesso centralizados orquestram a comunicação dos dispositivos LoRa com a Internet. Neste esquema, um gateway coleta dados de sensores, que estão dispersos pela paisagem. A implementação exigiu a formulação de um modelo onde os dispositivos

finais aderissem à classe A da especificação LoRaWAN. Esta classe é escolhida devido ao seu menor consumo de energia, essencial para uma operação sustentável, particularmente em aplicações *IoT*, com a vida útil prolongada do dispositivo e a substituição mínima da bateria são cruciais. Os dados dentro do sistema são armazenados em um servidor de armazenamento em rede, garantindo agregação e preservação eficiente de dados. Utilizar serviços prestados pela TTN, plataforma especializada em *IoT*, facilita a coleta, processamento e armazenamento de dados em um banco de dados com eficiência e economia.

A utilização dos serviços da TTN aproveita sua infraestrutura robusta e estabelecida, com ampla cobertura de rede LoRaWAN e recursos sofisticados de gerenciamento de dados. Esta escolha estratégica sustenta a aquisição e armazenamento contínuo de dados, salvaguardando a integridade e a disponibilidade dos dados para análise posterior. O Servidor de Aplicação, um servidor web HTTP, faz interface com o banco de dados implementado pelo TTN, modificando os dados em um formato legível. Além disso, os recursos de integração do TTN melhoram o processamento e a visualização de dados, permitindo o desenvolvimento de soluções e painéis sob medida para monitorar dados de sensores.

Um desafio fundamental no desenvolvimento de uma arquitetura abrangente de monitorização de desastres é garantir uma cobertura adequada em territórios extensos. A quantidade de sensores e seus custos de aquisição podem representar obstáculos para a implementação eficaz do sistema. É vital considerar estrategicamente o posicionamento dos sensores, levando em consideração a topografia, a vegetação e outros fatores importantes para otimizar a detecção precoce de desastres.

Outro desafio envolve impedimentos naturais e condições climáticas adversas que podem afetar a precisão e integridade do sensor. Mudanças meteorológicas repentinas, como ventos fortes, chuvas intensas ou temperaturas elevadas, podem prejudicar a funcionalidade do sensor e causar danos físicos. A implementação de medidas de proteção e a garantia da robustez dos sensores são fundamentais para garantir a sua eficácia operacional em condições climáticas variadas e terrenos desafiadores. Enfrentar estes desafios exige planejamento estratégico, seleção de tecnologia resiliente, incorporação de redundância e manutenção regular para garantir o desempenho ideal do sistema de monitoramento de desastres.

4.4 Modelo sem estado absorvente

No discurso que se segue, será delineada a arquitetura do modelo SPN (Stochastic Petri Net), acrescida de detalhes relativos ao seu fluxo de execução. A narrativa percorrerá metodicamente as camadas sequenciais do modelo, oferecendo uma descrição incisiva das características idiossincráticas inerentes a cada camada. Devido à complexidade do modelo,

as análises foram realizadas utilizando simulações estacionárias, com margem de erro aproximada de 2%. A ferramenta Mercury[1] foi utilizada para criar o modelo e, portanto, gerar os resultados utilizando a versão 5.0.1 (MACIEL et al., 2017).

Visão geral do modelo

A Figura 9 ilustra uma perspectiva abrangente do modelo. Este modelo é desenvolvido estrategicamente para facilitar aos engenheiros e analistas a avaliação crítica do desempenho do sistema, abrangendo uma variedade de recursos. O modelo particionado em cinco camadas principais: chegada de dados (alternativamente conhecidos como "admissões"), gateway, servidor TTN, servidor de aplicação e servidor de monitoramento (TALEB-BERROUANE; KHAN; AMYOTTE, 2020).

A camada de admissão é emblemática no início das solicitações do sistema, iniciadas a partir da detecção de anomalias pelos sensores, sinalizando posteriormente o Gateway. O Gateway assume um papel crucial na distribuição equitativa dos pedidos emanados dos sensores, alinhando-os com as suas admissões designadas. O Servidor TTN, servindo como base da Rede da Internet das Coisas, resume a fase inicial de processamento. Nesta fase, a autenticação do sensor é conduzida e os dados são armazenados provisoriamente antes de serem enviados ao Servidor de aplicação.

É imprescindível ressaltar que as solicitações propostas pelo Gateway são navegadas em direção à camada fog. O Servidor de aplicação tem a tarefa de assimilar os dados acumulados no Servidor TTN, processando-os ainda mais para criar uma interface gráfica que reflita os dados coletados. O Servidor de Monitoramento, em conclusão, significa o término das solicitações atravessadas pela camada fog, provisionando assim uma interface acessível ao usuário (LEE; MITICI, 2020).

O influxo de solicitação originado pelo sensor compreende vários constituintes. A Tabela 2 delineia os elementos que constituem este modelo, especificamente Pad1, Pad2 e Pad3. Estes elementos refletem a prontidão do sistema para assimilar novas solicitações emanadas de três zonas distintas representadas no layout arquitetônico mostrado na Figura 8. Sequencialmente, Parq1, Parq2 e Parq3 simbolizam a assimilação dessas solicitações dentro da fila de entrada do sistema. Os componentes acima mencionados englobam solicitações contendo dados de entrada, que são posteriormente divulgados através do gateway.

Além disso, os intervalos temporais entre chegadas sucessivas são denotados por transições denominadas Tad1, Tad2 e Tad3, que podem ser visualizadas na Tabela 3. Inicialmente, as solicitações navegam pelas transições T1, T2 e T3 antes de serem integradas na camada fog. A alocação dessas solicitações depende da distribuição executada pelo

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

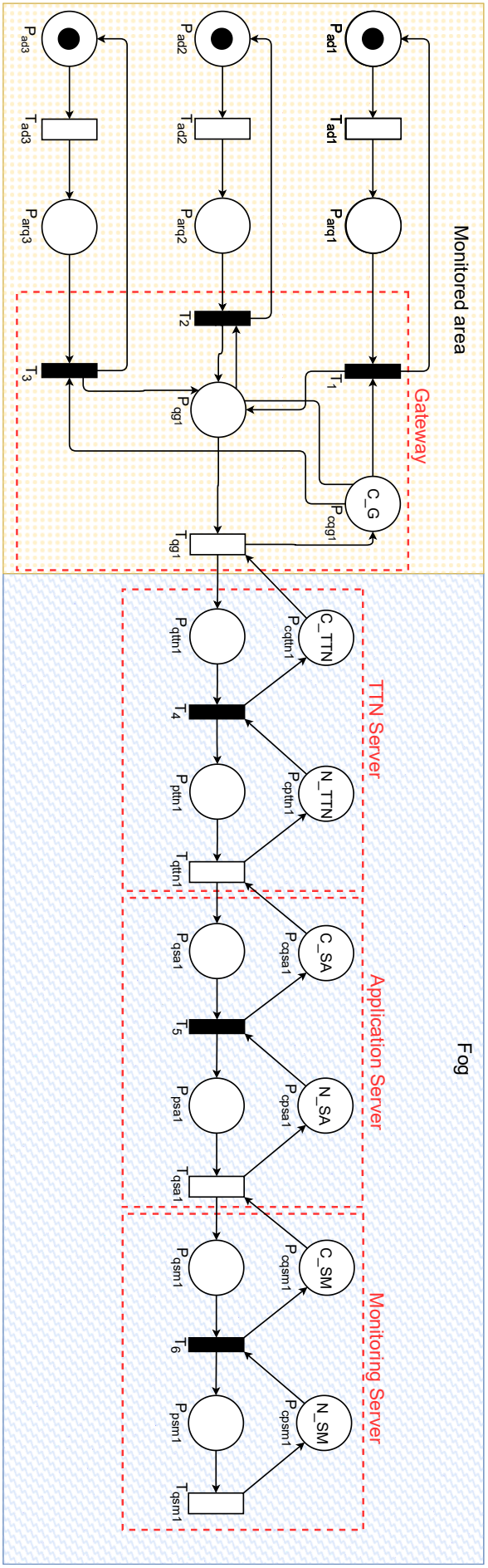


Figura 9 – Modelo SPN para monitoramento de desastres.

Tabela 2 – Parâmetros utilizados no modelo SPN.

Tipo	**Componentes**	**Descrição**
Lugares	Pad1, Pad2, Pad3	Novos pedidos de sensores.
	Parq1, Parq2, Parq3	Fila de dados dos sensores.
	Pqg1	Fila no gateway de coleta de dados.
	Pcqg1	Fila de capacidade do gateway de coleta de dados.
	Pcqttn1, Pcqsa1, Pcqsm1	Fila de capacidade do Servidor TTN, Servidor de Aplicação, e Servidor de Monitoramento, respectivamente.
	Pcpttn1, Pcpsa1, Pcpsm1	Capacidade de processamento do Servidor TTN, Servidor de Aplicação e Servidor de Monitoramento.
	Pqttn1, Pqsa1, Pqsm1	Fila do Servidor TTN, Servidor de Aplicação e Servidor de Monitoramento.
	Ppttn1, Ppsa1, Ppsm1	Processamento do Servidor TTN, Servidor de Aplicação, e Servidor de Monitoramento, respectivamente.
Transições Temporizadas	Tad1, Tad2,Tad3	Tempo entre chegadas de pedidos de sensores.
	Tqg1	Tempo de transição da fila do gateway para o Servidor TTN.
	Tqttn1	Tempo de transição de dados para o Servidor de Aplicação.
	Tqsa1	Tempo de transição de dados para o Servidor de Monitoramento.
	Tqsm1	Tempo de transição de dados no Servidor de Monitoramento.
Marcação de Lugares	C_G, C_TTN, C_SA, C_SM	Capacidade máxima das filas do Gateway, Servidor TTN, Servidor de Aplicação e Servidor de Monitoramento.
	N_TTN, N_SA, N_SM	Número de recursos para processamento.

gateway, simbolizado por Pqg1. Este gateway atua como um intermediário facilitando a comunicação entre diversos dispositivos ou redes, permitindo assim a transferência de dados entre eles. Após a transição Tqg1, os dados são transmitidos para a camada de Névoa para processamento subsequente.

As representações de fila para solicitações no Servidor TTN, no Servidor de aplicação e no Servidor de Monitoramento são designadas como Pcqttn1, Pcqsa1 e Pcqsm1, respectivamente. Estas filas são acessíveis através do gateway através da transição Tqg1. As durações de processamento e armazenamento dos servidores são indicadas por Pcpttn1, Pcpsa1 e Pcpsm1, refletindo as capacidades de processamento do Servidor TTN, do Servidor

de aplicação e do Servidor de Monitoramento. A transição Tqttn1 significa a transmissão de dados para o Servidor de Aplicação, enquanto Tqsa1 representa o encaminhamento de dados para o Servidor de Monitoramento, culminando com a transição Tqsm1. Por fim, os Locais de Marcação simbolizam a capacidade máxima das filas do Gateway, Servidor TTN, Servidor de Aplicação e Servidor de Monitoramento, bem como a contagem de recursos disponíveis para processamento.

Métricas do modelo

Este segmento elucida um conjunto de métricas pertinentes derivadas da resolução do modelo SPN. Os principais indicadores examinados incluem tempo médio de resposta, taxa de utilização de processamento de névoa, discrepâncias na comunicação de dados através da probabilidade de descarte na torre de transmissão e rendimento geral do sistema. Cada métrica é calculada utilizando o modelo SPN, com a metodologia de cálculo elaborada posteriormente.

A métrica *Mean Response Time* MRT é fundamental na avaliação da eficiência do sistema, estimando a duração média necessária para processar e endereçar solicitações. No âmbito dos incidentes de desastre, esta métrica, conforme definida na Equação 5.2, adquire maior importância, fornecendo uma medida quantificável da prontidão do sistema em reagir às situações, dado pela esperança (Esp) de haver requisições nos lugares de fila e de processamento, multiplicado pela Arrival Delay (AD). O AD representa o tempo que uma mensagem leva para ser recebida por um nó após ser enviada. Um tempo médio de resposta abreviado é imperativo para uma tomada de decisão rápida e eficaz, melhorando assim a segurança e as medidas de proteção para indivíduos e bens afetados. Respostas rápidas são fundamentais para mitigar os danos, conter a progressão do desastre e refinar as estratégias de combate a incêndios (VALENTIM et al., 2022). Consequentemente, o MRT surge como um indicador crucial para avaliar a eficácia e a capacidade de resposta de um sistema em cenários de emergência, como surtos de catástrofes.

$$\begin{aligned}\mathbf{MRT} = \frac{1}{1-DP} \times \Big(& Esp(Pqd1) + Esp(Pd1) \\ & + Esp(Pqd2) + Esp(Pd2) \\ & + Esp(Pqtc) + Esp(Pqgat) \\ & + Esp(Pqe1) + Esp(Pe1) + Esp(Pqe2) \\ & + Esp(Pe2) + Esp(Pqb) \Big) \times AD\end{aligned} \tag{4.1}$$

A equação 4.2 delineia a métrica de *Utilization* (UN), um parâmetro crítico para medir a eficiência e a distribuição de carga de processamento dentro de um sistema. Dentro do âmbito do modelo que está sendo analisado, a "Utilização"é calculada com ênfase na

camada de Névoa – chamada de "Névoa" – que é parte integrante da funcionalidade do sistema. A avaliação da utilização facilita a identificação de potenciais estrangulamentos sistêmicos ou casos de subutilização. Tais dados são inestimáveis para a configuração e dimensionamento ideais do *hardware* nos cenários avaliados, conforme discutido em (CHAHROUR et al., 2021). O cálculo do uso da camada de névoa envolve a divisão do número esperado de tokens na fila de processamento de névoa pela sua capacidade de processamento, com o quociente sendo então dimensionado para uma porcentagem. Essa métrica é indispensável para análise de desempenho e identificação de áreas de melhoria no sistema.

$$\mathbf{UN} = \frac{Esp(Pn1)}{CoreE} \times 100 \tag{4.2}$$

A *Drop Probability* (DP), representada na Equação 4.3, é uma métrica essencial para avaliar a capacidade de processamento de um sistema, especialmente no gerenciamento de grandes volumes de dados. Ela quantifica a probabilidade de solicitações ou dados serem descartados devido a uma sobrecarga do sistema ou a sua incapacidade de processá-los de forma eficiente. No contexto do presente estudo, a DP é particularmente examinada no gateway, um elemento central que faz a ponte entre os sensores e a camada de Névoa. O cálculo do DP envolve a análise da probabilidade (P) de tokens dentro de uma fila (Pcqtc) específica na torre de comunicação, sendo a métrica resultante expressa em porcentagem. Isto é conseguido multiplicando o valor probabilístico por 100%. Tal análise do DP é crucial na identificação de problemas sistémicos que podem resultar na perda de informação crítica, particularmente em cenários como o combate a incêndios. Compreender e mitigar estas questões é vital para aumentar a eficiência do sistema no processamento de dados e garantir a sua confiabilidade.

$$\mathbf{DP} = P(Pcqtc = 0) \times 100 \tag{4.3}$$

O *Trhoughput* (TP) é uma medida quantitativa crucial para avaliar a capacidade de um sistema no processamento de solicitações e na transferência de dados. No âmbito do modelo que está sendo avaliado, o TP é calculado para avaliar a esperaça do token em uma fila (Pqc1). A Vazão do sistema é determinada pelo número esperado de tokens nesta fila, dividido pela duração necessária para processar e limpar esses tokens da fila. A equação 4.4 descreve explicitamente a metodologia utilizada para calcular o fluxo no modelo. Esta métrica serve como um indicador tangível da capacidade do sistema de lidar com as demandas de processamento e transmissão de dados, sendo assim fundamental na avaliação de seu desempenho geral e na orientação de possíveis melhorias.

$$\mathbf{TP} = \frac{Esp(Pqc1)}{TempoDeServiço} \tag{4.4}$$

Tabela 3 – Valores atribuídos às transições e marcações do modelo SPN.

Tempo de Transição	Tad1, Tad12, Tad3	50.0 (ms)
	Tqg1	5.0 (ms)
	Tqttn1	25.0 (ms)
	Tqsa1	20.0 (ms)
	Tqsm1	1000.0 (ms)
Lugares	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

4.4.1 Resultados

Esta secção elucida e examina os resultados derivados de dois cenários distintos dentro do modelo de combate a desastres proposto. Testes experimentais foram conduzidos em ambientes variados, alterando a contagem de núcleos de processamento na camada de Névoa e a duração necessária para o processamento nesta camada. O objetivo principal desses testes foi avaliar o desempenho e a eficácia do sistema sob diversas configurações. Ao modular a quantidade de núcleos de processamento nos nós de nevoeiro, podemos discernir o impacto de tais ajustes nas capacidades de monitoramento e processamento de dados do sistema.

Através desta análise, o desempenho da camada de Névoa é examinado. Ao justapor resultados de cenários díspares, é possível verificar quais configurações geram desempenho mais eficiente e superior no monitoramento e na supressão de desastres. Essas informações são fundamentais para melhorar a arquitetura do sistema e informar as decisões na implementação de estratégias preventivas e de resposta a emergências de catástrofes. Os dados acumulados facilitam uma análise completa do desempenho do sistema, produzindo informações cruciais para refinar e otimizar a arquitetura existente. O aproveitamento desta abordagem baseada em dados permite a identificação de áreas para melhoria, ajustes de recursos e priorização na alocação de recursos, amplificando assim a eficácia do sistema de combate a catástrofes. Esta metodologia contribui significativamente para uma resposta a catástrofes mais proficiente, rápida e eficaz, reduzindo os danos e reforçando a proteção de vidas e propriedades.

Ao analisar os cenários propostos para monitoramento e supressão de desastres, foram empregadas métricas específicas para avaliar o desempenho do sistema. As métricas selecionadas abrangem o Tempo Médio de Resposta (MRT), Probabilidade de Descarte

(DP), Vazão (TP) e Utilização da camada de computação em névoa. Essas métricas oferecem uma perspectiva abrangente e objetiva sobre a funcionalidade do sistema, avaliando habilmente sua eficiência em termos de tempo de resposta, capacidade de processamento e utilização de recursos. Ao considerar estas métricas, podem ser identificadas áreas potenciais de melhoria, permitindo ações corretivas para otimizar o sistema de monitorização e supressão de desastres.

No cenário inaugural, o tempo de serviço de processamento foi fixado em 1000 ms, variando de forma empírica apenas no número de núcleos utilizados em cada elemento da camada de Névoa. Cinco valores distintos foram testados: 2 núcleos, 4 núcleos, 8 núcleos, 16 núcleos e 32 núcleos. No cenário subsequente, o número de núcleos de processamento foi consistentemente definido em 8, enquanto o tempo de serviço variou empiricamente entre 100 ms, 500 ms, 1.000 ms, 1.500 ms e 2.000 ms. Essas variações nos cenários facilitaram a análise de como a quantidade de recursos na camada de Névoa impacta a eficiência do sistema e a influência do tempo de serviço estendido no desempenho do sistema e na capacidade de processamento.

Cenário 1 - Variação no Número de Núcleos da Névoa.

O primeiro cenário experimental envolveu alterações sistemáticas no número de núcleos de processamento, dobrando progressivamente o poder de processamento: 2, 4, 8, 16 e 32 núcleos. Esta variação permitiu uma avaliação da influência direta do poder de processamento aumentado nas métricas do sistema. As descobertas resultantes estão representadas nas figuras abaixo, ilustrando a análise gráfica. Observou-se que um aumento no número de núcleos de processamento na camada de Névoa se correlaciona com um melhor desempenho do sistema. Especificamente, houve um aumento notável na taxa de processamento, na capacidade de lidar com grandes volumes de dados e na eficiência de resposta em tempo real do sistema.

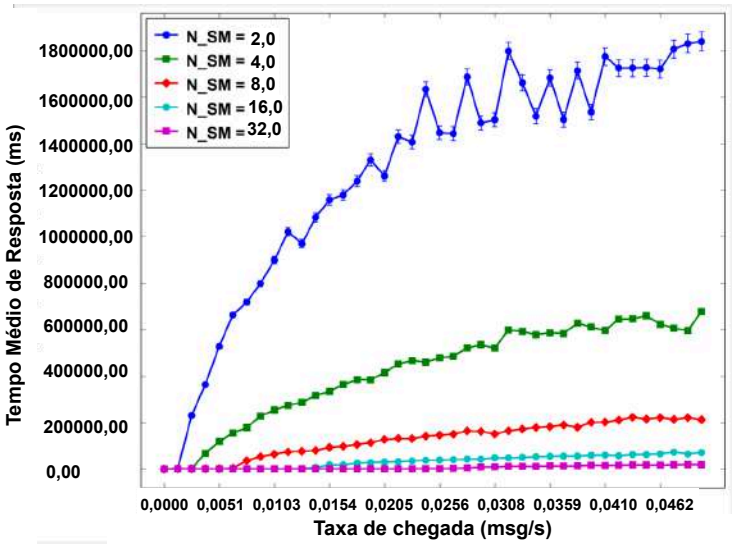
Resultados do Cenário 01

A Figura 10 exibe os resultados do Tempo Médio de Resposta (MRT) sob diversas configurações de recursos. Foi percebido que cenários com maior alocação de recursos apresentaram menores MRTs. Notavelmente, os cenários com 8, 16 e 32 nós de computação manifestaram resultados comparáveis, com apenas o cenário de 4 nós exibindo um MRT marginalmente elevado. No entanto, foi observado um aumento significativo no MRT quando a alocação de recursos foi reduzida a 2 nós, particularmente em relação à taxa de chegada (AR).

