

Marcos Dalessandro Cavalcante Lima
Orientador: Rayner Gomes Sousa

Algoritmo Genético no Mapeamento de Redes Virtualizadas: Otimização de QoS e Consumo Energético

Picos - PI
4 de junho de 2025

Marcos Dalessandro Cavalcante Lima
Orientador: Rayner Gomes Sousa

Algoritmo Genético no Mapeamento de Redes Virtualizadas: Otimização de QoS e Consumo Energético

Trabalho de conclusão do curso submetido
para Universidade Federal do Piauí para con-
clusão do curso de Bacharel em Sistemas de
Informação

Universidade Federal do Piauí
Campus Senador Heuvídio Nunes de Barros
Bacharelado em Sistemas de Informação

Picos - PI
4 de junho de 2025

FICHA CATALOGRÁFICA
Serviço de Processamento Técnico da Universidade Federal do Piauí
Biblioteca José Albano de Macêdo

L732a Lima, Marcos Dalessandro Cavalcante.

Algoritmo genético no mapeamento de redes virtualizadas: otimização de QoS e consumo energético / Marcos Dalessandro Cavalcante Lima – 2025.
37 f.

1 Arquivo em PDF.

Indexado no catálogo *online* da biblioteca José Albano de Macêdo-CSHNB
Aberto a pesquisadores, com restrições da Biblioteca.

Trabalho de Conclusão de Curso (Graduação) – Universidade Federal do Piauí, Curso de Bacharelado em Sistemas de Informação, Picos, 2025.
“Orientador: Rayner Gomes Sousa”.

1. Sistemas informacionais. 2. Recuperação da informação. 3. Mapeamento de redes. I. Lima, Marcos Dalessandro Cavalcante. II. Sousa, Rayner Gomes. III. Título.

CDD 005.7

Elaborada por Maria Letícia Cristina Alcântara Gomes
Bibliotecária CRB nº 03/1835

Inteligência Computacional Para o Mapeamento de Fatias de Redes Face a Demanda de Qualidade de Serviços e a Distribuição de Funções de Rede Virtualizadas Encadeadas

Marcos Dalessandro Cavalcante Lima

Monografia aprovada como exigência parcial para obtenção do grau de Bacharel em Sistemas
de Informação.

Data de Aprovação

Picos – PI, 26 de junho de 2025

Documento assinado digitalmente
 **RAYNER GOMES SOUSA**
Data: 04/07/2025 08:41:44-0300
Verifique em <https://validar.iti.gov.br>

Prof. Dr. Rayner Gomes Sousa

Documento assinado digitalmente
 **JOSE DENES LIMA ARAUJO**
Data: 04/07/2025 14:52:13-0300
Verifique em <https://validar.iti.gov.br>

Prof. Dr. José Denes Lima Araújo

Documento assinado digitalmente
 **JULIANA OLIVEIRA DE CARVALHO**
Data: 04/07/2025 09:01:33-0300
Verifique em <https://validar.iti.gov.br>

Prof. Dra. Juliana Oliveira de Carvalho

Agradecimentos

Primeiramente, quero agradecer a meus pais por tanto suporte para que tudo isso seja possível, além de sempre entenderem as eventuais ausências e momentos em que abdicamos de estarmos em família para que tudo fosse possível em diversos momentos dessa jornada.

A Carol que foi, é e será parte importante de minhas conquistas por estar sempre me animando em momentos ruins ou de baixa energia. Aos Miúdos que sempre garantiram que eu nunca estivesse ou me sentisse sozinho durante o tempo em que estivemos no curso, por não serem apenas colegas ou amigos, mas em boa parte do tempo foram família para mim. Ao meu orientador por acreditar em mim e compartilhar seu conhecimento.

De todas as pessoas, tudo isso só foi possível por conta das minhas avós que sempre estiveram me motivando, acima de todos isso é por vocês. Muito obrigado a todos supracitados e por tantos que foram importantes durante essa jornada.

Humanos não vivem por séculos. Não podemos esperar pelo progresso. Precisamos de uma liderança focada no futuro, não no passado.

