

Estratégias e Análises para Alocação Eficiente de Recursos em Smart Cities com Edge e Cloud Computing

Bruno L. R. e Silva¹, Viviane C. Santos¹, Alex V. Bastos², Samuel M. A. Araújo²

¹Curso de Engenharia Mecatrônica

Universidade Federal de São João del-Rei (UFSJ) - Ouro Branco, MG - Brasil

²Departamento de Tecnologia em Engenharia Civil, Computação, Automação, Telemática e Humanidades (DTECH) - Universidade Federal de São João del-Rei (UFSJ)
Ouro Branco, MG - Brasil

{brunoluizrsilva, vivianecostasantos18}@aluno.ufsj.edu.br,

{alexvbh, sabreu}@ufsj.edu.br

Abstract. The resource allocation problem in Smart Cities, integrating Edge and Cloud Computing resources, is addressed in this article. Three models based on Integer Linear Programming are proposed, aiming to: construct feasible solutions, minimize costs, and reduce the makespan. An analysis is presented in which a percentage increase in costs is allowed in exchange for a lower makespan. Experiments using exact algorithms were conducted. When considering the evolution of the network from 2G to 6G, it is observed that the rejection rate tends to increase due to reduced coverage. In cost minimization, still within the context of 6G technologies, costs can be reduced by up to $\approx 4.38\%$ compared to the model that only generates a feasible solution. When optimizing the makespan, the execution time increased from 0.03s to up to 2.35s, but with a reduction of $\approx 84.7\%$ in the makespan.

Resumo. O problema de alocação de recursos em Smart Cities, integrando recursos de Edge e Cloud Computing é abordado neste artigo. São propostos três modelos baseados em Programação Linear Inteira, que têm como objetivos: a construção de soluções viáveis, a minimização de custos e a redução do makespan. É apresentada uma análise na qual é permitido um percentual de aumento nos custos em troca de um menor makespan. Experimentos utilizando algoritmos exatos foram realizados. Ao considerar a evolução da rede do 2G ao 6G, observa-se que a taxa de rejeição tende a aumentar devido à redução da cobertura. Na minimização dos custos, ainda no contexto das tecnologias 6G, é possível reduzi-los em até $\approx 4.38\%$ em comparação com o modelo que apenas gera uma solução viável. Ao otimizar o makespan, o tempo de execução aumentou de 0.03s para até 2.35s, mas, com uma redução de $\approx 84.7\%$ no makespan.

1. Introdução

O aumento da densidade populacional em áreas urbanas tem intensificado a necessidade de infraestrutura e serviços adequados para atender às crescentes demandas da sociedade [Dahmane et al. 2024]. Nesse cenário, observa-se um crescimento no uso de dispositivos digitais, como smartphones e sensores, impulsionando novas oportunidades para a

aplicação da Internet das Coisas (*Internet of Things*, IoT) [Srirama 2024]. Sob essa perspectiva, a IoT é um paradigma tecnológico que conecta dispositivos à internet, permitindo a coleta, troca e análise de dados de forma autônoma, sem intervenção direta humana.

A integração de tecnologias digitais, baseadas em IoT, ao cotidiano urbano tem transformado diversos setores, consolidando o conceito de *Smart Cities* [Queiroz et al. 2022, Dahmane et al. 2024]. Diante disso, a IoT destaca-se por viabilizar a coleta, análise e transmissão de dados, promovendo a automação de processos e a melhoria da qualidade de vida [Santos et al. 2021]. Diversas cidades ao redor do mundo já apresentam um alto nível de desenvolvimento de aplicações de IoT, sendo respectivamente, Zurique, Oslo e Canberra¹ as cidades mais desenvolvidas neste aspecto.