A convergência dos valores do MRT em cenários com 8, 16 e 32 nós sugere o potencial de otimização de recursos. Especificamente, o emprego de 8 nós de computação

parece ser suficiente para lidar com uma demanda substancial, com o MRT flutuando entre 0 ms e 200.000 ms dentro de AR de até 1.800.000 ms. Isto sugere que aumentar o número de nós além de 8 não produz melhorias consideráveis no desempenho do MRT. O impacto do crescimento da AR foi notavelmente pronunciado em cenários com recursos limitados (2 nós). Em cenários com alocação aprimorada de recursos (8, 16 e 32 nós), o MRT apresentou escalonamento mínimo em resposta ao aumento da carga de trabalho, reforçando a adequação de 8 nós para uma demanda substancial sem comprometer o desempenho do sistema. Um padrão consistente identificado nos cinco cenários foi a estagnação do crescimento do MRT em relação a AR. Especificamente, o crescimento do MRT tendeu a estabilizar além de AR de 0,0308 msg/s. Consequentemente, esta faixa AR, abrangendo 0,0308 msg/s a 0,0462 msg/s, poderia oferecer um MRT médio estimado em cerca de 100.000 ms, especialmente ao utilizar 8 nós de computação.

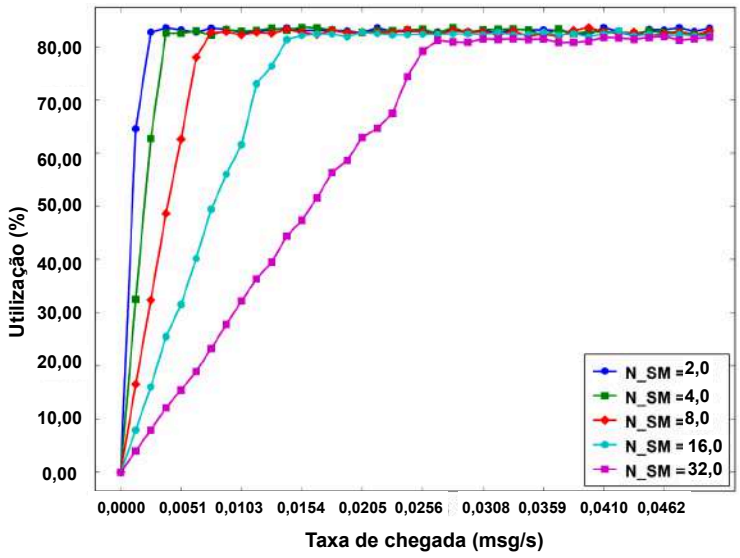
Figura 10 – Tempo Médio de Resposta.



A Figura 11 delineia o percentual de utilização desta camada, revelando uma escalada proporcional a AR dos dados. Observou-se que a utilização desta camada intensificou-se progressivamente, atingindo seu apogeu em 100%. Os dados indicaram que à medida que o volume de dados aumentava, a camada de processamento de nevoeiro era cada vez mais aproveitada, atingindo a sua capacidade máxima de processamento. Durante as avaliações com 2 núcleos de processamento, a utilização atingiu rapidamente 100% com AR superior a 0,0001 msg/s, mantendo esse pico durante todo o teste. Com 4 núcleos de processamento, notou-se um aumento gradual na utilização da camada, atingindo taxas máximas quando AR ultrapassou 0,0025 msg/s. No contexto de 8 núcleos de processamento, o sistema demonstrou um declínio na utilização além de AR de 0,0051 msg/s. Em testes com 16 e 32 núcleos de processamento, a AR dos dados foi de 0,0025 msg/s e 0,0256 msg/s, respectivamente. Os resultados sublinharam que um aumento nos núcleos de processamento levou

a capacidades melhoradas de processamento de dados, diminuindo subsequentemente as taxas de utilização.

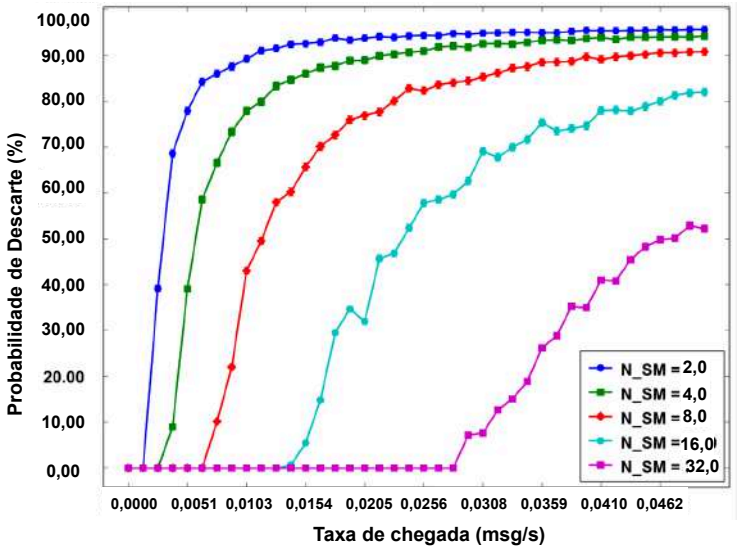
Figura 11 – Utilização.



A Figura 12 ilustra as probabilidades de descarte observadas em diferentes testes, correspondendo a respectiva AR dos dados. Na configuração de 2 núcleos, a probabilidade de descarte começou em AR superior a 0,0010 msg/s, aumentando acentuadamente para estabilizar entre 90% e 95%. Tais resultados eram esperados, dado o menor poder de processamento nesta configuração. Com 4 núcleos, os descartes iniciaram em uma AR acima de 0,0030 msg/s, com a taxa subindo acentuadamente de 85% para 90%, e aumentando ainda mais à medida que o teste avançava. O experimento com 8 núcleos de processamento indicou o início da probabilidade de descarte em um AR de 0,0060 msg/s, oscilando entre 80% e 85%. Esta constatação implica que um maior número de núcleos de processamento aumenta a capacidade do sistema para gerir pedidos, reduzindo assim a probabilidade de descarte. Na configuração de 16 núcleos, a AR foi de 0,0154 msg/s, com taxa de descarte variando de 70% a 80%. Embora um aumento nos núcleos tenha mitigado a probabilidade de descarte, a melhoria foi menos pronunciada em comparação com a configuração de 8 núcleos.

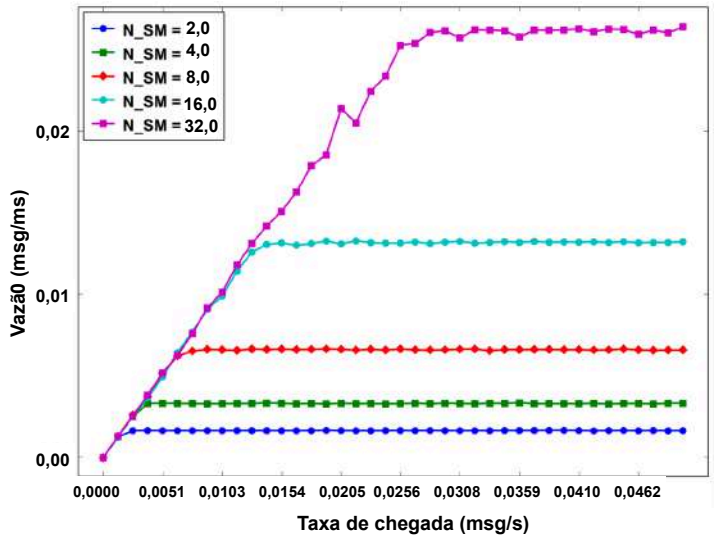
Na configuração de 32 núcleos, que proporcionou o desempenho ideal entre os testes, a taxa de entrada foi de 0,0286 msg/s com taxa de descarte de 50%, indicando uma redução considerável na probabilidade de descarte. Estas descobertas sugerem que um aumento nos núcleos diminui efetivamente a probabilidade de descartar solicitações, permitindo assim que o sistema acomode ARs mais elevados. Recomenda-se a utilização de sistemas com um número apropriado de núcleos para um desempenho ideal e uma probabilidade minimizada de descarte em cenários de alta demanda.

Figura 12 – Probabilidade de Descarte.



A Figura 13 explora a influência da capacidade de processamento no rendimento do sistema em vários cenários. O sistema alcançou seu maior rendimento com uma AR de 0,0051 msg/s em configurações com 2, 4 e 8 núcleos de processamento, com a Vazão permanecendo relativamente estável entre 0,001 msg/s e 0,005 msg/s durante todo o teste. Esses resultados indicam que o sistema pode gerenciar com eficiência as ARs de dados dentro dessa faixa, sem encontrar gargalos ou degradação significativa do desempenho. O sistema exibiu menor rendimento em um AR entre 0,0103 msg/s e 0,0154 msg/s, estabilizando em aproximadamente 0,01 msg/s. Isto sugere um limite de desempenho ideal

Figura 13 – Vazão.



Cenário 2 - Variação no Tempo de Serviço da Névoa.

O segundo cenário experimental investigou o desempenho do sistema sob tempos variados de serviço dentro dos dispositivos de Névoa. As durações do processamento foram aumentadas gradativamente para os seguintes valores: 100 ms, 500 ms, 1,000 ms, 1,500 ms e 2,000 ms. Esta metodologia sistemática permitiu uma avaliação abrangente da eficácia das métricas empregadas no sistema, visando um entendimento aprofundado do seu funcionamento. Através da representação gráfica, a interação entre o tempo de atendimento e a eficiência do sistema é elucidada, oferecendo uma perspectiva detalhada sobre as ramificações das variações nesta variável de controle. A alteração no tempo de serviço reflete mudanças na capacidade dos nós de processamento paralelo, e não na sua contagem numérica.

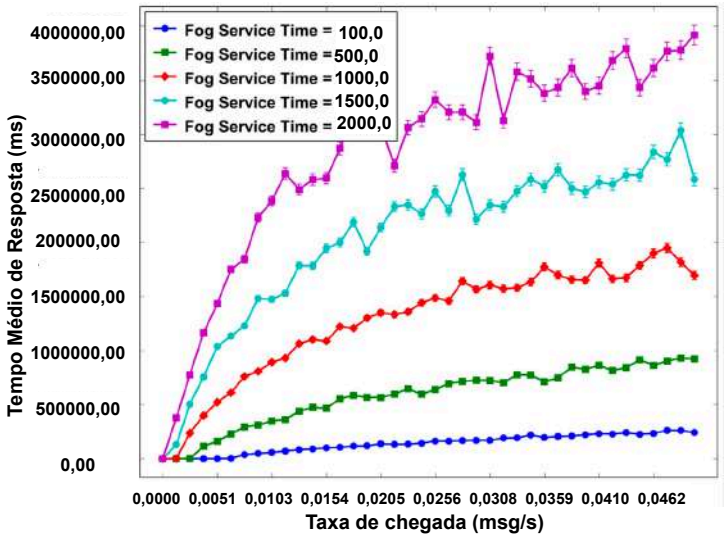
Resultados do Cenário 02

A Figura 14 delineia os resultados do Tempo Médio de Resposta (MRT) em vários cenários de tempo de serviço. Os dados recolhidos ilustram que os cenários com durações de serviço prolongadas exibiram MRTs significativamente elevados. Uma taxa de MRT notavelmente maior foi observada no cenário com tempo de serviço fixo de 2,000 ms, em correlação com a taxa de chegada (AR). Uma análise do crescimento da AR em diferentes cenários de tempo de serviço revelou um impacto substancial em configurações de tempo de serviço mais longos, especialmente em 2,000 ms. Em cenários com tempo de serviço de 1500 ms, notou-se um aumento acentuado na AR. Por outro lado, à medida que os tempos de atendimento foram reduzidos para 1000 ms, 500 ms e posteriormente para 100 ms, foi observada uma variação no MRT.

A análise destes cinco cenários indica uma tendência de estagnação no crescimento do MRT em relação a AR, particularmente além de uma taxa de AR de 0,0308 msg/s. Dentro desta faixa de AR, de 0,0308 msg/s a 0,0462 msg/s, pode-se esperar um MRT médio. Por exemplo, usando nós com tempo de serviço de 1.000 ms, o MRT médio foi estimado em 150.000 ms. Esses resultados sugerem que o tempo de atendimento exerce uma influência significativa na AR e no tempo médio de resposta de um sistema.

A Figura 15 apresenta o percentual de utilização da camada de Névoa, revelando um aumento proporcional com os dados AR. Observou-se que a utilização da camada subiu gradativamente, chegando a atingir valores superiores a 90%. Os dados indicam que com tempos de serviço mais longos, a camada de processamento de Névoa é utilizada de forma mais intensiva, atingindo sua capacidade máxima de processamento. Durante avaliações com tempos de serviço de 2.000 ms e 1.500 ms, a utilização da camada atingiu rapidamente 90% em um AR superior a 0,0001 msg/s, persistindo neste pico durante todo o teste. Esta tendência sugere que nestas condições, a camada de processamento de nevoeiro enfrenta uma elevada procura, operando na sua plena capacidade de processamento. Com um

Figura 14 – Tempo Médio de Resposta.

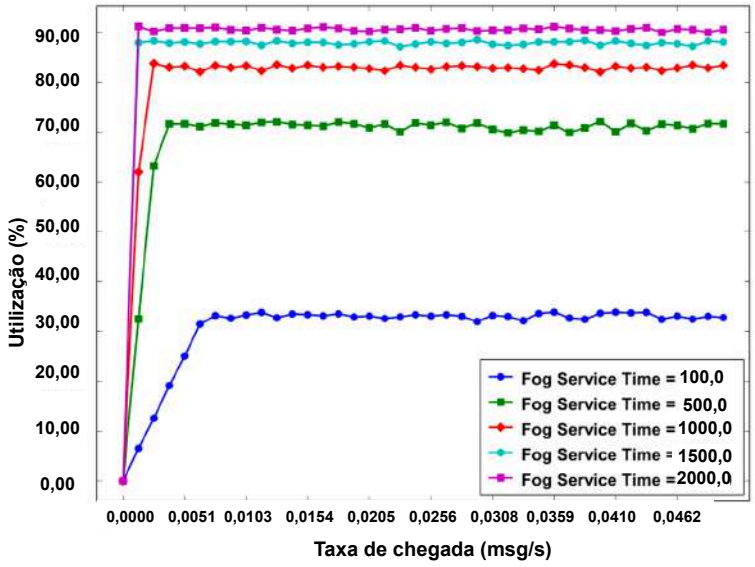


tempo de serviço de 1000 ms, notou-se uma escalada gradual na utilização da camada, atingindo um máximo de 85% em um AR superior a 0,0015 msg/s, indicando uma demanda comparativamente menor.

A redução do tempo de serviço para 500 ms, onde a AR excedeu 0,0040 msg/s, exibiu uma tendência decrescente na utilização da camada em comparação com cenários anteriores. Isto sugere que apesar de um AR aumentado, o menor tempo de serviço restringe a capacidade de processamento da camada de Névoa. No caso de um tempo de serviço de 100 ms, com um AR de 0,0025 msg/s, o sistema demonstrou uma capacidade aprimorada de processar mais informações, levando a uma menor dependência da camada de Névoa. As descobertas ressaltam que tempos de serviço mais longos resultam em uma utilização mais intensiva da camada, enquanto tempos de serviço mais curtos permitem que o sistema processe um volume maior de informações, reduzindo assim a demanda na camada de Névoa.

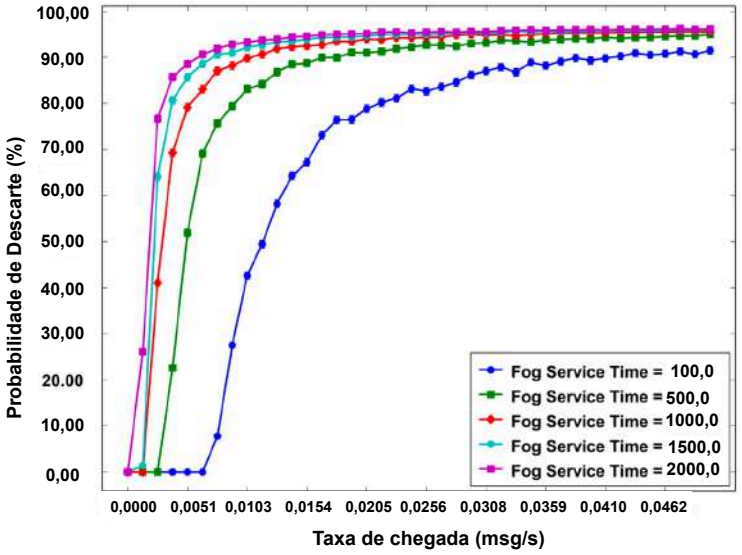
A Figura 16 explora a probabilidade de descarte em diferentes configurações. Na configuração de tempo de serviço de 100 ms, a probabilidade de descarte de solicitações começou quando AR ultrapassou 0,0051 msg/s, com taxas de descarte variando entre 75% e 85%. Estes resultados foram antecipados, dado o menor tempo de serviço desta configuração. Na configuração de 500 ms, foram observados descartes em ARs acima de 0,0025 msg/s, com taxas de descarte aumentando de 80% para 90%. Com um tempo de serviço de 1000 ms e um AR de 0,0015 msg/s, a probabilidade de descarte começou entre 85% e 90%, indicando que tempos de serviço mais longos levam a taxas de descarte mais altas. Para tempos de atendimento de 1.500 ms e 2.000 ms, com AR de 0,005 msg/s, as taxas de descarte oscilaram entre 90% e 95%. Esses resultados enfatizam a correlação

Figura 15 – Utilização.



direta entre tempo de atendimento, AR dos dados e probabilidade de descarte. O aumento dos tempos de serviço leva a um aumento significativo na probabilidade de descarte, independentemente da AR. Esta tendência pode ser atribuída a tempos de atendimento mais longos, implicando em redução da capacidade de processamento, resultando em acúmulo de solicitações e consequentes descartes.

Figura 16 – Probabilidade de Descarte.

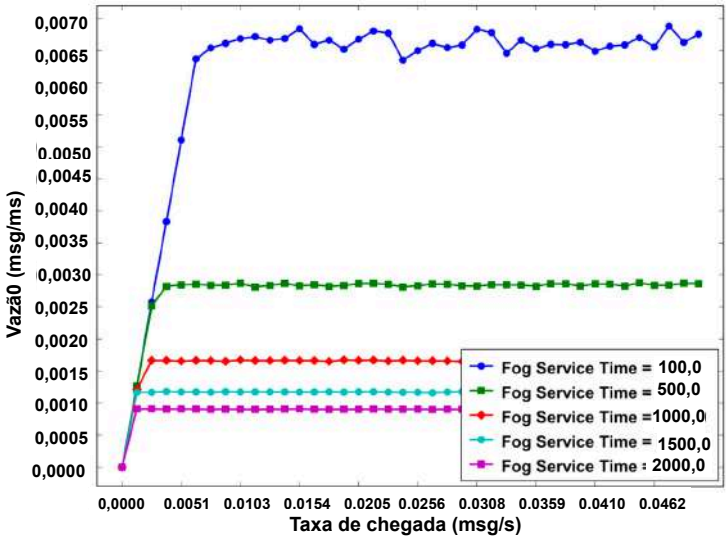


Na Figura 17, a Vazão do sistema foi avaliado sob diversos tempos de serviço, observando ARs e taxas de fluxo correspondentes. Estes parâmetros são cruciais para a análise e otimização de sistemas de comunicação. O sistema apresentou um bom desempenho com tempo de serviço de 100 ms, a mensagem chegou a 0,0051 msg/s, e o Vazão foi de 0,0065

msg/s. A Vazão denota a taxa de transmissão de mensagens bem-sucedida por unidade de tempo, enquanto AR reflete a frequência de recepção de mensagens pelo sistema.

Com um tempo de serviço aumentado para 500 ms, foi observado um declínio no rendimento nos ARs entre 0,0025 msg/s e 0,0040 msg/s, com uma Vazão caindo para aproximadamente 0,0030 msg/s. Além deste ponto, uma Vazão estável foi mantida. Nos cenários com tempos de atendimento de 2.000 ms, 1.500 ms e 1.000 ms, a menor vazão foi observada em um AR de 0,0010 msg/s, onde a vazão permaneceu entre 0,0001 msg/s e 0,0015 msg/s até a conclusão do teste . Estas descobertas destacam a importância de considerar o tempo de serviço e a AR na avaliação do desempenho dos sistemas de comunicação. Tais dados são fundamentais para projetar e dimensionar sistemas de comunicação, permitindo a seleção de parâmetros ideais para alcançar a máxima eficiência e desempenho. Compreender a interação entre tempo de serviço, AR e rendimento é fundamental para otimizar o desempenho dos sistemas de comunicação e garantir a entrega eficiente de mensagens.

Figura 17 – Vazão.



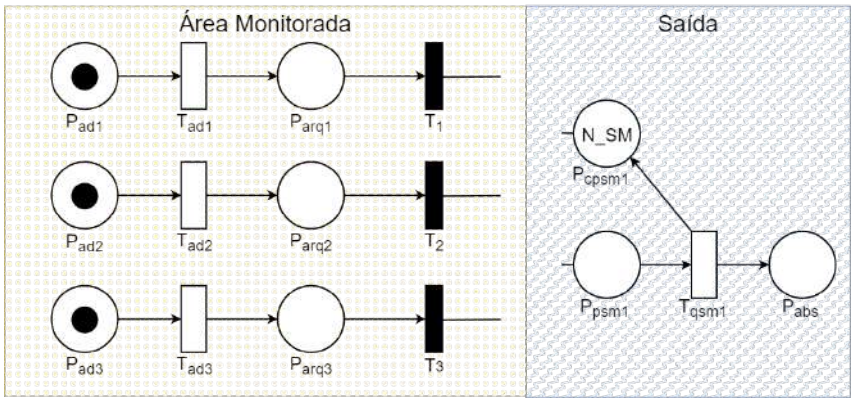
4.5 Modelo com estado absorvente

Esta seção investiga a configuração do sistema e os resultados derivados da integração do modelo em um estado absorvente. O modelo SPN, conforme detalhado na Seção 4.4, juntamente com seus parâmetros correspondentes, também é aplicável ao modelo aqui discutido. Este discurso descreve meticulosamente as modificações executadas para facilitar a transição do modelo para um estado absorvente.

