



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

Redes de sensores IoT em edifícios inteligentes: uma avaliação de desempenho usando modelos de filas

Brena Maia Santos

Número de Ordem PPGCC: M001

Teresina-PI, Outubro de 2021

Brena Maia Santos

Redes de sensores IoT em edifícios inteligentes: uma avaliação de desempenho usando modelos de filas

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. Francisco Airton Pereira da Silva

Coorientador: Prof. Dr. André Castelo Branco Soares

Teresina-PI

Outubro de 2021

FICHA CATALOGRÁFICA
Universidade Federal do Piauí
Centro de Ciências da Natureza
Serviço de Processamento Técnico

S237r Santos, Brena Maia.
Redes de sensores IoT em edifícios inteligentes: uma avaliação de desempenho usando modelos de filas / Brena Maia Santos. – 2021.
64 f. : il.

Dissertação (Mestrado) – Universidade Federal do Piauí, Programa de Pós-Graduação em Ciência da Computação, Mestrado em Ciência da Computação, Teresina, 2021.

“Orientador: Prof. Dr. Francisco Airton Pereira da Silva”.

“Co-orientador: Prof. Dr. André Castelo Branco Soares”.

1. Internet das coisas. 2. Dispositivos de internet embarcados. 3. Sistemas de monitoramento de edifícios. 4. Redes de sensores. I. Título.

CDD: 006.220 68

Brena Maia Santos

Redes de sensores IoT em edifícios inteligentes: uma avaliação de desempenho usando modelos de filas

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.



**Prof. Dr. Francisco Ailton Pereira da
Silva**

Orientador



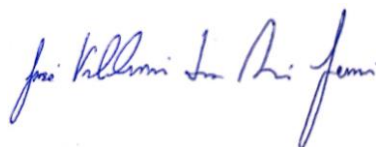
Profa. Dra. Patricia Takako Endo

Membro da Banca



Prof. Dr. Erico Meneses Leão

Membro da Banca



**Prof. Dr. José Valdemir dos Reis
Junior**

Membro da Banca

Teresina-PI

Outubro de 2021

*À minha mãe Jandira por
por estar comigo em todos os momentos.*

Agradecimentos

Agradeço a Deus por ter me abençoado com saúde e força, me permitido realizar este trabalho. Aos professores Dr. Francisco Airton e Dr. André Soares, pela orientação, incentivo e conhecimento repassado. A todos os demais professores, que foram muito importantes para a obtenção de todo o conhecimento que possuo. A toda a minha família, em especial a minha mãe Jandira, e meus irmãos Janaíra, Beatriz e Janilton Jr, por todo o apoio. Agradeço também ao meu pai Janilton, que me ensinou muito nos momentos em que esteve por perto e que, mesmo não estando mais presente fisicamente, tenho certeza que cuida de mim onde quer que esteja. A todos que de forma direta ou indireta fizeram parte da minha formação.

Resumo

A Internet das Coisas (IoT) engloba diversas tecnologias, como Wi-Fi e Bluetooth, que permitem o monitoramento constante de diferentes aspectos do dia a dia das pessoas, como o monitoramento de saúde e controle de acesso. A adoção de sensores e dispositivos IoT que permitem a interconexão de objetos está tornando edifícios em grandes cidades cada vez mais inteligentes. Nesse contexto, sensores e atuadores IoT podem monitorar parâmetros relacionados à segurança física, como questões referentes à controle de temperatura e detecção de fumaça. Dessa forma, o mau funcionamento de tais dispositivos de monitoramento pode causar danos à propriedade e talvez morte, tornando necessário avaliar o desempenho dos sistemas que compõem tal arquitetura computacional. Alguns estudos usaram modelos analíticos visando diferentes aspectos para avaliar o desempenho de sistemas de monitoramento de edifícios. No entanto, alguns pontos ainda não são explorados na literatura, como: (i) a análise da capacidade de recursos computacionais adequados à demanda de dados, ii) representação do número de núcleos por máquina e iii) o agrupamento de sensores por localização. Este trabalho propõe um modelo de rede de filas para avaliar o desempenho de uma infraestrutura de edifício inteligente com suporte à múltiplas camadas de processamento: borda e névoa. As avaliações de desempenho foram realizadas através de uma análise de sensibilidade e análises numéricas considerando o modelo proposto. Primeiramente, o modelo foi submetido à análise de sensibilidade utilizando o método Design of Experiments (DoE) para identificar gargalos na proposta. Em seguida, uma série de estudos de caso foi realizada com base nos resultados DoE. Os resultados do DoE nos permitiram concluir, por exemplo, que o número de nós pode ter mais impacto no tempo de resposta do que o número de núcleos desses nós. As análises numéricas permitem observar o comportamento das seguintes métricas: tempo médio de resposta, taxa de utilização de recursos, vazão, taxa de descarte e número de mensagens no sistema. Essas análises mostram que a taxa de chegada e o número de recursos disponíveis na névoa podem ser muito influentes no desempenho da rede. O modelo proposto e as análises realizadas podem ajudar os projetistas de sistemas a otimizar suas arquiteturas computacionais antes mesmo da implementação. O modelo foi validado através de um validador sintético que foi implementado utilizando a linguagem Java, seguindo todas as restrições impostas pelo modelo com relação aos seus componentes.

Palavras-chaves: Internet das Coisas (IoT), Edifício Inteligente, Modelo de Filas, Redes de Sensores.

Abstract

The Internet of Things (IoT) encompasses several technologies, such as Wi-Fi and Bluetooth, which allow constant monitoring of different aspects of people's daily lives, such as health monitoring and access control. The adoption of IoT sensors and devices that enable the interconnection of objects is making buildings in big cities increasingly smarter. In this context, IoT sensors and actuators can monitor parameters related to physical security, such as issues related to temperature control and smoke detection. Thus, the malfunction of such monitoring devices can cause property damage and perhaps death, making it necessary to evaluate the performance of the systems that make up such computational architecture. Some studies used analytical models targeting different aspects to assess the performance of building monitoring systems. However, some points are still not explored in the literature, such as: (i) the analysis of the capacity of computational resources adequate to the data demand, ii) representation of the number of cores per machine and iii) the clustering of sensors by location. This work proposes a queuing network model to evaluate the performance of an intelligent building infrastructure supporting multiple processing layers: edge and fog. Performance evaluations were carried out through a sensitivity analysis and numerical analysis considering the proposed model. First, the model was submitted to sensitivity analysis using the Design of Experiments (DoE) method to identify bottlenecks in the proposal. Then, a series of case studies were carried out based on the DoE results. The DoE results allowed us to conclude, for example, that the number of nodes can have more impact on response time than the number of cores on those nodes. Numerical analyzes allow us to observe the behavior of the following metrics: average response time, resource utilization rate, throughput, discard rate and number of messages in the system. These analyzes show that the arrival rate and number of resources available in fog can be very influential on network performance. The proposed model and the analyzes carried out can help system designers to optimize their computational architectures even before implementation. The model was validated through a synthetic validator that was implemented using the Java language, following all restrictions imposed by the model in relation to its components.

Keywords: Internet of Things (IoT), Smart Building, Queuing Model, Sensor Networks.

Lista de ilustrações

Figura 1 – Centro de serviço (fila de espera e um ou mais servidores).	11
Figura 2 – Componentes de um modelo de filas	12
Figura 3 – Gráfico de Pareto com Efeitos Padronizados	14
Figura 4 – Gráfico de Efeitos Principais da Temperatura na Resistência à Tração	15
Figura 5 – Gráfico de Interações Sinérgica e Antagônica	16
Figura 6 – Metodologia de trabalho deste projeto de avaliação de edifícios inteligentes.	22
Figura 7 – Visão geral de um cenário que permite monitorar vários sensores em um edifício inteligente.	24
Figura 8 – Modelo de filas de uma cenário de edifício inteligente composto por múltiplas camadas.	26
Figura 9 – Influência dos fatores na métrica MRT.	31
Figura 10 – Efeitos Principais para MRT.	31
Figura 11 – Interações de fatores considerando o MRT.	32
Figura 12 – Resultados da análise do modelo considerando diferentes números de núcleos.	34
Figura 13 – Resultados da análise do modelo considerando diferentes números de nós da névoa.	37
Figura 14 – Análise do tempo médio de resposta por dois fatores simultaneamente.	41
Figura 15 – Visão geral da infraestrutura utilizada no experimento de mensuração e validação do modelo proposto. Foram usados apenas quatro computadores por restrição de recursos.	44
Figura 16 – Diagrama de sequência destacando a comunicação e principais ações gerais realizadas em cada componente.	45
Figura 17 – Diagrama de classes com os principais componentes do modelo proposto. O validador sintético foi implementado em Java com uso de Sockets.	45
Figura 18 – Resultado do T-Test para a configuração de 1 núcleo de processamento.	46
Figura 19 – Resultado do T-Test para a configuração de 2 núcleos de processamento.	47
Figura 20 – Resultado do T-Test para a configuração de 3 núcleos de processamento.	47
Figura 21 – Resultado do T-Test para a configuração de 4 núcleos de processamento.	48

Lista de tabelas

Tabela 1 – Trabalhos Relacionados	17
Tabela 2 – Descrição dos componentes do modelo	26
Tabela 3 – Fatores e níveis do DoE	29
Tabela 4 – Combinações de fatores e níveis do DoE	30
Tabela 5 – Parâmetros Inseridos no Modelo	33
Tabela 6 – Combinações dos fatores número de núcleos e número de nós.	40
Tabela 7 – Dados Estatísticos do T-Test	46

Lista de abreviaturas e siglas

AR	<i>Arrival Rate</i>
DoE	<i>Design os Experiments</i>
GPS	Sistema de Posicionamento Global
IoT	<i>Internet of Things</i>
JMT	<i>Java Modeling Tools</i>
LAN	<i>Local Area Networks</i>
MRT	Tempo Médio de Resposta
NTP	<i>Network Time Protocol</i>
PAN	<i>Personal Area Networks</i>
SLA	<i>Service Level Agreement</i>
TIC	Tecnologias de Informação e Comunicação
WAN	<i>Wide Area Networks</i>

Sumário

1	INTRODUÇÃO	3
1.1	Motivação	4
1.2	Proposta e Contribuições	5
1.3	Organização do Trabalho	5
2	REFERENCIAL TEÓRICO	7
2.1	Internet das Coisas	7
2.1.1	Computação de Borda	8
2.1.2	Computação de Névoa	9
2.2	Edifícios Inteligentes	10
2.3	Introdução à Teoria de Filas	11
2.4	Análise de Sensibilidade usando DoE	13
2.4.1	Gráfico de Pareto	14
2.4.2	Gráfico de Efeitos Principais	14
2.4.3	Gráfico de Interação	15
3	TRABALHOS RELACIONADOS	17
4	DESENVOLVIMENTO DO MODELO DE FILAS	21
4.1	Metodologia	21
4.2	Cenário do edifício inteligente	23
4.3	Modelo de Rede de Filas	25
5	RESULTADOS E DISCUSSÃO	29
5.1	Análise de sensibilidade usando DoE	29
5.1.1	Desenho Experimental	29
5.1.2	Resultados do DoE	30
5.2	Resultados das Simulações	32
5.2.1	Cenário A: Variando o número de núcleos	33
5.2.2	Cenário B: Variação no número de nós na névoa	36
5.2.3	Cenário C: Variando o número de nós e núcleos simultaneamente	39
6	VALIDAÇÃO	43
7	CONCLUSÃO	49
8	PUBLICAÇÕES	51

REFERÊNCIAS **53**

1 Introdução

A Internet das Coisas (IoT) foi criada para permitir a conectividade onipresente entre diferentes “coisas” (MENA; PAPAPANAGIOTOU; YANG, 2018). A IoT conecta qualquer item à Internet por meio de interfaces de rede para identificação, posicionamento, rastreamento, monitoramento e gerenciamento (FU; YANG, 2021). O número de dispositivos IoT em todo o mundo está previsto para quase dobrar de 8,74 bilhões em 2020 para mais de 16,44 bilhões em 2025 (ANALYTICS, 2020). A IoT é promissora por causa de sua importância em muitas aplicações de comércio, indústria e educação (SAID; MASUD, 2013). Dispositivos IoT podem ser usados, por exemplo, para automatizar atividades em casas inteligentes (AJAO et al., 2020), tráfego de redes veiculares (WU et al., 2020) e monitoramento de saúde de idosos em suas casas (SOKULLU; AKKAŞ; DEMIR, 2020). Portanto, um sistema IoT concede serviços a cenários inteligentes em diferentes contextos, gerenciando com eficiência hardware, software e recursos de comunicação para reduzir custos em domínios específicos. Identificar comportamento anormal em um ambiente monitorado usando IoT tem sido muito útil em casas e edifícios inteligentes.

De acordo com as Nações Unidas, a população urbana mundial deverá crescer 2,5 bilhões de 2014 a 2050, quando deverá representar 66% da população global total (UN, 2015). O crescimento da população nas cidades aumenta a demanda pelas necessidades fundamentais das pessoas que vivem nelas, como habitação, serviços públicos, cuidados médicos, bem-estar, educação e emprego (TASCIKARAOGLU, 2018). O conceito de cidade inteligente foi idealizado para lidar com os desafios enfrentados durante o crescimento das cidades. Uma cidade inteligente denota “a integração efetiva de sistemas físicos, digitais e humanos no ambiente construído para oferecer um futuro sustentável, próspero e inclusivo para seus cidadãos” (GUREEV; DUNENKOVA; ONISHCHENKO, 2020). Como células de cidades inteligentes, os edifícios inteligentes integram inteligência, empresa, controle, materiais e construção para promover a eficiência energética, longevidade, conforto e satisfação daqueles que frequentam o edifício (BUCKMAN; MAYFIELD; BECK, 2014). Em ambos os contextos de cidades e edifícios inteligentes, o “inteligente” se refere ao desenvolvimento, integração e utilização de sistemas inteligentes baseados em Tecnologias de Informação e Comunicação (TIC). Originalmente, a tecnologia de casa inteligente era usada para controlar os sistemas ambientais, como iluminação e aquecimento. No entanto, há algum tempo, a IoT permite que quase todos os componentes elétricos da casa sejam conectados a um sistema central, tornando possível monitorar o ambiente. Uma casa inteligente pode fornecer serviços com base nas necessidades do usuário. A partir do conceito de casas inteligentes, a IoT pode ser estendida a um contexto mais amplo: edifícios inteligentes. O gerenciamento de edifícios inteligentes requer infraestruturas de

computação mais sofisticadas e prontidão para atender à uma grande demanda por geração de dados. O poder computacional da computação em nuvem é usado para processar essa demanda há um longo tempo.

A computação em nuvem tem sido a espinha dorsal para hospedar e oferecer serviços de computação e aplicativos orientados por assinatura. Ela também é usado para executar os aplicativos para diferentes sistemas ciberfísicos habilitados para IoT (ISLAM; KUMAR; HU, 2021). Usar a computação em nuvem às vezes é impossível devido ao posicionamento distante de dispositivos IoT. A computação de borda e névoa surgiu para colocar os serviços mais perto da fonte de dados (XU et al., 2020). O problema de pesquisa central é que tal adição de camadas de processamento na arquitetura de sistemas IoT requer avaliações de desempenho desde os primeiros estágios de desenvolvimento. No entanto, as avaliações com experimentos reais podem ser caras porque existem muitas possibilidades de configuração, principalmente no contexto de edifícios inteligentes, onde normalmente há vários andares com várias salas em cada um deles. A automação residencial simples com dispositivos IoT pode custar mais de cem mil dólares ¹. Despesas desnecessárias podem ser feitas para obter recursos que não serão usados. Os modelos analíticos podem ser uma solução neste contexto, permitindo previsões baseadas em cálculos probabilísticos (SILVA et al., 2017; PINHEIRO et al., 2019; RODRIGUES; ENDO; SILVA, 2019; SANTOS et al., 2020; SILVA; FÉ; GONÇALVES, 2020; CARVALHO et al., 2020; SILVA et al., 2018; PINHEIRO et al., 2018; PINHEIRO et al., 2018; SILVA et al., 2015). A teoria das filas é um método matemático popular para analisar diferentes sistemas e observar seu comportamento em relação ao desempenho do sistema. Os modelos de filas são simples, didáticos e eficientes (IEEE, ; MEMON; LI; AHMED, 2019; ARBIB et al., 2019). As aplicações da teoria das filas geralmente têm dois objetivos: prever o desempenho do sistema e encontrar um projeto de sistema para melhorar o desempenho na fase de planejamento (GHOMI; RAHMANI; QADER, 2019).

1.1 Motivação

Alguns trabalhos na literatura desenvolveram modelos analíticos para avaliar sistemas IoT em casas e edifícios inteligentes. Alguns estudos focaram em garantir apenas o funcionamento do sistema (NABIH et al., 2011; NOVÁK; BIÑAS; JAKAB, 2012; FANTI et al., 2014; GARCIA-CONSTANTINO; KONIOS; NUGENT, 2018), outros analisaram a eficiência energética (FANTI; MANGINI; ROCCOTELLI, 2014; FANTI; MANGINI; ROCCOTELLI, 2018; WANG; SECHILARIU; LOCMONT, 2013). Ajao et al. (AJAO et al., 2020), por exemplo, analisou o desempenho do sistema em termos de tempo de resposta e taxa de descarte de requisições. Entre os artigos citados acima, nenhum explorou

¹ Homeadvisor Service: <https://www.homeadvisor.com/cost/electrical/install-or-repair-a-home-automation-system/>

camadas de borda e névoa. Trabalhos anteriores também não exploraram a análise do impacto da variação da capacidade dos recursos no desempenho do sistema e nem o agrupamento de sensores por localização, característica essencial no monitoramento de mais de um ambiente. Desse modo, algumas questões podem surgir: (i) “*qual o impacto da taxa de chegada no desempenho de um sistema de edifício inteligente?*”; (ii) “*como uma configuração de capacidade de recursos específica impacta no desempenho do sistema de um edifício inteligente?*”; e (iii) “*quais recursos possuem maior impacto nas métricas de desempenho de um sistema de edifício inteligente?*”. Espera-se que a taxa de chegada seja um fator relevante no desempenho de um sistema de edifício inteligente, no entanto, é importante observar o nível de relevância de acordo com a configuração utilizada. Sendo assim, o principal objetivo deste trabalho é avaliar um modelo de filas desenvolvido com base no cenário de um edifício inteligente, identificando quais os fatores e configurações que possuem maior impacto no desempenho da rede.