Jayce Talis

Resumo

A virtualização de redes tem-se consolidado como uma tecnologia fundamental nas arquiteturas de comunicação modernas, especialmente com a evolução das redes 5G. O conceito de fatiamento permite a criação de redes virtuais adaptadas a diferentes perfis de aplicação, otimizando o uso da infraestrutura física. A introdução da Virtualização de Funções de Rede (NFV) possibilita a separação das funções de rede em componentes virtuais (VNFs), que podem ser alocados dinamicamente conforme a demanda. No entanto, a alocação eficiente dessas VNFs é um problema complexo, classificado como NP-Difícil, devido à necessidade de atender a requisitos simultâneos de qualidade de serviço (QoS), latência, largura de banda e disponibilidade, em um ambiente de recursos limitados e heterogêneos. Além disso, o impacto do consumo energético dessas redes virtualizadas compromete a sustentabilidade ambiental e os custos operacionais. Este trabalho propõe o uso de técnicas de inteligência computacional e heurísticas para resolver o problema de posicionamento de VNFs, visando o desenvolvimento de algoritmos de otimização multiobjetivo que aumentem a taxa de mapeamento das fatias de rede, reduzam o tempo de execução e minimizem o consumo de energia da infraestrutura.

Palavras-chaves: NFV, VNFs, 5G, Fatiamento, VNE, Mapeamento, NP-Difícil, Heurísticas, Eficiência Energética.

Abstract

Network virtualization has become a fundamental technology in modern communication architectures, particularly with the advancement of 5G networks. The concept of slicing allows the creation of virtual networks tailored to different application profiles, optimizing the use of physical infrastructure. The introduction of Network Function Virtualization (NFV) enables the decoupling of network functions into virtual components (VNFs), which can be dynamically allocated based on demand. However, the efficient allocation of these VNFs remains a complex problem, classified as NP-Hard, due to the need to simultaneously meet various quality of service (QoS) requirements, such as latency, bandwidth, and availability, within a resource-constrained and heterogeneous environment. Additionally, the growing energy consumption of virtualized networks poses a threat to environmental sustainability and increases operational costs. This work proposes the use of computational intelligence techniques and heuristics to address the VNF placement problem, aiming to develop multi-objective optimization algorithms that increase the network slice mapping rate, reduce execution time, and minimize energy consumption of the infrastructure.

Keywords: NFV, VNFs, 5G, Slicing, VNE, Mapping, NP-Hard, Heuristics, Energy Efficiency.

Lista de ilustrações

Figura 1 – Fluxograma da metodologia do trabalho.	23
Figura 2 – Consumo energético (W) da solução GA + QoS.	36
Figura 3 – Satisfação de requisitos de QoS.	37
Figura 4 – Taxa de sucesso no mapeamento de fatias de rede.	38
Figura 5 – Tempo de execução do algoritmo (s).	39

Lista de tabelas

Tabela 1 – Síntese dos Trabalhos Relacionados quanto à Arquitetura, Técnica, Energia e Disponibilidade	19
---	----

Sumário

1	Introdução	12
1.1	Objetivos	13
2	Referencial Teórico	14
2.1	Network Function Virtualization	14
2.2	Network Slicing	15
2.3	Internet - 5G/6G	15
2.4	Heurística e Inteligencia Computacional	16
2.5	Virtual Network Embedding (VNE)	16
2.6	Service Function Chaining (SFC)	17
3	Trabalhos Relacionados	18
4	Metodologia do Trabalho	23
4.1	Questão da Pesquisa	23
4.2	Classificação da Pesquisa	24
4.3	Detalhamento das Etapas Metodológicas	24
4.3.1	Etapa 1: Início e Definição do Escopo	25
4.3.2	Etapa 2: Revisão de Literatura Sistemática	25
4.3.3	Etapa 3: Arquitetura e Modelagem do Algoritmo	26
4.3.4	Etapa 4: Implementação do Algoritmo	27
4.3.5	Etapa 5: Definição e Realização dos Experimentos	28
4.3.6	Etapa 6: Avaliação da Eficiência e Análise dos Resultados	28
4.3.7	Limitações da Metodologia	29
4.3.8	Lacunas Atacadas pelo Trabalho	30
5	Solução e resultados	31
5.1	Descrição da Solução Implementada	31
5.2	Resultados	35
5.2.1	Consumo Energético	35
5.2.2	Satisfação de QoS	36
5.2.3	Taxa de Sucesso	37
5.2.4	Tempo de Execução	38
5.2.5	Discussão Geral	39
6	Conclusão	40
6.1	Implicações Práticas	40