Estrutura do Modelo

A Figura 18 ilustra os segmentos específicos do modelo que foram modificados para incorporar um estado absorvente eficiente. Essas modificações foram concentradas nas camadas de Admissão e Processamento de Saída, que são críticas no processo de absorção dos elementos que entram no sistema. A reformulação dos componentes do modelo possibilitou uma análise mais precisa da dinâmica de processamento no sistema, avaliando o tempo necessário para que os elementos em transição sejam completamente absorvidos pelo estado final. Além disso, permitiu calcular a probabilidade de absorção de uma determinada quantidade de elementos em um determinado prazo. Estas métricas são cruciais para compreender a capacidade e eficiência do sistema na gestão das demandas recebidas, fornecendo assim informações sobre o desempenho do sistema, especialmente em termos da sua capacidade de resposta e eficácia no processamento e assimilação dos elementos recebidos.

Figura 18 – Adaptação do modelo para estabelecer um estado absorvente.



As admissões nos locais Pad1, Pad2 e Pad3 foram alteradas. Essas mudanças determinaram as solicitações específicas que poderiam entrar no sistema a partir desses locais. A variável KEL foi introduzida para encapsular solicitações, denotando os K elementos que devem entrar no sistema. O influxo de elementos no sistema é limitado a um valor pré-determinado, conhecido como K elementos, originários dos locais mencionados. Por exemplo, se KEL for definido como 10, no máximo 10 elementos de cada local poderão entrar no sistema. Esta limitação desempenha um papel fundamental no controle do influxo de elementos no sistema e na garantia de um intercâmbio regulado e equilibrado entre lugares e transições. Além disso, esta metodologia auxilia na prevenção de sobrecargas e gargalos desnecessários no sistema (OROJLOO; AZGOMI, 2019).

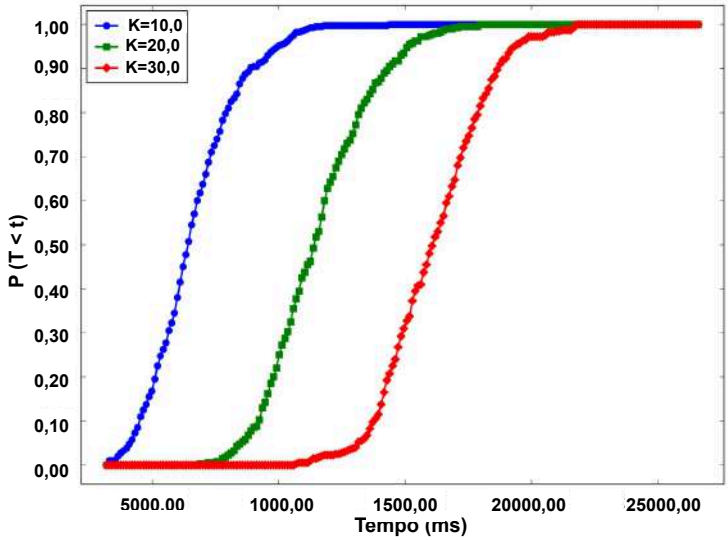
A camada Processamento de saída foi modificada para incluir o local Pabs (um local absorvente). Pabs serve para representar transações ou solicitações absorvidas pelo sistema. A integração deste elemento permite a determinação precisa do número de solicitações absorvidas pelo sistema. A absorção ocorre quando uma requisição navega pelo sistema e chega ao Pabs (CARVALHO et al., 2020). Os usuários são obrigados a incorporar a métrica

de absorção associada a este local para realizar os cálculos necessários. Ao correlacionar o número de elementos que entram no sistema (KEL) com o local dos Pabs, torna-se viável avaliar a proficiência do sistema no processamento desses elementos. Esta análise produz informações valiosas sobre o desempenho operacional e a capacidade de absorção do sistema.

4.5.1 Resultados do Modelo com Estado Absorvente

A análise representada na Figura 19 envolveu a coleta de dados de execução do sistema para valores variados de K (o número de solicitações), empiricamente entre 10, 20 e 30. A Função de Distribuição Cumulativa (CDF) foi empregada para representar a probabilidade de conclusão da execução dentro de um determinado prazo, enquanto o Tempo Médio de Absorção Total (MTTA) foi utilizado para avaliar a eficiência do sistema. A análise indicou que um menor número de elementos de entrada, como exemplificado por K definido em 10, correlacionou-se com uma maior probabilidade de conclusão dentro do intervalo de tempo desejado. Isto sugere que sistemas com um número reduzido de elementos de entrada tendem a ser mais eficientes em termos de conclusão de execução, com um intervalo que se estende de 0 a quase 10000 unidades de tempo. Por outro lado, observou-se que um aumento no número de elementos de entrada, denotados por valores K de 20 (variando de 7500 a quase 15000 unidades de tempo) e 30 (começando em torno de 11000 e estendendo-se até pouco mais de 20000 unidades de tempo), diminuiu o probabilidade de conclusão dentro do intervalo de tempo predefinido.

Figura 19 – Probabilidade de absorção baseada em k elementos processados.

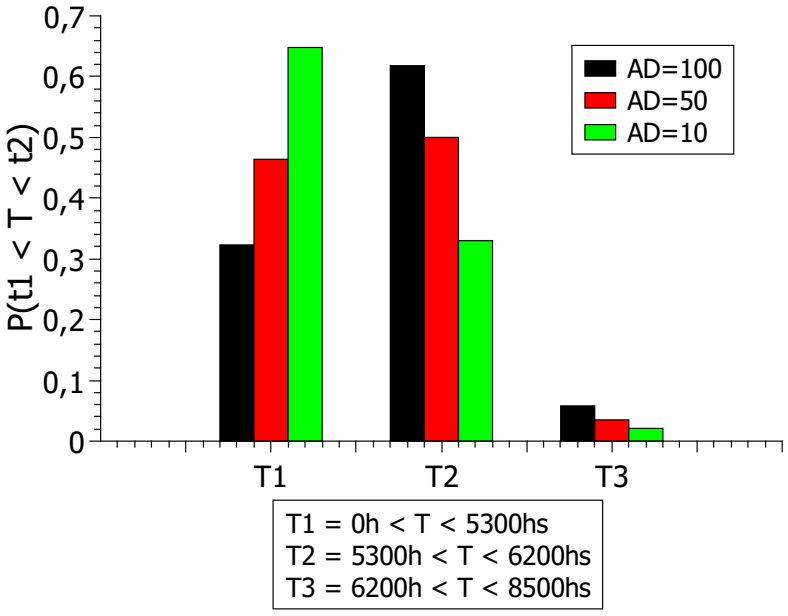


4.6 Análise da Absorção de Elementos e Tempo de Processamento no Modelo Estocástico de Rede de Petri.

A Figura 20 delineia três cenários – T1, T2 e T3 – cada um representando condições distintas para a absorção de elementos dentro de intervalos de tempo especificados. No cenário T1, o CDF é empregado para verificar a probabilidade de conclusão do processo de absorção no intervalo de tempo de 0 a 5300 horas, correlacionando-se ao MTTA. Esta investigação, realizada em modo transitório, considera taxas de chegada de elementos de 10, 50 e 100 unidades, com uma cota constante de 10 elementos designados para absorção.

No cenário T1, com taxa de chegada de elementos de 10 unidades, a probabilidade de conclusão da absorção variou entre 60% e 70%. Quando a taxa de chegada foi aumentada para 50 unidades, esta probabilidade oscilou entre 40% e 50%, e a uma taxa de chegada de 100 unidades, variou entre 30% e 40%. No cenário T2, que vai das 5300 às 6200 horas, observou-se que a uma AR de 10 unidades, a probabilidade de absorção completa situou-se entre 30% e 40%. Para taxas de chegada de 50 e 100 unidades, as probabilidades foram encontradas entre 50% e 60%, e 60% e 70%, respectivamente. O cenário T3, abrangendo o intervalo de 6,200 a 8,500 horas, demonstrou que a probabilidade de absorção completa era mínima, variando entre 0% e 10% para todas as taxas de chegada testadas (10, 50 e 1000 unidades). Isto indica que taxas mais altas de chegada de elementos levam a um aumento na probabilidade de absorção completa durante um período mais longo, proporcionando tempo suficiente para que os elementos sejam totalmente absorvidos pelo sistema.

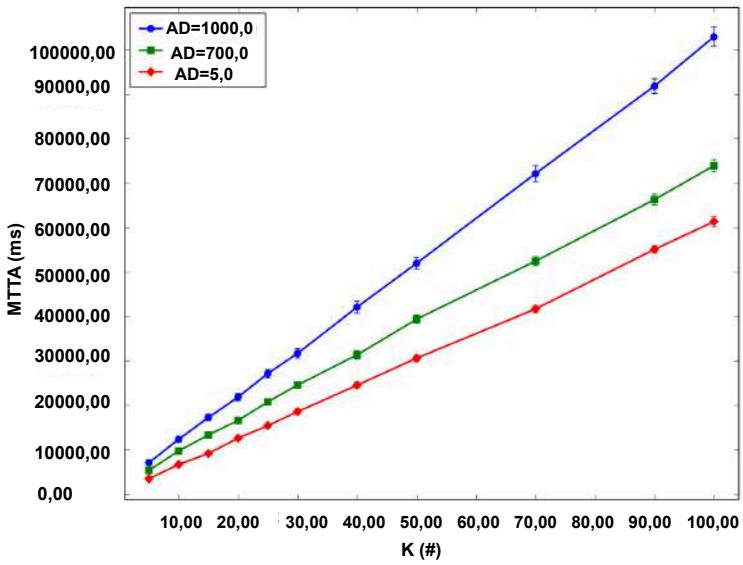
Figura 20 – Probabilidade de absorção baseada em um intervalo de tempo.



A análise da Figura 21 revela uma correlação direta entre a taxa de chegada do

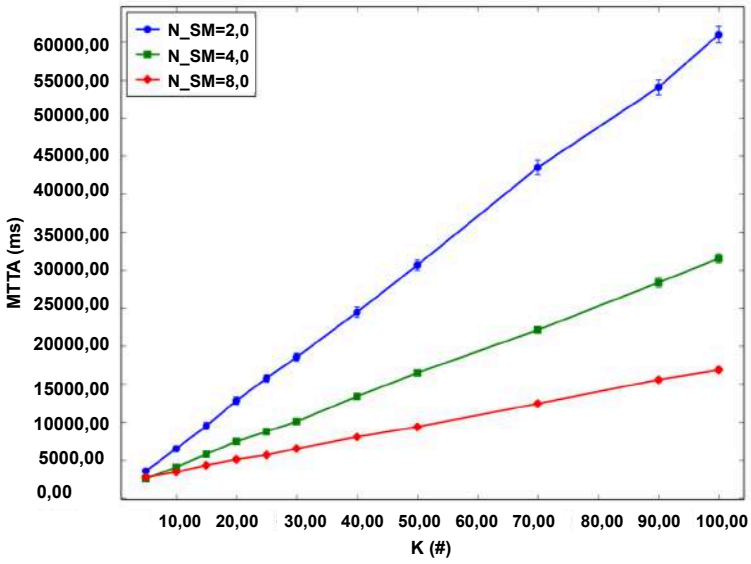
elemento e o MTTA. À medida que a taxa de chegada aumenta, o mesmo acontece com o MTTA, devido aos intervalos prolongados entre as entradas dos elementos, o que por sua vez afecta o tempo médio de processamento necessário para estes elementos. Por exemplo, processar 100 elementos a uma taxa de chegada de 50 unidades resulta num MTTA de aproximadamente 50,000 ms, significando que, em média, são necessários 50,000 ms para processar todos os 100 elementos desde a sua chegada. Por outro lado, se a taxa de chegada for aumentada para 7000 unidades, o MTTA aumenta para quase 70,000 ms para o mesmo volume de elementos, atribuível ao longo período de espera entre as chegadas. Notavelmente, a uma taxa de chegada de 1000 unidades, o MTTA atinge cerca de 100,000 ms. Este padrão enfatiza que taxas mais altas de chegada de elementos levam ao aumento de MTTAs, impactando a eficiência e a capacidade de resposta do sistema. Portanto, ao projetar e otimizar sistemas de processamento, especialmente aqueles que funcionam na camada de Névoa, é fundamental considerar a taxa de chegada dos elementos.

Figura 21 – Análise do MTTA variando a taxa de chegada.



A Figura 22 apresenta resultados que ilustram uma relação inversa entre o número de núcleos de processamento na camada fog e o MTTA. Com o aumento do número de núcleos, há uma diminuição acentuada do MTTA. Por exemplo, o processamento de 100 elementos com 2 núcleos resulta em um MTTA de cerca de 60,000 ms, enquanto com 4 núcleos, o MTTA é reduzido para aproximadamente 30,000 ms. A melhoria mais significativa é observada com 8 núcleos, onde o MTTA cai para aproximadamente 15,000 ms. Estas descobertas implicam que o número de núcleos influencia significativamente o tempo médio de transferência de tarefas dentro do sistema, sugerindo que aumentar o número de núcleos diminui o tempo médio de transferência. Tais informações são cruciais para otimizar a alocação de recursos em sistemas baseados em nevoeiro para melhorar o desempenho e a eficiência das operações de transferência de tarefas.

Figura 22 – Análise do MTTA variando o número de núcleos.



4.7 Projeto de Experimentos em Análise Estocástica de Redes de Petri

Esta seção realiza um exame analítico dos elementos que influenciam o desempenho do sistema em estudo, utilizando o modelo de Redes de Petri Estocásticas (SPN). Os SPNs são fundamentais na modelagem e análise de sistemas complexos, oferecendo uma compreensão profunda de seus comportamentos e atributos de desempenho. O foco desta análise centra-se no MRT, uma métrica crítica indicativa da eficiência e capacidade de resposta de um sistema. Ao examinar a interação entre vários componentes do sistema e seu impacto no MRT, pretendemos descobrir possíveis melhorias e caminhos para otimização (POLITIS et al., 2017).

Para facilitar esta análise, detalharemos a arquitetura do sistema, abrangendo seus processos, recursos e eventos. As interações entre esses elementos e seu efeito coletivo no tempo de resposta do sistema serão analisadas metodicamente. Além disso, discutiremos os resultados derivados da aplicação do modelo SPN, com o objetivo de colher informações que possam orientar modificações para melhorar o desempenho do sistema. Também nos aprofundaremos na metodologia DoE, uma técnica estruturada e estatística utilizada para estudar a influência de variáveis-chave dentro de um sistema. O DoE é particularmente valioso na exploração de diversos cenários e na extração de dados precisos sobre os efeitos das variáveis no desempenho do sistema. Espera-se que a integração do modelo SPN com metodologias DoE produza informações mais profundas, orientando nossas recomendações para otimização de maneira robusta e empírica.

Visão geral do projeto de experimentos

O presente estudo emprega métodos sofisticados de Design de Experimentos (DoE) para conduzir uma extensa análise de sensibilidade. DoE é uma abordagem metódica para planejar, projetar e analisar experimentos para obter conclusões válidas e objetivas de forma eficiente e eficaz. Este processo envolve a realização de uma série de testes em que variáveis experimentais são manipuladas para discernir os fatores causais das mudanças observadas na resposta de saída (FUKUDA, 2018). As técnicas DoE são celebradas por sua abordagem sistemática e estruturada à análise de sensibilidade. Eles permitem um exame minucioso dos fatores influentes no sistema, identificando variáveis-chave e seu impacto no resultado. Este estudo tem como objetivo apresentar os resultados em formato gráfico intuitivo e claro, conforme recomendado na literatura, para auxiliar na interpretação dos dados e na tomada de decisão (WEISSMAN; ANDERSON, 2015).

Neste experimento, uma arquitetura específica foi selecionada para análise de sensibilidade. Embora todas as camadas da arquitetura tenham sido examinadas, apenas os fatores que demonstram interações notáveis são apresentados. As interações foram avaliadas com base na métrica MRT devido à sua correlação direta com a percepção do usuário. A análise de sensibilidade priorizou fatores que influenciam significativamente o MRT, visando compreender como essas interações impactam o desempenho geral do sistema. Esse foco permitiu a identificação de fatores críticos para melhorar a experiência do usuário. A Tabela 4 descreve o design do experimento, listando os fatores selecionados e suas configurações de nível. Os fatores incluídos nesta análise são: (a) **C_G**, (b) **C_TTN**, (c) **C_SA**, (d) **C_SM** , e (e) **N_SM**, cada um com duas configurações: alta e baixa. A Tabela 5 exibe as diversas combinações desses fatores e suas configurações correspondentes, fornecendo uma estrutura abrangente para a análise experimental.

Tabela 4 – Design de Experimentos

Nome do Fatores	Configuração Baixa	Configuração Alta
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

4.7.1 Resultados: Análise dos Fatores que Afetam o Tempo Médio de Resposta.

Neste estudo, conduzimos uma análise abrangente para verificar os fatores que influenciam a métrica do MRT. A utilização de um gráfico de efeitos como ferramenta analítica proporcionou uma visualização lúcida e estruturada dos fatores, auxiliando na avaliação de sua significância e impacto. O gráfico de efeitos, representado na Figura 23, organiza metodicamente os fatores por sua influência, descendo do mais impactante para

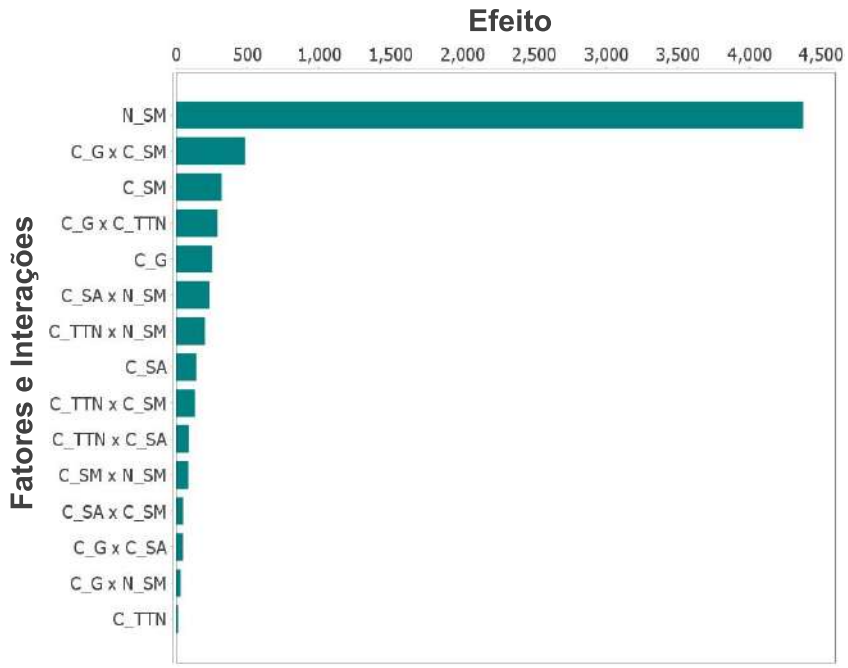
Tabela 5 – Combinação de fatores considerando a métrica MRT(ms).

C_G	**C_TTN**	**C_SA**	**C_SM**	**N_SM**	**MRT(ms)**
20,00	50,00	50,00	600,00	2,00	11789,14
20,00	50,00	50,00	600,00	4,00	8247,83
20,00	50,00	50,00	1200,00	2,00	12503,13
20,00	50,00	50,00	1200,00	4,00	6409,96
20,00	50,00	100,00	600,00	2,00	12054,15
20,00	50,00	100,00	600,00	4,00	9286,06
20,00	50,00	100,00	1200,00	2,00	11401,64
20,00	50,00	100,00	1200,00	4,00	6475,20
20,00	100,00	50,00	600,00	2,00	12560,43
20,00	100,00	50,00	600,00	4,00	7951,83
20,00	100,00	50,00	1200,00	2,00	10872,14
20,00	100,00	50,00	1200,00	4,00	6984,48
20,00	100,00	100,00	600,00	2,00	11816,65
20,00	100,00	100,00	600,00	4,00	6451,89
20,00	100,00	100,00	1200,00	2,00	11308,01
20,00	100,00	100,00	1200,00	4,00	7781,86
40,00	50,00	50,00	600,00	2,00	10984,09
40,00	50,00	50,00	600,00	4,00	7369,21
40,00	50,00	50,00	1200,00	2,00	11467,05
40,00	50,00	50,00	1200,00	4,00	8150,59
40,00	50,00	100,00	600,00	2,00	11622,22
40,00	50,00	100,00	600,00	4,00	5383,04
40,00	50,00	100,00	1200,00	2,00	12468,02
40,00	50,00	100,00	1200,00	4,00	6362,54
40,00	100,00	50,00	600,00	2,00	11385,80
40,00	100,00	50,00	600,00	4,00	7681,47
40,00	100,00	50,00	1200,00	2,00	11477,62
40,00	100,00	50,00	1200,00	4,00	7144,52
40,00	100,00	100,00	600,00	2,00	12135,27
40,00	100,00	100,00	600,00	4,00	7690,10
40,00	100,00	100,00	1200,00	2,00	10991,31
40,00	100,00	100,00	1200,00	4,00	7487,34

o menos impactante. Esta representação hierárquica facilita a identificação dos principais contribuintes para as variações observadas na métrica MRT em diferentes cenários de teste.

A partir da análise do gráfico de efeitos, concluiu-se que a capacidade do Servidor de Monitoramento afeta significativamente o desempenho do sistema, mais do que outros fatores. Essa informação ressalta o papel fundamental da capacidade do Servidor de Monitoramento no gerenciamento de demandas e fluxo de dados, influenciando diretamente a eficiência geral do sistema. Além disso, outros fatores que mostram efeitos substanciais incluem o número de núcleos no Servidor de Monitoramento e a interação entre o gateway e a capacidade do Servidor de Monitoramento. Alterações nesses fatores levaram a variações

Figura 23 – Visualização do impacto de vários fatores do modelo na métrica MRT.