A expansão da adoção da IoT nas cidades implica em desafios como a alta demanda por largura de banda, a necessidade de baixos atrasos e a exigência por capacidades de processamento e armazenamento. A integração da *Edge Computing* (EC) e *Cloud Computing* (CC) surge como uma solução promissora, equilibrando o processamento distribuído e o atendimento em tempo quase real [Queiroz et al. 2022]. No entanto, aspectos como escalabilidade, custos e o atendimento a métricas de Qualidade de Serviço tornam a alocação de recursos um tema central de pesquisa [Dahmane et al. 2024, Srirama 2024].

Apesar dos benefícios, a integração entre EC e CC apresenta desafios [Santos et al. 2021]. Enquanto a EC reduz atrasos ao aproximar recursos computacionais dos dispositivos IoT, a CC garante escalabilidade e desempenho para grandes volumes de dados. Essa integração exige uma alocação eficiente de recursos para minimizar custos e atender às necessidades dos usuários [Alsabah et al. 2021]. Além da infraestrutura, o bom desempenho das aplicações de IoT depende de redes com alta taxa de transferência e baixos atrasos [Queiroz et al. 2022]. Neste contexto, a Tabela 1 mostra algumas características da evolução das tecnologias móveis de rede. No entanto, apesar dos avanços e evoluções, ainda existem desafios: no 6G, por exemplo, a maior largura de banda é acompanhada de uma redução no alcance, limitando a cobertura em comparação com tecnologias anteriores, como o 5G [Alsabah et al. 2021].

Tabela 1. Evolução das Tecnologias Móveis^a

Geração	Aplicações	Taxa de transferência	Cobertura
2G	SMS, chamadas digitais, MMS (Multimedia Messaging Service), e-mails básicos	64 kbps	10 KM
3G	Navegação web, e-mails, vídeo chamadas, TV móvel, chamadas de vídeo	2 Mbps	5 KM
4G	Streaming HD, redes sociais, jogos, chamadas de vídeo em alta definição	100 Mbps - 1 Gbps	3 KM
5G	IoT quase em tempo real, automação industrial, <i>Smart Cities</i> , RA e RV	1 - 10 Gbps	0.6 KM
6G ^b	IoT em tempo real, veículos autônomos, comunicações holográficas,	100 Gbps - 1 Tbps	0.32 KM

^aA taxa de transferência e a cobertura são influenciadas por diversos fatores, incluindo frequência e interferências. A versão completa da Tabela e a referência dos valores utilizados estão disponíveis em [Silva et al. 2025].

^bOs valores para 6G são teóricos; os valores reais ainda dependem da implementação e da adoção prática no mercado.

No problema abordado, de modo semelhante à literatura, as demandas geradas por dispositivos e aplicações de IoT requerem processamento em servidores da infraestrutura de rede [Santos et al. 2021, Queiroz et al. 2022]. Assim como em [Alsabah et al. 2021], neste trabalho, o processamento dessas demandas é realizado por uma infraestrutura de rede composta por dispositivos localizados nas camadas de EC e CC. Neste contexto,

¹Dados do *Institute for Management Development*. Ranking das cidades mais inteligentes do mundo. Disponível em: <https://abrir.me/Siecp>. Acesso em: 20 jan. 2025.

os dispositivos IoT conectam-se a servidores da camada de EC com base na disponibilidade de recursos. Quando os servidores da camada EC não possuem recursos suficientes para processar uma demanda, ou quando é necessário otimizar métricas como custos, a solicitação pode ser redirecionada para os servidores da camada CC.

Neste cenário, como contribuição, são formulados modelos em Programação Linear Inteira (*Integer Linear Programming*, ILP) e algoritmos exatos baseados nas formulações. Os modelos têm como objetivo resolver a alocação de recursos, sendo: Modelo M1, tem como objetivo a construção de soluções viáveis, maximizando o atendimento das demandas de IoT, sem otimizar outras métricas; Modelo M2, visa maximizar o número de demandas de IoT atendidas, minimizando os custos operacionais da rede; Modelo M3, mantém o custo da rede do segundo modelo, mas foca na minimização do *makespan* (atraso máximo para o atendimento das demandas). Uma descrição complementar do problema e das questões de pesquisa investigadas é disponibilizada em [Silva et al. 2025].