1.2 Proposta e Contribuições

Este trabalho propõe um modelo de enfileiramento para avaliar sistemas IoT para edifícios inteligentes suportados por névoa. O modelo é baseado em um cenário onde é considerado o conforto climático de um edifício que possui mais de um andar e em cada um desses andares existem várias salas que são equipadas com sensores e dispositivos de borda. Os dispositivos de borda enviarão os dados coletados pelos sensores para processamento em nós da névoa. Em seguida, os dados processados ficarão disponíveis para acesso de clientes internos. As **contribuições** deste trabalho são as seguintes:

- Um **modelo de filas**, que é uma ferramenta útil para projetistas de sistemas para avaliar o desempenho de arquiteturas para edifícios inteligentes suportados por borda-névoa, antes mesmo de sua implementação. O modelo permite configurar parâmetros de acordo com a necessidade do projetista, incluindo a taxa de chegada de mensagens, número de nós, tempos de atendimento, tamanho da fila, entre outros. A partir disso, o projetista poderá analisar diversas métricas de desempenho, por exemplo, tempo médio de resposta (MRT) e taxa de descarte.
- Uma **análise de sensibilidade** com *Design of Experiments* (DoE), que permite analisar diferentes fatores e como as mudanças em seus níveis impactam o desempenho de um sistema de edifício inteligente. Quatro fatores serão observados: (i) tempo de serviço, (ii) número de nós de névoa, (iii) número de núcleos de processamento e (iv) tamanho da fila.
- Três **simulações** que servem como um guia para analisar o desempenho de um edifício inteligente. Os cenários monitorados analisam o comportamento do sistema

alterando o número de nós de névoa e o número de núcleos. Métricas como MRT, utilização de recursos, taxa de descarte e vazão foram consideradas.

1.3 Organização do Trabalho

O restante deste trabalho está organizado da seguinte forma: O capítulo 2 apresenta referencial teórico, com os principais conceitos necessários para a compreensão deste trabalho. O capítulo 3 apresenta os trabalhos relacionados, comparando-os com nossa proposta. O capítulo 4 apresenta o desenvolvimento do modelo de filas, explicando a metodologia aplicada, o cenário que foi usado como base para a construção do modelo e o modelo de filas proposto. O capítulo 5 descreve os experimentos realizados com Design of Experiments (DoE) e apresenta as simulações realizadas no modelo proposto. No capítulo 6 é apresentada a validação do modelo de filas proposto. O capítulo 7 apresenta as principais conclusões obtidas acerca dos estudos realizados. Finalmente, o capítulo 8 contém as publicações realizadas a partir deste trabalho.

2 Referencial Teórico

Esta seção apresenta conceitos essenciais para a compreensão da proposta e trabalhos relacionados. Inicialmente, algumas definições sobre Internet das Coisas são apresentadas na seção 2.1. Na seção 2.2, destacamos informações importantes sobre o funcionamento de edifícios inteligentes. A seção 2.3 apresenta uma introdução ao modelo de filas. Por fim, na seção 2.4, o funcionamento do DoE e a forma como seus resultados são geralmente interpretados são discutidos.

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, a Internet das Coisas 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 Internet das Coisas é a interligação de redes de dispositivos físicos incorporados com 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, e os interconectam logicamente com objetos virtuais (aplicativos da web, máquinas virtuais) sobre a infraestrutura de Internet existente (PAUL; JEYARAJ, 2019). A IoT conecta dispositivos por meio da Internet, permitindo monitoramento e controle remotos por meio de sistemas de controle baseados em nuvem. O objetivo da IoT é automatizar cada vez mais aspectos de nossas vidas e, ao mesmo tempo, aumentar a eficiência dos processos. Os sistemas IoT podem ser inseridos em várias áreas sociais, e cada uma tem um impacto diferente. A indústria e a manufatura são os maiores segmentos no espaço geral da IoT, considerando o número de dispositivos conectados ao valor que esses serviços agregam à indústria (LEA, 2018).

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, 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. Na fase de armazenamento e processamento, algumas plataformas que podem ser utilizadas são: computação em nuvem, computação em névoa, computação em grade e computação de borda. 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 vários novos aplicativos de computação, como realidade virtual e ambientes 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 aplicativos sensíveis a atraso apresentam alto nível de exigência para as infraestruturas computacionais convencionais. Dessa forma, o uso de serviços de computação em nuvem, névoa e borda tornaram-se partes importantes para aplicativos IoT. A computação em nuvem tem como principal objetivo fazer melhor uso dos recursos distribuídos e resolver problemas de computação em grande escala. Por exemplo, a computação em nuvem pode concentrar o poder de milhares de computadores em um problema, permitindo que os pesquisadores façam seu trabalho mais rápido do que nunca (SADIKU; MUSA; MOMOH, 2014). No entanto o paradigma da computação em nuvem não consegue atender aos requisitos de baixa latência e reconhecimento de local. Nesse contexto, pode-se introduzir a computação de borda e névoa.

2.1.1 Computação de Borda

Com a crescente quantidade de dados gerados na borda da rede, a velocidade do transporte de dados está se tornando o gargalo para o paradigma da computação baseada em nuvem (SHI et al., 2016). Nesse caso, os dados precisam ser processados na borda para um tempo de resposta mais curto, processamento mais eficiente e menor pressão de rede. A computação de borda se refere às tecnologias que permitem que a computação seja realizada na borda da rede para que ela aconteça perto das fontes de dados. Dessa forma, os provedores de conteúdo e desenvolvedores de aplicativos podem usar os sistemas de computação de borda, oferecendo aos usuários serviços mais próximos a eles. Os casos de uso da computação de borda 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 borda, de acordo com o tipo de aplicação:

- Os aplicativos de navegação podem mover os serviços de navegação ou pesquisa para a borda de uma área local, nos casos 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.

- Aplicativos 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 de borda, seus dispositivos podem ser qualquer recurso de computação ou rede que resida entre fontes de dados e data centers baseados em nuvem. Por exemplo, um dispositivo de borda pode ser um smartphone que se encontra entre os sensores e a nuvem. Na computação de borda, o dispositivo final não apenas consome dados, mas também produz dados. E na extremidade da rede, os dispositivos não apenas solicitam serviços e informações da nuvem, mas também lidam com tarefas de computação - incluindo processamento, armazenamento, armazenamento em cache e balanceamento de carga - em dados enviados de e para a nuvem. A borda deve ser projetada bem 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 que precisam ser atendidos para utilizar os sistemas de computação de borda podem ser classificados em mecanismo de faturamento inteligente, suporte a aplicativos 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 de Névoa

Computação de névoa 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 à nuvem, a névoa fornece serviços de dados, computação, armazenamento e aplicativos para usuários finais. A névoa fornece processamento e armazenamento de dados IoT localmente em dispositivos IoT em vez de enviá-los para a nuvem. Em contraste com a nuvem, a névoa fornece serviços com resposta mais rápida e maior qualidade. Na computação em névoa, os serviços podem ser hospedados em dispositivos finais, como decodificadores ou pontos de acesso. A infraestrutura desta computação distribuída permite que os aplicativos sejam executados o mais próximo possível de dados acionáveis e massivos detectados, vindos de pessoas, processos e coisas.

A computação em névoa oferece muitas vantagens importantes desejadas pelos aplicativos de hoje, como processamento em tempo real, escalonamento rápido e acessível, e conteúdo local e pool de recursos. Como tal, a computação em névoa atraiu rapidamente muita atenção da indústria e da academia. Naturalmente, ele conecta a IoT com a infraestrutura de computação da internet. Em (CHEN; ZHANG; SHI, 2017) os autores citam alguns aplicativos que exigem computação com névoa, são eles: veículos conectados, veículos de piloto automático, redes inteligentes, redes de sensores e atuadores sem fio,

casas inteligentes, cidades inteligentes, manufatura conectada, sistemas de óleo e gás conectados e sistemas móveis de saúde.

Os sistemas IoT apresentam alguns requisitos-chave que devem ser cumpridos pelos ambientes de névoa e computação em nuvem para atingir uma operação correta e 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 Edifícios Inteligentes

Para (KUBBA, 2012), um edifício inteligente é aquele que fornece fácil acesso, mantém as pessoas confortáveis, ambientalmente satisfeitas, seguras e fornece serviços para manter os ocupantes produtivos para seus fins no edifício. Conforme denotado pelo termo edifício “inteligente”, ele possui sistemas de controle inteligentes e dispositivos inteligentes e interconectados além da estrutura e função do edifício tradicional. Segundo (JIA et al., 2018), um edifício inteligente pode ser caracterizado por três aspectos: componentes, funções e resultados. Os componentes compreendem várias peças interconectadas de equipamentos e aparelhos técnicos, incluindo sistemas tradicionais, como aquecimento, ventilação e ar condicionado (HVAC), iluminação, rede e sistemas elétricos, juntamente com sua infraestrutura de detecção e controle associada e tecnologias emergentes, como geração e armazenamento de energia no local.

A partir da IoT, quase todos os componentes elétricos de um ambiente podem ser conectados à um sistema central, possibilitando o monitoramento desse ambiente. Como a IoT está integrada ao edifício inteligente, é necessário armazenar, processar e analisar as informações obtidas das entidades que interagem (inquilinos, outros edifícios, sensores, etc.) Edifícios inteligentes abrangem uma ampla variedade de sensores, atuadores, dispositivos e sistemas de controle que são interconectados e funcionam em conjunto para melhorar o serviço para seus ocupantes (JIA et al., 2019).

As capacidades dos sistemas presentes num edifício avaliam-se pelas funções que executam. Essas funções, que podem ser bastante diversificadas, possuem no entanto características (tais como a sua natureza, o seu âmbito, ou os seus objetivos) que as permitem agrupar em conjuntos (NUNES; SÊRRO, 2014). Segundo (ZAFARI; PAPAPANAGIOTOU; CHRISTIDIS, 2015), as características gerais de um edifício inteligente são as seguintes:

- vários sistemas de negócios interconectados;
- equipar os inquilinos ou pessoas com tecnologia;
- ligação a vários outros edifícios;

- conexão às redes inteligentes.

Os edifícios inteligentes têm como objetivo fornecer soluções que sejam eficientes em termos de energia, ecológicas, gerenciáveis em caso de desastres e confortáveis. Portanto, qualquer solução que possa potencialmente aumentar o nível de conforto e fornecer os serviços mencionados pode ser incorporada em edifícios inteligentes. Tais soluções podem ser alcançadas através de serviços como: detectar a ocupação de um ambiente e a partir disso adaptar o tratamento do espaço interno (iluminação, térmico e qualidade do ar interno) de acordo com as mudanças de ocupação para economizar energia (AIN et al., 2018; SHAH et al., 2020; NABIH et al., 2011); realizar reconhecimento facial para permitir/bloquear acesso a determinados locais, a partir de imagens coletadas por câmeras de segurança (ISMAEL; IRINA, 2020; MACIEL et al., 2017; PRIYANKA et al.,); realizar atividade de resposta à demanda de energia quando o edifício é equipado com medidores de energia e um sistema de armazenamento de energia (CUTSEM et al., 2020; ZHANG et al., 2019; HAIDAR et al., 2018).

Aquecimento, ventilação e ar condicionado (sistemas HVAC) são responsáveis pela maior parte da energia usada em edifícios. Usar um equipamento HVAC altamente eficiente pode levar a economias significativas de energia e emissões. Toda a estrutura do edifício em combinação com a zona de conforto aprimorada pode produzir uma economia muito maior. Além disso, HVAC de alta eficiência pode fornecer conforto térmico aprimorado para as pessoas, bem como contribuir para a melhoria da qualidade ambiental da sala (OMAROV; ALTAYEVA; CHO, 2017).

2.3 Introdução à Teoria de Filas

Uma fila é a implementação de uma lista de espera de trabalhos a fim de obter um serviço. A teoria das filas é usada quando você tem muitos trabalhos, recursos limitados e, como resultado, longas filas e atrasos. Em outras palavras, a teoria das filas se aplica a qualquer lugar em que surjam filas (GHOMI; RAHMANI; QADER, 2019). Um ou mais servidores podem fornecer determinados serviços. Este conjunto (fila e um ou mais servidores capaz de fornecer serviços) é denominado centro de serviços. Se for feita uma analogia com o mundo real, pode-se imaginar uma ou mais recepções que são acessadas por meio de uma fila. Todas as mesas frontais funcionam em paralelo e prestam o mesmo serviço, independentemente da sua posição. Na teoria das filas, um centro de serviço é esquematicamente representado como na Figura 1 (HARAGOS; CERNAZANU-GLAVAN, 2012).

Os centros de serviço pode ser configurado com um ou mais servidores e possuem uma fila de espera, que pode ser definida com tamanho finito ou infinito, onde os dados aguardam processamento caso os servidores disponíveis estejam sendo utilizados. Um

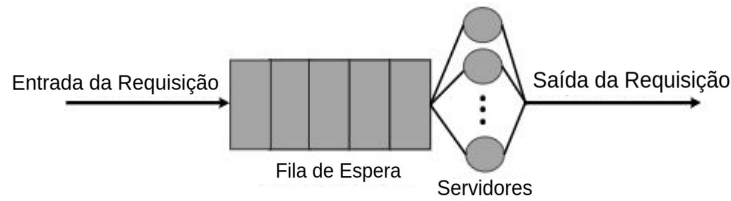


Figura 1 – Centro de serviço (fila de espera e um ou mais servidores).

modelo que usa um ou mais centros de serviço é chamado de modelo de filas. No entanto, além dos centros de serviço o modelo de filas possui vários outros componentes que podem ser utilizados. A Figura 2 apresenta alguns desses componentes.

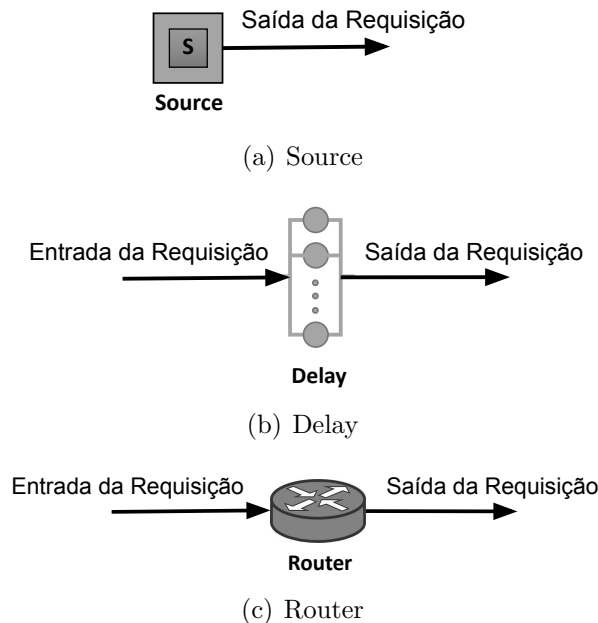


Figura 2 – Componentes de um modelo de filas

Na Figura 2(a) temos o componente "Source", que é responsável pela geração dos dados que serão inseridos no modelo. Para esse componente é possível definir a taxa de chegada dos dados e o nível de prioridade do mesmo. A Figura 2(b) apresenta o componente "Delay", que permite adicionar uma pausa na transmissão dos dados, podendo ser útil em casos em que é necessário a simulação de um atraso de rede. A Figura 2(c) apresenta o componente "Router", que realiza a transmissão de dados para os centros de serviço.

Várias métricas de desempenho são usadas para medir o desempenho de um modelo de filas. Abaixo, as métricas de desempenho mais críticas são apresentadas e eles podem caracterizar esse modelo de filas. O número de clientes em um sistema de tempo é denotado por n_s . Uma variável aleatória n , cujos valores médios não podem ser calculados usando uma probabilidade distribuição, é usada. O valor esperado para o número de clientes que estão em um determinado momento no sistema é chamado n^{th} sobre a origem de n e é

calculado como segue:

$$E[n^k] = \sum_{i=0}^{\infty} i^k \text{Prob}\{n = i\} \quad (2.1)$$

Se for considerado que $\text{Prob}\{n = i\}$ e $E[n^k]$ são iguais às médias ao longo de um intervalo de tempo infinitamente longo (médias de tempo de longa duração), temos:

$$\lim_{s \rightarrow \infty} (\text{primeiro tempo } s \text{ quando } i \text{ clientes estão no sistema}) \quad (2.2)$$

$$E[n^k] = \lim_{s \rightarrow \infty} \frac{1}{S} \int_0^S [n_u]^k du \quad (2.3)$$

Cada cliente j que chega ao sistema deve passar algum tempo antes de ser atendido. Este tempo é definido como o tempo de resposta para o cliente j e foi anotado com r_j . Se for considerado que $\text{Prob}\{r \leq t\}$ e $E[r^k]$ são iguais às médias sobre um número infinito de clientes (médias de clientes de longo prazo), a seguinte equação para o função de distribuição de probabilidade e valor esperado.

$$\text{Prob}\{r \leq t\} = \lim_{j \rightarrow \infty} (Z) \quad (2.4)$$

A variável Z significa a fração dos primeiros J clientes a chegar cujo tempo de resposta é menor ou igual a t .

$$E[r^k] = \lim_{j \rightarrow \infty} \frac{1}{J} \sum_{j=1}^J r_j^k \quad (2.5)$$

Se o sistema for estável ($E[n]$ e $E[r]$ são números finitos), a vazão T deve ser igual à taxa de longo prazo para as chegadas de clientes, onde T é definido como:

$$T = \lim_{s \rightarrow \infty} \frac{1}{S} (\text{número de clientes que passaram no tempo } s) \quad (2.6)$$

O tempo em que um servidor está ocupado é denominado de utilização da central de atendimento. Se for denotado por b_s , o número de servidores ocupados no tempo s , U (utilização) é definido como segue:

$$T = \lim_{s \rightarrow \infty} \frac{1}{S} \int_0^s b_u du \quad (2.7)$$

2.4 Análise de Sensibilidade usando DoE

A análise de sensibilidade é uma medida do efeito de um determinado dado de entrada sobre os dados de saída, com o objetivo de delinear os elos fracos dos sistemas computacionais, e a partir daí, buscar adotar um conjunto de técnicas que visam melhorar esses sistemas em diferentes cenários (CAMPOLONGO; TARANTOLA; SALTELLI, 1999). Alguns trabalhos usam análise de sensibilidade para fornecer a segurança necessária e encaminhar a perspectiva dos administradores do sistema (COSTA et al., 2016; FERREIRA et al., 2019). Neste trabalho, aplicamos uma análise de sensibilidade com DoE.

O *Design of Experiments* (DoE) corresponde a um conjunto de técnicas estatísticas que aprofundam o conhecimento sobre o produto ou processo em estudo (KLEIJNEN, 1995). Também pode ser definido por uma série de testes nos quais o pesquisador altera o conjunto de variáveis ou fatores de entrada para observar e identificar os motivos das mudanças na resposta de saída.