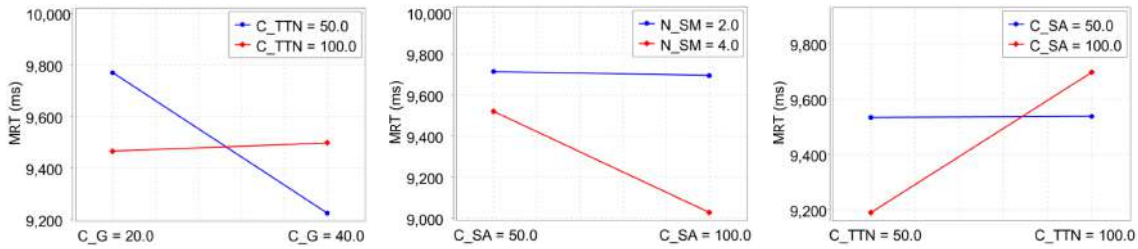
notáveis no MRT, enfatizando a criticidade de seu controle e otimização para melhorar o desempenho do sistema. Os dados sugerem que estratégias focadas para aumentar a capacidade do Servidor de Monitoramento e melhorar sua interação com o gateway podem aumentar significativamente o tempo de resposta do sistema.

Por outro lado, descobriu-se que a capacidade do servidor TTN exerce uma influência relativamente pequena no tempo de resposta do sistema. Embora ainda seja um fator importante, seu impacto nas variações do tempo de resposta foi menos pronunciado em comparação com outras variáveis analisadas. Esta observação realça a necessidade de discernir o impacto relativo de cada fator, permitindo uma tomada de decisão mais informada e eficaz. Concluindo, é crucial priorizar melhorias na capacidade do Servidor de Monitoramento e na sua interação com o gateway, com base na sua influência significativa no MRT. Esses esforços direcionados são fundamentais para otimizar o desempenho do sistema e garantir uma experiência de usuário satisfatória. Compreender e abordar estes fatores críticos contribuirá substancialmente para melhorar a eficácia global do sistema em estudo.

4.7.2 Resultados: Análise de Sensibilidade de Fatores Influentes no Desempenho do Sistema.

O estudo analisou meticulosamente os fatores que impactam a métrica MRT, utilizando gráficos de efeitos como instrumento visual para avaliar sua significância e magnitude. O gráfico de efeitos, conforme mostrado na Figura 24, organiza sistematicamente os fatores com base em seu impacto na métrica MRT, elucidando os fatores mais influentes.

Figura 24 – Interação entre fatores quanto ao seu impacto na métrica MRT.



(a) Cap. do Gateway x Cap. do Servidor TTN.

(b) Cap. do Servidor de Aplicação x Nucleos do Servidor de Monitoramento.

(c) Cap. do Servidor TTN x Cap. do Servidor de Aplicação.

A análise revelou que a capacidade do Servidor de Monitoramento foi o fator mais influente , afetando diretamente o desempenho do sistema. Essa descoberta sugere que melhorias na capacidade do Servidor de Monitoramento podem melhorar significativamente a eficiência do sistema.

A Figura 24(b) destaca a interação entre a capacidade do Servidor de Aplicação e a contagem de núcleos do Servidor de Monitoramento. Notavelmente, um aumento na capacidade do Servidor de Aplicação, aliado a um maior número de núcleos do Servidor de Monitoramento, levou a uma redução significativa no MRT, ressaltando a importância desses fatores na otimização do desempenho do sistema. Por fim, a Figura 24(c) ilustra a interação entre as capacidades do Servidor TTN e do Servidor de Aplicação. Essa interação revela um padrão diferenciado: o aumento da capacidade do servidor TTN resultou em um MRT mais alto, indicando um declínio no desempenho do sistema, enquanto as alterações na capacidade do servidor de aplicação tiveram um efeito limitado. Isto sublinha a complexidade das interações dentro do sistema e a necessidade de uma consideração cuidadosa ao implementar mudanças.

Este estudo elucida a importância de configurar sistemas de prevenção de eventos de desastres de forma otimizada em uma arquitetura LoRaWAN. Aproveitar modelos SPN permite a avaliação e o refinamento de vários aspectos do sistema, como números de nós de processamento na camada de névoa e tempos de processamento de informações. Essas informações são cruciais para aumentar a eficiência do sistema e evitar desperdício de recursos, garantindo uma resposta rápida e eficaz a desastres. O modelo SPN desenvolvido nesta pesquisa calcula métricas-chave para avaliar e otimizar o sistema de prevenção de eventos de desastres. Entre elas, o tempo médio de resposta é primordial, pois reflete a prontidão do sistema na detecção e resposta a desastres. Os resultados deste estudo oferecem orientação sobre cenários ideais e configurações de máquina para evitar desperdício de recursos em aplicações do mundo real.

5 Arquitetura LoraWan com Tolerância a Falhas: Uma Avaliação de Desempenho Com Foco na Camada de Sensores

Este capítulo apresenta o uso de modelagem estocástica para simular operações de sensores a nível individual. O capítulo aborda uma avaliação de desempenho de uma arquitetura LoRaWAN com tolerância a falhas utilizando a métrica de vazão.

5.1 Contextualização

O surgimento de novas tecnologias, como LoRa [1], Sigfox [2], Weightless [3] e Wi-fi HaLow [4], permitiu o desenvolvimento de novas aplicações que demandam comunicação sem fio eficiente de longa distância. Os nós de sensores sem fio permitem uma ampla gama de aplicações, como segurança de infraestrutura (HESSEL; ALMON; HOLLICK, 2023), monitoramento ambiental (BRITO et al., 2023), sensoriamento em cidades inteligentes (PIECHOWIAK; ZWIERZYKOWSKI; MUSZNICKI, 2023), etc. Tais aplicações visam coletar informações sobre um determinado fenômeno ou evento. Esses nós sensores são geralmente implantados em condições severas e ambientes inacessíveis. Por estas razões, os nós sensores devem operar durante longos períodos sem intervenção humana.

Diante desta alta imprevisibilidade, é necessário adotar um método avaliativo que oriente configurações mínimas por sensor. SPNs podem ser utilizadas para modelar e predizer probabilisticamente o desempenho de sensores LoRaWAN. SPNs são conhecidas pelo alto grau de representatividade, sendo mais intuitivos que opções convencionais, como cadeias de Markov, para representar concorrência, paralelismo e sincronização em sistemas (SILVA et al., 2023; FÉ et al., 2023; SOUSA et al., 2023). Alguns trabalhos utilizam modelos analíticos relacionados ao problema aqui abordado, porém focando na camada de servidores. Este trabalho apresenta um modelo SPN para modelar uma rede de sensores LoRaWAN incluindo a camada do servidor de rede. As contribuições mais específicas deste trabalho são:

- **Avaliação de sensores ao nível individual**. Com o propósito de obter informações detalhadas de desempenho, os sensores foram modelados para permitir a identificação

[1] http://www.semtech.com/wireless-rf/lora.html
[2] https://www.sigfox.com/en
[3] http://www.weightless.org/
[4] https://www.wi-fi.org/discover-wi-fi/wi-fi-certified-halow

das métricas de cada sensor, detalhando sua sensibilidade, capacidade de transmissão e possíveis peculiaridades em suas respostas a diferentes estímulos.

- **Análise de tolerância a falhas com mecanismo de retransmissão**. Busca-se compreender como a arquitetura responde a falhas no envio de dados, examinando a eficácia e eficiência do mecanismo de retransmissão. Essa análise visa identificar padrões de comportamento diante de falhas, contribuindo para o aprimoramento da arquitetura, promovendo maior robustez e confiabilidade em ambientes propensos a falhas na comunicação. O sistema pode ser comprometido tanto por falhas na comunicação dos sensores, resultando na indisponibilidade ou imprecisão dos dados coletados.

- **Foco na métrica de vazão por representar bem o impacto das características individuais dos sensores**. Busca-se explorar o impacto de cada sensor na vazão do sistema. Essa abordagem visa aprofundar a compreensão do impacto direto das particularidades dos sensores na eficiência da arquitetura, contribuindo para uma análise mais abrangente e informada.

5.2 Trabalhos Relacionados

Esta seção apresenta estudos relacionados ao tema deste trabalho, identificados na literatura. A busca abrangeu artigos que exploram arquiteturas LoRaWAN, com foco em aspectos como tempo intergeracional, taxa de falhas, taxa de retransmissão e distância entre sensores e gateway, bem como entre gateway e servidor. O estudo de (CHALL; LAHOUD; HELOU, 2019) propõe investigar o canal de rádio LoRaWAN na faixa de 868 MHz por meio de extensas campanhas de medição realizadas em ambientes internos e externos, abrangendo localidades urbanas e rurais. Usando os resultados empíricos dessas medições, eles avaliaram o desempenho e desenvolveram modelos de perda do percurso para comunicações LoRaWAN, comparando-os com modelos empíricos amplamente adotados. O trabalho de (RIBEIRO et al., 2022) propõe uma arquitetura segura e tolerante a falhas para resolver problemas de segurança e disponibilidade na tecnologia LoRaWAN. Além disso, eles conduziram análises de desempenho da rede blockchain em um ambiente de nuvem, considerando diversas cargas de trabalho, e realizaram experimentos de tolerância a falhas para entender o impacto das falhas na rede.

O trabalho de (ALI et al., 2019) propõe avaliar o desempenho da tecnologia LoRaWAN utilizando Aloha sincronizado. Para fazer isso, eles realizam simulações extensas, analisando métricas como taxa de erros de pacotes, taxa de transferência, atraso e consumo de energia em diferentes cenários. O estudo de (KHAN; JURDAK; PORTMANN, 2019) propõe um modelo analítico para calcular o atraso e a energia consumida na entrega confiável de dados no enlace de subida em redes LoRaWAN Classe A. A aplicação prática

do modelo inclui a avaliação do impacto do número de dispositivos finais e do número máximo de retransmissões de quadros de dados na entrega confirmada de dados Uplink em redes LoRaWAN. O trabalho de (INGABIRE; LARIJANI; GIBSON, 2020) busca avaliar o desempenho da rede LoRaWAN em ambientes urbanos, utilizando diversos modelos de propagação, e comparar os resultados da simulação com medições reais. O estudo de (COTRIM; KLEINSCHMIDT, 2021) propõe um modelo analítico para redes LoRaWAN multi-rotas e investiga consumo de energia, throughput, atraso e taxa de entrega de pacotes em redes single-hop relay e multi-rota.

O trabalho de (MARTINEZ et al., 2020a) investiga a possibilidade de haver tráfego confirmado em redes LoRaWAN sob interferência de canal não reportada. Os resultados foram obtidos por meio de simulações utilizando um simulador de rede de código aberto, fornecendo informações para operadoras LoRaWAN na otimização de suas redes contra interferências antes de implantações reais. O estudo de (CITONI et al., 2021) propõe uma análise abrangente das causas conhecidas de perda de pacotes em uma rede LoRaWAN somente uplink. Além disso, investigam limitações do ciclo de trabalho, colisões de pacotes, cobertura insuficiente e de demodulação do receptor. O trabalho de (MUKHIA; JAYARATHNA; LERTSINSRUBTAVEE, 2023) propõe introduzir um mecanismo de retransmissão na camada de aplicação LoRaWAN, para garantir a integridade dos dados do sensor. A extensão visa melhorar a robustez da comunicação entre os nós finais e a nuvem da rede, garantindo uma transmissão de dados contínua e confiável.

A Tabela 6 compara pontos importantes deste estudo com outros estudos da área. O primeiro critério comparativo é o método de avaliação. Os estudos pesquisados mostraram uma ligeira tendência à utilização de simulações para avaliar suas propostas. Porém, nosso estudo utiliza Modelagem SPN para prever probabilisticamente algumas situações de vazão que podem ocorrer e fornecer adaptabilidade nas escolhas de componentes para um sistema real. A modelagem SPN tende a ser bem utilizada em situações onde a proposta sistema exigiria um alto custo para testes reais. As métricas avaliadas neste contexto podem ser diversas, porém a métrica de vazão torna-se crucial para um sistema deste tipo. As métricas estudadas auxiliam na compreensão do comportamento do sistema e na parametrização no envio dos pacotes. Este estudo procura apresentar um modelo estocástico inovador para a avaliação do desempenho de arquiteturas LoRaWAN, com ênfase na análise de vazão de pacotes e tolerância a falhas. A proposta visa capturar padrões de comportamento, permitindo a mensuração do impacto da modificação de parâmetros na eficiência da arquitetura. De forma mais específica, a pesquisa abrange o intervalo entre gerações, a taxa de falhas, a taxa de retransmissão, a distância entre sensores e o gateway, bem como a distância entre o gateway e o servidor. O último fator comparado é se o estudo considera a análise de sensibilidade. Na pesquisa realizada, nenhum outro trabalho buscou utilizar este mecanismo para verificar qual fator causa mais impacto no sistema.

Tabela 6 – Trabalhos Relacionados

Trabalhos	**Métricas**	**Método de Avaliação**	**Parâmetros Configuráveis da Simulação**	**DOE**
(CHALL; LAHOUD; HELOU, 2019)	Indicador de intensidade do sinal	Modelo PL	Tempo entre gerações	✗
(RIBEIRO et al., 2022)	Latência e vazão	Experimento	Taxa de falha	✗
(ALI et al., 2019)	Taxa de erro, vazão	Simulação	Tempo entre gerações	✗
(KHAN; JURDAK; PORTMANN, 2019)	Atraso de dados e consumo de energia	Simulação	Taxa de retransmissão	✗
(INGABIRE; LARIJANI; GIBSON, 2020)	Força do sinal recebido	Modelo de propagação	Tempo entre gerações	✗
(COTRIM; KLEINSCHMIDT, 2021)	Fluxo, atraso e taxa de entrega de pacotes	Simulação	Taxa de retransmissão	✗
(MARTINEZ et al., 2020a)	Tempo de resposta	Simulação	Taxa de retransmissão	✗
(CITONI et al., 2021)	Perda de pacotes e o peso de cada condição de interrupção	Simulação	Tempo de transmissão entre nós	✗
(MUKHIA; JAYARATHNA; LERTSINSRUBTAVEE, 2023)	Força do sinal recebido, a relação sinal-ruído, a entrega de pacotes e a taxa de integridade dos dados	Experimento	Taxa de retransmissão	✗
(SAFI et al., 2022)	Tempo de resposta e o atraso geral da rede	Experimento	Tempo entre gerações	✗
Este trabalho	Vazão	Modelo SPN	Tempo entre gerações, taxa de falha, taxa de retransmissão e tempo de transmissão entre nós	✓

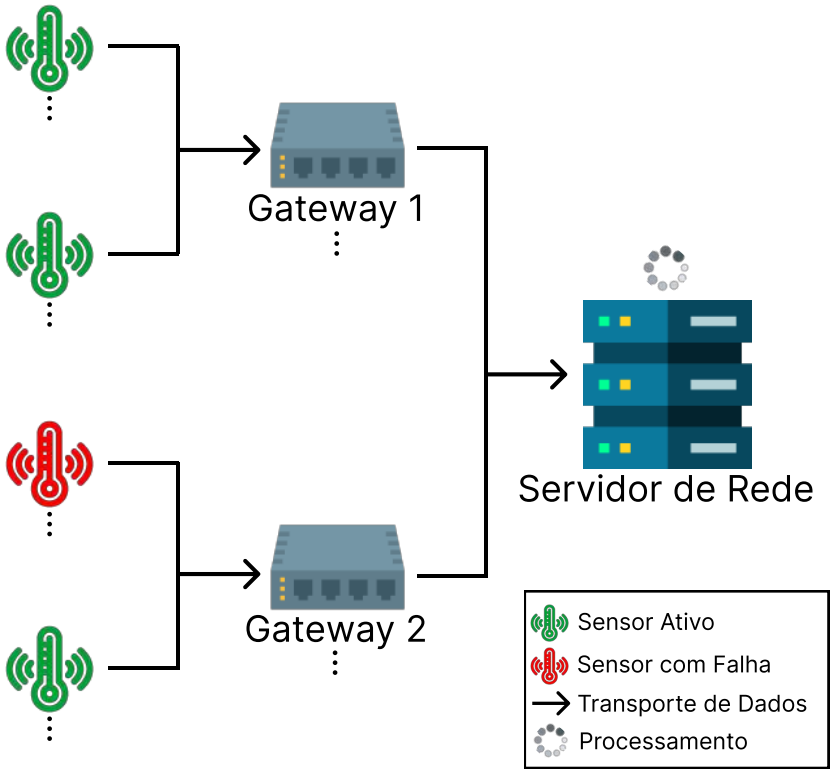
5.3 Arquitetura de Rede

A especificação LoRa inclui duas camadas: a camada física (LoRa RF) e a camada de enlace. O baixo consumo de energia e a comunicação por longas distâncias são características alcançadas pelo LoRa graças ao esquema de modulação *Chirp Spread Spectrum* (CSS), utilizado na camada física, e da utilização do LoRaWAN, baseada no ALOHA puro (ELBSIR et al., 2020). Porém, o aumento de colisões devido ao maior tempo no ar (*Time on Air– ToA*) das transmissões de pacotes tem se tornado uma desvantagem da tecnologia (ILARIZKY et al., 2021). Isso está relacionado a seleção inadequada de potências de transmissão e de fatores de espalhamento por dispositivos finais (CHEN et al., 2023). De forma simplificada, enquanto maiores potências de transmissão diminuem as oportunidades de reaproveitamento espacial, maiores fatores de espalhamento tornam o tempo no ar dos sinais transmitidos mais longos (MARTINEZ et al., 2020b). Portanto, para minimizar a probabilidade de colisões e, consequentemente, aprimorar o acesso ao meio, deve-se buscar a configuração apropriada de fatores de espalhamento e potências de transmissão (ALTAYEB et al., 2021). Após fazer tais configurações nos sensores deve-se posicioná-los de forma estratégica para obter um desempenho desejado. Uma questão muito importante é a distância que ficará cada sensor em relação ao seu gateway, como também a distância do gateway até o servidor de rede.

Esta seção apresenta a arquitetura adotada como base para criação do modelo SPN em questão, conforme ilustrado na Figura 25. A arquitetura representa a comunicação individual de cada sensor que envia seus dados pelo gateway até chegar ao servidor de rede. A rede é baseada na tecnologia LoRaWAN, a qual é uma arquitetura de rede na qual há pontos de acesso centralizados (gateways) que gerenciam a comunicação dos dispositivos LoRa com a Internet. O diferencial no presente trabalho é uma análise de tolerância a falhas com mecanismo de retransmissão. Quando ocorre uma falha no envio dos dados pelo sensor, os dados podem ser retransmitidos ao destino ou descartados, otimizando o uso de recursos e evitando desperdícios.

Os sensores utilizados neste sistema podem ser de diferentes tipos, cada um projetado para medir uma grandeza física específica do ambiente. Os nós LoRa, desempenham um papel crucial na recolha de dados através dos diferentes sensores utilizados. O gateway é responsável pela coleta de dados dos sensores, enquanto os sensores são distribuídos por todo o cenário. A implementação da topologia requer um design no qual os dispositivos finais utilizem a classe A da especificação LoRaWAN, permitindo que os dispositivos transmitam dados em intervalos regulares. Essa escolha foi feita porque os dispositivos classe A consomem menos energia. Para garantir uma recolha e armazenamento eficiente dos dados, a solução adotada consiste no armazenamento dos dados num servidor de rede, que pode fornecer as ferramentas necessárias para recolher, processar e armazenar

Figura 25 – Arquitetura LoRaWAN proposta com tolerância a falhas.



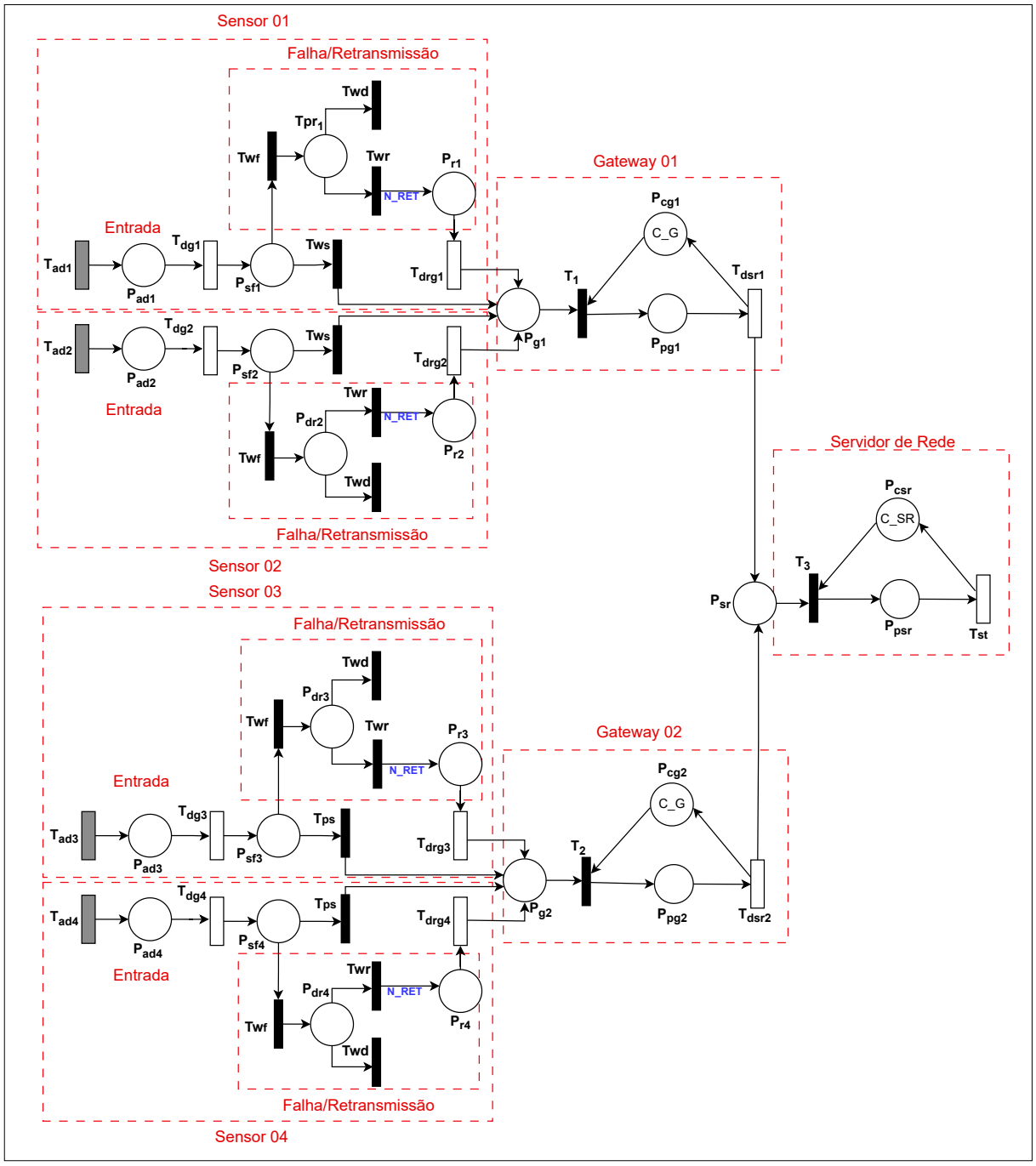
posteriormente os dados dos sensores numa base de dados.