6.2 Sugestões para Futuros Trabalhos	41
Referências	42

1 Introdução

A ascensão da tecnologia 5G tem impulsionado a demanda por serviços de rede mais flexíveis, adaptáveis. Essa evolução introduz a necessidade de arquiteturas mais inteligentes, capazes de fornecer qualidade de serviço (QoS) personalizada a diferentes perfis de aplicações, como Internet das Coisas (IoT), veículos autônomos, realidade aumentada e aplicações críticas em saúde e segurança (SUBEDI et al., 2021). Para atender a esses cenários, o conceito de *network slicing* torna-se essencial, pois permite a criação de redes lógicas sobre uma infraestrutura física compartilhada, cada uma com suas próprias características e garantias de serviço (SOUSA, 2023).

A concretização dessa flexibilidade depende de tecnologias emergentes como SDN (*Software Defined Networking*), NFV (*Network Function Virtualization*) e computação em nuvem, que juntas viabilizam uma infraestrutura programável, escalável e resiliente (MIJUMBI et al., 2015). Em particular, a NFV permite que as funções de rede — como *firewall*, *NAT* ou balanceador de carga — sejam virtualizadas e executadas sob demanda em hardware comum, sob a forma de *Virtual Network Functions* (VNFs). A orquestração adequada dessas funções, bem como seu posicionamento eficiente sobre a rede física, são tarefas fundamentais que impactam diretamente no desempenho, custo operacional e, cada vez mais, no consumo energético da rede.

Nesse contexto, a *eficiência energética* se destaca como um dos principais desafios da próxima geração de redes. A operação contínua de *datacenters* e infraestruturas distribuídas exige estratégias que otimizem o uso de energia sem comprometer os requisitos de QoS. Estudos recentes destacam que abordagens de orquestração e alocação de recursos que considerem o perfil energético dos servidores e dos fluxos de tráfego podem reduzir significativamente o consumo global de energia da rede (CHEN et al., 2020; XIANG et al., 2021).

Além disso, a adoção de *Network Slicing as A Service* (NSaaS) possibilita a entrega de fatias de rede personalizadas sob demanda, aumentando a complexidade da gestão e exigindo mecanismos dinâmicos e autônomos para alocação de recursos (DHANASEKARAN; PING; GOMEZ, 2023a). Cada fatia de rede deve ser isolada e capaz de atender aos requisitos específicos de seu cliente, incluindo segurança, confiabilidade, desempenho e, quando aplicável, critérios de sustentabilidade energética (GUO et al., 2022b).

Assim, o presente trabalho propõe o uso de técnicas de inteligência computacional e heurísticas adaptativas para o problema de mapeamento de fatias de rede (*Virtual Network Embedding* — VNE), considerando múltiplos critérios de otimização, incluindo latência, largura de banda, confiabilidade e consumo energético. Ao incorporar a eficiência energética como uma métrica adicional no processo de alocação de VNFs, busca-se promover uma operação mais sustentável e inteligente das redes virtualizadas no contexto 5G e além

(HUANG et al., 2018).

1.1 Objetivos

Essa seção tem como principal finalidade apresentar os objetivos gerais e específicos do estudo. O presente trabalho visa principalmente realizar uma revisão sistemática da literatura com foco no entendimento do estado da arte relacionado ao mapeamento de redes e ao posicionamento de funções de rede virtualizadas. A intenção é identificar as principais abordagens, técnicas e desafios existentes nessa área, com especial atenção às soluções que consideram critérios de eficiência energética e qualidade de serviço (QoS). A revisão será a base teórica fundamental para subsidiar o desenvolvimento de uma solução prática, cujo foco será mitigar o problema do elevado consumo energético em redes virtualizadas, garantindo, ao mesmo tempo, níveis satisfatórios de desempenho e qualidade nos serviços oferecidos.