Os projetistas de sistemas costumam adotar análises de sensibilidade para avaliar o quão “sensível” uma métrica é a mudanças no modelo (SANTOS et al., 2021). Os parâmetros a serem alterados são definidos por meio de um plano de experimento. O objetivo é gerar a quantidade mais significativa de informações com o mínimo de experimentos possíveis. O comportamento do sistema com base nas mudanças de parâmetro pode ser observado usando conjuntos de saídas. Na literatura, existem três categorias de gráficos geralmente adotados para experimentos com DoE:

2.4.1 Gráfico de Pareto

O gráfico de Pareto permite detectar qual o efeito da interação de fatores é mais importante para o processo ou estudo de otimização de projeto com o qual se deve lidar. Exibe os valores absolutos dos efeitos e desenha uma linha de referência no gráfico. Qualquer efeito que ultrapasse essa linha de referência é potencialmente importante. Um gráfico de Pareto é construído como na Figura 3, por exemplo. O gráfico mostra que os fatores B (tool geometry) e C (cutting angle) e a interação AC possui maior impacto.

2.4.2 Gráfico de Efeitos Principais

O “Gráfico de Efeitos Principais” é o gráfico dos valores médios de resposta em cada nível de um parâmetro de projeto ou variável de processo. Pode-se usar esse gráfico para comparar a força relativa dos efeitos de vários fatores. O sinal e a magnitude de um efeito principal nos diriam o seguinte:

- O sinal de um efeito principal nos diz a direção do efeito, ou seja, se o valor médio da resposta aumenta ou diminui.

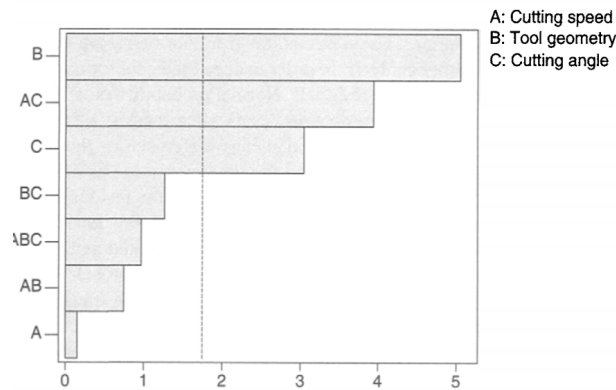


Figura 3 – Gráfico de Pareto com Efeitos Padronizados

- A magnitude nos diz a força do efeito.

Se o efeito do parâmetro de projeto ou processo é positivo, isso implica que a resposta média é maior em nível alto do que em nível baixo da configuração do parâmetro. Em contraste, se o efeito for negativo, isso significa que a resposta média em nível baixo de configuração do parâmetro é maior que em nível alto. A Figura 4 ilustra o efeito principal da temperatura na resistência à tração de uma amostra de aço. Como pode ser visto na Figura, a resistência aumenta quando as configurações da temperatura variam para o nível baixo (ou seja, 1 para 1). O efeito de um parâmetro de projeto ou processo (ou fator) pode ser calculado matematicamente usando a simples Equação 2.8:

$$E_f = \bar{F}_{(+1)} - \bar{F}_{(-1)} \quad (2.8)$$

Quando $\bar{F}_{(+1)}$ significa a resposta média na configuração de alto nível de um fator, e $\bar{F}_{(-1)}$ significa a resposta média na configuração de baixo nível de um fator.

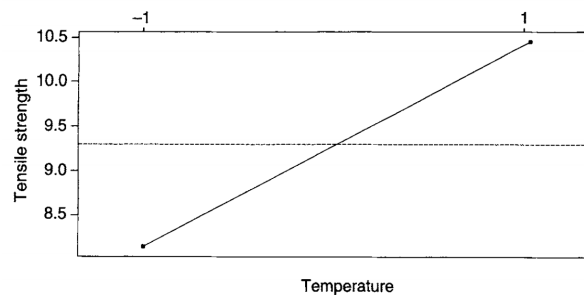


Figura 4 – Gráfico de Efeitos Principais da Temperatura na Resistência à Tração

2.4.3 Gráfico de Interação

O processo de “Interações”, identifica efeitos importantes e determina sua magnitude, logo, as interações entre os efeitos são cruciais. As interações ocorrem quando o efeito de um fator depende do nível de outro fator. Uma medida de design sempre aborda vários

fatores. Entender como esses fatores interagem em que magnitude permite para escolher a melhor combinação de medidas, revelando combinações de fatores com efeito cumulativo ou degradante. A interação entre os fatores A e B pode ser calculada usando a Equação 2.9.

$$I_{A,B} = \frac{1}{2}(E_{A,B(+1)} - E_{A,B(-1)}) \quad (2.9)$$

O $E_{A,B(+1)}$ é o efeito do fator 'A' no nível alto do fator 'B' e $E_{A,B(-1)}$ é o efeito do fator 'A' no nível baixo do fator 'B'.

Para determinar se dois parâmetros de processo estão interagindo ou não, pode-se usar uma ferramenta gráfica simples, porém poderosa, chamada de gráficos de interação. Se as linhas no gráfico de interação forem paralelas, não haverá interação entre os parâmetros do processo. Isso implica que a mudança na resposta média do fator 'A' não depende dos níveis do fator 'B'. Por outro lado, se as linhas não são paralelas, existe uma interação entre os fatores. Quanto maior o grau de afastamento de ser paralelo, mais forte o efeito de interação. Para interação sinérgica, as linhas no gráfico não se cruzam. Por exemplo, a Figura 5(a) é um exemplo de interação sinérgica.

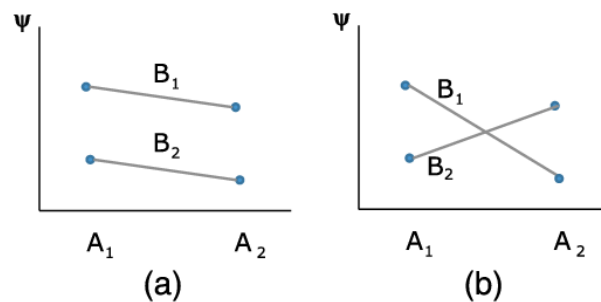


Figura 5 – Gráfico de Interações Sinérgica e Antagônica

Na interação antagônica, as linhas no gráfico se cruzam. Isso pode ser ilustrado na Figura 5(b). Neste caso, a mudança na resposta média para o fator A no nível 'A₁' é alta em comparação ao nível 'A₂'. As mudanças nos níveis do fator 'A' para a resposta média, indica uma dependência do fator 'A' em relação aos níveis do fator 'B'.

3 Trabalhos Relacionados

Este capítulo apresenta trabalhos relacionados com abordagens semelhantes à nossa proposta. Oito estudos foram selecionados e classificados por ano, todos fazem uso de algum modelo analítico (por exemplo, Redes de Petri, Cadeia de Markov e Redes de Fila) e contexto de aplicação em casas ou edifícios inteligentes.. A Tabela 1 apresenta uma comparação dos estudos coletados, destacando o contexto da aplicação, as métricas utilizadas, a análise da capacidade dos recursos, agrupamento de sensores por localização e representação do número de núcleos por máquina. Em seguida, todos os trabalhos relacionados são discutidos.

Tabela 1 – Trabalhos Relacionados

Referência	Contexto de Aplicação	Métricas	Análise da Capacidade de Recursos	Sensores Agrupados por Localização	Representa Número de Núcleos por Máquina
NABIH et al.	Casa inteligente	Eficácia do sistema	Não	Não	Não
NOVÁK; BIÑAS; JAKAB	Casa inteligente	Eficácia do sistema	Não	Não	Não
WANG; SECHILARIU; LOCMENT	Edifício inteligente	Eficiência energética	Não	Não	Não
FANTI; MANGINI; ROCCOTELLI	Edifício inteligente	Eficiência energética	Não	Não	Não
FANTI et al.	Casa inteligente	Eficácia do sistema, vivacidade, limitação e reversibilidade	Não	Não	Não
FANTI; MANGINI; ROCCOTELLI	Edifício inteligente	Eficiência energética	Não	Não	Não
GARCIA-CONSTANTINO; KONIOS; NUGENT	Casa inteligente	Eficácia do sistema	Não	Não	Não
AJAO et al.	Casa inteligente	Porcentagem de erro, Tempo de resposta.	Não	Não	Não
Este trabalho	Edifício inteligente	Utilização de recursos, Tempo Médio de Resposta, Taxa de descarte, Vazão, Número de trabalhos no sistema.	Sim	Sim	Sim

Alguns trabalhos possuem contexto de aplicação em casas inteligentes (NABIH et al., 2011; NOVÁK; BIÑAS; JAKAB, 2012; FANTI et al., 2014; GARCIA-CONSTANTINO; KONIOS; NUGENT, 2018; AJAO et al., 2020). Nabih et al. (2011) utilizam modelo de rede de petri em um cenário que envolve os seguintes módulos de casa inteligente: sistema de iluminação, sistema de controle de cortina, sistema de controle de temperatura, sistema de controle multimídia, sistema de acesso remoto e sistema de segurança. Como forma de avaliar o modelo proposto, os autores apenas realizaram testes para garantir o funcionamento do sistema, sem se preocupar com recursos computacionais. Novák, Biñas e Jakab (2012) utilizam modelo de Markov, considerando o uso de sensores para monitorar diferentes cômodos e detectar acidentes domésticos. Foi apresentado um breve estudo experimental realizado em um conjunto de dados fornecido pelo projeto MavHome. Para avaliação do modelo, os autores realizaram testes para verificar se o sistema se comporta de maneira esperada de acordo com determinadas situações. O comportamento inesperado foi adicionado artificialmente ao sistema como meio de teste. Fanti et al. (2014) buscaram monitorar a vida de idosos que residem em casas inteligentes, para detectar inatividade e acidentes, além de avaliar a gravidade do acidente para comunicar familiares e serviços de emergência. Os autores propõem um modelo de rede de petri em uma abordagem modular, a fim de descrever efetivamente as ações e as atividades do sistema, e executam uma simulação que permite verificar se os protocolos de emergência são executados corretamente e se as propriedades de vitalidade, limitação e reversibilidade são preservadas.

Garcia-Constantino, Konios e Nugent (2018) desenvolveram um modelo de rede de petri que considera atividades de um dia típico de uma pessoa idosa, por exemplo, preparação de chá e/ou café. Sensores térmicos e de contato são considerados com o objetivo de detectar comportamentos anormais no contexto dessas atividades. Os autores avaliam o modelo através de testes apenas para garantir o funcionamento do sistema, ou seja, não é considerada a utilização de recursos computacionais. Ajao et al. (2020) projetaram uma sala inteligente com controle de janela automatizado, que pode abrir e fechar automaticamente com base nas mudanças das condições climáticas, com o objetivo de auxiliar pessoas idosas ou com deficiência. Um modelo de rede de petri foi desenvolvido e um protótipo de laboratório foi construído e testado usando diferentes condições meteorológicas predefinidas. As métricas tempo de resposta e porcentagem de erro foram observadas. Embora alguns dos trabalhos citados tenham observado a comunicação de mensagens para ambientes externos, os autores não destacam como a comunicação é realizada, nem focam no processamento remoto. Nossa proposta explora a IoT e as camadas de borda e névoa como recursos complementares que ajudam a otimizar o processamento de dados.

Os demais trabalhos relacionados possuem contexto de aplicação em edifícios inteligentes (WANG; SECHILARIU; LOCMENT, 2013; FANTI; MANGINI; ROCCOTELLI, 2014; FANTI; MANGINI; ROCCOTELLI, 2018). Wang, Sechilariu e Locment (2013) apresentam uma modelagem usando redes de Petri, para gerenciamento de energia de um

sistema de energia de múltiplas fontes. A estratégia de controle visa gerenciar o fluxo de energia através da carga em relação à sua demanda e às restrições da rede pública. [Fanti, Mangini e Roccotelli \(2014\)](#) monitoraram o consumo de energia do distrito e envia sinais de redução de demanda para os edifícios inteligentes. A estratégia de gerenciamento do edifício é descrita em uma abordagem de resposta à demanda e a unidade de controle é modelada em uma estrutura de rede de petri. [Fanti, Mangini e Roccotelli \(2018\)](#) analisaram dispositivos domésticos inteligentes e realizam análise de eficiência energética, buscando obter uma redução no consumo de energia dos dispositivos do edifício Um controlador para gerenciar o uso da carga é projetado em uma estrutura de rede de Petri. Um estudo de caso real valida e testa a eficácia do simulador aplicado aos aparelhos considerados. Este trabalho também possui contexto de aplicação em edifício inteligente, porém os trabalhos citados acima estão focados em garantir a eficiência energética, enquanto nosso foco está na utilização dos recursos computacionais.

Entre os oito trabalhos relacionados, apenas um analisou uma das métricas utilizadas neste trabalho. [Ajao et al. \(2020\)](#) observaram o tempo de resposta, principal métrica estudada nesta proposta de dissertação. No entanto, apenas este trabalho analisa as seguintes métricas: taxa de descarte, rendimento, uso de recursos computacionais e o número de trabalhos no sistema. Essas métricas são importantes para analisar se o sistema está funcionando corretamente. O tempo médio de resposta é importante para verificar se a configuração resulta em um tempo de transmissão e processamento satisfatório. A taxa de descarte permite observar a quantidade de requisições que são descartadas de acordo com a configuração da rede. A vazão mostra a taxa de tráfego no sistema. A utilização de recursos computacionais permite observar a configuração necessária para atender às necessidades do sistema, evitando a sobrecarga ou ociosidade de recursos. O número de trabalhos no sistema mostra o número de solicitações nas filas do sistema. Além disso, este trabalho não se limita a avaliar sensores ou condições ambientais específicas. O modelo proposto é configurável para qualquer tipo de dado coletado por um conjunto de sensores.

Três parâmetros utilizados para a comparação dos trabalhos aumentam significativamente a contribuição deste estudo, pois são características encontradas apenas neste trabalho. **Análise da capacidade do recurso** refere-se a avaliar como o sistema se comporta de acordo com o número de recursos disponíveis. Essa análise permite prever se o sistema atenderá às solicitações de forma satisfatória, além de evitar o desperdício de recursos computacionais. Neste trabalho, o comportamento do sistema alterando o número de nós na névoa é analisado. **Sensores agrupados por localização** refere-se a como o modelo representa diferentes conjuntos de sensores. Nosso modelo permite atribuir diferentes taxas de chegada dependendo da localização. Em nosso modelo, esses locais podem ser vistos como salas de um edifício. Esse recurso visa tornar o modelo mais realista, pois, dependendo da localização, os dados gerados podem ser diferentes. Nosso modelo também é o único que representa o **número de núcleos de processamento**

por máquina. A camada de névoa possui máquinas multi-core. Quanto mais núcleos um nó possui, mais solicitações ele pode processar em paralelo. O modelo permite variar a capacidade de névoa alterando o número de máquinas e o número de núcleos em cada uma delas. Acreditamos que esse recurso é fundamental para representar arquiteturas com precisão.

4 Desenvolvimento do modelo de filas

Este capítulo apresenta como foi feito o desenvolvimento do modelo e filas, explicando a metodologia aplicada, o cenário considerado, e por fim apresenta o modelo de filas que foi desenvolvido para avaliações de desempenho.

4.1 Metodologia

O objetivo principal deste trabalho é desenvolver um modelo de enfileiramento que seja capaz de avaliar sistemas IoT para edifícios inteligentes suportados pelas camadas de borda e névoa. Para a avaliação do modelo, alguns cenários foram construídos e, a partir deles, foram desenvolvidas uma análise de sensibilidade e três análises numéricas para determinar quais fatores mais impactam os resultados das métricas estudadas. A Figura 6 apresenta um fluxograma que resume a estratégia usada neste trabalho.

Entendimento da Aplicação: É importante entender o funcionamento da aplicação, definindo quantos e quais são os componentes envolvidos, e o fluxo de dados do sistema, por exemplo, para onde os dados serão enviados após passar pelo componente 'x'.

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: MRT, utilização de recursos, taxa de descarte e vazão.

Definição dos Parâmetros: Os parâmetros que serão inseridos no modelo são definidos. Esses parâmetros definem o comportamento e capacidade de recursos de cada componente. Neste trabalho, os parâmetros adicionados foram o número de núcleos, número de nós, taxa de serviço e tamanho da fila.

Geração do Modelo Analítico: Um modelo de desempenho usando modelo de filas é desenvolvido. Nesta parte, o modelo é construído considerando as métricas e parâmetros definidos, e os resultados esperados. A escolha do modelo de filas se dá porque o cenário de edifícios inteligentes considerado possui poucos componentes e o modelo de filas atende satisfatoriamente à sistemas de baixa complexidade.

Validação do Modelo: Implementação da validação do modelo utilizando linguagem de programação considerando os componentes inseridos no modelo. Os resultados que forem coletados na validação são comparados com os resultados retornados pelo modelo, se forem valores semelhantes o modelo está validado, do contrário haverá necessidade de ajuste no modelo. Se após a validação for detectada a necessidade de realizar ajuste no

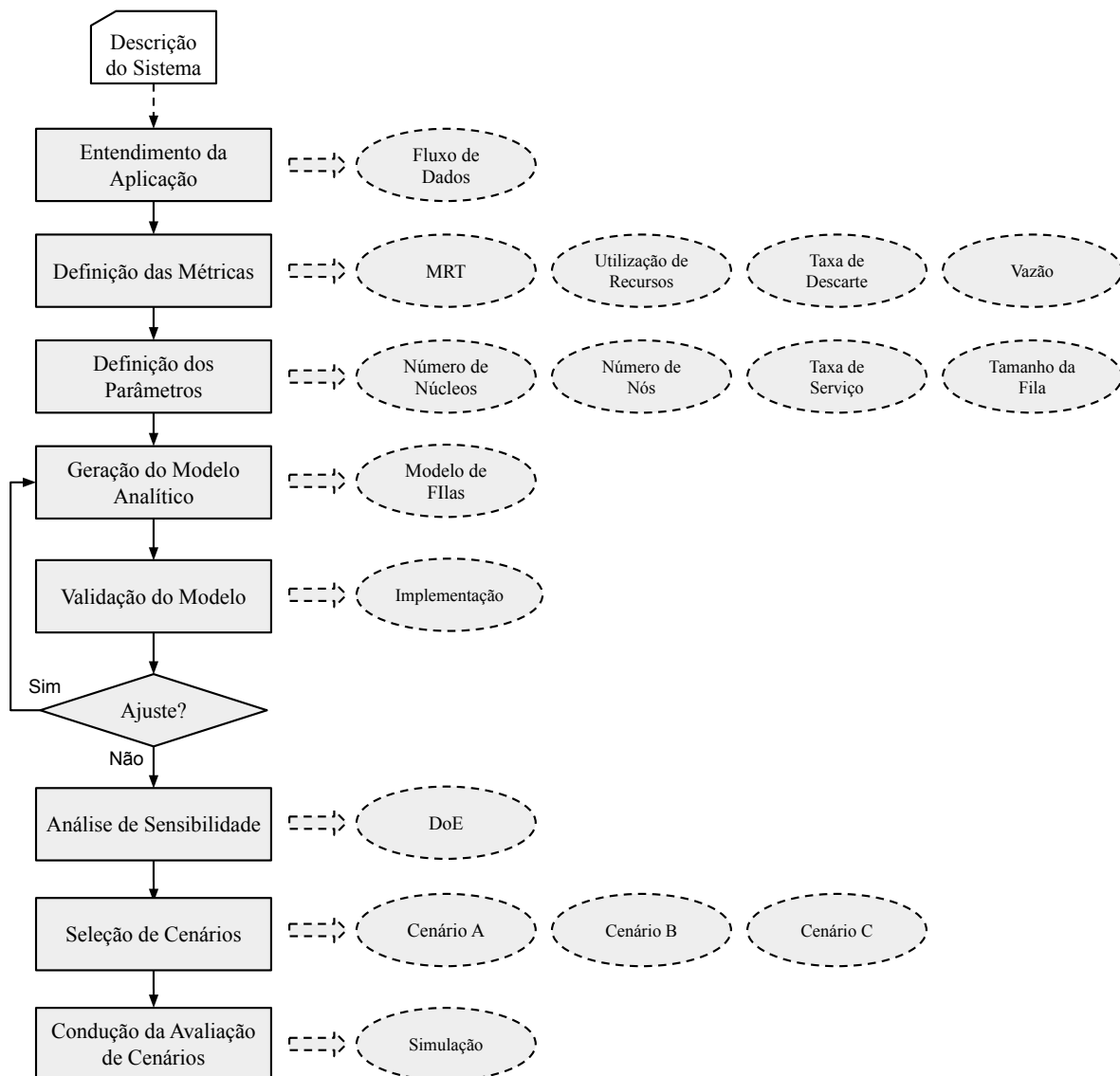


Figura 6 – Metodologia de trabalho deste projeto de avaliação de edifícios inteligentes.

modelo, deve-se retornar para a etapa de geração do modelo analítico.