Neste artigo, são realizados experimentos computacionais por meio de simulações baseadas em cenários realistas, com o objetivo de comparar os diferentes algoritmos propostos. Os experimentos consideram a variação na cobertura das tecnologias de rede, abrangendo do 2G ao 6G (Tabela 1). O artigo está organizado da seguinte forma: Seção 2 apresenta os trabalhos relacionados; Seção 3, os modelos propostos; Seção 4, os experimentos; e Seção 5, as conclusões e perspectivas futuras.

2. Referencial Teórico

O artigo de Dahmane *et al.* revisa serviços e soluções em *Smart Cities*, analisando IoT, EC e CC [Dahmane et al. 2024]. Os autores destacam benefícios em transporte, saúde e segurança, além de desafios como densidade populacional, custos e gestão de dados. Também identifica lacunas na literatura, incluindo otimização de alocação de recursos e eficiência energética, enfatizando a interoperabilidade e abordagens centradas no usuário.

Por sua vez, Khan *et al.* apresentam um levantamento sobre a evolução das pesquisas relacionadas às *Smart Cities*, com foco na integração de CC e EC [Khan et al. 2020]. Os autores destacam problemas do setor e deixam como recomendações para pesquisas futuras o desenvolvimento de modelos eficientes para integração de EC em cenários reais. Aspectos estes abordados neste artigo, que propõe gerar modelos e analisar os *trade-offs* envolvidos nos cenários avaliados, abordando métricas como o balanceamento de carga em servidores e os atrasos. Segundo Khan *et al.*, a adoção de EC em *Smart Cities* envolve diversos desafios, sendo um deles abordado neste estudo, mas que não é discutido em trabalhos da literatura [Santos et al. 2021, Queiroz et al. 2022]: a análise de algoritmos em conjunto com as tecnologias de rede emergentes, como o 5G e o 6G.

Li *et al.* exploram estratégias para alocação de recursos na camada de EC, visando atender às demandas de dispositivos IoT e maximizar a satisfação do usuário [Li et al. 2022]. O estudo aborda um problema de otimização pertencente à classe NP-difícil, similar ao tratado neste artigo, com foco na maximização da satisfação do usuário com base no atraso. Neste trabalho, aspectos relacionados ao atraso também são analisados, considerando comparações entre a evolução das tecnologias de rede, bem como atrasos no processamento e na transmissão de dados. O referencial teórico completo, utilizado como base para este trabalho, é disponibilizado em [Silva et al. 2025].

3. Modelagem Matemática do Problema com ILP

A Tabela 2 descreve os conjuntos, parâmetros e variáveis utilizados nos modelos. Em seguida, são apresentadas as restrições. Por fim, os modelos são definidos com suas respectivas funções objetivo (Tabela 3).