5.4 Modelo

Esta seção apresenta o modelo SPN desenvolvido considerando as peculiaridades da arquitetura proposta. A Figura 26 apresenta o modelo base. O modelo está dividido em três partes, sendo elas: (1) Sensor, (2) Gateway, e (3) Servidor de Rede. A Tabela 7 ilustra os principais componentes do modelo.

A primeira parte do modelo é composta pelos sensores. Cada sensor contém três partes: a entrada, a transmissão, que faz parte do sensor em si, e a retransmissão. A entrada é composta por duas transições, T_{adx} e T_{dgx}, e por um lugar P_{adx}, onde x indica a numeração do sensor. T_{adx} é uma transição de distribuição de tempo determinística e representa o tempo entre a chegada de requisições em cada sensor. Quando T_{adx} é ativada, um *token* é depositado no lugar P_{adx}, que representa a chegada de requisições no sensor. T_{dgx} representa o tempo médio levado para transmitir uma requisição e, quando ativada, deposita um *token* no lugar P_{sfx}. P_{sfx} é um lugar auxiliar que indicará se houve a probabilidade de sucesso ou fracasso na transmissão. Caso a transmissão ocorra com sucesso, a transição T_{ws} é ativada e deposita o token no lugar P_{gx}. Caso a transmissão

Figura 26 – Modelo SPN desenvolvido a partir da arquitetura LoRaWAN.



falhe, a transição T_{wf} é ativada e o *token* é depositado no lugar P_{drx}, onde será feita a retransmissão.

O lugar P_{drx} representa o estado em que é feita uma tentativa de reconexão e retransmissão com o gateway. Caso a conexão falhe, a transição T_{wd} é ativada e o *token* é descartado. Caso a conexão seja reestabelecida com sucesso, a transição T_{wp} é ativada e a condição de guarda `N_RET` é acionada fazendo a multiplicidade de réplicas e o token é depositado no lugar P_{rx}. O lugar P_{rx} representa o estado onde o sensor está retransmitindo

Tabela 7 – Descrição dos principais componentes do modelo.

Tipo	**Componente**	**Definição**
Lugares	Pad1, Pad2, Pad3, Pad4	Novos pacotes no sistema
	Psf1, Psf2, Psf3, Psf4	Pacotes na fila para o gateway
	Pdr1, Pdr2, Pdr3, Pdr4	Pacotes na fila para descarte ou retransmissão
	Pr1, Pr2, Pr3, Pr4	Pacotes na fila para retransmissão
	Pg1, Pg2	Capacidade de processamento do gateway
	Pcg1, Pcg2	Fila no gateway de coleta de pacotes
	Ppg1, Ppg2	Fila de capacidade do gateway
	Psr	Fila do gateway
	Pcrs	Fila de capacidade do servidor de rede
	Ppsr	Fila do servidor de rede
Transições Determinísticas	Tad1, Tad2, Tad3, Tad4	Tempo de chegada dos pacotes
Transições Imediatas	Twf	Tempo de transmissão falhada
	Tws	Tempo de transmissão bem-sucedida
	Twd	Tempo de transmissão com descarte
	Twp	Tempo de retransmissão
	T1, T2, T3	Tempo entre chegadas de pacotes de sensores
Transições Exponenciais	Tdg1, Tdg2, Tdg3, Tdg4	Tempo de transmissão do nó para o gateway
	Tdrg1, Tdrg2, Tdrg3, Tdrg4	Tempo de retransmissão do nó para o gateway
	Tdrs1, Tdrs2	Tempo de transmissão do gateway para o servidor de rede
	Tst	Tempo de transição do pacote no servidor de rede
Lugares de Marcação	AD	Chegada de pacotes
	C_G	Capacidade do gateway
	C_SR	Capacidade do servidor de rede

a requisição. A transição T_{drgx} representa o tempo de retransmissão da requisição e, quando é ativada, o *token* é depositado no lugar P_{gx}. A segunda parte do modelo é composta pelos gateways, que recebem requisições de n sensores. Na figura apresentada, cada gateway recebe requisições de dois sensores. O primeiro lugar do componente é o P_{gx}, que representa a fila de entrada do gateway x. Neste modelo, é assumido que o gateway tenha grande

capacidade de enfileiramento, logo não foi considerada uma fila de tamanho limitado para o armazenamento de requisições.

Quando uma requisição chega no gateway e há capacidade de processamento disponível no lugar P_{cgx}, a transição T_x é ativada e um *token* é depositado no lugar P_{pgx}. O lugar P_{pgx} representa o estado onde a requisição está sendo encaminhada para o servidor. A transição T_{dsrx} representa o tempo que o gateway leva para transmitir uma requisição para o gateway e, quando ativada, ela deposita o *token* no lugar P_{sr}. A terceira parte do modelo é formada pelo componente servidor de rede. O lugar P_{sr}, que representa a fila de entrada do servidor. Assim como o gateway, a capacidade de enfileiramento de requisições do servidor não foi limitada. Quando há capacidade de processamento no servidor, representado pelo lugar P_{csr}, a requisição é encaminhada para processamento pela ativação da transição T_3. O lugar P_{psr} representa o estado no qual a requisição está sendo processada pelo servidor. Por fim, a transição T_{st} representa o tempo necessário para o servidor processar uma requisição.

As métricas utilizadas neste estudo foram o TP, o qual é uma medida quantitativa crucial para avaliar a capacidade de um sistema no processamento de solicitações e na transferência de dados. No âmbito do modelo que está sendo avaliado, o TP é baseado na lei de *Little* (JAIN, 1990), que é calculado considerando a quantidade de requisições em processamento dividido pelo tempo de processamento de cada uma das requisições. A Equação (5.1) apresenta a sintaxe da métrica utilizada no Mercury (SILVA et al., 2015), sendo medida pela esperança de *tokens* no lugar de processamento do servidor de rede, dividido pelo tempo que a requisição leva para ser processada. A relevância da vazão neste cenário é destacada, uma vez que ela é indicativa da habilidade da arquitetura se expandir para satisfazer a crescente demanda de tráfego. A vazão se torna relevante para esse contexto por permitir identificar o desempenho do sistema em relação à mudança dos diversos fatores a que está sujeito o sistema, como variação do erro, aumento do atraso de entrega de mensagens, aumento da quantidade de mensagens no sistema, entre outros.

$$\mathbf{TP} = \frac{\mathrm{Esp}(P_{adx})}{Tst} \tag{5.1}$$

A Equação 5.2 mostra como o MRT é obtido. A Lei de *Little*, um dos teoremas mais utilizados na teoria das filas, relaciona o número médio de tarefas em um sistema ao tempo médio gasto no sistema. De acordo com a Lei de *Little*: "Número médio de tarefas no sistema = taxa de chegada × tempo médio de resposta"(LITTLE, 1961). Portanto, o MRT indica o tempo médio necessário para uma mensagem ser processada e respondida pelo sistema. Essa métrica permite calcular o tempo que uma mensagem sai de um sensor até ser processada no servidor de rede, contribuindo para uma tomada de decisão ágil e eficaz. Mais especificamente, o cálculo do MRT é obtido pela soma dos tokens em todos os lugares que não tem capacidade, dividido pela vazão do sistema.

$$\mathbf{MRT} = \frac{\mathrm{Esp}(P_{adx}) + \mathrm{Esp}(P_{sfx}) + \mathrm{Esp}(P_{drx}) + \mathrm{Esp}(P_{rx}) + \mathrm{Esp}(P_{gx})}{TP} \tag{5.2}$$

5.5 Design of Experiments

Nesta seção aprofundaremos na análise dos elementos que podem afetar o desempenho do sistema em estudo, utilizando o modelo SPN. As Redes de Petri Estocásticas são uma ferramenta poderosa para modelar e analisar sistemas complexos, permitindo uma compreensão abrangente do seu comportamento e desempenho. Nosso foco principal estará na métrica Flow, pois ela representa bem o impacto das características individuais dos sensores. Ao compreender como diferentes componentes interagem e influenciam o Flow, seremos capazes de identificar oportunidades de melhoria e otimização (POLITIS et al., 2017).

Para realizar esta análise, descreveremos detalhadamente a estrutura do sistema em estudo, considerando os diferentes processos, recursos e eventos que compõem o seu funcionamento. Também analisaremos as interações entre tais elementos e como elas afetam o tempo de resposta geral do sistema. Além disso, discutimos os resultados obtidos com a aplicação do modelo SPN. Com base nestas análises, poderemos obter sobre como determinadas modificações nos componentes do sistema podem impactar o seu desempenho, permitindo-nos propor soluções e melhorias específicas.

A seguir, abordaremos o método DoE, que é uma ferramenta extremamente útil para explorar diferentes cenários e obter informações precisas sobre como as variáveis influenciam o desempenho do sistema. Com a combinação do modelo SPN e do DoE, seremos capazes de compreender e basear as nossas recomendações de otimização de forma robusta e bem fundamentada.

Visão geral do projeto de experimentos

O presente estudo emprega técnicas avançadas de DoE para realizar uma análise de sensibilidade abrangente. DoE, ou design de experimentos, é um processo estruturado de planejamento, projeto e análise de experimentos que visa obter conclusões válidas e objetivas de forma eficiente e eficaz. Esta abordagem envolve a realização de uma série de testes nos quais um engenheiro pesquisador modifica as variáveis ou fatores de entrada para identificar as causas das mudanças observadas na resposta de saída (FUKUDA, 2018).

É essencial destacar que o uso de técnicas DoE proporciona uma abordagem sistemática e estruturada para análise de sensibilidade. Estas técnicas permitem uma avaliação abrangente dos fatores que influenciam o sistema em estudo, permitindo a

identificação dos fatores mais significativos e o seu impacto na resposta de saída. Com base nas categorias gráficas recomendadas pela literatura especializada, busca-se fornecer uma representação visual intuitiva dos resultados, facilitando a interpretação dos dados e o processo de tomada de decisão (WEISSMAN; ANDERSON, 2015).

O experimento foi conduzido usando a arquitetura proposta e seus componentes. A análise de sensibilidade examinou esses componentes; no entanto, apenas os fatores com interações significativas são discutidos. A interação entre os fatores foi avaliada com base em seu impacto na métrica vazão, selecionada por sua relevância direta para a percepção do usuário final. Os fatores analisados neste estudo incluem cinco componentes do sistema: (i) *C_G*, (ii) *C_SR*, (iii) *Tempo de transmissão*, (iv) *Twf* e (v) *Twp*. Esses fatores foram avaliados em dois níveis de configuração: Os fatores têm dois níveis, que são configurações baixas e altas. A tabela 8 apresenta os valores gerados a partir de um valor padrão, considerando um aumento de 50% (valor máximo) e uma redução de 50% (valor mínimo). A Tabela 9 ilustra as combinações mais relevantes desses valores e exibe os resultados das 32 combinações relacionadas às combinações utilizadas para criar o DoE. A combinação desses fatores permite a aplicação da técnica DOE, que possibilita identificar quais componentes da arquitetura são mais relevantes e como suas variações podem impactar os resultados.

Tabela 8 – Design dos Experimentos.

Nome do Fator	**Configuração Baixa**	**Configuração Alta**
C_G	50,0 (qdt)	150,0 (qdt)
C_SR	10,0 (qdt)	100,0 (qdt)
Tempo de Transmissão	1000,0 (ms)	3000,0 (ms)
Twf	0,1 (%)	0,9 (%)
Twp	0,1 (%)	0,9 (%)

5.5.1 Análise de resultados

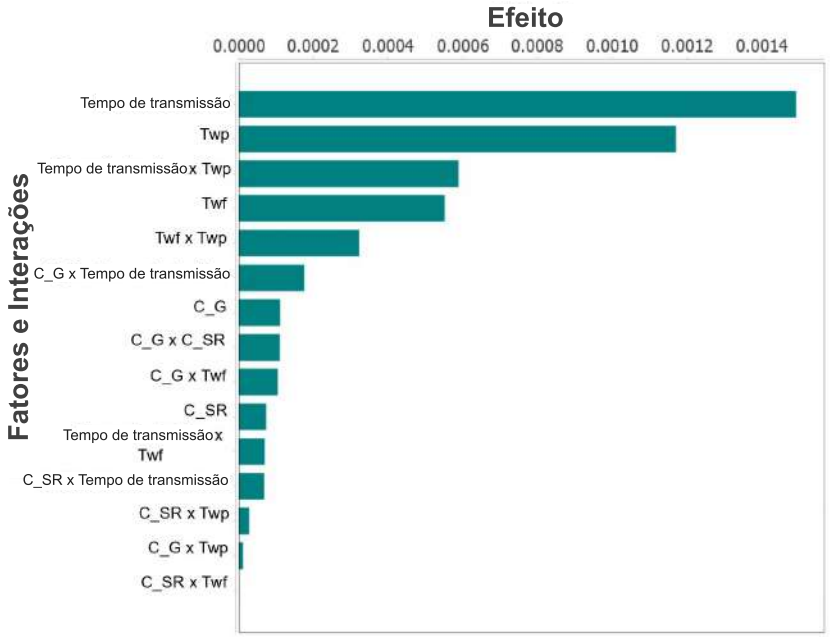
Esta subseção apresenta os resultados da análise de sensibilidade com DoE. A Figura 27 apresenta o gráfico do efeito do fator. O efeito de influência de cada fator individualmente e os efeitos das interações entre os fatores compõem o gráfico. O fator *Tempo de transmissão* teve o maior impacto no TP, seguido pelo fator *Twp*. O fator *Tempo de transmissão* teve um impacto maior que 0,0014, e *Twp* teve um impacto de aproximadamente 0,0012. A interação *Transmission time* X *Twp* teve impactos semelhantes de aproximadamente 0,006. As demais interações e fatores tiveram baixo impacto na métrica TP em comparação aos mencionados anteriormente. Os resultados são relevantes para o desenvolvimento de estratégias para melhorar o TP. Os dados apresentados permitem ao administrador do sistema identificar e priorizar os fatores mais críticos.

A Figura 29a mostra a interação entre *C_G* e *C_SR*. Há interação entre os níveis 10 e 100 de *C_R* em relação a *C_G*. Caso o avaliador tenha que optar pensando em

Tabela 9 – Combinação de fatores considerando a métrica de Vazão

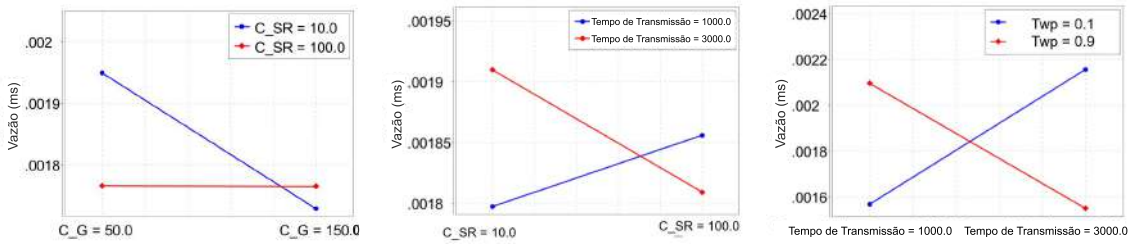
C_G(qtd)	C_SR(qtd)	Tempo de Transmissão(ms	Twf(%)	Twp(%)	TP(pct/ms)
50,00	10,00	1000,00	0,10	0,10	0,002175519
50,00	10,00	1000,00	0,10	10,00	0,003225104
50,00	10,00	1000,00	10,00	0,10	0,000694403
50,00	10,00	1000,00	10,00	10,00	0,003270508
50,00	10,00	3000,00	0,10	0,10	0,001070262
50,00	10,00	3000,00	0,10	10,00	0,001836415
50,00	10,00	3000,00	10,00	0,10	0,000773032
50,00	10,00	3000,00	10,00	10,00	0,001109548
50,00	100,00	1000,00	0,10	0,10	0,002118254
50,00	100,00	1000,00	0,10	10,00	0,002986661
50,00	100,00	1000,00	10,00	0,10	0,001462720
50,00	100,00	1000,00	10,00	10,00	0,003804096
50,00	100,00	3000,00	0,10	0,10	0,001196712
50,00	100,00	3000,00	0,10	10,00	0,001361617
50,00	100,00	3000,00	10,00	0,10	0,000603426
50,00	100,00	3000,00	10,00	10,00	0,001192422
150,00	10,00	1000,00	0,10	0,10	0,002481466
150,00	10,00	1000,00	0,10	10,00	0,002993262
150,00	10,00	1000,00	10,00	0,10	0,001253853
150,00	10,00	1000,00	10,00	10,00	0,003546888
150,00	10,00	3000,00	0,10	0,10	0,001092394
150,00	10,00	3000,00	0,10	10,00	0,001390318
150,00	10,00	3000,00	10,00	0,10	0,000551892
150,00	10,00	3000,00	10,00	10,00	0,000960975
150,00	100,00	1000,00	0,10	0,10	0,002521636
150,00	100,00	1000,00	0,10	10,00	0,003334495
150,00	100,00	1000,00	10,00	0,10	0,001274768
150,00	100,00	1000,00	10,00	10,00	0,003675449
150,00	100,00	3000,00	0,10	0,10	0,000915030
150,00	100,00	3000,00	0,10	10,00	0,001569367
150,00	100,00	3000,00	10,00	0,10	0,000471951
150,00	100,00	3000,00	10,00	10,00	0,001248960

Figura 27 – Visualização do impacto de vários fatores do modelo na métrica TP.



maximizar o TP, o ideal é a combinação de $C_SR = 10$ e $C_G = 50$. Essa combinação apresenta um TP maior. Porém, vale ressaltar que o fator C_SR tem maior impacto na métrica TP, pois apresenta maior diferença entre os níveis em comparação ao fator C_G,

Figura 28 – Interação entre fatores quanto ao seu impacto na métrica TP.



(a) Cap. do Gateway x Cap. do Servidor de Rede.

(b) Cap. do Servidor de Rede x Tempo de Transmissão.

(c) Tempo de Transmissão x Tempo de Retransmissão.

que permanece quase constante entre seus dois níveis.

A Figura 29b mostra a interação entre *DISTANCE* e *C_SR*. Há uma interação entre os fatores, e a combinação *DISTANCE* = 3000 e *C_SR* = 10 resultou em um TP maior, onde o TP é aproximadamente 0,0019. O fator que mais impactou a métrica TP foi *DISTANCE*, portanto este fator pode ser ajustado de acordo com a necessidade e objetivo do TP para o sistema. A Figura 29c mostra a interação entre *Tempo de transmissão* e *Twp*. Há uma interação significativa entre os fatores, em que a combinação dos níveis para obtenção do maior TP é com *Twf* = 10 e *Tempo de transmissão* = 1000. A combinação apresentou um TP de aproximadamente 0,0022. Porém, é importante destacar que o impacto na métrica TP de ambos os fatores é semelhante, uma vez que até a inclinação das retas é semelhante.

5.6 Análise de Impacto de Diferentes Métricas

Esta seção apresenta os resultados gerados com o modelo proposto. Devido à complexidade do modelo, as análises foram conduzidas por meio de simulações estacionárias, com uma margem de erro aproximada de 2%. A ferramenta Mercury foi empregada na criação do modelo e na geração dos resultados (MACIEL et al., 2017). A Tabela 10 detalha os valores que alimentam o modelo para todos os estudos de casos. Esses valores foram derivados da realização de experimentos práticos. As transições T_{adx} e T_{dgx} são as únicas do tipo *single server*, o restante são do tipo *infinite server*. As subseções subsequentes mostram seis estudos de caso como resultados de simulações. Todos os resultados focam na vazão de dados do sistema. Nos quatro primeiros resultados, variou-se o número de nós na rede (de 1 a 4), considerando a distância equidistante entre os nós (N), o gateway (G) e o servidor (S) (mantendo a mesma distância, N-G=G-S). A limitação de quatro nós foi estabelecida para facilitar a análise do cenário proposto. É importante destacar que o modelo foi desenhado para ser escalável, permitindo adaptações conforme necessário para avaliar cenários com um número maior de nós.

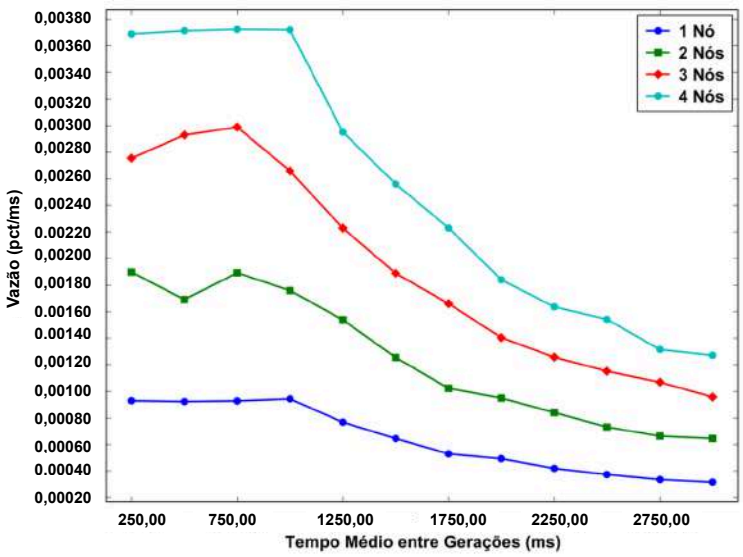
Tabela 10 – Parâmetros usados no modelo.