Análise de Sensibilidade: Usando DoE, a análise apresenta resultados considerando fatores e níveis predefinidos. A partir disso, é possível identificar os fatores mais relevantes para os resultados das métricas escolhidas, e o modo como a interação entre os fatores e as variações dos seus níveis impactam no desempenho.

Seleção de Cenários: Alguns cenários são construídos para as análises de desempenho. Nesta parte, serão definidos quais cenários podem representar a realidade de um edifício inteligente. Os cenários serão escolhidos analisando os fatores mais importantes considerando os resultados da análise de sensibilidade.

Condução da Avaliação de Cenários: Os cenários construídos são avaliados usando o modelo de filas, por meio de simulação. Em cada cenário os fatores terão seus

níveis variados e as métricas escolhidas serão analisadas, permitindo perceber com quais configurações o sistema possui desempenho satisfatório.

4.2 Cenário do edifício inteligente

Esta seção apresenta o cenário considerado para monitoramento de conforto climático em edifícios inteligentes. O cenário é discutido considerando três aspectos: visão geral do sistema, ciclo de vida da mensagem e suposições.

Visão geral do sistema - A Figura 7 ilustra o cenário do sistema. Os sensores podem ser diferentes e ter finalidades diferentes, mas exemplificamos alguns tipos possíveis de sensores como ilustração. A temperatura mais adequada para o corpo humano em um ambiente interno é de aproximadamente 18°C a 24°C, e a umidade mais adequada é de aproximadamente 45% a 65% (SUNG; HSIAO, 2020). Buscando garantir o conforto climático, o edifício pode utilizar sensores para detectar temperatura e umidade em determinados locais, e ao identificar que as medidas se encontram fora do intervalo satisfatório, o gerente do edifício pode investigar a causa desse acontecimento.

O cenário é composto por duas camadas de computação: (i) camada de computação de borda em salas de edifícios, para integração de dados de sensores de temperatura e umidade; (ii) camada de computação de névoa em data centers para acesso de cliente interno (por exemplo, gerentes do edifício). A camada de computação de borda permite o monitoramento e a agregação de dados de climatização, usando sensores de temperatura e umidade para coletar e processar dados sobre o ambiente de cada sala periodicamente. A camada de computação de borda é projetada com vários nós para realizar a coleta e o processamento de dados em todas as salas de todos os andares do edifício. A camada de computação de névoa é composta por: um gateway borda-névoa para agrupamento de dados e balanceamento de carga entre as duas camadas; um conjunto de nós de névoa para processamento paralelo de dados; e uma estação onde os dados são disponibilizados para acesso do cliente interno.

Ciclo de vida da mensagem - O cenário também indica o ciclo de vida dos pacotes de dados e o comportamento operacional do sistema de monitoramento de conforto climático. Os dados são coletados periodicamente por detectores de temperatura e umidade e, em seguida, enviados para um dispositivo de borda para que sejam agrupados e encapsulados como um alerta. Se o dispositivo estiver ocupado, os dados poderão ser adicionados à fila, onde serão atendidas de acordo com a ordem de chegada. No entanto, se a fila do dispositivo de borda estiver completamente ocupada, os dados serão descartados. Esses alertas são então transmitidos para a névoa por meio de um gateway. O gateway borda-névoa desempenha um papel de gateway para distribuição de dados e balanceamento de carga para os nós de névoa. O balanceamento de carga é executado para que todos

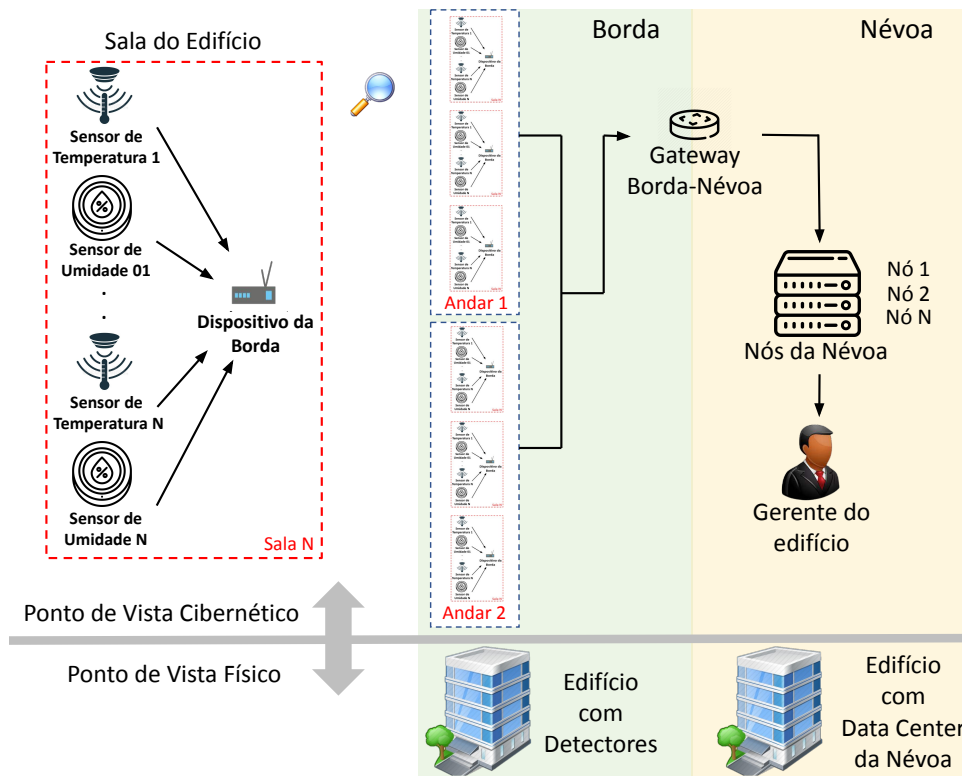


Figura 7 – Visão geral de um cenário que permite monitorar vários sensores em um edifício inteligente.

os nós da névoa recebam a mesma quantidade de solicitações de processamento, o que é importante para evitar sobrecarregar e enfileirar um nó enquanto outros nós estão disponíveis. As mensagens são processadas em nós de névoa por aplicativos especializados, que são personalizados para o tipo de alerta. Assim como acontece no dispositivo da borda, os nós da névoa também possuem um limite de requisições em fila, e caso esse limite seja atingido, os dados serão descartados. Os dados processados na camada de névoa são entregues diretamente aos clientes internos (gerentes do edifício). Diante disso, podem surgir questões como: “ qual é o impacto da taxa de chegada de solicitações nas métricas de desempenho de um sistema de edifício inteligente? ”; “ como uma configuração de capacidade de recursos específicos impacta nas métricas de desempenho de um sistema de edifício inteligente? ”; e “ quais recursos possuem maior impacto nas métricas de desempenho de um sistema de edifício inteligente? ”.

Suposições e Argumentos - Algumas suposições sobre o cenário em consideração são fornecidas abaixo para simplificar a modelagem.

- *Camada de borda*
 - [b1]: A geração de dados foi modelada para todos os sensores ativos em uma sala, conectados a um dispositivo de borda que também está instalado na sala.
 - [b2]: Não levamos em consideração a latência de comunicação entre sensores e

dispositivos de ponta. Na prática, a conexão é formada por comunicação sem fio. No entanto, simplificamos o impacto negativo da comunicação de curta distância na camada de ponta nas métricas de desempenho geral.

- [b3]: A latência de comunicação da conexão entre as camadas de borda e névoa é assumida como um atraso na propagação de dados de cada nó de borda para a camada de névoa.
- [b4]: A coleta de dados de cada sensor é independente uma da outra. Porém, a taxa de cada entrada é determinística, ou seja, com uma taxa fixa.
- Camada de névoa
 - [n1]: Não consideramos balanceamento de carga sofisticado na camada de névoa. Os trabalhos recebidos no gateway da névoa são distribuídos uniformemente para cada um dos nós na camada de névoa. Para simplificar a modelagem, não focamos no problema de balanceamento de carga.
 - [n2]: Consideramos nós com configurações iguais, mas o modelo permite ao avaliador configurar os nós de forma heterogênea.
- *Infraestrutura IoT*
 - [i1]: O desempenho das transações de dados entre sensores IoT e clientes internos (gerente do edifício) é o foco principal da modelagem. Portanto, o envolvimento de componentes físicos e sua disponibilidade operacional é minimizado. Não consideramos falhas de componentes e comportamentos de recuperação na modelagem de avaliação de desempenho.
 - [i2]: Nosso foco principal é (i) explorar o gargalo na transmissão de dados de temperatura e umidade e (ii) explorar o impacto da alteração da configuração da camada de névoa nas métricas de desempenho.

4.3 Modelo de Rede de Filas

Esta seção apresenta um modelo baseado na teoria das filas para o cenário apresentado, que é ilustrado na Figura 8. O modelo possui vários pontos de entrada e um ponto de saída. O *Java Modeling Tools* (JMT) foi utilizado para modelar e avaliar o cenário proposto. JMT é um kit de ferramentas de código aberto para analisar e avaliar o desempenho de sistemas de comunicação com base na teoria de filas (FISHMAN, 2013). O arquivo de modelo (na extensão .jsimg) pode ser baixado através do seguinte url: <<https://tinyurl.com/queuemodel>>. A Tabela 2 descreve todos os elementos do modelo.

O fluxo de dados no modelo ocorre da esquerda para a direita. As múltiplas entradas no modelo correspondem aos quartos do edifício.

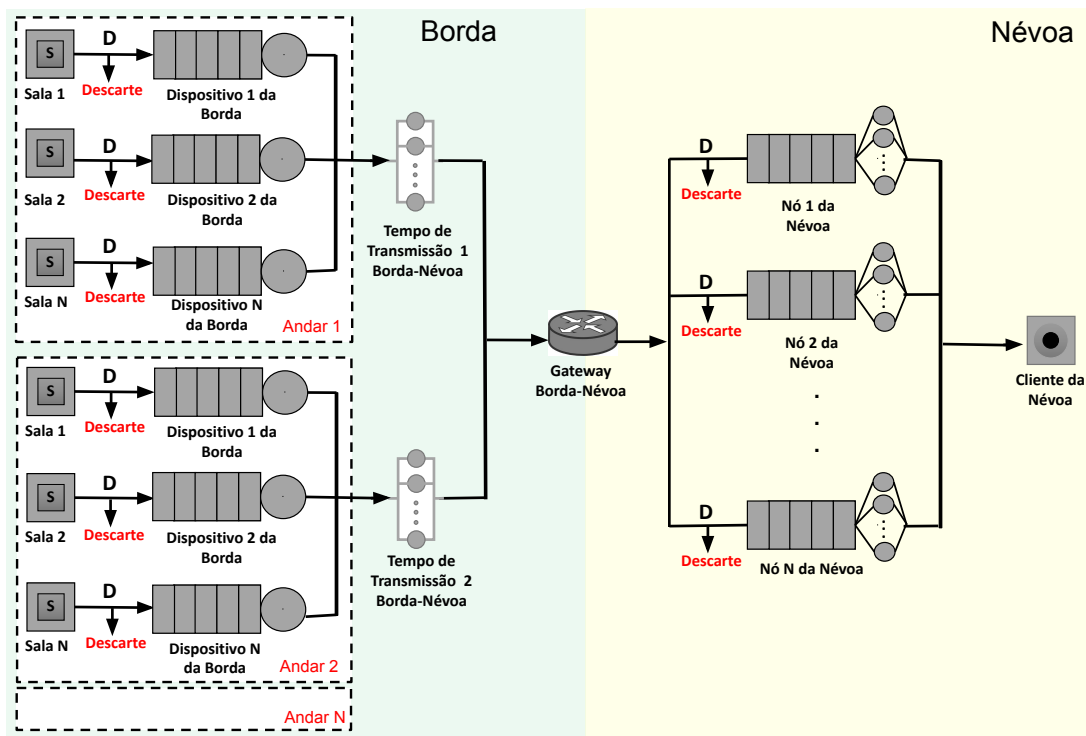


Figura 8 – Modelo de filas de uma cenário de edifício inteligente composto por múltiplas camadas.

Tabela 2 – Descrição dos componentes do modelo

Tipo	Elemento	Descrição
Processamento	Sala	Responsável pela geração dos dados.
	Dispositivo da Borda	Responsável pelo pré-processamento dos dados.
	Gateway Borda-Névoa	Responsável pelo encaminhamento dos dados da borda para a névoa.
Comunicação	Nó da Névoa	Salva os dados, processa e encaminha informações e/ou alertas para gerente do edifício.
	Tempo de Propagação Borda-Névoa	Atraso na propagação dos dados da borda.
Estado de Execução	Cliente da Névoa	Representa o fim do processamento dos dados e disponibilidade para gerente do edifício.

O componente Sala representa os sensores presentes em uma sala do edifício, que geram solicitações dentro de um intervalo de tempo predefinido e as encaminham para um dispositivo de borda que atua como um portal entre a borda e a camada de névoa. A taxa de chegada definida para o componente Sala dependerá do número de sensores e da distribuição de geração de dados. As salas têm n sensores, uma quantidade que pode variar dependendo do tamanho da sala, pois salas muito grandes podem precisar de vários sensores para cobrir toda a sua área. Quando a taxa de chegada é maior do que o sistema pode suportar, os dados são descartados. O dispositivo de borda é representado por uma fila e um servidor interno exclusivo, que pode ser configurado com um ou mais núcleos de processamento que executam tarefas de forma paralela. Além disso, é possível definir o

tempo de serviço do dispositivo de borda, ou seja, o tempo que a solicitação leva até ser enviada para o gateway. Considera-se que as salas do edifício estão organizadas em pisos. Cada andar tem uma certa distância da camada de névoa. Desta forma, há um atraso ("Tempo de propagação da névoa da borda") de cada andar para a camada de névoa. Os componentes do tempo de propagação não têm um serviço específico; ou seja, é apenas um componente que causa um atraso na propagação de uma solicitação, emulando um atraso de rede.

Na camada de névoa, há um gateway que é usado como ponto de entrada. Ao chegar ao gateway, as mensagens podem ser distribuídas seguindo uma estratégia de balanceamento de carga específica. Neste trabalho, na seção de simulações, consideramos a estratégia de distribuição igualitária. Pela notação Kendall (FERREIRA et al., 2011), a rede segue o padrão $D/M/c/K/FCFS$. A taxa de geração segue um padrão determinístico (D), pois os sensores são calibrados para um intervalo de geração fixo. No entanto, os tempos de serviço do nó seguem uma distribuição exponencial (BALAKRISHNAN, 2019). As estações de serviço têm um número (c) de servidores, que em nosso trabalho chamamos de "núcleos". As respectivas filas têm um tamanho fixo (K) e uma política de ordem de chegada (FCFS). Caso as filas dos nós estejam congestionadas, as requisições que chegarem serão descartadas. A névoa também tem uma estação *sink* ("Cliente da Névoa") correspondente ao local onde os gerentes do edifício podem acessar dados confidenciais em tempo real.

5 Resultados e Discussão

Este capítulo apresenta os resultados obtidos através da análise de sensibilidade realizada usando DoE, e os três cenários de simulação que foram analisados.

5.1 Análise de sensibilidade usando DoE

Esta seção descreve os experimentos realizados para analisar os fatores que podem influenciar o desempenho de um edifício inteligente. Também foi analisado como as mudanças em seus níveis impactam o desempenho do sistema. Os resultados obtidos são discutidos com base na métrica do tempo médio de resposta.

5.1.1 Desenho Experimental

A métrica do tempo médio de resposta (MRT) é analisada por meio do DoE. A escolha do MRT se deve ao seu impacto mais direto na percepção do usuário final. O nível de utilização de recursos, por exemplo, é uma métrica considerada do tipo secundário. Quatro fatores foram adotados neste estudo: taxa de serviço, número de nós, número de núcleos e tamanho da fila. Todos os fatores têm dois níveis, que foram escolhidos a partir de sucessivos testes com níveis de fatores mínimos inseridos no modelo, para garantir um controle sobre as análises. O fator de taxa de serviço se refere à taxa que o servidor leva para executar uma solicitação e seus níveis são 0,033msg/ms e 0,044msg/ms. O número de nós refere-se ao número de servidores na névoa, definido como 5 e 10. O número de núcleos é definido pelo número de núcleos de processamento disponível em cada servidor da névoa, definido como 2 e 4. O tamanho da fila se refere ao número de solicitações que serão adicionadas à fila do servidor. Seus dois níveis são 50 e 100. A Tabela 3 resume os fatores e níveis escolhidos para realizar o DoE usando a métrica MRT.

Tabela 3 – Fatores e níveis do DoE

Fatores	Nível 1	Nível 2
Taxa de Serviço	0.033	0.044
Número de Nós	2	10
Número de Núcleos	4	8
Tamanho da Fila	50	100

Eles devem ser combinados para definir como os experimentos devem ser realizados com todos os fatores e níveis definidos. A Tabela 4 apresenta as combinações possíveis entre os fatores e níveis escolhidos.