- Levantar e analisar publicações relevantes sobre mapeamento de redes e alocação de funções de rede virtualizadas;
- Identificar abordagens que tratam da eficiência energética nesse contexto;
- Avaliar estratégias que conciliem eficiência energética e manutenção da QoS;
- Fornecer subsídios para o desenvolvimento de uma solução prática baseada nos resultados da revisão.

2 Referencial Teórico

A crescente complexidade das redes de comunicação modernas, impulsionada pelo avanço das redes 5G e pela perspectiva do 6G, demanda soluções flexíveis, escaláveis e eficientes para a provisão de serviços. Neste contexto, conceitos como Network Function Virtualization (NFV), fatiamento de redes (Network Slicing), Virtual Network Embedding (VNE) e funções de serviço encadeadas (SFC) emergem como pilares fundamentais. Esta seção apresenta os fundamentos teóricos que sustentam o desenvolvimento deste trabalho.

2.1 Network Function Virtualization

A *Network Function Virtualization* (NFV) é uma tecnologia transformadora que substitui as funções de rede baseadas em hardware por programas de software, permitindo o provisionamento sob demanda e a redução de custos (UMRAO; YADAV, 2023). É um conceito de arquitetura de rede que propõe utilizar tecnologias relacionadas à virtualização de computadores para visualizar as diferentes funções dos nós da rede, para poderem ser conectados para criar serviços de informação. Com as NFVs, é possível criar várias redes em um único computador. Ao utilizar a NFV, os serviços de rede podem ser implantados de forma eficiente, com custos operacionais otimizados (QURESHI et al., 2022). Simplificando, uma rede virtual é composta por uma ou duas máquinas virtuais que executam vários aplicativos e processos em servidores de alta capacidade, dispositivos de rede, armazenamento e infraestrutura de computação em nuvem e em névoa, em vez de apenas implementar ferramentas de rede.

Com a crescente necessidade de transparência, escalabilidade e flexibilidade, a virtualização da função de rede móvel (NFV) tornou-se uma ferramenta essencial para a maioria dos provedores de rede móvel. A NFV separa as funções de rede dos dispositivos de *hardware*. Essa separação permite que os serviços de rede, chamados de funções de rede virtuais (VNFs), sejam hospedados em *hardware* comum, simplificando a entrega e o gerenciamento de serviços para os clientes, melhorando a flexibilidade e levando ao aumento da utilização, eficiência e escala de recursos. O posicionamento adequado do VNF é um desafio técnico crítico, pois afeta diretamente o desempenho, a confiabilidade e os custos operacionais da rede (ATTAOUI et al., 2023). O posicionamento pode afetar significativamente o desempenho, a confiabilidade e os custos operacionais da rede.

2.2 Network Slicing

O fatiamento da rede é uma técnica que permite a criação de zonas dedicadas na rede para diferentes finalidades, como o acesso prioritário a serviços essenciais. Essa abordagem garante alta disponibilidade e qualidade de serviço mesmo em situações de congestionamento, por meio da alocação de recursos virtuais sobre uma mesma infraestrutura física (DHANASEKARAN; PING; GOMEZ, 2023b). Se a infraestrutura for suficientemente robusta, é possível criar múltiplas fatias. Por exemplo, uma zona pode ser destinada a serviços críticos, como comunicações de emergência ou aplicações de saúde em tempo real. Ao separar os serviços em zonas específicas, é possível garantir que cada um receba os recursos necessários para funcionar de forma eficiente e confiável (LI et al., 2023).

O fatiamento da rede desempenha um papel crucial no isolamento do tráfego para vários grupos de usuários ou tipos de dispositivos, garantindo serviços personalizados e medidas de segurança aprimoradas (GUO et al., 2022a). Cada uma dessas aplicações pode ter requisitos diferentes em termos de largura de banda, latência e confiabilidade. Ao atribuir uma zona dedicada a cada tipo de serviço, é possível otimizar a experiência do usuário, garantindo que cada aplicação receba os recursos necessários para operar de maneira eficiente e confiável. Outra base importante é a capacidade de isolar o tráfego de determinados usuários ou tipos de dispositivos. Isto é útil em situações onde o tráfego de diferentes grupos de utilizadores (como clientes residenciais e empresariais) precisa ser separado, ou onde políticas de segurança específicas precisam ser aplicadas a dispositivos *IoT*, protegendo assim a integridade e a privacidade dos dados transmitidos.