Componente	Valor	Descrição
Tadx	1.500	Tempo de chegada do pacote
Tdgx	1.000	Tempo de transmissão do nó para o gateway
Tdrgx	1.000	Tempo de retransmissão do nó para o gateway
Tdsrx	1.000	Tempo de transmissão do gateway para o servidor de rede
Tx	1.0	Tempo entre chegadas de pacotes de sensores
Tws	0.9	Tempo de transmissão bem-sucedida
Twf	0.1	Tempo de transmissão falhada
Twd	0.9	Tempo de transmissão com descarte
Twp	0.1	Tempo de retransmissão
N-RET	5	Número de pacotes retransmitidos
C-G	50	Capacidade do gateway
C-SR	10	Capacidade do servidor de rede
Tst	8	Tempo de serviço do servidor de rede

Estudo de caso 1 – Tempo entre gerações.

Esta seção apresenta o tempo médio entre gerações do sistema, que se refere ao intervalo de tempo entre o envio sucessivo de pacotes pelos nós de uma rede. Este conceito é equivalente ao tempo de chegada do pacote, que representa o período entre a geração de um pacote e o próximo. A manipulação do número de nós na rede (de 1 a 4) revelou informações acerca da interação sobre a taxa entre gerações e o tempo entre gerações. A Figura 29 visualiza o impacto dessa variação no tempo entre gerações, desempenhando um papel crucial na influência direta sobre a vazão do sistema.

Figura 29 – Tempo médio entre as gerações.

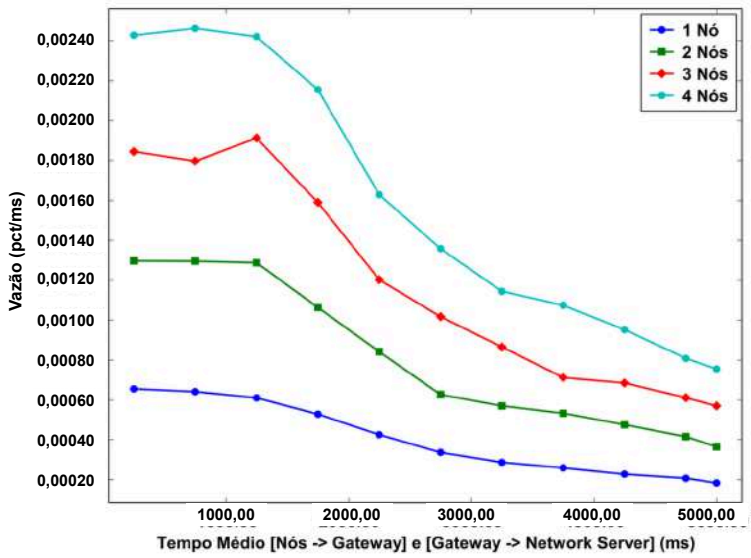


A Figura 29 mostra o impacto da variação no tempo entre gerações. À medida que

T_{adx} é incrementado, observa-se uma diminuição na vazão, indicando que um aumento no tempo entre transmissões resulta em uma redução efetiva da quantidade de dados entregue ao destino. Essa correlação se traduz na queda progressiva do desempenho das linhas de comunicação à medida que o intervalo entre a origem e o destino se torna mais extenso. A influência da quantidade de nós na rede também é evidente, sendo que um acréscimo no número de nós contribui para um aumento na vazão. Isso é causado pelo aumento do tempo de espera entre as requisições, o que leva a uma saturação da capacidade de processamento da rede. Quando o tempo de chegada do pacote T_adx excede certos limites, como 1.000 milissegundos, uma redução no vazão é observada. Isso indica que intervalos de chegada mais longos não sobrecarregam o sistema, mas reduzem a frequência dos pacotes processados, o que impacta diretamente o desempenho geral. Conforme mostrado no gráfico, a taxa de transferência diminui consistentemente à medida que o tempo entre as gerações aumenta, independentemente do número de nós. Essa mudança de cenário sugere um ponto de inflexão, onde a rede atinge sua capacidade máxima e, a partir desse ponto, o aumento do tempo entre conexões específicas começa a impactar os níveis de vazão. O aumento da taxa de transferência com a expansão do número de nós na rede é justificado pelo aumento das comunicações de múltiplas fontes.

Estudo de caso 2 - Tempo médio de transmissão do nó ao gateway e do gateway ao servidor da rede.

Figura 30 – Tempo de transmissão do nó para o gateway e do gateway para o servidor da rede.



A Figura 30 apresenta o impacto da variação da distância entre os componentes da rede e a vazão do sistema. Os resultados são relativos às configurações variadas dos parâmetros T_{dgx}, T_{drgx} e T_{dsrx}, a fim de considerar o impacto da distância no tempo de transmissão.

Os parâmetros T_{dgx}, T_{drgx} e T_{dsrx} foram manipulados em intervalos específicos, variando empiricamente entre 1.000 ms, 2.000 ms, 3.000 ms, 4.000 ms e 5.000 ms. Observou-se uma tendência na diminuição da vazão à medida que o tempo de transmissão dos nós aumentava. Destaca-se uma redução expressiva, especialmente ao empregar 4 nós, onde a queda é significativa. A partir de 1.200 ms, essa diminuição torna-se mais acentuada, evidenciando um ponto crítico no qual o aumento de T_{dgx}, T_{drgx} e T_{dsrx} impacta consideravelmente na eficiência da transmissão. Mais nós na rede aumenta a vazão, indicando uma relação positiva entre a quantidade de nós e a eficiência do sistema.

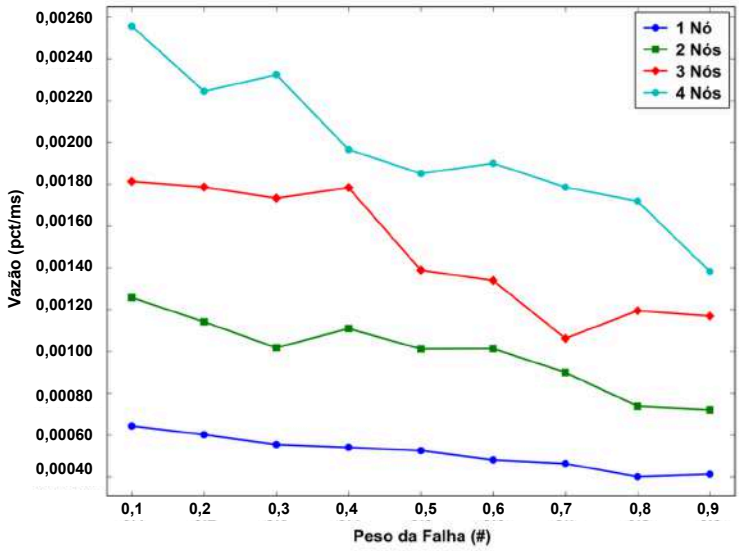
Estudo de caso 3 – Taxa de falha.

A Figura 31 mostra o impacto de T_{wf} no sistema. Neste estudo, o peso atribuído às falhas foi ajustado para analisar o efeito das probabilidades de falha no desempenho do sistema em diferentes configurações de nós. Em sistemas de nó único, o TP diminui à medida que o peso da falha aumenta, porém a redução é relativamente menor em comparação a sistemas com mais nós. Isso ocorre porque o desempenho depende exclusivamente do nó único, que continua transmitindo dados, mesmo que de forma limitada, devido à falta de redundância. Em sistemas multi-nós, como mostrado para configurações com 2, 3 e 4 nós, o impacto das falhas é mais evidente. No entanto, o uso de vários sensores na mesma área fornece maior redundância, compensando parcialmente o aumento nas probabilidades de falha. Por exemplo, mesmo com uma probabilidade de falha de 90%, sistemas com 4 nós ainda mantêm uma taxa de transferência significativamente maior do que a de sistemas com apenas 1 nó. O estudo também revelou uma tendência geral de redução do TP conforme T_{wf} aumenta, sugerindo uma relação inversamente proporcional. Em sistemas multi-nós, essa tendência é menos linear devido às interações entre nós e redundância, que introduzem variações no comportamento. Esse padrão demonstra que o aumento do peso da falha está associado a uma diminuição na eficiência da transmissão de dados, destacando a importância de considerar o número de nós ao projetar sistemas para ambientes com altas taxas de falha.

Estudo de Caso 4 – Taxa de Retransmissão.

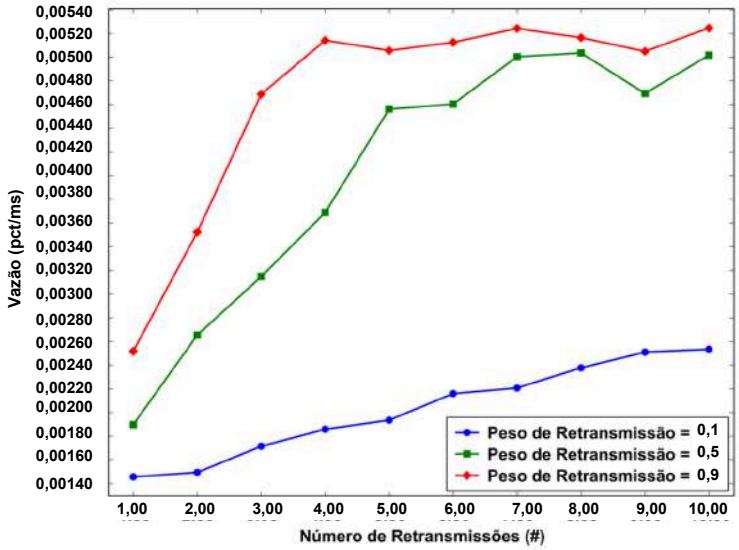
A Figura 32 mostra o impacto da variação da taxa de retransmissão sobre a vazão do sistema. O resultado refere-se a T_{wp}, sendo que T_{wf} foi ajustado para 0,5 com o propósito de induzir maior incidência de falhas, resultando em um aumento nas retransmissões. O peso de retransmissão foi variado em 0,1, 0,5 e 0,9, com a alteração do número de retransmissões variando de 1 a 10. Há correlação entre o número de retransmissões e a probabilidade de ocorrência de transmissões adicionais. À medida que o peso da retransmissão aumenta, temos uma maior vazão no sistema. Isso é esperado, pois teremos mais pacotes trafegando no sistema, o que por um lado aumenta a chance de entrega de uma informação, por

Figura 31 – Peso de variação da falha.



outro lado, também pode sobrecarregar o sistema. Quando o peso de retransmissão é baixo, especificamente 0,1, a vazão é reduzida. No entanto, ao aumentar para valores intermediários, como 0,5, e valores mais altos, como 0,9, observa-se uma tendência de aproximação das linhas de transmissão. Este fenômeno sugere que, com o aumento do número de retransmissões, as diferenças na vazão entre diferentes pesos de retransmissão se tornam menores.

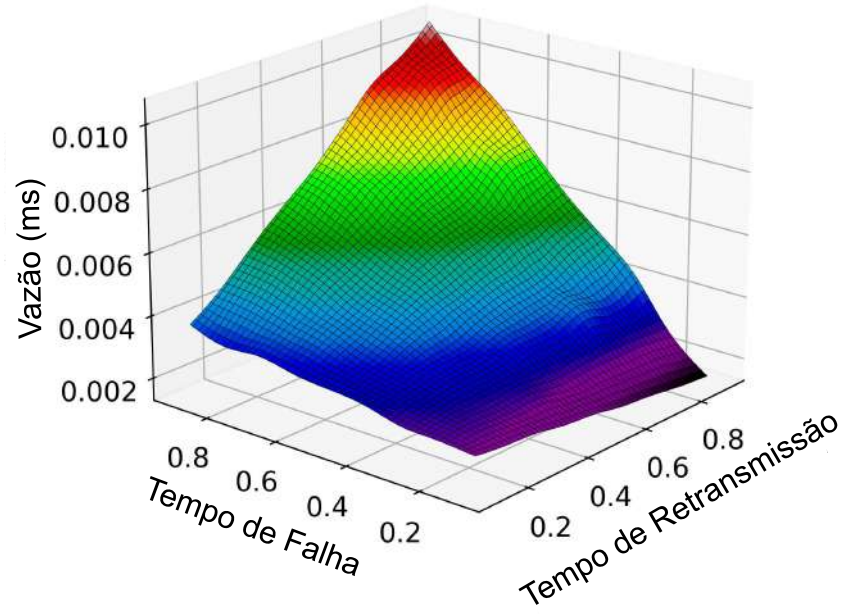
Figura 32 – Impacto da variação da taxa de retransmissão na taxa de transferência do sistema.



Estudo de Caso 5 – Tempo de Falha x Tempo de Retransmissão.

Esta seção apresenta o resultado da variação do Tempo de Falha (FT) em relação ao Tempo de Retransmissão (RT) para a vazão do sistema. A Figura 33 mostra o resultado da variação de FT em conjunto com RF para Vazão do sistema. Para esta análise, um intervalo FT foi definido entre 0.1 e 0.9. O RT foi definido para a análise variando entre 0.1 e 0.9. Esses valores podem ser encontrados na Tabela 10 nas transições Twf e Twp.

Figura 33 – Variação simultânea do tempo de falha e do tempo de retransmissão.



O gráfico exibe o aumento na Vazão à medida que o FT e o RT aumenta. Quando existe a taxa de falha máxima (nesta análise FT =0.9 e RT=0.9) a Vazão é superior a 0.010 ms. Quando as taxas começam a ser reduzidas, a Vazão tende a diminuir. Isso é devido à diminuição da quantidade de falhas do sistema, com isso, impactando seu desempenho. Na melhor das hipóteses deste cenário, onde FT=0.1 e RT=0.9, a Vazão tem seu pico mínimo de 0.002 ms. Portanto, observou-se que conforme a taxa de falha aumenta, então mais pacotes serão processadas e por consequência deste aumento a Vazão do sistema tende a ser maior.

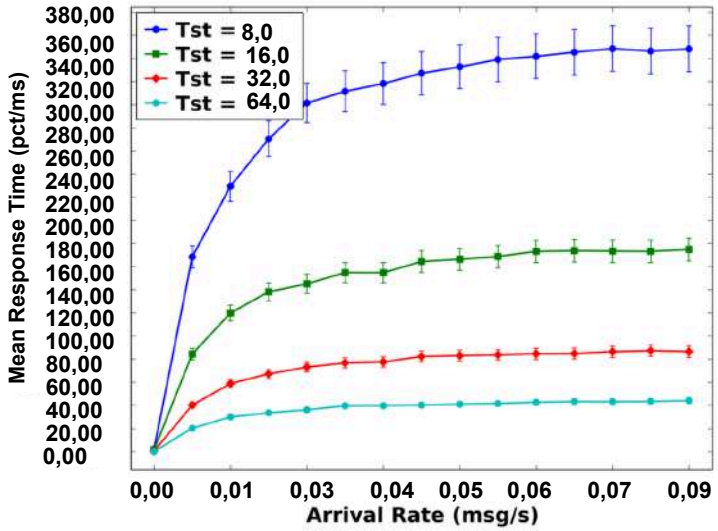
Nesta análise, o gráfico expõe algumas informações bem úteis para o avaliador em um ambiente que tem variação permanente, pois permite perceber comportamentos do sistema em dada circunstância. Por exemplo, se aumentar o número de retransmissões irá diminuir na taxa de falhas ocasionando em um aumentada na vazão do sistema. Nos testes realizados para este gráfico, o crescimento da Vazão não é excessivo, mas indica que a tolerância a falhas garante o desempenho do sistema para evitar o desperdício de recursos.

Os resultados também corroboram descobertas anteriores do DoE, onde o FT e o RT são fatores que impactam na capacidade do sistema. Se esta capacidade for muito baixa, pode acabar prejudicando o desempenho do sistema por não passar pacotes suficientes. Por outro lado, se a capacidade for elevada em um momento de baixo tráfego de pacotes, recursos serão desperdiçados.

Estudo de caso 6 – Tempo Médio de Resposta.

A Figura 34 ilustra a relação entre o tempo médio de resposta (MRT) e a taxa de chegada de mensagens para diferentes tempos de serviço no servidor de rede T_{st}, destacando o impacto da carga do sistema no desempenho da comunicação. Observa-se que, para T_{st} menores (8s e 16s), o MRT cresce de forma acentuada à medida que a taxa de chegada aumenta, sugerindo um acúmulo de requisições e maior incidência de retransmissões. A eficiência do acesso ao meio pode ser comprometida por configurações inadequadas. Em contrapartida, T_{st} mais elevados (32s e 64s) resultam em um crescimento mais moderado do MRT, sugerindo um regime de operação mais estável, ainda que potencialmente saturado.

Figura 34 – Impacto do tempo médio de resposta no sistema.



No contexto da arquitetura proposta, o gráfico reforça a necessidade de otimização dinâmica dos parâmetros de transmissão para reduzir o impacto das colisões e melhorar a eficiência do sistema. A análise dos diferentes tempos de serviço indica que a estratégia de retransmissão possibilita uma redução nos tempos de resposta sem comprometer a eficiência e desempenho. A adaptação de fatores podem diminuir os efeitos da latência e da carga excessiva, garantindo maior confiabilidade na comunicação da rede.

6 Validação

Este capítulo apresenta a validação do modelo SPN proposto, onde os dados de desempenho do modelo são comparados com os dados de um experimento de comunicação entre os componentes em uma rede.

6.1 Processo de Validação

Um modelo SPN representa a simplificação de um conceito ou problema do mundo real. Esses modelos permitem simular diversas situações relacionadas ao problema em análise, bem como estimar o comportamento esperado através de métricas. Os dados obtidos são usados como base para a tomada de decisões, seja quanto ao desenvolvimento de uma nova solução, seja para a adaptação da solução vigente. Tempo de resposta e custo de implementação da solução são fatores que pesam na decisão. Algumas soluções podem usar serviços em nuvem, cujos valores podem ser elevados, mas permitem ajustes para diminuir os custos. Todavia, outras soluções requerem aquisição e implantação de estrutura física.

Em projetos de grande escala o custo de implementação pode ser proibitivo. Mesmo com dados adquiridos a partir das simulações, surge a necessidade de validar as informações obtidas, para mitigar os riscos e custos envolvidos. Implementar uma solução em pequena escala envolve menores custos, ao passo que permite comparar com as métricas obtidas e validar o modelo. Construir um modelo simplificado também é outra possibilidade, pois permite validar com poucos recursos de software e hardware. Assim, a validação consiste em criar um experimento para determinar se um modelo SPN representa minimamente o comportamento esperado do sistema real ou simulado.

A Figura 35 apresenta o fluxograma que descreve a estratégia usada nesta validação para assegurar que o experimento seja validado corretamente:

- Montagem do experimento conforme o modelo original: O experimento é configurado de acordo com a arquitetura do modelo, considerando parâmetros como capacidade de processamento e tempos de serviço. Isso inclui a configuração de computadores conectados à mesma rede. Caso a complexidade do modelo torne a execução inviável, será necessária sua simplificação.

- Simplificação do modelo: Verificando-se a inviabilidade de gerar o experimento a partir do modelo original, deriva-se um novo modelo para encapsular algumas características, mas mantém a essência da arquitetura. Desta forma, reduz-se a

quantidade de componentes replicáveis apresentados no cenário e no modelo original. Devido à restrição de recursos computacionais, a simplificação do modelo permitirá que o experimento reflita componentes, parâmetros de tempos de serviços. Do ponto de vista estatístico, validar uma parte do modelo validará também o todo.

- Sincronização dos relógios: A sincronização dos relógios das máquinas físicas é fundamental para garantir a precisão na medição dos tempos de resposta. Diferenças nos relógios podem introduzir discrepâncias nos cálculos do MRT, comprometendo a comparação entre o experimento e o modelo. Garantir essa sincronia minimiza erros e melhora a confiabilidade dos resultados..

- Execução do experimento com baixa carga (0 drop): Esta etapa avalia se a comunicação entre os componentes ocorre conforme esperado. As requisições são enviadas com baixa taxa de chegada, pois minimiza a perda de dados ou eventos (0 drop).

- Coleta dos tempos intermediários do experimento: A execução do experimento coleta os Tempos Intermediários Externos (TIEs) entre os componentes. Um tempo intermediário é a diferença entre o momento que a requisição sai de um componente e chega em outro componente. Os Tempos Intermediários Internos (TIIs) ao componente correspondem aos processos internos prévios ao efetivo processamento.

- Alimentação do modelo SPN com os tempos intermediários do experimento: Os TIEs e TIIs obtidos alimentarão o modelo, seja o original, seja o simplificado. Os TIIs fazem parte do tempo de resposta do sistema, mesmo que sejam valores baixos para serem descartados no contexto do modelo. Valores elevados de TIIs podem indicar gargalo no modelo ou no experimento. Nesse caso é importante reavaliar a etapa de simplificação do modelo.