Tabela 4 – Combinações de fatores e níveis do DoE

Nº	Taxa de Serviço	Número de Nós	Número de Núcleos	Tamanho da Fila
1	0.033	2	4	50
2	0.033	2	4	100
3	0.033	2	8	50
4	0.033	2	8	100
5	0.033	10	4	50
6	0.033	10	4	100
7	0.033	10	8	50
8	0.033	10	8	100
9	0.044	2	4	50
10	0.044	2	4	100
11	0.044	2	8	50
12	0.044	2	8	100
13	0.044	10	4	50
14	0.044	10	4	100
15	0.044	10	8	50
16	0.044	10	8	100

5.1.2 Resultados do DoE

O gráfico de Pareto determina a magnitude e a importância dos fatores. A Figura 9 apresenta o gráfico de Pareto para os fatores relacionados à métrica MRT. Quando um fator tem alto impacto nos testes, significa que valores muito diferentes são obtidos ao alterar seu nível. As barras que cruzam a linha de referência vermelha (efeito 366,1) são consideradas estatisticamente significativas. Esses fatores são estatisticamente significativos considerando a confiança estatística de 95% com os termos do modelo atual. O fator número de núcleos é o de maior relevância entre os fatores deste estudo. Portanto, o número de núcleos por nós de névoa é decisivo na eficiência de monitoramento do edifício. O fator número de nós também tem alta relevância. O tamanho da fila e a taxa de serviço mostraram-se muito menos influentes. Como o gráfico de Pareto exhibe o valor absoluto dos efeitos, você pode determinar quais efeitos são grandes, mas não pode determinar quais efeitos aumentam ou diminuem o tempo de resposta.

A Figura 10 apresenta o gráfico dos principais efeitos da métrica MRT. O gráfico divide os valores médios resultantes para cada nível. Quanto mais horizontal a linha, menos influência esse fator tem, pois significa que os diferentes níveis do fator influenciam de forma semelhante o resultado. Todos os níveis de fator interferem com a métrica MRT de alguma forma. Os fatores número de nós e número de núcleos têm o maior efeito. Em relação ao fator número de nós, com 5 nós, obteve-se o maior tempo médio de resposta (485ms), enquanto com 10 nós esse tempo foi bem menor (53ms). Portanto, o edifício será mais eficiente usando 10 nós na névoa. Em relação ao fator número de núcleos, pode-se observar que o MRT é muito maior quando se usa 2 núcleos em vez de 4. Ao usar 4 núcleos,

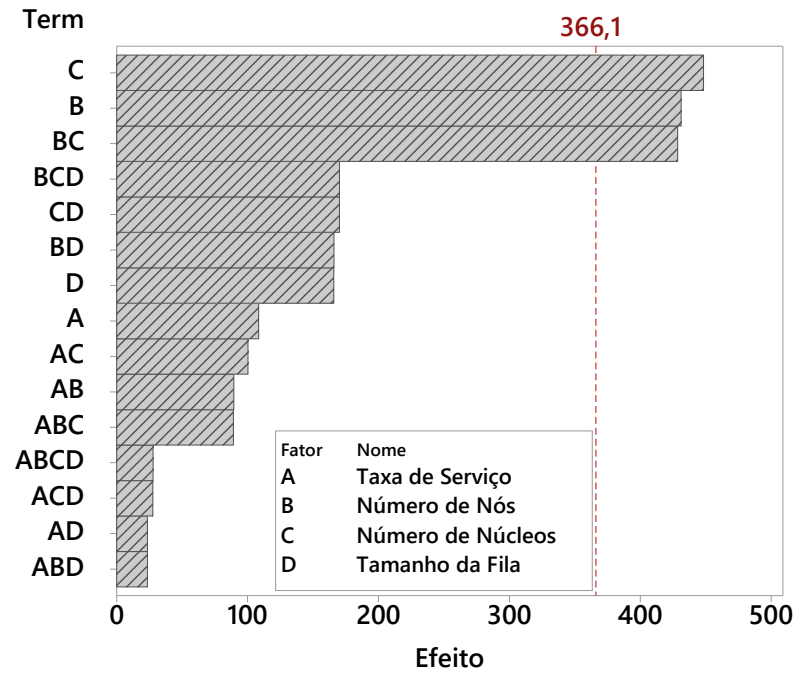


Figura 9 – Influência dos fatores na métrica MRT.

a velocidade de processamento é duplicada. Assim, o MRT é drasticamente reduzido.

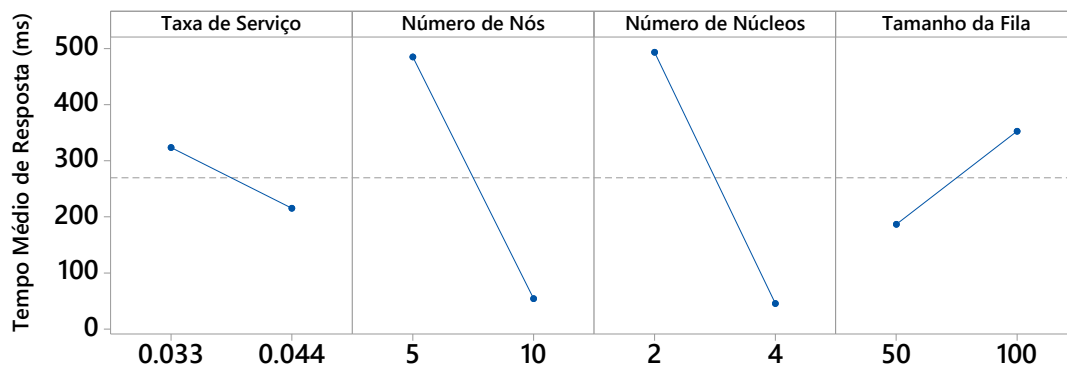


Figura 10 – Efeitos Principais para MRT.

A Figura 11 mostra as interações para cada combinação possível de fatores. Dois fatores interagem entre si se o efeito depender da variação nos efeitos um do outro. Existe uma interação entre todos os fatores, embora a variação dos efeitos seja baixa em alguns casos. Na interação entre a taxa de serviço e o número de nós, a maior variação da taxa ocorre quando o número de nós é igual a 5. Quando a taxa de serviço é igual a 0,033msg/ms, o MRT atinge 584ms. Com uma taxa de serviço de 0,044msg/ms, o MRT é igual a 386ms. Na interação entre a taxa de serviço e o número de núcleos, a maior variação da taxa ocorre quando o número de núcleos é igual a 2. O MRT é igual a 598ms quando a taxa de serviço atinge 0,033msg/ms e 389ms quando a taxa é 0,044msg/ms. A interação entre a taxa de serviço e o tamanho da fila é relativamente baixa (em comparação com outras interações), sem variação no MRT ao alterar os níveis desses fatores. Na interação entre o

número de nós e núcleos, a maior variação ocorre quando o número de núcleos é igual a 2. Com 5 nós na névoa, o MRT chega a 923ms. Com 10 nós, o MRT é igual a 63ms. Na interação entre o número de nós e o tamanho da fila, a maior variação ocorre com um tamanho de fila igual a 100. O MRT atinge 651ms para 5 nós, e com 10 nós, o MRT é igual a 53ms. Na interação entre o número de núcleos e o tamanho da fila, a maior variação também ocorre quando o tamanho da fila é igual a 100. Com esta configuração, o MRT atinge 661ms quando os nós têm 2 núcleos. Com 4 núcleos, o MRT é igual a 43ms.

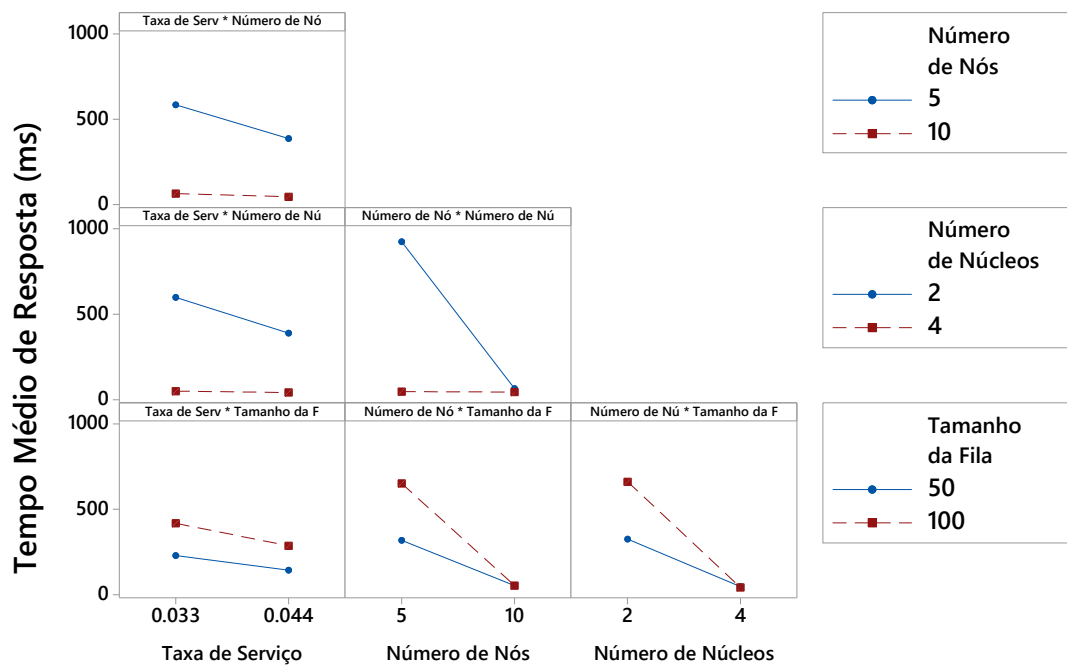


Figura 11 – Interações de fatores considerando o MRT.

5.2 Resultados das Simulações

Esta seção apresenta as simulações realizadas no modelo de filas proposto considerando a variação nos dois fatores que mais influenciaram o desempenho do sistema de monitoramento predial. Esses fatores foram definidos com base no DoE realizado anteriormente, e são o número de nós na névoa e o número de núcleos em cada nó. A Tabela 5 apresenta a configuração utilizada para os experimentos. A tag **X** indica que o componente não tem definição de capacidade de fila. A coluna de tempo representa o tempo de serviço para os componentes da fila. O tempo de propagação de componentes representa o tempo de comunicação entre uma camada e outra. Nesta seção, três cenários são apresentados: O cenário A apresenta uma variação no número de nós; O cenário B apresenta uma variação no número de núcleos; e O cenário C examina os dois fatores juntos.

Tabela 5 – Parâmetros Inseridos no Modelo

Tipo do componente	Componente	Tempo (ms)	Tamanho da Fila
Máquina com Fila	Dispositivos da Borda	5	50
	Nós da névoa	30	50
Tempo de Propagação	Borda-névoa 1	6	X
	Borda-névoa 2	12	X

5.2.1 Cenário A: Variando o número de núcleos

Nesse cenário, foi variado o fator número de núcleos, que obteve a maior relevância segundo o DoE. O número de núcleos na névoa foi variado em 1, 2, 4 e 8 núcleos, enquanto o número de núcleos dos nós de borda permaneceu fixo (1 núcleo). A quantidade de nós de borda e névoa foi fixada em 6 e 5 nós, respectivamente.

A Figura 12 apresenta os resultados considerando diferentes números de núcleos nos nós de névoa: 1, 2, 4 e 8. A Figura 12(a) mostra o MRT de todo o sistema. Quanto maior a quantidade de recursos, menor tende a ser o MRT, mas esse padrão nem sempre é aparente. Assim, quando existe apenas 1 núcleo, o MRT é muito maior do que para as outras configurações, mesmo para valores de baixa taxa de chegada. Com 1 núcleo, nota-se que o crescimento permanece estagnado entre 1400ms e 1500ms. Essa estagnação ocorre devido à alta utilização de recursos com pouco poder de processamento (ver Figura 12(b)). A estagnação também ocorre com 2 núcleos, mas apenas a partir de $AR = 0,062\text{msg/ms}$, atingindo um MRT de aproximadamente 700ms. Com 4 e 8 núcleos, os MRTs obtidos são muito próximos. Essa proximidade sugere que talvez seja mais vantajoso e econômico usar apenas 4 núcleos. Considerando o menor AR ($0,04\text{msg/ms}$), temos um MRT de 1434ms, 77ms, 46ms e 45ms para 1, 2, 4 e 8 núcleos, nessa ordem. Assumindo o MRT desejado em torno de 100ms, as configurações que atendem a essa restrição são aquelas com 2 ou mais núcleos. Considerando o ponto extremo ($AR = 0,08\text{msg/ms}$), existe um MRT de 1492ms, 731ms, 59ms e 47ms para 1, 2, 4 e 8 núcleos, respectivamente. Assumindo um Acordo de Nível de Serviço (SLA) de 100ms, essa restrição de tempo seria satisfeita com 4 ou 8 núcleos.

A Figura 12(b) mostra a utilização da névoa. Quanto maior o número de núcleos, menor tende a ser a taxa de utilização dos recursos. Assim, o uso com 1 núcleo é muito maior do que com 8 núcleos, mesmo com valores baixos de AR. Quando os nós de névoa têm apenas 1 núcleo, a utilização é 100% para todas as taxas de chegada analisadas. Quando há 2 núcleos, a utilização para $AR = 0,04\text{msg/ms}$ é de aproximadamente 70% e atinge 100% AR igual ou superior a $0,058\text{msg/ms}$. Com 4 e 8 núcleos, a utilização cresce conforme a taxa de chegada aumenta, mas não esgota os recursos disponíveis. Considerando o ponto de menor demanda ($AR = 0,04\text{msg/ms}$), temos uma taxa de utilização de 100%, 70%, 36% e 17% para 1, 2, 3 e 4 núcleos, nesta ordem. Assumindo uma premissa de SLA onde

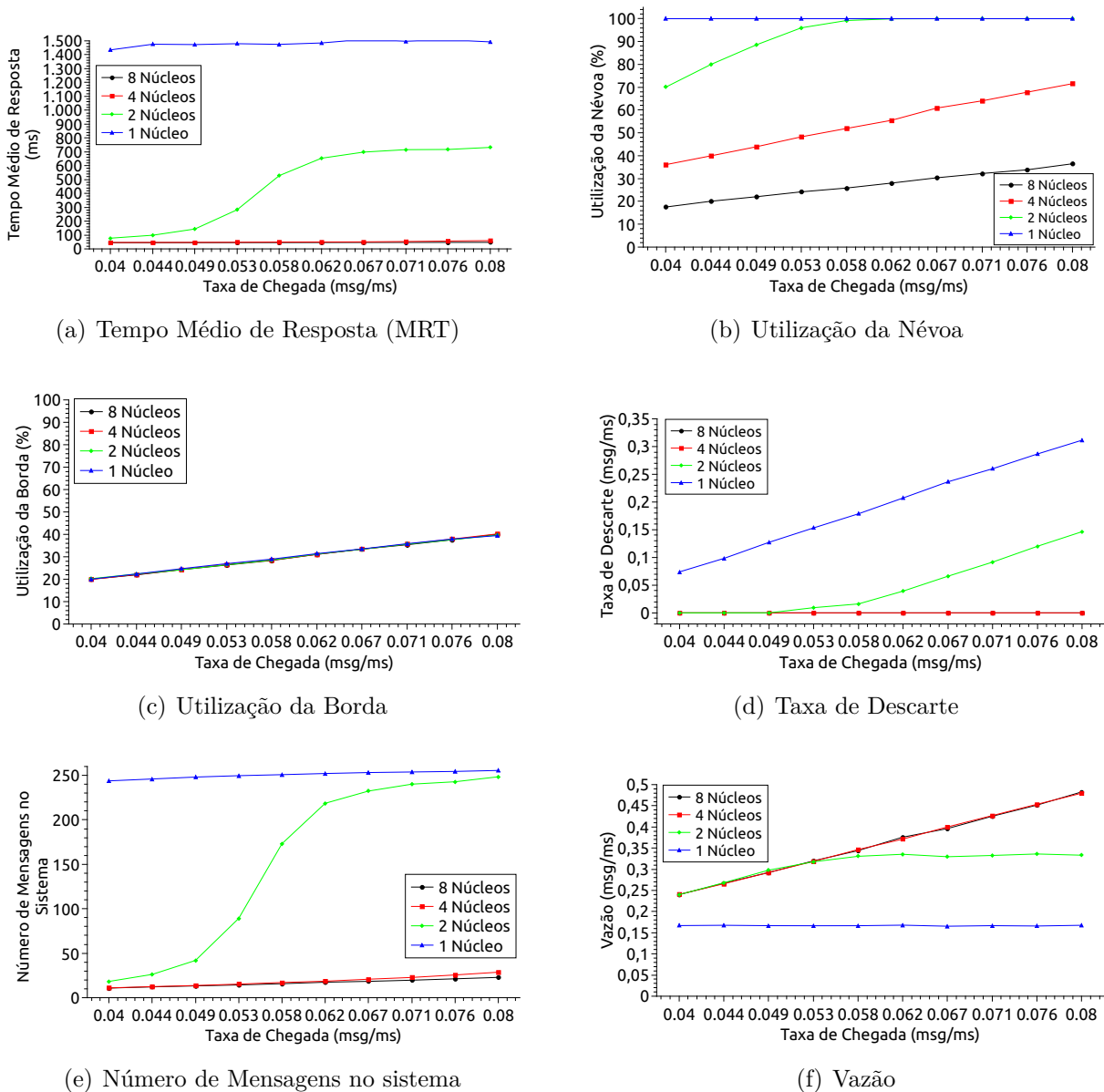


Figura 12 – Resultados da análise do modelo considerando diferentes números de núcleos.

a utilização máxima de 80% é aceita, tal restrição pode ser atendida quando os nós de névoa principais são configurados com 2 ou mais núcleos. Considerando o ponto extremo ($AR = 0,08\text{mg/ms}$), observa-se uma utilização de 100%, 100%, 71% e 36% para 1, 2, 4 e 8 núcleos, respectivamente. Com uma suposição de SLA de 80%, esta opção pode ser atendida com 4 ou 8 núcleos em cada nó de névoa.

A Figura 12(c) mostra a utilização da borda. Conforme a taxa de chegada aumenta, a taxa de utilização da borda também aumenta. No entanto, alterar o número de núcleos em nós de borda não influencia o resultado porque a borda não depende do processamento em nós de névoa para realizar suas tarefas. Independentemente da premissa do SLA na utilização da borda, é possível escolher qualquer número de nós na névoa. O número de nós de névoa não interfere na utilização da borda.

A Figura 12(d) mostra a taxa de descarte. A taxa de descarte tende a diminuir à medida que o número de núcleos aumenta. Assim, configurações com 4 ou mais núcleos ficam bem abaixo das demais quando a taxa de chegada é alta. Com 1 núcleo, o sistema tem uma taxa de descarte de pelo menos 0,1msg/ms para AR baixo, e a taxa aumenta conforme o AR cresce. Com 2 núcleos, não há descarte de mensagem até $AR = 0,049\text{msg/ms}$, mas com taxas de chegada mais altas, ele pode descartar aproximadamente 0,15msg/ms. O sistema não descarta nenhuma mensagem para configurações com 4 e 8 núcleos, mesmo com $AR = 0,08\text{msg/ms}$. Considerando o menor AR (0,04msg/ms), o descarte existe apenas quando o sistema estiver configurado com 1 núcleo, com taxa de descarte igual a 0,07msg/ms. Presumindo que o SLA não exija a eliminação de mensagens, a configuração precisa ter 2 ou mais núcleos em cada nó na névoa. Considerando $AR = 0,08\text{msg/ms}$, há um descarte de 0,34msg/ms e 0,14msg/ms para 2 e 4 nós, e nenhum descarte com 4 e 8 núcleos. Para o SLA onde não deve haver descarte, as configurações que atendem a esse requisito possuem 4 ou 8 núcleos.