Tabela 2. Conjuntos, Variáveis e Parâmetros do Modelo

Descrição dos Conjuntos		p^d	Demanda de processamento (GHz) da aplicação $d \in D$
D	Conjunto de aplicações IoT a serem processadas	nc^d	Demanda de número de cores pela aplicação $d \in D$
E	Conjunto de servidores de EC disponíveis	st^d	Demanda de armazenamento (GB) da aplicação $d \in D$
C	Conjunto de servidores de CC disponíveis	s^d	Quantidade de dados enviados (Mbps) por $d \in D$
$V = E \cup C$	Conjunto de todos os servidores disponíveis na rede	OC_i	Custo de operação (€) do servidor $i \in V$
$A^d \subseteq V$	Conjunto de servidores de EC que cobrem o dispositivo $d \in D^1$, ou de CC que podem ser acessados através da EC	B_i	Largura de banda oferecida (Mbps) pelo servidor $i \in V$
		M_i	Oferta de memória (GB) no servidor $i \in V$
		P_i	Oferta de processamento (GHz) no servidor $i \in V$
		NC_i	Número cores disponíveis no servidor $i \in V$
		ST_i	Oferta de armazenamento (GB) no servidor $i \in V$
		MC	Custo máximo (€) de um modelo
		t_i^p	Atraso de processamento (ms) no servidor $i \in V$
		t_i^d	Atraso de comunicação (ms) entre $d \in D$ e $i \in V^1$
		α	Parâmetro de controle do custo
Descrição das Variáveis			
T_{max}	Máximo de comunicação e processamento (<i>makespan</i>)		¹ Definido pela fórmula de Haversine.
w^d	0 se a aplicação $d \in D$ foi atendida (1, caso contrário)		² Considera-se o atraso de transmissão, em relação a s^d , e o atraso de propagação no meio.
x_i^d	1 se a aplicação $d \in D$ foi processada no servidor $i \in V$		
z_i	1 se o servidor $i \in V$ está ativo		
Descrição dos Parâmetros			
c^d	Custo de não atendimento (€) da aplicação $d \in D$		
b^d	Demandas de largura de banda (Mbps) da aplicação $d \in D$		
m^d	Demandas de memória (GB) da aplicação $d \in D$		

$$\sum_{i \in V | i \in A^d} x_i^d = (1 - w^d), \forall d \in D \quad (1)$$

$$\sum_{d \in D | i \in A^d} b^d x_i^d \leq B_i z_i, \forall i \in V \quad (2)$$

$$\sum_{d \in D | i \in A^d} m^d x_i^d \leq M_i z_i, \forall i \in V \quad (3)$$

$$\sum_{d \in D | i \in A^d} nc^d x_i^d \leq NC_i z_i, \forall i \in V \quad (4)$$

$$\sum_{d \in D | i \in A^d} p^d x_i^d \leq P_i z_i, \forall i \in V \quad (5)$$

$$\sum_{d \in D | i \in A^d} st^d x_i^d \leq ST_i z_i, \forall i \in V \quad (6)$$

$$(t_i^d + t_i^p) x_i^d \leq T_{max}, \forall d \in D, i \in A^d \quad (7)$$

$$\sum_{i \in V} C_i z_i + \sum_{d \in D} w^d c^d \leq MC\alpha \quad (8)$$

As Restrições (1) definem o local de atendimento das demandas. Elas garantem que, se uma demanda $d \in D$ for atendida em um servidor $i \in V$, a variável w^d receberá o valor 0. O conjunto de restrições (2), (3), (4), (5) e (6) garante, respectivamente, que a soma das demandas de recursos, em termos de largura de banda, memória, cores, processamento e armazenamento demandado, não ultrapasse as capacidades oferecidas. O conjunto de Restrições (7) garante que o tempo de resposta total, somando o atraso na comunicação, não excede T_{max} . As Restrições (8) asseguram que o custo OC_i , incluindo os custos de ativação dos servidores z_i e o custo de não atendimento das demandas, não ultrapasse o limite $MC\alpha$. Neste caso, o parâmetro α é empregado para permitir um certo grau de aumento ou redução desse parâmetro, possibilitando avaliar o impacto de variações de custo em relação aos atrasos de comunicação e processamento. O parâmetro MC é definido como o custo máximo da rede, estipulado com o modelo M2.

Modelo 1 (M1): Garante a construção de uma solução factível, sem a otimização de outras métricas (Tabela 3). O objetivo é gerar soluções com um menor tempo de execução do algoritmo de alocação. A função objetivo penaliza as demandas rejeitadas ao forçar a variável w^d a assumir o valor 0, utilizando a penalidade c^d , associada a um Acordo de Nível de Serviço (*Service Level Agreement*, SLA) por demanda não atendida.