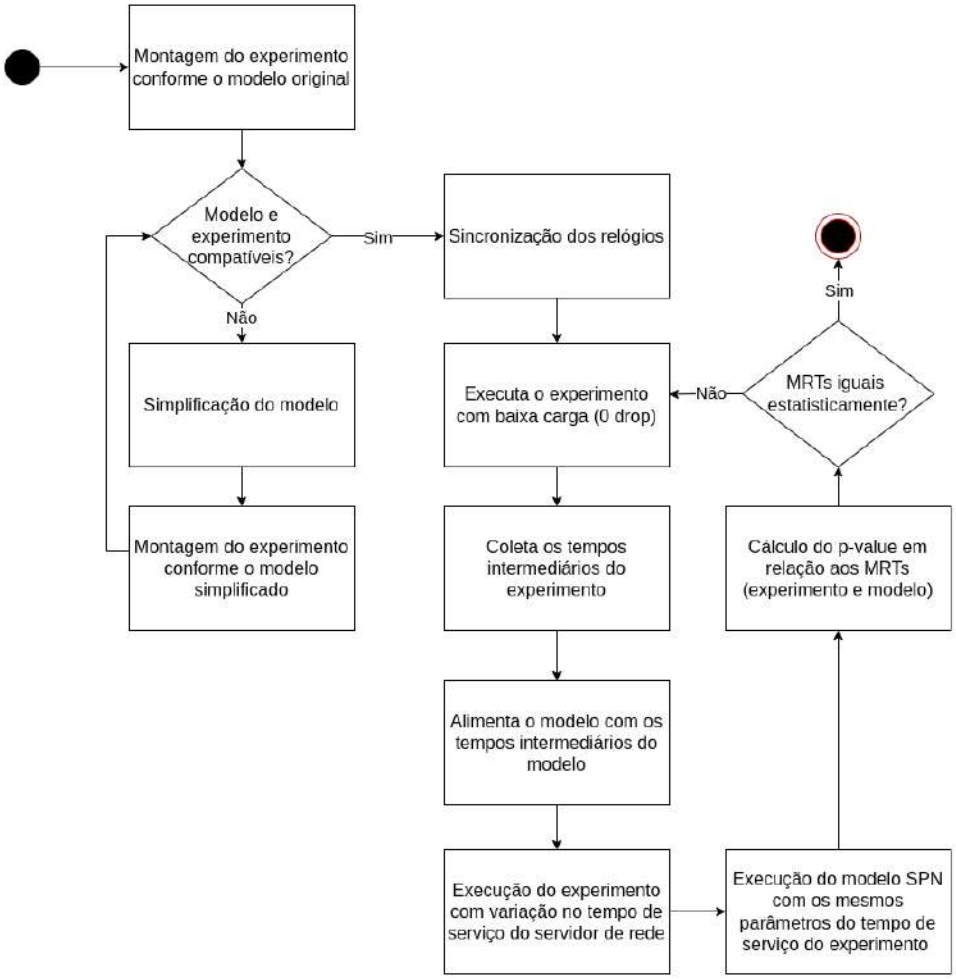
- Execução do experimento com variação no tempo de serviço: O componente é configurado com até 3 componentes, conforme o modelo SPN. No experimento, o componente alvo terá variação de tempos de serviços. Nesta etapa o experimento é executado com alta carga, onde o MRT e o Desvio Padrão são coletados. O experimento realiza 3 ciclos de envio, de forma que em cada ciclo do tempo de serviço é incrementado no componente alvo.

- Execução do modelo SPN com os mesmos parâmetros do tempo de serviço do experimento: O modelo simplificado é configurado para ser executado sob as mesmas condições de carga (taxa de chegada). Nesta etapa também são coletados o MRT e Desvio Padrão correspondente para cada variação no tempo de serviço.

- Cálculo do p-value em relação aos MRTs (experimento e modelo): A partir dos dados obtidos da simulação e do experimento, calcula-se o p-value entre os MRT em cada cenário, afim de que essa confiança estatística seja maior/igual a 0,05.

- Verificação estatística dos MRTs: Verificando-se que os MRTs são estatisticamente iguais, o processo é finalizado com sucesso. Caso contrário, retorna-se à etapa de coleta dos tempos intermediários do experimento para realizar novos ajustes no modelo.

Figura 35 – Fluxo de Validação.

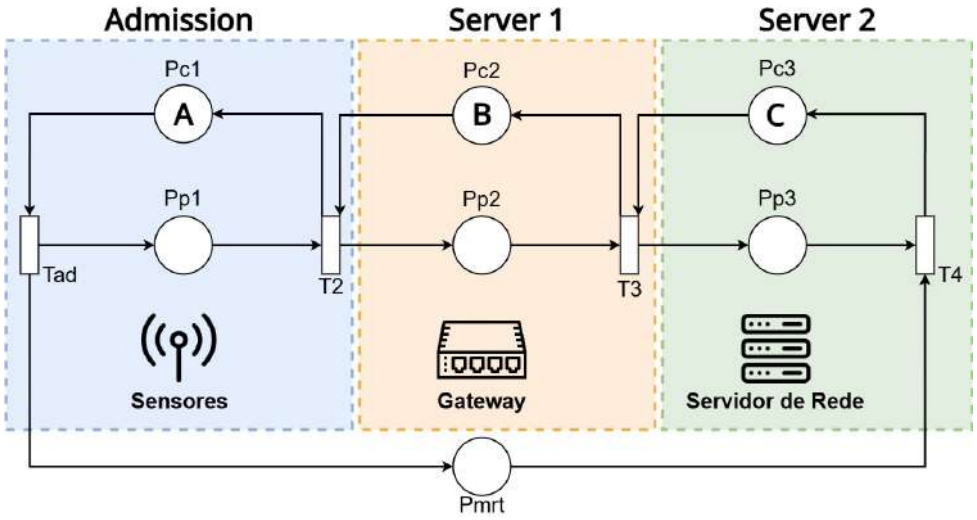


6.2 Modelo Simplificado

A Figura 36 apresenta o modelo SPN simplificado, derivado do modelo base da Figura 26. Diferente do modelo inicial, o modelo simplificado utiliza três camadas bem definidas: Admissão (Sensor), Servidor 1 (Gateway) e Servidor 2 (Servidor de Rede). Essa abordagem visa representar um sistema distribuído em uma rede local, onde o processamento ocorre de forma hierárquica entre diferentes dispositivos. Esse modelo reflete um cenário onde múltiplos computadores compartilham a carga de processamento dentro da mesma rede.

Na camada de admissão, os sensores geram pacotes que são transmitidos para o gateway. Diferente do modelo base, onde os sensores eram modelados individualmente, a unificação das entradas de dados foi aplicada para simplificar o processo de avaliação de desempenho. Não foi adotada uma transição determinística para a admissão dos pacotes, permitindo que o modelo mantenha a estocasticidade e represente variações reais nas taxas de chegada dos pacotes. No Servidor 1, representado pelo gateway, os pacotes recebidos são encaminhados para o Servidor 2, que representa o servidor de rede responsável pelo processamento final das mensagens. O fluxo do modelo segue uma sequência bem definida, permitindo a avaliação de desempenho do encaminhamento de pacotes na arquitetura proposta.

Figura 36 – Modelo SPN da Validação.



Conforme o fluxograma da Figura 35, o experimento foi executado visando encontrar variações significativas do MRT. A calibragem do experimento envolve ajustes em diversos parâmetros como a quantidade de elementos de processamento e tempos de processamento em cada componente. Ressalta-se que a montagem do experimento envolve previamente a estruturação de equipamentos físicos, máquinas virtuais, rede de comunicação e quaisquer outros recursos necessários. Após essa calibragem, os tempos coletados alimentaram o modelo simplificado. As simulações do modelo coletam os valores de MRT, os quais, junto com os dados experimentais, são submetidos a uma verificação estatística. Esse fluxo se repetiu até que os valores apresentados na Tabela 11 fossem definidos, garantindo a validação da abordagem proposta.

6.3 Ambiente do Experimento

O experimento foi desenvolvido utilizando uma infraestrutura física composta por três computadores interconectados em rede, eliminando camadas de virtualização que poderiam introduzir overhead e impactar a precisão das medições. Todos os dispositivos

Tabela 11 – Principais transições temporizadas utilizadas no modelo SPN da Figura 36

Elemento	Descrição	Valor	Tipo Servidor
Tad	Intervalo de tempo entre a chegada de requisições	100 ms	Single Server
T2	Tempo associado às transmissão da admissão até o Server 1	80 ms	Single Server
T3	Tempo associado ao processamento de uma requisição no Server1	90 ms	Single Server
T4	Tempo associado ao processamento de uma requisição no Server2	500 ms	Single Server

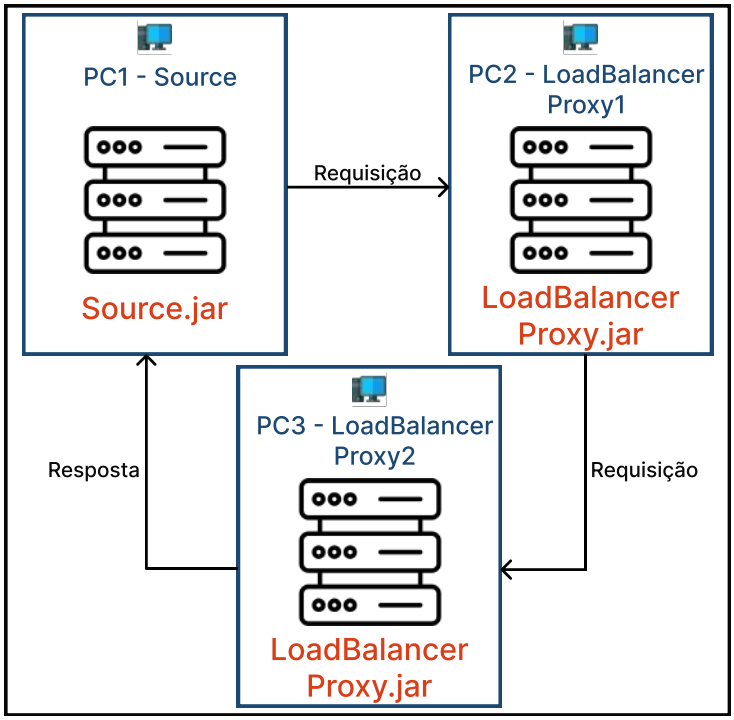
executam o sistema operacional Ubuntu, onde as demais características estão listadas na Tabela 12, garantindo um ambiente homogêneo e compatível com ferramentas de medição de desempenho de rede. O objetivo principal dessa configuração foi proporcionar um ambiente de testes controlado, onde o tráfego de dados e a distribuição de carga pudessem ser analisados de forma detalhada sem interferências externas, como variações de latência introduzidas por soluções de nuvem pública ou privadas. Essa abordagem também permitiu um controle rigoroso sobre os recursos computacionais utilizados, evitando a influência de políticas de provisionamento dinâmico que poderiam impactar a estabilidade do experimento. O uso de computadores físicos assegura que a comunicação entre os nós ocorra diretamente via rede local (LAN), reduzindo variáveis não controláveis.

A arquitetura da solução, ilustrada na Figura 37, consiste em três nós distintos: PC1, PC2 e PC3. O primeiro nó (PC1) hospeda a aplicação Source.jar, responsável por iniciar o fluxo de requisições e registrar métricas associadas ao tempo de resposta total do sistema. O segundo e o terceiro nós (PC2 e PC3) executam instâncias da aplicação LoadBalancerProxy.jar, desempenhando a função de balanceamento e encaminhamento de requisições de forma sequencial. O tráfego segue um fluxo unidirecional, onde PC1 encaminha as requisições para PC2, que realiza o roteamento para PC3. Após o processamento final, a resposta retorna diretamente para PC1, permitindo a mensuração precisa da métrica MRT. Essa topologia foi escolhida para avaliar o comportamento de encaminhamento de requisições em um ambiente distribuído sem dependência de arquiteturas em nuvem ou infraestrutura virtualizada, garantindo que os tempos de resposta sejam determinados exclusivamente pela capacidade de processamento dos nós físicos e pela eficiência da comunicação na rede local.

Tabela 12 – Configuração do experimento

Máquinas	Sistema Operacional	Memória	CPU
PC1	Ubuntu 22.04.5 LTS	32Gi	11th Gen Intel(R) Core(TM) i5-11400H @ 2.70GHz
PC2	Ubuntu 22.04.5 LTS	16Gi	Intel(R) Core (TM) i3-6100 CPU @3.70GHz
PC3	Ubuntu 22.04.5 LTS	16Gi	Intel(R) Core (TM) i3-6100 CPU @3.70GHz

Figura 37 – Ambiente de execução da validação.



6.4 Ferramenta de Validação

O experimento de validação utilizou o software PASID Validator, uma implementação desenvolvida especificamente para esta dissertação, em Java, na versão 11. Essa ferramenta foi projetada para certificar os resultados obtidos a partir dos modelos em SPN, permitindo a configuração de parâmetros de execução e de comunicação em rede, conforme o modelo SPN em avaliação.

O PASID Validator consiste em dois artefatos executáveis inseridos no ambiente de rede do experimento: `Source.jar` e `LoadBalancerProxy.jar`. O artefato `Source` é responsável por enviar requisições, receber as respostas e calcular as métricas ao final do experimento. O `LoadBalancerProxy` recebe as requisições, enfileira e distribui entre seus `ServiceProxy`. Cada `ServiceProxy` processa uma requisição e a encaminha para o próximo `LoadBalancer`. O último `LoadBalancer` da cadeia processa a requisição (via `ServiceProxy`) e devolve a resposta ao `Source`. Os artefatos `Source` e `LoadBalancerProxy` comunicam-se via sockets TCP/IP. Durante a configuração do experimento, são definidos os endereços e portas de cada artefato, garantindo que cada um conheça o próximo elemento da cadeia. O uso de sockets possibilita uma comunicação direta e contínua entre os componentes do experimento, assegurando confiabilidade na troca de informações.

O `Source` envia requisições em intervalos de tempo previamente definidos na configuração. As requisições são encaminhadas ao primeiro `LoadBalancerProxy`, que as distribui para um `ServiceProxy` utilizando um algoritmo de fila circular. Caso a fila de admissão atinja sua capacidade máxima, novas requisições são descartadas. O

`ServiceProxy` executa um pseudo processamento da requisição, simulando o tempo de serviço por meio do comando `Thread.sleep()`. O tempo de espera é determinado pelo tempo médio de serviço, especificado na configuração. Quando finaliza, o `ServiceProxy` encaminha a requisição para o próximo elemento da rede. O último `LoadBalancerProxy` devolve a resposta ao `Source`. Este calcula o tempo de resposta da requisição, valor que será utilizado para determinar o MRT do experimento. O timestamp de saída de uma máquina e de chegada à próxima é concatenado ao corpo da requisição. Essa estratégia permite mapear a rota da requisição e identificar possíveis gargalos.

6.5 Cenário de Validação

Neste cenário, os `LoadBalancerProxy` foram configurados conforme a Tabela 13. Ressalta-se que o termo `ServiceProxy` é uma abstração para representar um elemento de processamento na capacidade dos `LoadBalancerProxy`. De acordo com o contexto em estudo, esse elemento pode significar o núcleo de processamento. Da mesma forma, o `LoadBalancerProxy` pode representar um gateway e o servidor de rede. A estratégia de descarte da fila é *drop* ao atingir o limite estabelecido para o tamanho da fila, que é de 1 em cada *load balancer*. A estratégia de fila usada nos *load balancers* é o Primeiro a Chegar, Primeiro a ser Servido (FCFS). Em cada configuração, a duração do processamento dos serviços no último *load balancer* é variado entre 500(ms), 700(ms) e 900(ms). Em cada configuração são enviadas 500 requisições, onde o tempo de resposta de cada mensagem respondida será usado para calcular o MRT e o desvio padrão da configuração. O modelo SPN foi simulado com taxa de erro de 2%, onde o MRT e desvio padrão foram coletados para análise estatística.

Tabela 13 – Parâmetros para validação

Configuração	LoadBalancer1	LoadBalancer2
X	2 Services	3 Services
Y	2 Services	3 Services
Z	2 Services	3 Services

6.6 Resultados

O gráfico da Figura 38 apresenta a comparação entre o MRT do experimento e o MRT do modelo SPN para diferentes tempos de serviço. Observa-se que, em cada configuração analisada (500 ms, 700 ms e 900 ms), os valores médios obtidos no experimento são similares aos obtidos pelo modelo, com pequenas variações. A tendência geral indica que o MRT aumenta proporcionalmente ao tempo de serviço, o que está de acordo com a expectativa teórica.

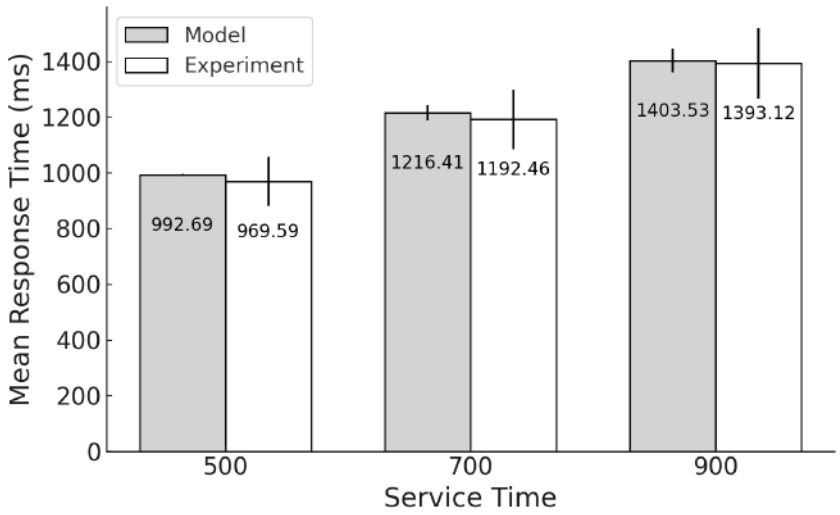
Os valores experimentais apresentam uma variação maior em relação ao modelo, evidenciada pelas barras de erro. Essa variabilidade pode estar associada a fatores experimentais, como oscilações no tempo de processamento e pequenas incertezas no ambiente de teste. No entanto, a diferença entre os valores médios do modelo e do experimento é relativamente pequena, indicando que o modelo representa bem o comportamento do sistema.

Para validar a similaridade entre os resultados do modelo e do experimento, aplicamos um T-Test de uma amostra. O T-Test é um teste estatístico usado para comparar médias de dois grupos, desde que as amostras satisfaçam condições de normalidade, igualdade de variância e independência (KIM, 2015). Os resultados estatísticos são apresentados na Tabela 14. Em cada configuração, foram analisadas múltiplas requisições, resultando no cálculo do MRT e seu desvio padrão para ambas as abordagens.

Os valores de p obtidos foram superiores a 0,05 para todas as configurações. Isso significa que não há evidências estatísticas para rejeitar a hipótese nula de igualdade entre os dois grupos amostrais, com um nível de confiança de 95%. Dessa forma, considerando o modelo como referência, a análise sugere que o experimento consegue reproduzir os valores de MRT com boa precisão, apesar da variabilidade observada.

Por fim, esta validação reforça que o modelo SPN é uma ferramenta adequada para prever e analisar o comportamento do sistema. Os resultados indicam que o modelo pode ser utilizado para planejar e otimizar a arquitetura analisada, minimizando a necessidade de experimentação em larga escala. Contudo, recomenda-se um refinamento adicional no controle da variabilidade experimental para garantir uma calibração mais precisa em futuras análises.

Figura 38 – Resultado da Validação.



Service Time	**Model Mean**	**Model SD**	**Experiment Mean**	**Experiment SD**	**t Value**	**p Value**
500	992.69	4.32	969.59	88.14	1.17	0.45
700	1216.41	26.93	1192.46	107.50	1.67	0.33
900	1403.53	42.60	1393.12	126.65	0.35	0.79

Tabela 14 – Resultados do Teste-T para diferentes tempos de serviço, incluindo médias e desvios padrão do modelo e do experimento.

7 Conclusões

Esta dissertação explorou o uso de Redes de Petri Estocásticas (SPNs) como uma abordagem formal e analítica para a modelagem, avaliação e otimização de arquiteturas LoRaWAN aplicadas à detecção de desastres. Em um cenário marcado por alta volatilidade de tráfego, escassez de recursos e a necessidade de respostas rápidas, o uso de SPNs demonstrou ser uma ferramenta eficaz para entender o comportamento dinâmico da arquitetura sob diferentes condições de operação.

A modelagem proposta permitiu não apenas avaliar métricas tradicionais, como Tempo Médio de Resposta (MRT) e vazão (TP), mas também investigar o impacto de falhas e retransmissões sobre a estabilidade do sistema. Através da simulação de diversos cenários, foi possível demonstrar como ajustes no tempo de processamento e no número de nós na camada de névoa influenciam diretamente o desempenho e a eficiência da rede — orientando o dimensionamento ideal da infraestrutura.

A pesquisa também conferiu destaque à análise de tolerância a falhas, revelando que o uso de estratégias adaptativas pode aumentar a robustez do sistema. A análise de sensibilidade, por sua vez, foi essencial para identificar os parâmetros mais críticos da arquitetura e fornecer subsídios para decisões de planejamento e manutenção preventiva.

Seis estudos de caso deram suporte prático às análises, evidenciando, por exemplo, que a redefinição de parâmetros de transmissão pode melhorar significativamente a vazão, reduzir falhas e diminuir a latência — transformando arquiteturas sensíveis em redes resilientes e responsivas. A relação entre o MRT e a taxa de chegada de mensagens também foi discutida em profundidade, reforçando como o comportamento emergente do sistema pode ser previsto e ajustado com base em princípios formais.

Além disso, a validação experimental demonstrou uma boa correlação estatística entre os dados simulados e os obtidos em ambiente controlado, mesmo com o modelo de validação ainda em desenvolvimento. Esse alinhamento reforça a aplicabilidade da abordagem proposta, não apenas como ferramenta de simulação, mas como suporte concreto ao planejamento de sistemas reais, com potencial para reduzir custos operacionais e acelerar tomadas de decisão.

Dessa forma, este trabalho contribui com um arcabouço metodológico que combina rigor formal e aplicabilidade prática, oferecendo diretrizes claras para o projeto de redes LoRaWAN mais eficientes, resilientes e preparadas para operar em contextos críticos, como o monitoramento de desastres naturais.

Como perspectivas futuras, propõe-se a evolução do modelo para abarcar estratégias

mais avançadas, como o descarte adaptativo de pacotes redundantes e a introdução de múltiplos gateways, visando maior escalabilidade e capacidade de recuperação frente a eventos de alta demanda. Espera-se que essas evoluções fortaleçam ainda mais o papel da modelagem estocástica como elemento estratégico no desenvolvimento de soluções inteligentes para cidades conectadas e sistemas ciberfísicos críticos.

7.1 Fora de Escopo

Alguns assuntos estão fora do escopo deste trabalho e, portanto, não foram abordados neste documento. A seguir serão listados alguns desses assuntos:

- Detalhes de Modulação Específicos: As redes LoRaWAN utilizam parâmetros de modulação, como o fator de espalhamento (SF), o canal de frequência (CF) e a largura de banda (BW), por exemplo. Embora esses parâmetros sejam críticos para o desempenho real das redes LoRaWAN, a modelagem opta por abstraí-los. Essa abstração visa simplificar o processo de modelagem e análise, permitindo um foco mais claro nas interações de alto nível e nos comportamentos emergentes do sistema sem se perder nas complexidades técnicas detalhadas de cada parâmetro de modulação.