A Figura 12(e) exibe o número de mensagens no sistema. Observe que as linhas do gráfico têm curvas semelhantes às do MRT (ver Figura 12(a)). Quanto maior o número de mensagens no sistema, mais longa será a fila e mais tempo será atendido. Na configuração de 1 núcleo, o sistema fica sempre com aproximadamente 250 mensagens, independente da taxa de recebimento. Quando há 2 núcleos, o número de mensagens é menor que 50 até $AR = 0,049\text{msg/ms}$. Para configurações de 4 e 8 núcleos, o número de mensagens é inferior a 50 para todas as taxas de chegada.

A Figura 12(f) mostra a vazão do sistema. A vazão deve aumentar à medida que a taxa de chegada aumenta, pois mais mensagens chegam. Pode-se observar tal comportamento em configurações com 2, 4 e 8 núcleos. No entanto, um ponto de inflexão com 2 núcleos e $AR = 0,58\text{msg/ms}$ é observado quando o fluxo começa a se estabilizar. A configuração de 1 núcleo tem uma vazão muito menor do que as outras e permaneceu estável para todas as taxas de chegada. A estabilidade acontece porque o sistema fica sobrecarregado e começa a perder mensagens (veja a Figura 12(d)). Considerando $AR = 0,04\text{msg/ms}$, a vazão é 0,16, 0,23, 0,24 e 0,24 para 1, 2, 4 e 8 núcleos, nessa ordem. Assumindo uma premissa de SLA em que uma vazão maior que 0,2msg/ms é necessária, é possível atender a essa restrição com 2 ou mais núcleos. Considerando $AR = 0,08\text{msg/ms}$, a vazão é 0,16, 0,33, 0,47 e 0,48 para 1, 2, 4 e 8 núcleos, respectivamente. Considerando um SLA de vazão mínima de 0,04msg/ms, as configurações que atendem a esse requisito são aquelas que têm 4 ou 8 núcleos em cada nó na névoa.

Em conclusão, a borda não é afetada pela mudança no número de núcleos na névoa. A borda não depende do processamento de névoa para funcionar. Para métricas MRT, o sistema se comporta de forma estável em todas as configurações de névoa, exceto com 2 núcleos. Com esta configuração, a partir de $AR = 0,053\text{msg/ms}$, o MRT aumenta

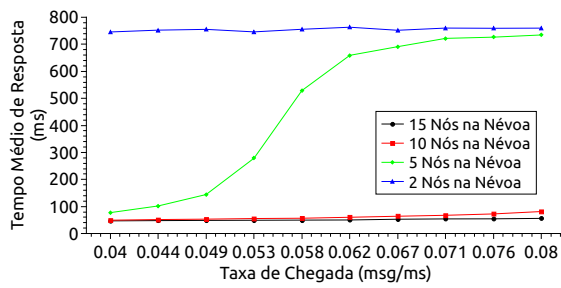
porque os recursos de névoa também crescem, chegando perto de 100%. O aumento do MRT também aumenta o número de mensagens no sistema, tornando os gráficos das duas métricas semelhantes. Além disso, a taxa de descarte só existe quando os nós de névoa têm apenas 1 ou 2 núcleos.

5.2.2 Cenário B: Variação no número de nós na névoa

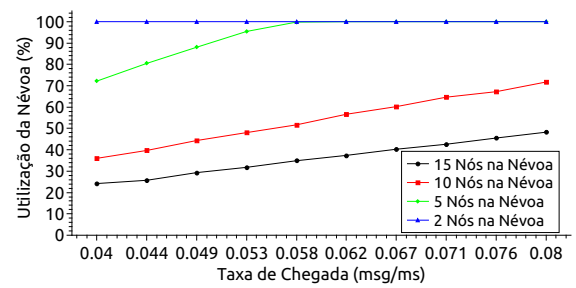
Nesse cenário, o segundo fator mais relevante no DoE foi variado: o número de nós na névoa. O número de nós na névoa foi variado em 2, 5, 10 e 15 nós, enquanto o número de nós na borda permaneceu fixo (6 nós). Cada nó de borda tem 1 núcleo de processamento, enquanto os nós de névoa têm 2 núcleos.

A Figura 13 apresenta os resultados considerando um número diferente de nós na névoa. A Figura 13(a) mostra o MRT de todo o sistema para as 4 variações: 2, 5, 10 e 15 nós de névoa. Espera-se que quanto maior o número de nós, menor o MRT, pois o sistema realiza o balanceamento de carga. Esse comportamento pode ser visto nas configurações observadas nesta análise. No entanto, com 10 ou mais nós na névoa, a diferença no MRT pode não ser significativa. Somente quando os recursos foram reduzidos para 5 e 2 nós, o MRT foi muito maior devido ao AR, atingindo 734ms e 749ms, respectivamente. A proximidade do MRT para 10 e 15 nós sugere que pode ser mais vantajoso e econômico usar apenas 10 nós na névoa. Em relação ao crescimento da taxa de chegada, observa-se que o AR tem maior impacto quando há 5 nós de névoa. O MRT tem um crescimento pouco significativo para 2, 10 e 15 nós conforme a carga de trabalho aumenta. Porém, com 2 nós, o MRT está sempre acima de 700ms, independente da taxa de chegada. Um padrão observado nos quatro cenários é sobre a estagnação do crescimento do MRT em função do AR. Em todas as configurações, existe uma certa estabilidade no MRT. No entanto, para 5 nós, isso só pode ser observado com AR de 0,071msg/ms. O intervalo $AR = [0,071\text{msg/ms} - 0,08\text{msg/ms}]$ pode ser usado para oferecer aos clientes de infraestrutura um MRT médio de 700ms. Para 15 nós, essa média seria de 55ms. Considerando o menor AR ($AR = 0,04\text{msg/ms}$), há um MRT de 744ms, 77ms, 49ms e 46ms para 2, 5, 10 e 15 nós, respectivamente. Considerando uma suposição de SLA que relata um requisito de $MRT > 100$ ms, tal restrição de tempo seria satisfeita com configurações de 5 nós na névoa. Considerando o ponto extremo, com a maior demanda ($AR = 0,08\text{msg/ms}$), existe um MRT de 759ms, 734ms, 81ms e 55ms para 2, 5, 10 e 15 nós, respectivamente. Para uma suposição de SLA de 100ms, essa restrição de tempo só será atendida a partir de 10 nós de névoa.

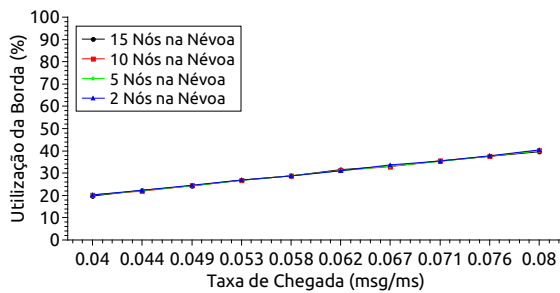
A Figura 13(b) mostra a taxa de utilização da névoa. Quanto maior o número de nós na névoa, menor tende a ser o uso de recursos. Assim, a utilização com 2 núcleos é muito maior do que com 5, 10 e 15 nós. Um padrão observado em todos os cenários, exceto para 2 nós, é que a utilização aumenta à medida que a taxa de chegada aumenta.



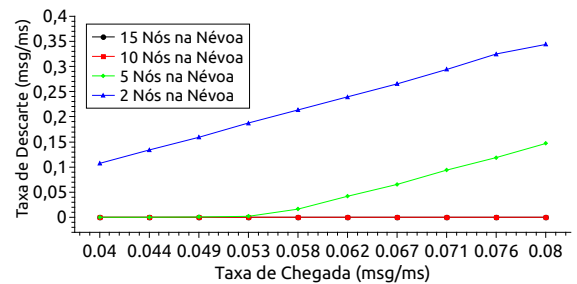
(a) Tempo Médio de Resposta (MRT)



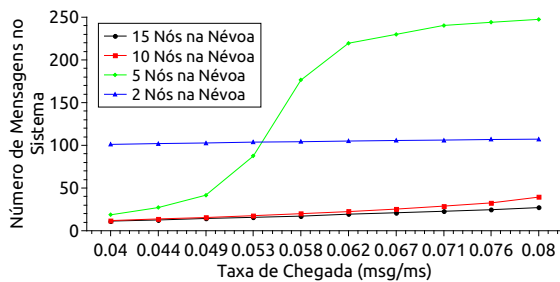
(b) Utilização da Névoa



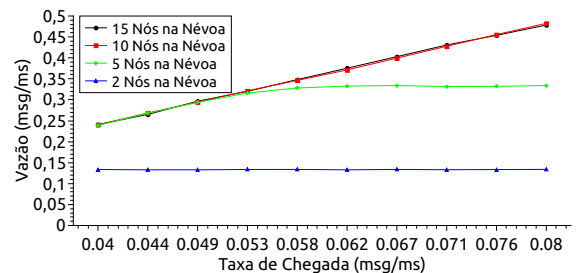
(c) Utilização da Borda



(d) Taxa de Descarte



(e) Número de Mensagens no sistema



(f) Vazão

Figura 13 – Resultados da análise do modelo considerando diferentes números de nós da névoa.

Com 2 núcleos, o crescimento do uso permanece estável em 100% para todas as taxas de chegada. Quando há 5 nós, a utilização cresce até atingir 100%, com AR igual ou superior a 0,058msg/ms. Com 10 e 15 nós, a maior utilização alcançada é 71% e 48%, respectivamente. Considerando o menor AR (0,04msg/ms) há uma utilização de 100%, 72%, 36% e 24% para 2, 5, 10 e 15 nós, nessa ordem. Um SLA que considera uma utilização de até 80% pode ser atendido com configurações com 5 ou mais nós. Considerando o ponto de maior demanda (AR = 0,08msg/ms), a taxa de utilização é de 100%, 100%, 71% e 48%, respectivamente. Para a premissa de SLA de uso de até 80%, essa restrição pode ser atendida a partir de 10 nós na névoa.

A Figura 13(c) mostra a taxa de utilização da borda. Como esperado, conforme a taxa de chegada aumenta, a taxa de utilização da borda também aumenta. No entanto, alterar o número de nós na névoa não influencia muito o resultado. A borda não depende

do processamento de nós de névoa para realizar suas tarefas, pois essa camada vem antes da camada de névoa. Independentemente da preferência pelo uso da borda, o projetista do sistema pode escolher qualquer número de nós na névoa, pois isso não interfere no uso dos recursos da borda.

A Figura 13(d) mostra a taxa de descarte de mensagens. Quanto maior a capacidade do sistema, menor tende a ser a taxa de descarte. Assim, para configurações com 10 nós ou mais, as taxas de descarte ficam bem abaixo das demais quando a taxa de chegada é alta. O sistema não descarta nenhuma mensagem para configurações com 10 ou mais nós, mesmo com $AR = 0,08\text{msg/ms}$. Com 5 nós, não há descarte de mensagem até $AR = 0,053\text{msg/ms}$, mas com taxas de chegada mais altas, pode-se descartar mais de $0,1\text{msg/ms}$. Com 2 nós, o sistema tem uma taxa de descarte de pelo menos $0,1\text{msg/ms}$ para AR baixo e a taxa aumenta com o crescimento de AR . Considerando o menor AR ($0,04\text{msg/ms}$), descarte somente quando o sistema estiver configurado com 2 nós de névoa, com taxa de descarte igual a $0,1\text{msg/ms}$. Presumindo que um SLA não exija descarte, a configuração deve ter 5 ou mais nós na névoa. Considerando o ponto de maior demanda ($AR = 0,08\text{msg/ms}$), há um descarte de $0,34\text{msg/ms}$ e $0,14\text{msg/ms}$ para 2 e 4 nós, nessa ordem, sem descarte para 10 e 15 nós. Imaginando o SLA onde não deve haver descarte, as configurações que atendem a esse requisito passam a ter 10 ou mais nós.

A Figura 13(e) exhibe o número de mensagens (solicitações) em estado de processamento dentro do sistema. As linhas do gráfico têm curvas semelhantes às do gráfico MRT (ver Figura 13(a)). Essa semelhança ocorre porque a quantidade de mensagens no sistema está relacionada ao MRT. Quando o MRT aumenta, haverá mais mensagens no sistema porque as filas ficam mais longas. No entanto, com 2 nós na névoa, o MRT observado é maior do que com a configuração de 5 nós. Ainda assim, para a configuração com 2 nós, há menos mensagens no sistema a partir de $AR = 0,58\text{msg/ms}$. O baixo número de mensagens com 2 nós deve-se à maior taxa de descarte que as demais configurações, pois possui menor capacidade de processamento e menos filas (ver Figura 13(d)).

A Figura 13(f) mostra a vazão do sistema. Espera-se que a vazão seja diretamente proporcional à taxa de chegada. O mesmo comportamento pode ser observado com configurações incluindo 5, 10 e 15 nós. Porém, há um ponto de inflexão com 5 nós e $AR = 0,58\text{msg/ms}$ quando a taxa começa a se estabilizar. A configuração de 2 nós tem uma vazão muito menor do que as outras e permaneceu estável para todas as taxas de chegada. A estabilidade acontece porque o sistema fica sobrecarregado e começa a perder mensagens (veja a Figura 13(d)). Considerando $AR = 0,04\text{msg/ms}$, a vazão é $0,13\text{msg/ms}$, $0,23\text{msg/ms}$, $0,24\text{msg/ms}$ e $0,24\text{msg/ms}$ para 2, 5, 10 e 15 núcleos, nessa ordem. Assumindo uma premissa de SLA em que uma vazão maior que $0,2\text{msg/ms}$ é necessária, é possível atender a essa restrição com 5 ou mais nós. Quando $AR = 0,08\text{msg/ms}$, há uma vazão de $0,13\text{msg/ms}$, $0,34\text{msg/ms}$, $0,47\text{msg/ms}$ e $0,48\text{msg/ms}$ para 2, 5, 10 e 15

nós, respectivamente. Considerando um SLA de pelo menos 0,04msg/ms para vazão, as configurações que atendem a esse requisito são aquelas que têm 10 ou mais nós na névoa.

Analisando os resultados desta seção de forma mais ampla, pode-se ver que o uso da borda da rede não é impactado pela mudança no número de nós na névoa. Para a métrica MRT, o sistema se comporta de forma estável em todas as configurações de névoa, exceto com 5 nós. Com essa configuração, a partir de $AR = 0,053\text{msg/ms}$, o MRT tem aumentos consideráveis. Nesse caso, o uso de recursos de névoa também cresce, chegando perto de 100%. O aumento do MRT faz com que o número de mensagens no sistema também aumente. Com AR de 0,058msg/ms, a configuração com 5 nós passa a ter mais mensagens no sistema do que a configuração com 2 nós. Com 2 nós, o sistema possui a menor vazão e a menor capacidade de processamento, o que também significa filas mais curtas, causando um aumento na taxa de descarte.

5.2.3 Cenário C: Variando o número de nós e núcleos simultaneamente

Nos cenários anteriores (A e B), os fatores número de nós de névoa e número de núcleos em cada nó foram analisados separadamente. Tais análises nos permitiram observar de forma bastante detalhada como cada fator interfere em todas as métricas. No entanto, além de ter um impacto isolado no comportamento do sistema, a análise DoE mostrou que existe uma forte interação entre os dois fatores no tempo médio de resposta, conforme mostrado no gráfico de Pareto (Figura 9) e gráfico de interação (Figura 11). Esses gráficos indicam apenas a existência e a magnitude da interação, mas não com precisão. Portanto, esta seção mostra a variação dos dois fatores com valores iguais a 1, 2, 4 e 8 nós/núcleos. O número de nós na borda foi fixado com 6 nós com 1 núcleo cada. A Tabela 6 apresenta as combinações entre os fatores analisados neste cenário.

A Figura 14 apresenta um gráfico de superfície 3D para mostrar o comportamento do sistema considerando o MRT variando dois fatores com alto impacto no desempenho. Em primeiro lugar, é fundamental dizer que as cores estão relacionadas ao resultado do MRT. A barra à direita indica a magnitude dos resultados. A parte superior indica MRTs maiores e a parte inferior indica MRTs menores. Portanto, a cor roxa representa o MRT mais baixo e a cor vermelha significa o MRT mais alto. No gráfico, vale destacar a presença de uma projeção no topo que facilita a interação dos fatores. Alterar o número de núcleos gera um impacto maior do que alterar o número de nós. A cor roxa está presente na maior parte da projeção. A cor roxa corresponde aos MRTs na posição superior de $1,9\text{E}+06\text{ms}$, ou seja, se for adotado número de nó com número de núcleo maior que 2, o MRT ficará abaixo de $1,9\text{E}+06\text{ms}$. Observando a parte vermelha do gráfico (MRT maior), o fator dominante é o número de núcleos, se esse número for pequeno, o MRT sempre tende a ser alto, e não adianta alterar o número de nós. Portanto, o resultado indica que muitas vezes é mais benéfico investir na capacidade de um único nó de servidor com maior poder de

Combinação	Número de Núcleos	Número de Nós
#1	1	1
#2	1	2
#3	1	4
#4	1	8
#5	2	1
#6	2	2
#7	2	4
#8	2	8
#9	4	1
#10	4	2
#11	4	4
#12	4	8
#13	8	1
#14	8	2
#15	8	4
#16	8	8

Tabela 6 – Combinações dos fatores número de núcleos e número de nós.

processamento do que adquirir vários servidores menores. Este estudo de caso mostra que ao comprar um número de 8 nós com 8 núcleos, haverá o mesmo desempenho de 6 nós com 6 núcleos para este estudo de caso.