Modelo 2 (M2): Minimiza os custos de operação, considerando o custo de ativação dos servidores e o custo associado às demandas não atendidas. A função objetivo força a variável w^d a assumir o valor 0 para que o custo c^d não seja contabilizado (Tabela 3).

Modelo 3 (M3): Este modelo minimiza o *makespan*, definido como o tempo de resposta máximo T_{max} necessário para atender todas as demandas. Esse princípio visa reduzir o maior tempo de resposta necessário para atender uma demanda, favorecendo os usuários do serviço. A função objetivo é representada na Tabela 3.

Tabela 3. Funções Objetivo e Restrições dos Modelos

	M1	M2	M3
Função objetivo:	$\min\left(\sum_{d \in D} w^d c^d\right)$	$\min\left(\sum_{i \in V} OC_i z_i + \sum_{d \in D} w^d c^d\right)$	$\min(T_{max})$
Sujeito a:	Restrições: (1) a (6)	Restrições: (1) a (6)	Restrições: (1) a (8)

4. Experimentos computacionais

Os experimentos foram conduzidos em um computador com processador Intel Core i9-12900HX, 32 GB de RAM DDR5 e sistema operacional Ubuntu 24.04.1 LTS. O simulador foi implementado em C++ e a API IBM ILOG CPLEX V12.6.3, com uma *thread*.²

4.1. Ambiente de simulação

Topologia de rede física: De forma similar a [Queiroz et al. 2022], a topologia de rede utilizada é inspirada na cidade de Módena (na Itália). Ela possui, respectivamente, 100 e 5 servidores de EC e CC. Os servidores de CC estão localizados na cidade de Ohio (Estados Unidos), e podem receber dados de qualquer servidor da camada de EC. Os dados utilizados para parametrizar os servidores de EC e CC estão detalhados em [Silva et al. 2025]. A cobertura e a taxa de transferência da rede são definidas com valores da Tabela 1.

Dispositivos de IoT: Os custos associados às demandas de aplicações oriundas de dispositivos IoT estão detalhados em [Silva et al. 2025]. Os valores utilizados para p^d , m^d , st^d e s^d foram baseados no trabalho de [Santos et al. 2021].

Métricas de rede avaliadas: tempo de execução do algoritmo em segundos (s); taxa de rejeição, definida como a soma dos percentuais de Rejeição por Falta de Cobertura (RFC) e Rejeição por Falta de Recursos (RFR); espalhamento de servidores, calculado como o percentual de servidores ativos para atender às demandas; custo de operação; e T_{max} .

4.2. Análise de Desempenho

Os algoritmos implementados são exatos e baseados nos modelos M1, M2 e M3. O solver foi configurado com um tempo de execução máximo de 30 minutos. Todas as soluções geradas são viáveis, entretanto, algumas não são ótimas devido ao limite de tempo.

Esse limite de tempo foi estabelecido com o objetivo de encontrar soluções ótimas ou o mais próximo possível delas, permitindo análises mais precisas dos resultados. Os experimentos que não geraram resultados garantidamente ótimos são indicados nos gráficos com um asterisco (*). No modelo M3, é proposta uma variação de parâmetro

²O código-fonte e as instâncias estão disponíveis em <https://abrir.link/JXNHY>.

$\alpha \in \{1.1, 1.2, 1.3\}$. O número de dispositivos de IoT processados variou entre {300, 500, 700}. Em cada gráfico apresentado, observa-se no eixo x, a variação do número de dispositivos (β), modelo (λ) e α , representados, respectivamente, na tripla $[\beta, \lambda, \alpha]$.

O tempo de execução do algoritmo baseado no modelo M1 é inferior ao dos demais em todos experimentos (Figura 1). Esse comportamento é atribuído ao fato de que tal algoritmo foca na viabilidade e atendimento das demandas, enquanto os algoritmos baseados nos modelos M2 e M3 consideram otimizar aspectos adicionais. Neste caso, para uma mesma instância, o modelo M1 possui menos variáveis que o modelo M2, e ambos menos variáveis e restrições a serem resolvidas que M3, o que reduz o tempo de execução.