- Interferência e ruído de rádio: não foram abordados na modelagem. Embora esses fatores sejam críticos no desempenho de redes LoRaWAN, este trabalho foca diretamente na vazão e na eficiência da rede, adotando uma modelagem focada na capacidade de transmissão e processamento dos servidores. A exclusão desses fatores permite uma análise mais isolada da influência da arquitetura e dos parâmetros internos da rede sobre o desempenho, sem os efeitos externos do ambiente. Através da análise de sensibilidade, pode-se investigar os fatores que tem um impacto mais relevante, a taxa de transmissão de dados e a melhor combinação de medida que afetam essas métricas. Esse enfoque simplificado evita a complexidade adicional introduzida pelas interferências e pelo ruído de rádio, permitindo uma compreensão mais clara e direta das principais influências sobre o desempenho da rede.

- Movimentação e Mobilidade dos Dispositivos: não foram levadas em conta na modelagem. Este trabalho analisa o desempenho em redes LoRaWAN utilizando Redes de Petri Estocásticas (SPN). Incorporar a movimentação e a mobilidade dos dispositivos adicionaria um nível de complexidade que desviaria o foco das métricas de desempenho específicas que estamos avaliando. No entanto, ao não considerar esses aspectos, a modelagem pode se concentrar mais diretamente nas métricas de desempenho essenciais, como Tempo Médio de Resposta (MRT) e vazão. Essa abordagem permite uma análise detalhada e precisa dos fatores estáticos que influenciam o desempenho

da rede, sem a interferência das variações dinâmicas que a movimentação e a mobilidade introduziriam. Assim, este trabalho pode fornecer informações claras sobre a eficiência e a capacidade de redes LoRaWAN em um cenário controlado.

Referências

AFANASYEV, I. et al. Towards the internet of robotic things: Analysis, architecture, components and challenges. *2019 12th International Conference on Developments in eSystems Engineering (DeSE)*, p. 3–8, 2019. 23

ALI, Z. et al. Performance evaluation of lorawan for green internet of things. *IEEE Access*, IEEE, v. 7, p. 164102–164112, 2019. 50, 52

ALTAYEB, M. et al. Turbolora: Enhancing lorawan data rate via device synchronization. *2021 IEEE 18th Annual Consumer Communications and Networking Conference (CCNC)*, p. 1–4, 2021. 53

ANTUNES, M. et al. Low-cost system for early detection and deployment of countermeasures against wild fires. *2019 IEEE 5th World Forum on Internet of Things (WF-IoT)*, p. 418–423, 2019. 21, 22

ASTERIOU, V.; PAPADIMITRIOU, G.; NICOPOLITIDIS, P. Adaptive mac protocols for iot edge computing architectures with event-triggered traffic. *2021 IEEE International Black Sea Conference on Communications and Networking (BlackSeaCom)*, p. 1–6, 2021. 21, 22

BELLAVISTA, P. et al. A survey on fog computing for the internet of things. *Pervasive and mobile computing*, Elsevier, v. 52, p. 71–99, 2019. 9

BHATIA, L. et al. Dataset: Loed: The lorawan at the edge dataset. *arXiv preprint arXiv:2010.14211*, 2020. 10

BRIEF, I. L. Internet of things. *Future Generation Computer Systems*, 2019. 7

BRITO, T. et al. Data acquisition filtering focused on optimizing transmission in a lorawan network applied to the wsn forest monitoring system. *Sensors*, MDPI, v. 23, n. 3, p. 1282, 2023. 49

CAMPOLONGO, F.; TARANTOLA, S.; SALTELLI, A. Tackling quantitatively large dimensionality problems. *Computer Physics Communication*, Institute Jbr Systems, lnformatics and Safety. Joint Research Centre of the European Commission, v. 117, n. 1, p. 75–85, 1999. 14

CARVALHO, D. et al. Edge servers placement in mobile edge computing using stochastic petri nets. *International Journal of Computational Science and Engineering*, Inderscience Publishers (IEL), v. 23, n. 4, p. 352–366, 2020. 39

CHAHROUR, N. et al. Deterioration modeling and maintenance assessment using physics-informed stochastic petri nets: Application to torrent protection structures. *Reliability Engineering & System Safety*, Elsevier, v. 210, p. 107524, 2021. 29

CHALL, R. E.; LAHOUD, S.; HELOU, M. E. Lorawan network: Radio propagation models and performance evaluation in various environments in lebanon. *IEEE Internet of Things Journal*, IEEE, v. 6, n. 2, p. 2366–2378, 2019. 50, 52

CHAUDHARI, B. S.; ZENNARO, M.; BORKAR, S. Lpwan technologies: Emerging application characteristics, requirements, and design considerations. *Future Internet*, MDPI, v. 12, n. 3, p. 46, 2020. 3

CHEN, M. et al. Probabilistic model checking for unconfirmed transmission in lorawan on the mac layer. *GLOBECOM 2023 - 2023 IEEE Global Communications Conference*, p. 2457–2462, 2023. 53

CHEN, P.-Y. et al. Contact-aware opportunistic data forwarding in disconnected lorawan mobile networks. In: IEEE. *2020 IEEE 40th International Conference on Distributed Computing Systems (ICDCS)*. [S.l.], 2020. p. 574–583. 10

CHEN, S.; ZHANG, T.; SHI, W. Fog computing. *IEEE Internet Computing*, IEEE, v. 21, n. 2, p. 4–6, 2017. 9

CITONI, B. et al. Impact of inter-gateway distance on lorawan performance. *Electronics*, MDPI, v. 10, n. 18, p. 2197, 2021. 51, 52

COSTA, I. et al. Availability evaluation and sensitivity analysis of a mobile backend-as-a-service platform. *Quality and Reliability Engineering International*, Wiley Online Library, v. 32, n. 7, p. 2191–2205, 2016. 14

COTRIM, J. R.; KLEINSCHMIDT, J. H. An analytical model for multihop lorawan networks. TechRxiv, 2021. 51, 52

DEBDAS, S. et al. Long range (lora) fire fighter in dense forest. *2022 International Conference on Computer, Power and Communications (ICCPC)*, p. 286–293, 2022. 19, 21, 22

DUISEBEKOVA, K. S. et al. Environmental monitoring system for analysis of climatic and ecological changes using lora technology. In: *Proceedings of the 5th International Conference on Engineering and MIS*. [S.l.: s.n.], 2019. p. 1–6. 10

ELBSIR, H. E. et al. Evaluation of lorawan class b efficiency for downlink traffic. *2020 16th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, p. 105–110, 2020. 53

FÉ, I. et al. Model-driven dependability and power consumption quantification of kubernetes based cloud-fog continuum. *IEEE Access*, IEEE, 2023. 49

FEITOSA, L. et al. Performance evaluation of message routing strategies in the internet of robotic things using the d/m/c/k/fcfs queuing network. *Electronics*, MDPI, v. 10, n. 21, p. 2626, 2021. 14

FERREIRA, V. et al. Redes corporais sem fio e suas aplicações em saúde. *Jornadas de Atualização em Informática*, v. 23, 2018. 7

FUKUDA, I. M. e. a. Design of experiments (doe) applied to pharmaceutical and analytical quality by design (qbd). *Science of the total environment*, Brazilian journal of pharmaceutical sciences, v. 54, 2018. 44, 58

GERMAN, R. *Performance analysis of communication systems - modelling with non-Markovian stochastic Petri nets*. [S.l.]: Wiley, 2000. (Wiley-Interscience series in systems and optimization). ISBN 978-0-471-49258-0. 11

GERMAN, R. *Performance analysis of communication systems with non-Markovian stochastic Petri nets*. [S.l.]: John Wiley & Sons, Inc., 2000. 13

GONZáLEZ-PALACIO, M. et al. Machine-learning-based combined path loss and shadowing model in lorawan for energy efficiency enhancement. *IEEE Internet of Things Journal*, v. 10, n. 12, p. 10725–10739, 2023. 3

HAJOVSKY, R. et al. Design of an iot-based monitoring system as a part of prevention of thermal events in mining and landfill waste disposal sites: A pilot case study. *IEEE Transactions on Instrumentation and Measurement*, v. 72, p. 1–14, 2023. 21, 22

HAN, X.; JIE, L.; ZUYAN, S. Non-autonomous coloured petri net-based methodology for the dispatching process of urban fire-fighting. *Fire Safety Journal*, v. 35, n. 4, p. 299–325, 2000. ISSN 0379-7112. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0379711200000394>. 21

HASEGAWA, Y.; SUZUKI, K. A multi-user ack-aggregation method for large-scale reliable lorawan service. In: IEEE. *ICC 2019-2019 IEEE International Conference on Communications (ICC)*. [S.l.], 2019. p. 1–7. 11

HESSEL, F.; ALMON, L.; HOLLICK, M. Lorawan security: An evolvable survey on vulnerabilities, attacks and their systematic mitigation. *ACM Transactions on Sensor Networks*, ACM New York, NY, v. 18, n. 4, p. 1–55, 2023. 49

HRISTOV, G. et al. Emerging methods for early detection of forest fires using unmanned aerial vehicles and lorawan sensor networks. *2018 28th EAEEIE Annual Conference (EAEEIE)*, p. 1–9, 2018. 21, 22

ILARIZKY, A. et al. Performance analysis of adaptive data rate scheme at network-server side in lorawan network. *2021 2nd International Conference on ICT for Rural Development (IC-ICTRuDev)*, p. 1–5, 2021. 53

INGABIRE, W.; LARIJANI, H.; GIBSON, R. M. Performance evaluation of propagation models for lorawan in an urban environment. *2020 International Conference on Electrical, Communication, and Computer Engineering (ICECCE)*, p. 1–6, 2020. 51, 52

IQBAL, M.; ABDULLAH, A. Y. M.; SHABNAM, F. An application based comparative study of lpwan technologies for iot environment. *2020 IEEE Region 10 Symposium (TENSYMP)*, p. 1857–1860, 2020. 10

JAIN, R. *The art of computer systems performance analysis: techniques for experimental design, measurement, simulation, and modeling*. [S.l.]: John Wiley & Sons, 1990. 57

KHAN, F. H.; JURDAK, R.; PORTMANN, M. A model for reliable uplink transmissions in lorawan. *2019 15th International Conference on Distributed Computing in Sensor Systems (DCOSS)*, p. 147–156, 2019. 50, 52

KHAN, W. Z. et al. Edge computing: A survey. *Future Generation Computer Systems*, Elsevier, v. 97, p. 219–235, 2019. 8, 9

KIM, T. K. T test as a parametric statistic. *Korean journal of anesthesiology*, The Korean Society of Anesthesiologists, v. 68, n. 6, p. 540–546, 2015. 76

LEA, P. *Internet of Things for Architects: Architecting IoT solutions by implementing sensors, communication infrastructure, edge computing, analytics, and security*. [S.l.]: Packt Publishing Ltd, 2018. 7

LEE, J.; MITICI, M. An integrated assessment of safety and efficiency of aircraft maintenance strategies using agent-based modelling and stochastic petri nets. *Reliability Engineering & System Safety*, Elsevier, v. 202, p. 107052, 2020. 25

LEE, J. J. et al. Building a private lorawan platform. BEIESP, 2019. 10

LIMA, J. W. S. d.; CALLOU, G.; ANDRADE, E. Queueing theory and stochastic petri net: A tutorial. *Research, Society and Development*, v. 10, n. 3, p. e2810312826, Mar. 2021. Disponível em: <https://rsdjournal.org/index.php/rsd/article/view/12826>. 11

LITTLE, J. D. A proof for the queuing formula: L= λ w. *Operations research*, INFORMS, v. 9, n. 3, p. 383–387, 1961. 57

LUNA, P.; GUTIéRREZ, S.; ESPINOSA, R. Design and implementation of a node geolocation system for fire monitoring through lorawan. *2020 IEEE International Autumn Meeting on Power, Electronics and Computing (ROPEC)*, v. 4, p. 1–6, 2020. 21, 22

MACIEL, P. et al. Mercury: Performance and dependability evaluation of systems with exponential, expolynomial, and general distributions. *2017 IEEE 22nd Pacific Rim international symposium on dependable computing (PRDC)*, p. 50–57, 2017. 25, 61

MACIEL, P. R. M. *Performance, reliability, and availability evaluation of computational systems, Volume 2: Reliability, availability modeling, measuring, and data analysis*. [S.l.]: CRC Press, 2023. 11

MACIEL, P. R. M. *Performance, reliability, and availability evaluation of computational systems, volume I: performance and background*. [S.l.]: CRC Press, 2023. 11

MAGRO, S. Z. D. et al. Dispositivo para retransmissão de mensagens lorawan. *Revista Brasileira de Computação Aplicada*, v. 14, n. 2, p. 47–55, 2022. 5

MARSAN, M. A. et al. *Modelling with Generalized Stochastic Petri Nets*. USA: John Wiley & Sons, Inc., 1994. ISBN 0471930598. 11

MARSAN, M. A. et al. Modelling with generalized stochastic petri nets. *ACM SIGMETRICS performance evaluation review*, ACM New York, NY, USA, v. 26, n. 2, p. 2, 1998. 14

MARTINEZ, I. et al. On the performance evaluation of lorawan with re-transmissions under jamming. *2020 IEEE Symposium on Computers and Communications (ISCC)*, p. 1–7, 2020. 51, 52

MARTINEZ, I. et al. On the performance evaluation of lorawan with re-transmissions under jamming. *2020 IEEE Symposium on Computers and Communications (ISCC)*, p. 1–7, 2020. 53

MASEK, P. et al. Performance analysis of different lorawan frequency bands for mmtc scenarios. *2022 45th International Conference on Telecommunications and Signal Processing (TSP)*, p. 417–420, 2022. 4

MAUDET, S. et al. Refined node energy consumption modeling in a lorawan network. *Sensors*, v. 21, n. 19, 2021. ISSN 1424-8220. Disponível em: <https://www.mdpi.com/1424-8220/21/19/6398>. 4

MAYER, P. et al. Lora vs. lora: In-field evaluation and comparison for long-lifetime sensor nodes. *2019 IEEE 8th International Workshop on Advances in Sensors and Interfaces (IWASI)*, p. 307–311, 2019. 10

MOLLOY, M. K. Performance analysis using stochastic petri nets. *IEEE Trans. Comput.*, IEEE Computer Society, Washington, DC, USA, v. 31, p. 913–917, September 1982. ISSN 0018-9340. 11

MOUHA, R. A. R. A. et al. Internet of things (iot). *Journal of Data Analysis and Information Processing*, Scientific Research Publishing, v. 9, n. 02, p. 77, 2021. 3

MUKHIA, R.; JAYARATHNA, K. G. S.; LERTSINSRUBTAVEE, A. Performance evaluation of lorawan forest fire monitoring network in the wild. *Proceedings of the 18th Asian Internet Engineering Conference*, p. 96–104, 2023. 51, 52

MURATA, T. Petri nets: Properties, analysis and applications. *Proceedings of the IEEE*, IEEE, v. 77, n. 4, p. 541–580, 1989. 11

NING, H. et al. Performance analysis for emergency response capability based on stochastic petri net. *2010 IEEE International Conference on Emergency Management and Management Sciences*, p. 61–64, 2010. 22

NOURA, H. et al. Lorawan security survey: Issues, threats and possible mitigation techniques. *Internet of Things*, Elsevier, v. 12, p. 100303, 2020. 10

OROJLOO, H.; AZGOMI, M. A. Modelling and evaluation of the security of cyber-physical systems using stochastic petri nets. *IET Cyber-Physical Systems: Theory & Applications*, Wiley Online Library, v. 4, n. 1, p. 50–57, 2019. 39

PASTÓRIO, A.; RODRIGUES, L.; CAMARGO, E. de. Uma revisao sistemática da literatura sobre tolerância a falhas em internet das coisas. *Anais Estendidos do X Simpósio Brasileiro de Engenharia de Sistemas Computacionais*, SBC, p. 57–64, 2020. 3

PASTÓRIO, A. F.; CAMARGO, E. T. de. Técnicas de geolocalização em redes lorawan como abordagem de tolerância a falhas para dispositivos iot baseados em gps. *Anais do XXII Workshop de Testes e Tolerância a Falhas*, p. 29–42, 2021. 5

PAUL, A.; JEYARAJ, R. Internet of things: A primer. *Human Behavior and Emerging Technologies*, Wiley Online Library, v. 1, n. 1, p. 37–47, 2019. 7

PETÄJÄJÄRVI, J. et al. Evaluation of lora lpwan technology for indoor remote health and wellbeing monitoring. *International Journal of Wireless Information Networks*, Springer, v. 24, p. 153–165, 2017. 3

PETERSON, J. L. Petri nets. *ACM Computing Surveys (CSUR)*, ACM New York, NY, USA, v. 9, n. 3, p. 223–252, 1977. 11

PIECHOWIAK, M.; ZWIERZYKOWSKI, P.; MUSZNICKI, B. Lorawan metering infrastructure planning in smart cities. *Applied Sciences*, MDPI, v. 13, n. 14, p. 8431, 2023. 49

POLITIS, S. N. et al. Design of experiments (doe) in pharmaceutical development. *Drug development and industrial pharmacy*, Taylor & Francis, v. 43, n. 6, p. 889–901, 2017. 43, 58

RAMSON, S. J. et al. A self-powered, real-time, lorawan iot-based soil health monitoring system. *IEEE Internet of Things Journal*, IEEE, v. 8, n. 11, p. 9278–9293, 2021. 10

RIBEIRO, V. et al. A fault-tolerant and secure architecture for key management in lorawan based on permissioned blockchain. *IEEE Access*, IEEE, v. 10, p. 58722–58735, 2022. 50, 52

ROMEO, L. et al. Internet of robotic things in smart domains: Applications and challenges. *Sensors*, MDPI, v. 20, n. 12, p. 3355, 2020. 23

SABZALI A., e. a. A survey on lorawan technology: Architecture, applications, and open issues. IEEE Communications Surveys & Tutorials, v. 20, n. 4, p. 3417–3452, 2017. 4

SADIKU, M. N.; MUSA, S. M.; MOMOH, O. D. Cloud computing: opportunities and challenges. *IEEE potentials*, IEEE, v. 33, n. 1, p. 34–36, 2014. 8

SAFI, A. et al. A fault tolerant surveillance system for fire detection and prevention using lorawan in smart buildings. *Sensors*, MDPI, v. 22, n. 21, p. 8411, 2022. 52

SANTOS, L. et al. Data processing on edge and cloud: a performability evaluation and sensitivity analysis. *Journal of Network and Systems Management*, Springer, v. 29, n. 3, p. 27, 2021. 14

SENDRA, S. e. a. Lorawan network for fire monitoring in rural environments. Electronics, v. 9, n. 3, p. 531, 2020. 4

SHI, W. et al. Edge computing: Vision and challenges. *IEEE internet of things journal*, IEEE, v. 3, n. 5, p. 637–646, 2016. 8

SILVA, B. et al. Mercury: An integrated environment for performance and dependability evaluation of general systems. *Proceedings of the Industrial Track at 45th Dependable Systems and Networks Conference, DSN*, p. 1–4, 2015. 57

SILVA, L. G. et al. Urban advanced mobility dependability: A model-based quantification on vehicular ad hoc networks with virtual machine migration. *Sensors*, MDPI, v. 23, n. 23, p. 9485, 2023. 49

SOUSA, R. d. et al. Performability evaluation and sensitivity analysis of a video streaming on demand architecture. *Applied Sciences*, MDPI, v. 13, n. 2, p. 998, 2023. 49

TALEB-BERROUANE, M.; KHAN, F.; AMYOTTE, P. Bayesian stochastic petri nets (bspn)-a new modelling tool for dynamic safety and reliability analysis. *Reliability Engineering & System Safety*, Elsevier, v. 193, p. 106587, 2020. 25

TAO, Y.; XIA, D.; ZHU, Y. Research on emergency decision-making model for major disasters based on stochastic petri nets. *2019 9th International Conference on Fire Science and Fire Protection Engineering (ICFSFPE)*, p. 1–6, 2019. 20, 21

TAYEH, G. B. et al. A personal lpwan remote monitoring system. *2021 International Wireless Communications and Mobile Computing (IWCMC)*, p. 80–85, 2021. 21, 22

TSAKMAKIS, A. et al. Learning-automata-based hybrid model for event detection in lorawan networks. *2022 IEEE 21st Mediterranean Electrotechnical Conference (MELECON)*, p. 1183–1188, 2022. 21, 22

ULLAH, M. A.; MIKHAYLOV, K.; ALVES, H. Analysis and simulation of lorawan lr-fhss for direct-to-satellite scenario. *IEEE Wireless Communications Letters*, v. 11, n. 3, p. 548–552, 2022. 4

VALENTIM, R. A. et al. Stochastic petri net model describing the relationship between reported maternal and congenital syphilis cases in brazil. *BMC medical informatics and decision making*, Springer, v. 22, n. 1, p. 40, 2022. 28

WEISSMAN, S. A.; ANDERSON, N. G. Design of experiments (doe) and process optimization. a review of recent publications. *Organic Process Research & Development*, ACS Publications, v. 19, n. 11, p. 1605–1633, 2015. 44, 59

ZHANG, C. et al. Interval-valued intuitionistic uncertain linguistic cloud petri net and its application to risk assessment for subway fire accident. *IEEE Transactions on Automation Science and Engineering*, v. 19, n. 1, p. 163–177, 2022. 20, 21

ZHOU, J.; RENIERS, G. Petri-net based simulation analysis for emergency response to multiple simultaneous large-scale fires. *Journal of Loss Prevention in the Process Industries*, v. 40, p. 554–562, 2016. ISSN 0950-4230. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0950423016300249>. 21

ZHOU, J.; RENIERS, G.; COZZANI, V. A petri-net approach for firefighting force allocation analysis of fire emergency response with backups. *Reliability Engineering and System Safety*, v. 229, p. 108847, 2023. ISSN 0951-8320. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0951832022004641>. 21