2.3 Internet - 5G/6G

Uma das características-chave das redes 5G é o fatiamento da rede; as redes 5G aproveitam o fatiamento de rede, um recurso fundamental que permite a criação de várias redes virtuais em uma única infraestrutura física para atender a diversas necessidades de forma eficiente (BONDRE; SHARMA; BONDRE, 2023). Isso exige infraestrutura de TI avançada para criar múltiplas redes virtuais em infraestruturas físicas públicas. Essas redes virtuais podem ser adaptadas para atender às necessidades de diferentes aplicativos, serviços e clientes em todo o mundo. Por meio da Virtualização de Funções de Rede (NFV), as redes 5G podem se particionar em redes virtuais distintas, oferecendo suporte a serviços personalizados para vários clientes e setores (YA; JUAN, 2022).

Os componentes da rede são isolados, proporcionando uma experiência semelhante à de redes físicas separadas. Um elemento de rede (NS) é definido como uma combinação de *software* especializado, rede e VNF que pode ser implantado em diferentes locais e conectado administrativamente a contratos e serviços. A progressão do 6G pode aprimorar o fatiamento da rede e a virtualização das funções de rede, permitindo maior personalização

de serviços e eficiência na alocação de recursos (OUYANG et al., 2023). Isso resulta em redes mais flexíveis e adaptáveis para atender a uma variedade de casos de uso, desde comunicações críticas até Internet das Coisas avançada e realidade aumentada/virtual.

2.4 Heurística e Inteligencia Computacional

Na ciência da computação, as heurísticas desempenham um papel crucial nas tarefas de busca, otimização e tomada de decisão, oferecendo soluções eficientes quando os métodos clássicos falham devido à complexidade ou restrições de recursos (SAHARIA, 2022). Em vez de seguir um conjunto preciso de regras, a heurística utiliza princípios práticos, bom senso e experiência para orientar o processo de resolução de problemas. Na computação, as heurísticas são frequentemente utilizadas em processos de busca, otimização e decisão, onde é importante encontrar uma solução ótima em tempo e recursos. Embora não garantam soluções perfeitas, as heurísticas são importantes pela sua capacidade de encontrar soluções ótimas de forma rápida e eficiente, e são utilizadas em diversos domínios, desde jogos até roteamento de rede.

Inteligência Computacional (CI) é um campo multidisciplinar que alavanca paradigmas computacionais motivados biologicamente e linguisticamente (APPLICATIONS... , 2023), além de uma disciplina ampla que visa desenvolver sistemas e algoritmos capazes de simular o comportamento humano inteligente. Utilizando modelos inspirados em processos biológicos, como redes neurais artificiais e algoritmos genéticos, a ciência da computação visa criar sistemas autônomos que aprendam, raciocinem e tomem decisões como as pessoas. Esses sistemas são usados em muitas áreas, como reconhecimento de padrões, previsão de dados, otimização de processos e robótica. Ao utilizar métodos de otimização global, aprendizado de máquina e raciocínio difuso, a CI lida com tarefas computacionais complexas de forma eficaz (BESOZZI et al., 2020).

2.5 Virtual Network Embedding (VNE)

O *Virtual Network Embedding* (VNE) refere-se ao processo de alocar, de maneira eficiente, os recursos de uma rede física (substrato) para suportar redes virtuais com requisitos específicos de desempenho. Essa alocação deve respeitar restrições como topologia, capacidade de banda, latência e confiabilidade da infraestrutura física (FISCHER et al., 2013).