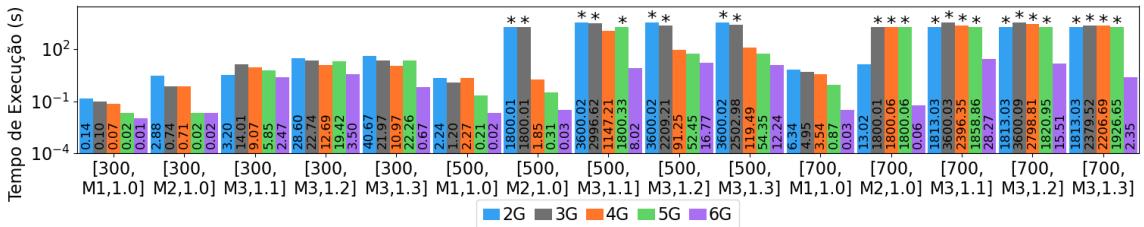


Figura 1. Tempo de execução dos algoritmos

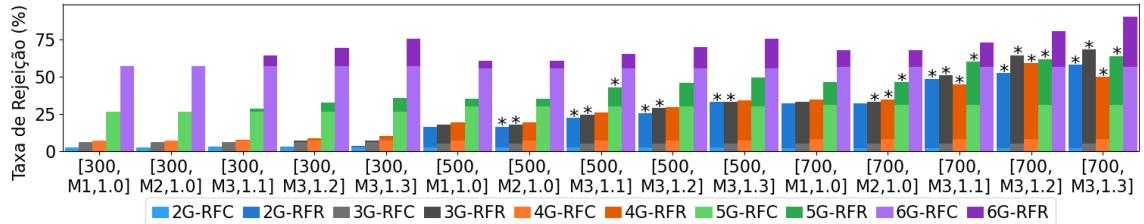


Figura 2. Taxa de rejeição, estratificada por RFC e RFR

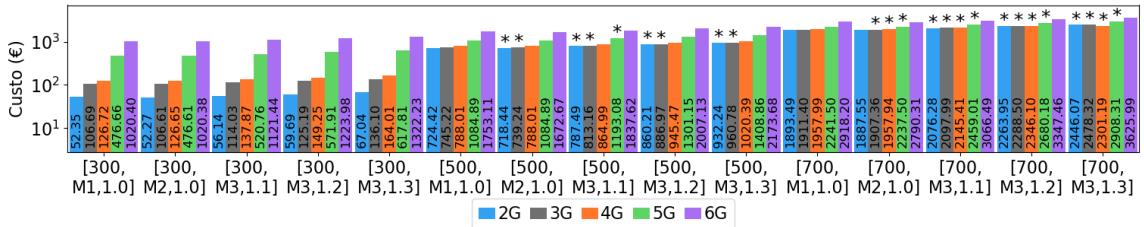


Figura 3. Custos operacionais

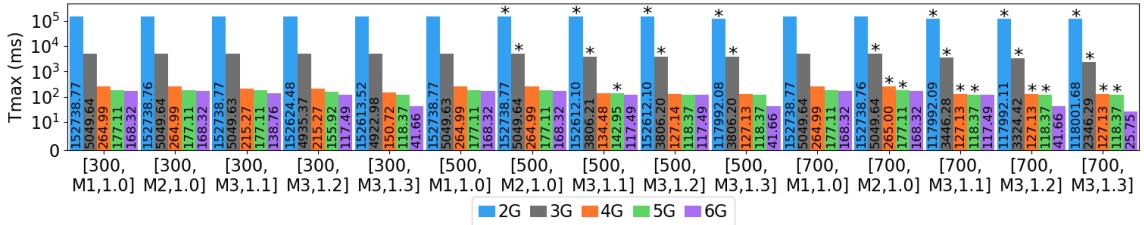


Figura 4. Tmax por tecnologia e número de dispositivos

Ainda considerando os tempos de execução (Figura 1), nos cenários avaliados com a tecnologia 6G, todos os algoritmos apresentam um tempo inferior em comparação com as demais tecnologias. Esse comportamento decorre da menor cobertura dos servidores da rede, o que, na topologia examinada, resulta em um aumento da taxa de rejeição por inviabilidade de cobertura (RFC, Figura 2). Como consequência, menos variáveis e restrições são geradas no modelo, reduzindo o tempo de execução do respectivo algoritmo.