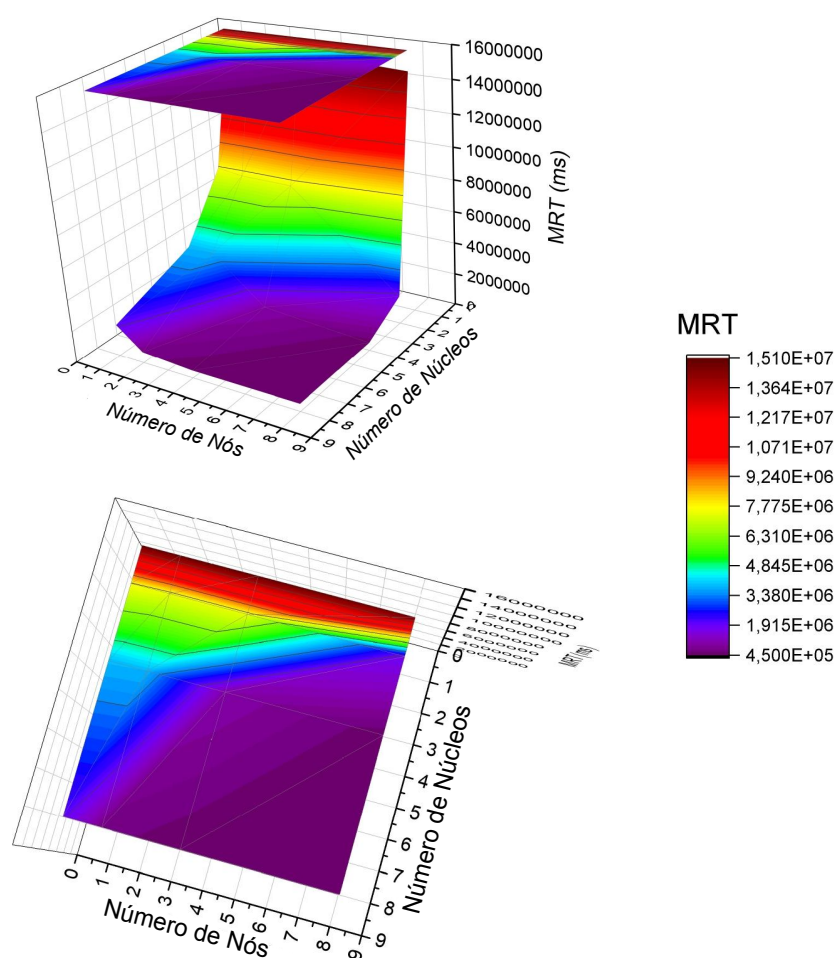


Figura 14 – Análise do tempo médio de resposta por dois fatores simultaneamente.

6 Validação

Este capítulo apresenta a validação do modelo através de uma comparação entre resultados obtidos na resolução do modelo de filas e mensurações em um ambiente de computadores reais. O processo de validação ocorreu seguindo as seguintes etapas:

1. Simplificação do modelo. Tivemos que diminuir o número de componentes replicados que foram apresentados no cenário e no modelo por questões de restrição de recursos computacionais.
2. Implementação do validador sintético. Implementamos, utilizando a linguagem Java, um validador seguindo todas as restrições impostas pelo modelo. A métrica usada na validação foi o MRT pois é a métrica que possui maior impacto sob a visão do usuário final e por ser uma das métricas mais adotadas neste tipo de trabalho de avaliação de desempenho.
3. Executamos o validador sintético em um conjunto de computadores de forma distribuída e coletamos os tempos de execução em cada um dos componentes incluindo transmissão e processamento final. Estas “fatias de tempo médio”, considerando uma requisição em um nó de processamento (núcleo do nó da névoa), foram utilizadas para alimentar os tempos de serviço do modelo proposto.
4. Depois, executamos o validador sintético variando o número de núcleos do nó da névoa e comparamos o MRT do experimento com os resultados das mesmas configurações no modelo.

Antes de continuar a descrição mais detalhada das etapas deste processo de validação, é importante destacar que a pandemia impôs restrições de acesso a recursos de laboratório e por conta disso a validação foi realizada com um número reduzido de máquinas. Assim, diferentemente do cenário e modelo apresentados anteriormente, não consideramos nesta etapa múltiplas salas, dispositivos da borda e nós da névoa. Apesar desta restrição focamos no paralelismo resultante do uso de múltiplos núcleos de processamento em um nó da névoa. Por restrições técnicas todas as máquinas possuíam a mesma configuração, tratando-se de computadores de mesa (Intel Core i5, 4GB). Obviamente, a princípio, o dispositivo da borda deveria ser representado por uma espécie de switch ou roteador mas optamos por padronizar todos os componentes físicos. O processamento realizado nos núcleos do nó da névoa só foram possíveis de ocorrer devido à implementação paralelizável com Java threads. Uma thread é como uma CPU virtual que permite a execução de um código Java de forma paralela à aplicação principal (OAKS; WONG,

2004). Dessa forma, é possível que um programa opere com mais eficiência, fazendo várias coisas ao mesmo tempo. A Figura 15 apresenta a visão geral de implantação da disposição dos quatro computadores utilizados no experimento, espelhando o modelo simplificado. A Sala é responsável pela geração de dados para processamento, os quais são repassados pelo Dispositivo de Borda e Gateway, até chegar no Nó da Névoa.

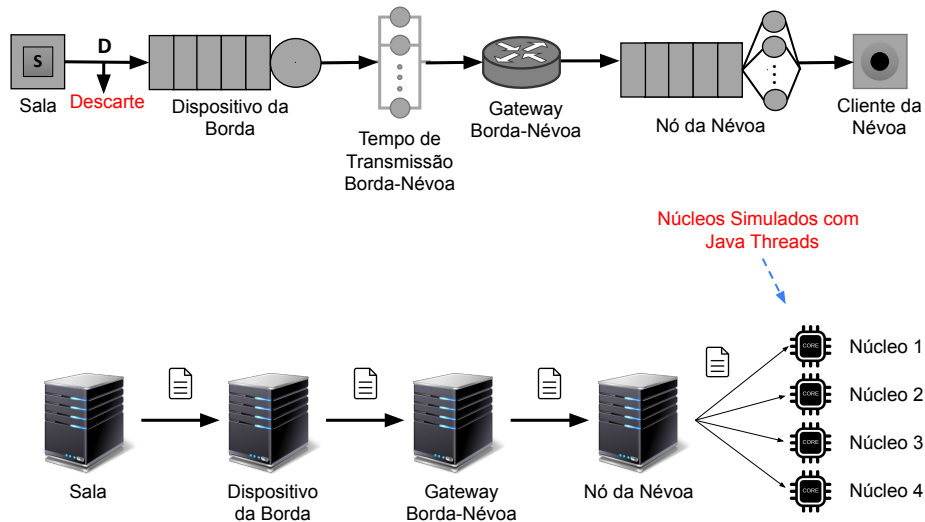


Figura 15 – Visão geral da infraestrutura utilizada no experimento de mensuração e validação do modelo proposto. Foram usados apenas quatro computadores por restrição de recursos.

A Figura 16 complementa a informação sobre a comunicação dos componentes onde destacamos o registro de tempo de transmissão ou processamento em cada um dos componentes. O ciclo de comunicação inicia na Sala e termina no núcleo da névoa. Como os componentes são executados de forma distribuída, foi necessário adotar um relógio global com o protocolo NTP para registrar as janelas de tempo em cada componente.

Como já mencionado anteriormente, o validador sintético foi implementado em Java. A Figura 17 apresenta o diagrama de classes do validador sintético. A classe Source é responsável por enviar um arquivo com os dados de interesse. Neste trabalho de validação utilizamos um algoritmo clássico que conta o número de palavras em um texto e agrupa-os por ocorrências individuais. Acreditamos que a adoção de um algoritmo genérico como este não impôs um aspecto negativo sobre o experimento, uma vez que o objetivo era apenas imprimir um certo peso de processamento na névoa. Aplicamos um certo grau de abstração no código em algumas classes. Como o Gateway e EdgeDevice possuem uma função de retransmissor de dados tais classes tiveram uma relação de herança, o que permite que uma classe seja criada a partir de outra classe previamente criada. A classe EdgeDevice herda de Gateway. A comunicação entre os componentes foi implementada por uso de Java Socket. Por fim a classe Server é quem representa uma máquina para processamento, a qual possui n núcleos (cores). O processamento de fato ocorre na classe Core, que ao

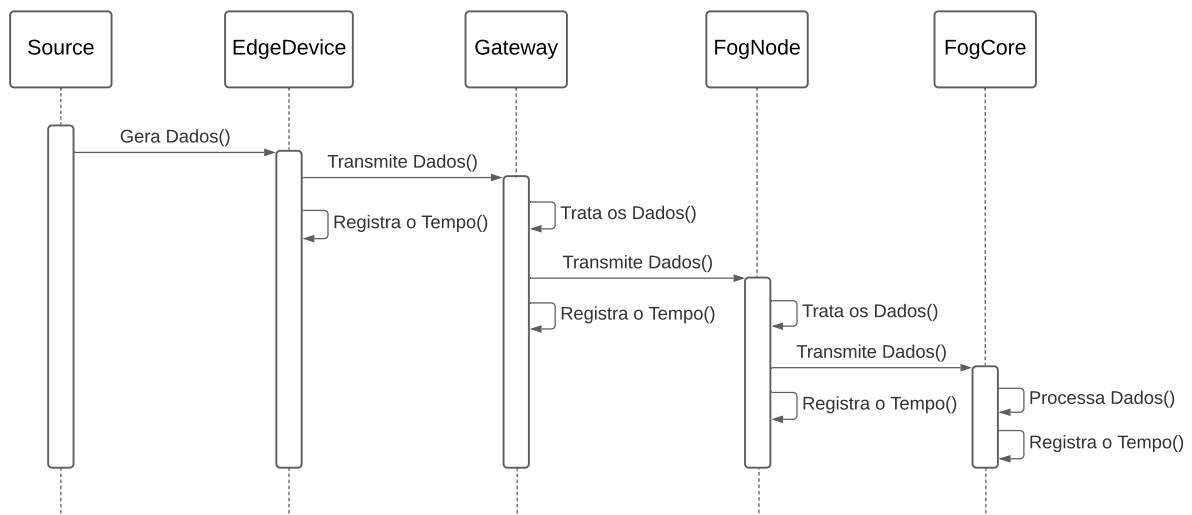


Figura 16 – Diagrama de sequência destacando a comunicação e principais ações gerais realizadas em cada componente.

finalizar apenas imprime o resultado.

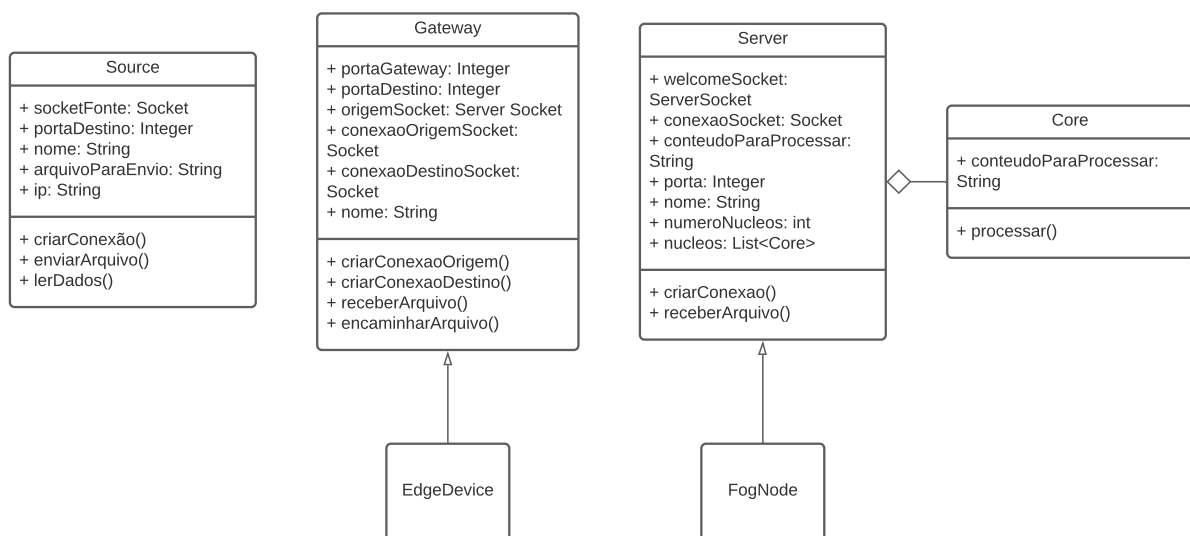


Figura 17 – Diagrama de classes com os principais componentes do modelo proposto. O validador sintético foi implementado em Java com uso de Sockets.

Como mencionado anteriormente, variamos apenas o número de núcleos de um nó da névoa. Aplicamos o T-Test de uma amostra para comparar o MRT gerado pelo modelo com o MRT obtido nos experimentos. O T-Test é um tipo de teste estatístico usado para comparar as médias de dois grupos, ele pode ser usado quando as amostras satisfazem as condições de normalidade, igual variância e independência (KIM, 2015). Os resultados estatísticos são elencados na Tabela 7. Para cada configuração (1, 2, 3, 4 núcleos da névoa), foi repetido o experimento 30 vezes. O modelo resultou em um valor único que foi comparado com o intervalo de confiança do experimento. Todas as amostras apresentaram distribuição normal. Para verificar a significância do T-Test, observamos

o valor p . A Tabela 11 resume os resultados. O valor p é superior a 0,05 em todos os casos. Portanto, não podemos refutar a hipótese nula de igualdade em todos os casos, com 95% de confiança. Podemos concluir que os resultados gerados pelo modelo são estatisticamente equivalentes ao experimento. O modelo reflete o ambiente real e é útil para o planejamento de arquiteturas IoT no contexto deste trabalho. Por fim, as Figuras 18, 19, 20 e 21 apresentam visualmente por meio de Boxplots a comparação das médias do experimento com o modelo em cada uma das quatro configurações.

Tabela 7 – Dados Estatísticos do T-Test

Número de Núcleos	Número de Amostras	Média do MRT (ms) - Experimento	Média do MRT (ms) - Modelo	Intervalo de Confiança	Desvio Padrão	P-Value
1	30	8973,4	8961	[8879,8; 9067,0]	75,4	0,732
2	30	6941	7001	[6662; 7220]	225	0,581
3	30	5843	5810	[5461; 6225]	308	0,823
4	30	5193	5173	[4913; 5474]	226	0,851

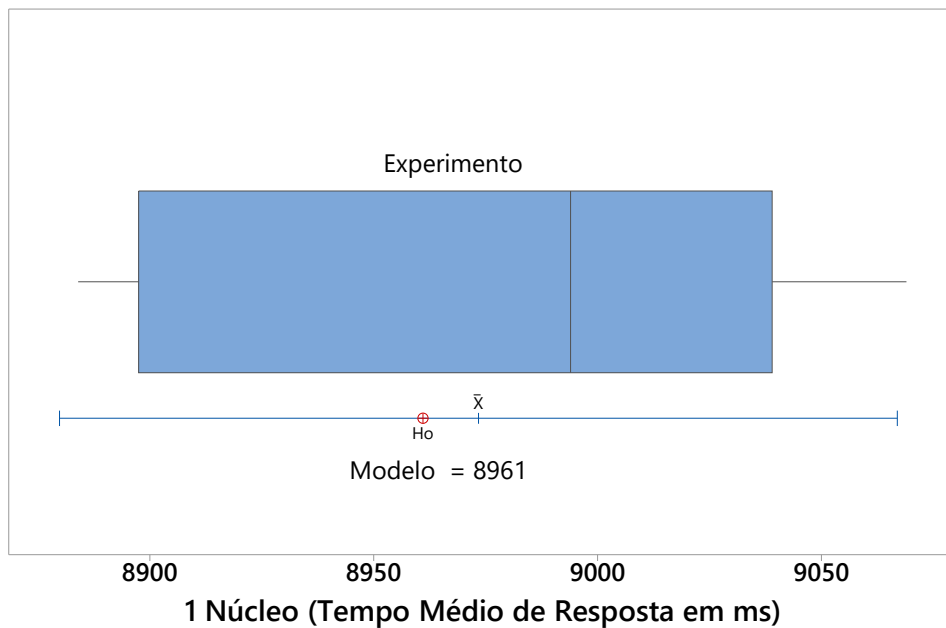


Figura 18 – Resultado do T-Test para a configuração de 1 núcleo de processamento.

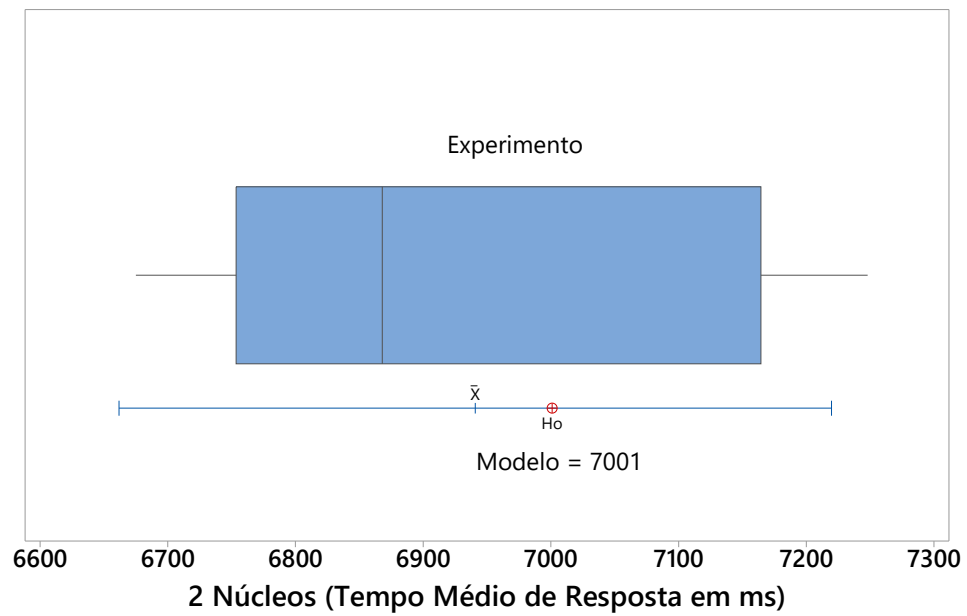


Figura 19 – Resultado do T-Test para a configuração de 2 núcleos de processamento.

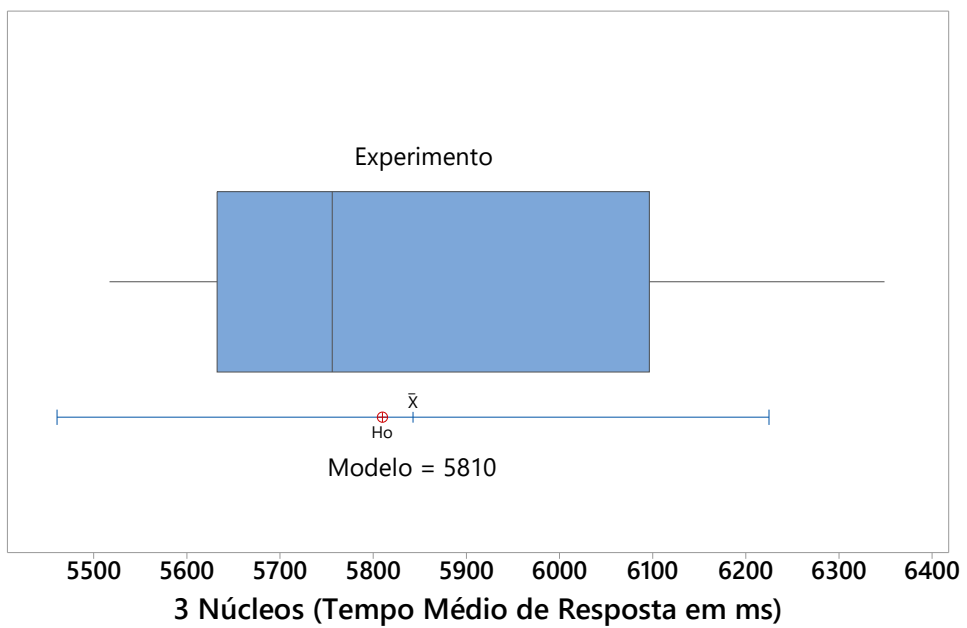


Figura 20 – Resultado do T-Test para a configuração de 3 núcleos de processamento.