A abordagem para resolver o VNE pode ser dividida em duas etapas — mapeamento dos nós virtuais para nós físicos e, posteriormente, mapeamento dos enlaces — ou pode ser feita de forma conjunta. O principal desafio está na complexidade combinatória do problema e na necessidade de cumprir requisitos rigorosos de Qualidade de Serviço (QoS). Para lidar com isso, a literatura propõe desde soluções exatas, como aquelas baseadas em programação inteira, até métodos mais flexíveis como algoritmos heurísticos, meta-

heurísticos e técnicas de aprendizado de máquina (MEHRABI; HABIBI; SIRJANI, 2019; HUANG et al., 2018).

Nos últimos anos, têm-se destacado abordagens híbridas que combinam técnicas de inteligência computacional com heurísticas adaptativas, permitindo obter soluções próximas do ótimo com menor tempo de processamento. Essas soluções mostram-se especialmente eficazes em ambientes altamente dinâmicos, com múltiplas requisições de redes virtuais concorrendo por recursos limitados.

2.6 Service Function Chaining (SFC)

O Service Function Chaining (SFC) é uma técnica que permite encadear funções de rede, como *firewalls*, balanceadores de carga e sistemas de detecção de intrusão, em uma sequência lógica para compor serviços personalizados sobre uma infraestrutura virtualizada. Cada fluxo de tráfego é direcionado por essa cadeia conforme os requisitos específicos da aplicação, respeitando a ordem e as dependências entre as funções (LI et al., 2023).

O mapeamento eficiente de SFCs exige que as funções de rede virtuais (VNFs) sejam alocadas em nós físicos com capacidade adequada, levando em conta não apenas os requisitos de CPU e memória, mas também as exigências de latência, banda e disponibilidade. A complexidade do problema aumenta à medida que múltiplas cadeias coexistem, competindo por recursos limitados na infraestrutura subjacente.

Assim como no VNE, o SFC é um problema combinatório de difícil resolução, que pode ser abordado por estratégias sequenciais (alocação das VNFs seguida de roteamento dos fluxos) ou integradas. Diferentes abordagens vêm sendo aplicadas, incluindo algoritmos exatos, heurísticas, meta-heurísticas e modelos baseados em aprendizado de máquina (MEHRABI; HABIBI; SIRJANI, 2019). Em cenários dinâmicos, soluções híbridas com inteligência adaptativa têm-se mostrado eficazes por equilibrar desempenho, escalabilidade e tempo de resposta, especialmente quando há necessidade de reconfiguração rápida das cadeias diante de variações no tráfego ou falhas na infraestrutura (DHANASEKARAN; PING; GOMEZ, 2023b).

3 Trabalhos Relacionados

Esta seção visa apresentar os trabalhos relacionados, explorando diversas abordagens e contribuições no contexto das redes 5G, virtualização de funções de rede (NFV) e inteligência computacional; ainda apresenta a revisão sistemática desenvolvida. Visando identificar os principais avanços relacionados ao mapeamento de redes virtuais e encaideamento de funções de serviço (SFC), realizamos uma revisão sistemática da literatura seguindo uma abordagem estruturada e progressiva. O processo iniciou-se com a definição da *string* de busca, cuidadosamente elaborada para abranger os termos mais relevantes da área. A *string* adotada foi:

(Network Slicing or Virtual Network Embedding or Virtualization or NS) and (NFV or Network Function Virtualization) and (SFC or Service Function Chain)

Essa formulação visou recuperar trabalhos que discutem desde a virtualização e faturamento de redes até técnicas de alocação de funções em cadeias de serviços, abrangendo diferentes perspectivas terminológicas e tecnológicas. A busca foi realizada na base de dados *Scopus*, considerando publicações no intervalo de 2020 a 2024. Como critério adicional, incluímos a exigência de conter o termo “Service Function Chain (SFC)” como palavra-chave. Esse procedimento resultou inicialmente em 51 artigos.

A seguir, aplicamos uma série de filtros para refinar o conjunto de trabalhos. O primeiro filtro eliminou todos os artigos não redigidos em inglês, reduzindo o total para 47. Em seguida, removemos os artigos que não apresentavam classificação *Qualis*, resultando em 32 publicações. O terceiro filtro consistiu na leitura dos resumos e conclusões com o intuito de excluir os trabalhos que não apresentavam explicitamente algum algoritmo de mapeamento, etapa que reduziu o corpus final para 10 artigos.