Em relação à taxa de rejeição (Figura 2), os experimentos mostram que o aumento na carga de dispositivos processados resulta em variações significativas nessa taxa entre diferentes gerações de redes. No cenário 2G, com [300,M1,1.0], a taxa de rejeição é de 2.67%, enquanto, com [700,M1,1.0], aumenta para 56.86%. Esse comportamento evidencia que tecnologias mais antigas, embora ofereçam maior alcance de cobertura, apresentam um aumento mais acentuado na taxa de rejeição devido ao RFR em cenários de média a alta demanda. Esse crescimento ocorre devido ao maior número de dispositivos processados simultaneamente, o que, mesmo com mais opções de servidores para atendimento, leva ao esgotamento dos recursos da rede. Por outro lado, nos cenários como o 5G e 6G, o aumento na taxa de rejeição acontece pela RFC, e não RFR. Neste caso, a baixa cobertura causa a rejeição prematura da demanda, não esgotando os recursos da rede. Este fator indica que a topologia de rede avaliada deve ser repensada quanto à distribuição geográfica dos servidores, para atender de forma eficiente dispositivos baseados em tecnologias 6G.

Com relação ao espalhamento de servidores (por questões de espaço, Figura disponibilizada em [Silva et al. 2025]), os experimentos com as tecnologias 2G e 3G apresentam índices mais elevados em quase todos os cenários. Enquanto tecnologias mais avançadas, como o 6G, registram os menores valores, variando entre 50% e 70%. Esse comportamento é atribuído à diferença no alcance de cobertura: tecnologias com maior alcance tendem a apresentar maior espalhamento, enquanto aquelas com menor alcance resultam em servidores mais concentrados geograficamente.

Analizando os custos (Figura 3), observa-se que, em todos os experimentos, o custo aumenta com o número de dispositivos processados. Esse comportamento está relacionado à taxa de rejeição (Figura 2), uma vez que a minimização de custos leva em consideração a soma dos custos dos servidores ativos e os custos associados às demandas não atendidas. A medida que a rejeição aumenta, o custo também cresce, especialmente em cenários de média e alta demanda, onde a taxa de rejeição é elevada (Figura 2).

Analizando o *makespan* (T_{max} , Figura 4), as tecnologias mais antigas, como o 2G, apresentam taxas de transferência mais limitadas (Tabela 1), resultando em um *makespan* elevado, superior a 150000 ms em todos os cenários investigados. Em contraste, tecnologias mais modernas, como o 5G e o 6G, registram tempos significativamente menores, abaixo de 200 ms, devido à maior taxa de transferência. Ao variar o parâmetro α , nos cenários [700,M3,1.1] e [700,M3,1.3], considerando a tecnologia 6G, o T_{max} diminui de 117.49 ms para 25.75 ms (redução de $\approx 78\%$). Esse comportamento indica um *trade-off*, evidenciando que é possível reduzir o *makespan*, fator que beneficia o usuário final; porém, em contrapartida, implicando em maiores custos para os provedores. Essa redução é causada pela realocação e rejeição de dispositivos, que ocorre à medida que o valor de α aumenta. Um maior custo possibilita o atendimento das demandas em servidores de forma a otimizar o *makespan*. Em contraste com o espalhamento, ao analisar o mesmo cenário 6G com [700,M3,1.3], observa-se a ausência de servidores CC, com todos os dispositivos alocados em servidores EC, devido ao menor atraso fim a fim nessa camada.