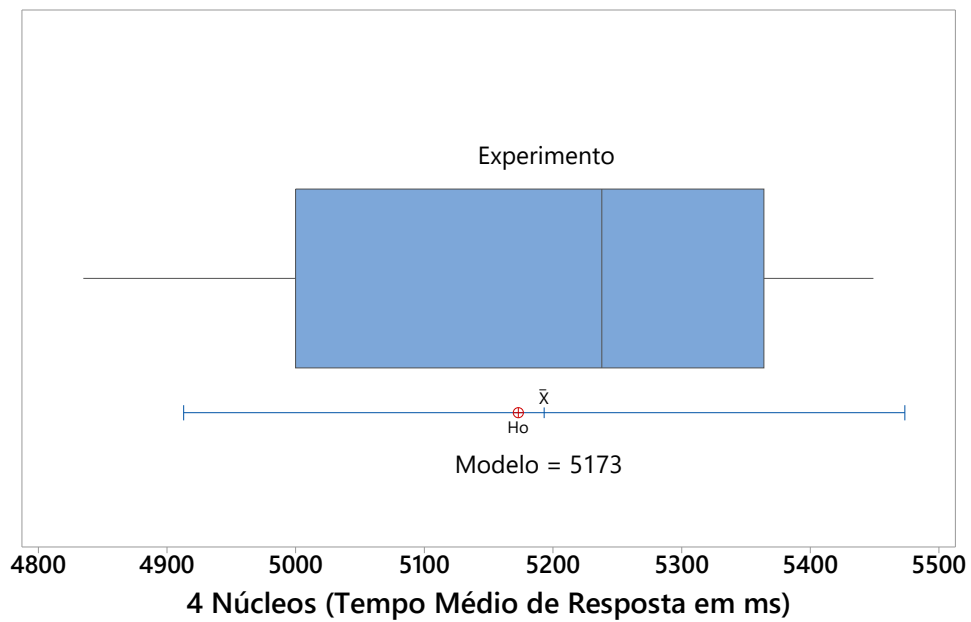


Figura 21 – Resultado do T-Test para a configuração de 4 núcleos de processamento.

7 Conclusão

Este trabalho propôs uma arquitetura de envio de mensagens baseada em rede de filas (D/M/c/K/FCFS) para avaliar o desempenho de infraestruturas de edifícios inteligentes. O cenário avaliado inclui sensores IoT e componentes de borda e névoa.

O modelo permite a análise de várias métricas, como nível de utilização, taxa de descarte, tempo médio de resposta e taxa de vazão. Além disso, o modelo possui recursos que não são encontrados em trabalhos relacionados, como análise de capacidade de recursos, sensores agrupados por localização e o número de núcleos por máquina.

As análises de desempenho foram realizadas considerando a arquitetura de envio de mensagens baseada em rede de filas proposta por meio de uma análise de sensibilidade usando DoE e simulações. Na análise com DoE, alguns fatores que podem afetar o desempenho do edifício foram estudados: número de nós disponíveis na névoa, número de núcleos em cada nó, tamanho da fila e taxa de serviço. Os dois fatores mais relevantes foram explorados nas simulações: o número de nós na névoa e núcleos.

Os resultados das simulações mostram que a taxa de chegada e o número de recursos disponíveis na névoa podem ser muito influentes no desempenho do sistema. À medida que a taxa de chegada aumenta, pode ser necessário usar mais recursos de névoa para que o sistema funcione satisfatoriamente. Nos resultados para a taxa de descarte no cenário A, por exemplo, quando temos 2 núcleos de processamento e $AR = 0,04$ msg/ms, não houve descarte de mensagem, mas quando $AR > 0,049$ msg/ms, o número de mensagens descartadas aumenta com o incremento da taxa de chegada. A validação de um modelo analítico é uma etapa fundamental de um trabalho como este.

A validação permite ter uma confiança maior que o modelo está consistente e que pode calcular corretamente as métricas de interesse. Na validação apresentada, apesar de ter sido sobre uma simplificação do modelo, foi possível observar com um certo grau de confiança estatística que o resultado do modelo é semelhante ao resultado do experimento e portanto o modelo pode ser utilizado para os devidos fins propostos.

Este trabalho pode ser útil para projetistas de sistemas neste contexto para definir melhor as configurações do sistema de monitoramento de edifícios. A limitação deste trabalho está na falta de suporte à nuvem. Portanto, pretendemos estender o modelo, incluindo a camada de nuvem e seus componentes, avaliando novos cenários como trabalhos futuros.

8 Publicações

Este projeto de pós-graduação obteve duas publicações dentro do período do mestrado. O primeiro, um artigo em evento, possui Qualis B3. O segundo artigo, de revista, possui Qualis A1, destacados abaixo:

Santos, B., Silva, F. A., Soares, A. (2021, July). Redes de sensores IoT em edifícios inteligentes: uma avaliação de desempenho usando modelos de filas. In *Anais do XX Workshop em Desempenho de Sistemas Computacionais e de Comunicação* (pp. 25-36). SBC.

Santos, B., Soares, A., Nguyen, T. A., Min, D. K., Lee, J. W., Silva, F. A. (2021). IoT Sensor Networks in Smart Buildings: A Performance Assessment Using Queuing Models. *MDPI Sensors*, 21(16), 5660.

Referências

- AIN, Q.-u. et al. Iot operating system based fuzzy inference system for home energy management system in smart buildings. *Sensors*, Multidisciplinary Digital Publishing Institute, v. 18, n. 9, p. 2802, 2018. [11](#)
- AJAO, L. A. et al. Modeling and implementation of smart home and self-control window using fpga and petri net. In: IEEE. *2020 IEEE PES/IAS PowerAfrica*. [S.l.], 2020. p. 1–5. [3](#), [4](#), [17](#), [18](#), [19](#)
- ANALYTICS, S. Number of internet of things (iot) connected devices worldwide in 2018, 2025 and 2030 (in billions). *Statista Inc.*, <https://www.statista.com/statistics/802690/worldwide-connecteddevices-by-access-technology>, last accessed, v. 7, p. 17, 2020. [3](#)
- ARBIB, C. et al. Real-time emergency response through performant iot architectures. In: *International Conference on Information Systems for Crisis Response and Management (ISCRAM)*. [S.l.: s.n.], 2019. [4](#)
- BALAKRISHNAN, K. *Exponential distribution: theory, methods and applications*. [S.l.]: Routledge, 2019. [27](#)
- 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](#), [10](#)
- BRIEF, I. L. Internet of things. 2019. [7](#)
- BUCKMAN, A. H.; MAYFIELD, M.; BECK, S. B. What is a smart building? *Smart and Sustainable Built Environment*, Emerald Group Publishing Limited, 2014. [3](#)
- CAMPOLONGO, F.; TARANTOLA, S.; SALTELLI, A. Tackling quantitatively large dimensionality problems. *Computer Physics Communication*, Institute Jbr Systems, Informatics and Safety. Joint Research Centre of the European Commission, v. 117, n. 1, p. 75–85, 1999. [13](#)
- CARVALHO, D. et al. Mobile edge computing performance evaluation using stochastic petri nets. In: IEEE. *2020 IEEE Symposium on Computers and Communications (ISCC)*. [S.l.], 2020. p. 1–6. [4](#)
- CHEN, S.; ZHANG, T.; SHI, W. Fog computing. *IEEE Internet Computing*, IEEE, v. 21, n. 2, p. 4–6, 2017. [9](#)
- 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. [13](#)
- CUTSEM, O. V. et al. Cooperative energy management of a community of smart-buildings: A blockchain approach. *International Journal of electrical power & energy systems*, Elsevier, v. 117, p. 105643, 2020. [11](#)

- FANTI, M. P.; MANGINI, A. M.; ROCCOTELLI, M. A petri net model for a building energy management system based on a demand response approach. In: IEEE. *22nd Mediterranean Conference on Control and Automation*. [S.l.], 2014. p. 816–821. [4](#), [17](#), [18](#)
- FANTI, M. P.; MANGINI, A. M.; ROCCOTELLI, M. A simulation and control model for building energy management. *Control Engineering Practice*, Elsevier, v. 72, p. 192–205, 2018. [4](#), [17](#), [18](#), [19](#)
- FANTI, M. P. et al. A petri net model of an integrated system for the health care at home management. In: IEEE. *2014 IEEE international conference on automation science and engineering (CASE)*. [S.l.], 2014. p. 582–587. [4](#), [17](#), [18](#)
- FERREIRA, L. et al. Optimizing resource availability in composable data center infrastructures. In: IEEE. *2019 9th Latin-American Symposium on Dependable Computing (LADC)*. [S.l.], 2019. p. 1–10. [13](#)
- FERREIRA, M. A. M. et al. Statistical queuing theory with some applications. ExcelingTech Publisher, 2011. [27](#)
- FISHMAN, G. S. *Discrete-event simulation: modeling, programming, and analysis*. [S.l.]: Springer Science & Business Media, 2013. [25](#)
- FU, X.; YANG, Y. Modeling and analyzing cascading failures for internet of things. *Information Sciences*, Elsevier, v. 545, p. 753–770, 2021. [3](#)
- GARCIA-CONSTANTINO, M.; KONIOS, A.; NUGENT, C. Modelling activities of daily living with petri nets. In: IEEE. *2018 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*. [S.l.], 2018. p. 866–871. [4](#), [17](#), [18](#)
- GHOMI, E. J.; RAHMANI, A. M.; QADER, N. N. Applying queue theory for modeling of cloud computing: A systematic review. *Concurrency and Computation: Practice and Experience*, Wiley Online Library, v. 31, n. 17, p. e5186, 2019. [4](#), [11](#)
- GUREEV, P. M.; DUNENKOVA, H. N.; ONISHCHENKO, S. I. Smart technologies of the “smart city”. In: SPRINGER. *Institute of Scientific Communications Conference*. [S.l.], 2020. p. 1052–1062. [3](#)
- H AidAR, N. et al. New consumer-dependent energy management system to reduce cost and carbon impact in smart buildings. *Sustainable Cities and Society*, Elsevier, v. 39, p. 740–750, 2018. [11](#)
- HARAGOS, I.-M.; CERNAZANU-GLAVAN, C. Modelling road traffic using service center. *AECE 2012*, v. 2, 2012. [11](#)
- Dynamic qoe / qos-aware queueing for heterogeneous traffic in smart home. v. 7. [4](#)
- ISLAM, M. S. U.; KUMAR, A.; HU, Y.-C. Context-aware scheduling in fog computing: A survey, taxonomy, challenges and future directions. *Journal of Network and Computer Applications*, Elsevier, p. 103008, 2021. [4](#)
- ISMAEL, K. D.; IRINA, S. Face recognition using viola-jones depending on python. *Indonesian Journal of Electrical Engineering and Computer Science*, v. 20, n. 3, p. 1513–1521, 2020. [11](#)

- JIA, M. et al. Adopting internet of things for the development of smart buildings: A review of enabling technologies and applications. *Automation in Construction*, Elsevier, v. 101, p. 111–126, 2019. 10
- JIA, R. et al. Design automation for smart building systems. *Proceedings of the IEEE*, IEEE, v. 106, n. 9, p. 1680–1699, 2018. 10
- 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*, Korean Society of Anesthesiologists, v. 68, n. 6, p. 540, 2015. 45
- KLEIJNEN, J. P. Sensitivity analysis and optimization in simulation: design of experiments and case studies. In: IEEE. *Winter Simulation Conference Proceedings, 1995*. [S.l.], 1995. p. 133–140. 14
- KUBBA, S. Chapter 9 - impact of energy and atmosphere. In: KUBBA, S. (Ed.). *Handbook of Green Building Design and Construction*. Boston: Butterworth-Heinemann, 2012. p. 385–492. ISBN 978-0-12-385128-4. Disponível em: <<https://www.sciencedirect.com/science/article/pii/B9780123851284000093>>. 10
- 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
- MACIEL, H. S. et al. An integrated access control and lighting configuration system for smart buildings. *Journal of communications software and systems*, Udruga za komunikacijske i informacijske tehnologije, Fakultet . . . , v. 13, n. 2, p. 101–108, 2017. 11
- MEMON, R. A.; LI, J. P.; AHMED, J. Simulation model for blockchain systems using queuing theory. *Electronics*, Multidisciplinary Digital Publishing Institute, v. 8, n. 2, p. 234, 2019. 4
- MENA, D. M.; PAPAPANAGIOTOU, I.; YANG, B. Internet of things: Survey on security. *Information Security Journal: A Global Perspective*, Taylor Francis, v. 27, n. 3, p. 162–182, 2018. 3
- NABIH, A. K. et al. Modeling, simulation, and control of smart homes using petri nets. *International Journal of Smart Home*, v. 5, n. 3, p. 1–14, 2011. 4, 11, 17, 18
- NOVÁK, M.; BIŇAS, M.; JAKAB, F. Unobtrusive anomaly detection in presence of elderly in a smart-home environment. In: IEEE. *2012 ELEKTRO*. [S.l.], 2012. p. 341–344. 4, 17, 18
- NUNES, R.; SÊRRO, C. Edifícios inteligentes: Conceitos e serviços. *Jornal O Projeto*, Ano XV, n. 9, p. 3–6, 2014. 10
- OAKS, S.; WONG, H. *Java Threads: Understanding and Mastering Concurrent Programming*. [S.l.]: "O'Reilly Media, Inc.", 2004. 43
- OMAROV, B.; ALTAYEVA, A.; CHO, Y. I. Smart building climate control considering indoor and outdoor parameters. In: SPRINGER. *IFIP International Conference on Computer Information Systems and Industrial Management*. [S.l.], 2017. p. 412–422. 11

- 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
- PINHEIRO, T. et al. Performance and data traffic analysis of mobile cloud environments. In: IEEE. *2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. [S.l.], 2018. p. 4100–4105. 4
- PINHEIRO, T. et al. Performance and resource consumption analysis of elastic systems on public clouds. In: IEEE. *2019 IEEE International Conference on Systems, Man and Cybernetics (SMC)*. [S.l.], 2019. p. 2115–2120. 4
- PINHEIRO, T. F. da S. et al. Performance prediction for supporting mobile applications' offloading. *The Journal of Supercomputing*, Springer, v. 74, n. 8, p. 4060–4103, 2018. 4
- PRIYANKA, D. et al. Smart door using facial recognition. 11
- RODRIGUES, L.; ENDO, P. T.; SILVA, F. A. Stochastic model for evaluating smart hospitals performance. In: IEEE. *2019 IEEE Latin-American Conference on Communications (LATINCOM)*. [S.l.], 2019. p. 1–6. 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
- SAID, O.; MASUD, M. Towards internet of things: Survey and future vision. *International Journal of Computer Networks*, v. 5, n. 1, p. 1–17, 2013. 3
- SANTOS, G. L. et al. The internet of things for healthcare: optimising e-health system availability in the fog and cloud. *International Journal of Computational Science and Engineering*, Inderscience Publishers (IEL), v. 21, n. 4, p. 615–628, 2020. 4
- 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. 1–24, 2021. 14
- SHAH, A. S. et al. Dynamic user preference parameters selection and energy consumption optimization for smart homes using deep extreme learning machine and bat algorithm. *IEEE Access*, IEEE, v. 8, p. 204744–204762, 2020. 11
- 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, F. A.; FÉ, I.; GONÇALVES, G. Stochastic models for performance and cost analysis of a hybrid cloud and fog architecture. *JOURNAL OF SUPERCOMPUTING*, Springer, 2020. 4
- SILVA, F. A. et al. Mobile cloud performance evaluation using stochastic models. *IEEE Transactions on Mobile Computing*, IEEE, v. 17, n. 5, p. 1134–1147, 2017. 4
- SILVA, F. A. et al. Mobile cloud performance evaluation using stochastic models. *IEEE Transactions on Mobile Computing*, IEEE, v. 17, n. 5, p. 1134–1147, 2018. 4
- SILVA, F. A. et al. Planning mobile cloud infrastructures using stochastic petri nets and graphic processing units. In: IEEE. *2015 IEEE 7th International Conference on Cloud Computing Technology and Science (CloudCom)*. [S.l.], 2015. p. 471–474. 4

- SOKULLU, R.; AKKAŞ, M. A.; DEMIR, E. Iot supported smart home for the elderly. *Internet of Things*, Elsevier, v. 11, p. 100239, 2020. [3](#)
- SUNG, W.-T.; HSIAO, S.-J. The application of thermal comfort control based on smart house system of iot. *Measurement*, Elsevier, v. 149, p. 106997, 2020. [23](#)
- TASCIKARAOGLU, A. Evaluation of spatio-temporal forecasting methods in various smart city applications. *Renewable and Sustainable Energy Reviews*, Elsevier, v. 82, p. 424–435, 2018. [3](#)
- UN, D. World urbanization prospects: The 2014 revision. *United Nations Department of Economics and Social Affairs, Population Division: New York, NY, USA*, v. 41, 2015. [3](#)
- WANG, B. C.; SECHILARIU, M.; LOCMONT, F. Power flow petri net modelling for building integrated multi-source power system with smart grid interaction. *Mathematics and Computers in Simulation*, Elsevier, v. 91, p. 119–133, 2013. [4](#), [17](#), [18](#)
- WU, T. et al. Multi-agent deep reinforcement learning for urban traffic light control in vehicular networks. *IEEE Transactions on Vehicular Technology*, IEEE, v. 69, n. 8, p. 8243–8256, 2020. [3](#)
- XU, H. et al. Prediction of thermal energy inside smart homes using iot and classifier ensemble techniques. *Computer Communications*, Elsevier, v. 151, p. 581–589, 2020. [4](#)
- ZAFARI, F.; PAPAPANAGIOTOU, I.; CHRISTIDIS, K. Microlocation for internet-of-things-equipped smart buildings. *IEEE Internet of Things Journal*, IEEE, v. 3, n. 1, p. 96–112, 2015. [10](#)
- ZHANG, X. et al. An iot-based thermal model learning framework for smart buildings. *IEEE Internet of Things Journal*, IEEE, v. 7, n. 1, p. 518–527, 2019. [11](#)