5. Considerações finais

Este trabalho analisou estratégias para a alocação de recursos em *Smart Cities*, integrando dispositivos IoT com infraestruturas de EC e CC. Foram avaliados três modelos baseados em ILP, com restrições da literatura e objetivos distintos: *i*) gerar soluções viáveis (M1),

ii) minimizar custos de operação (M2), e *iii)* reduzir o tempo máximo de resposta (M3). O modelo (M3) propôs aumentar o custo em troca de um *makespan* menor.

Dentre outros aspectos, demonstrou-se experimentalmente que abrir mão de um percentual da taxa de aceitação pode ser uma abordagem eficiente para reduzir os atrasos fim a fim, resultando em uma redução no *makespan*. Essa estratégia é vantajosa em contextos onde a prioridade está nos atrasos, como em aplicações críticas de IoT. Trabalhos futuros pretendem investigar métodos adaptativos que ajustem dinamicamente o percentual de aceitação e custos, com base em métricas como densidade de dispositivos, demandas em processamento e aspectos de mercado.

A análise mostrou que tecnologias mais recentes, como o 6G, reduzem o *makespan*, mas aumentam a taxa de rejeição devido à menor cobertura. Isso sugere a necessidade de repensar o posicionamento dos dispositivos na camada de EC para garantir uma cobertura mais ampla. Além disso, a otimização de métricas como custos operacionais e *makespan* (modelos M2 e M3) resultou em maior tempo de execução dos algoritmos. Para trabalhos futuros, busca-se desenvolver heurísticas e métodos baseados em IA para melhorar o tempo de convergência e reduzir a dimensionalidade do problema.

Agradecimentos

Os autores agradecem à UFSJ, com menção aos programas PIDAC-Af/UFSJ e PIBIC/UFSJ, que viabilizaram esta pesquisa por meio de bolsas de Iniciação Científica.

Referências

- Alsabah, M., Naser, M. A., Mahmmud, B. M., Abdulhussain, S. H., Eissa, M. R., Al-Baidhani, A., Noordin, N. K., Sait, S. M., Al-Utaibi, K. A., and Hashim, F. (2021). 6G Wireless Communications Networks: A Comprehensive Survey. *IEEE Access*, 9.
- Dahmane, W. M., Ouchani, S., and Bouarfa, H. (2024). Smart cities services and solutions: A systematic review. *Data and Information Management*, page 100087.
- Khan, L. U., Yaqoob, I., Tran, N. H., Kazmi, S. M. A., Dang, T. N., and Hong, C. S. (2020). Edge-Computing-Enabled Smart Cities: A Comprehensive Survey. *IEEE Internet of Things Journal*, 7(10):10200–10232.
- Li, J., Liang, W., Xu, W., Xu, Z., Jia, X., Zhou, W., and Zhao, J. (2022). Maximizing user service satisfaction for delay-sensitive iot applications in edge computing. *IEEE Transactions on Parallel and Distributed Systems*, 33(5):1199–1212.
- Queiroz, T. A. d., Canali, C., Iori, M., and Lancellotti, R. (2022). *An Optimization View to the Design of Edge Computing Infrastructures for IoT Applications*, pages 1–30. Springer International Publishing, Cham.
- Santos, J., Wauters, T., Volckaert, B., and De Turck, F. (2021). Towards end-to-end resource provisioning in fog computing over low power wide area networks. *Journal of Network and Computer Applications*, 175:102915.
- Silva, B. L. R., Santos, V. C., Bastos, A. V., and Araújo, S. M. A. (2025). Dados e Análises Complementares. Disponível em: <https://abrir.link/kwaou>.
- Srirama, S. N. (2024). A decade of research in fog computing: Relevance, challenges, and future directions. *Software: Practice and Experience*, 54(1):3–23.