A partir desses 10 trabalhos selecionados, realizamos uma análise detalhada visando extrair informações essenciais para caracterização e comparação das abordagens. Foram coletados dados sobre a infraestrutura utilizada, como o número de nós e enlaces reais, e os requisitos das redes virtuais, como a quantidade de nós e enlaces virtuais solicitados.

Também foram considerados aspectos da arquitetura proposta (centralizada, paralela, distribuída ou técnica mista), os requisitos computacionais dos nós (CPU e memória) e as características das cadeias de serviço (quantidade de funções e quantidade de serviços). Por fim, verificamos se os artigos abordavam aspectos relacionados à eficiência energética e se as soluções apresentadas estavam publicamente disponíveis, por exemplo, em serviços de repositórios de código como o GitHub.

Esse processo sistemático permitiu consolidar uma base sólida para a compreensão do estado da arte, evidenciando lacunas e oportunidades de pesquisa, especialmente no que diz respeito à eficiência energética e à aplicação de técnicas computacionais avançadas no

problema de mapeamento de SFCs. A tabela 1 apresenta os resultados obtidos do quarto filtro da nossa revisão.

Tabela 1 – Síntese dos Trabalhos Relacionados quanto à Arquitetura, Técnica, Energia e Disponibilidade

Referência	Arquitetura	Técnica	Leva energia em conta?	Solução Pública?
(LI et al., 2021)	Centralizado	Lagrangian Relaxation Heuristic (LRHSA)	Não	Não
(CHEN et al., 2022)	Paralelismo	Q-learning baseada em MDP	Não	Não
(BASU et al., 2023)	Paralelismo	ML, SFC-DRIVE, VNF-READ	Não	Não
(LIU et al., 2020)	Paralelismo	Deep Reinforcement Learning (DDPG)	Não	Não
(LEYVA-PUPO; CERVELLÓ-PASTOR; ANAGNOSTOPOULOS, 2022)	Distribuído	ILP, DPC-UPCR	Sim	Sim
(BHAMARE; SAMDANIS; TALEB, 2022)	Distribuído	MILP, Simulated Annealing (SARG)	Não	Não
(SUN et al., 2019)	Distribuído	Breadth-First Search (BFS), SFCDO	Não	Não
(ZHANG et al., 2023)	Centralizado	ILP, Heuristic	Não	Não
(WANG et al., 2021)	Distribuído	PPPS, ILP, Heuristic	Não	Não
(LI et al., 2020)	Distribuído	ILP, Heuristic para VNFs	Sim	Não
(CHEN et al., 2021)	Distribuído	Tabu Search, Heurística para Subchains	Não	Não
Este Trabalho	Distribuído	GA	Sim	Sim

(LI et al., 2021) aborda o problema de orquestração dinâmica de cadeias de funções de serviço (SFC) na Internet das Coisas (IoT). Os autores propõem uma abordagem baseada em relaxação Lagrangiana para otimizar a alocação de funções de rede virtualizadas (VNFs), visando minimizar a latência e melhorar a eficiência da rede. A metodologia envolve a formulação de um modelo de programação inteira para representar o problema de orquestração, seguido pela aplicação de técnicas heurísticas para encontrar soluções próximas do ótimo. Os resultados experimentais demonstram que a abordagem proposta é eficaz na redução do tempo de orquestração e na melhoria do desempenho geral da rede.

(CHEN et al., 2022) os autores apresentam um framework leve para o mapeamento de SFCs em redes sem fio habilitadas para SDN/NFV, utilizando aprendizado por reforço baseado em Processos de Decisão de Markov (MDP). A proposta visa melhorar a eficiência do mapeamento de funções de serviço, reduzindo o tempo de processamento e aumentando a taxa de aceitação de SFCs. A abordagem considera as características dinâmicas das redes sem fio e adapta-se às mudanças nas condições da rede. Os experimentos realizados mostram que o framework proposto supera métodos tradicionais em termos de desempenho e eficiência.

(BASU et al., 2023) introduz o sistema DRIVE, que utiliza aprendizado de máquina para introspecção dinâmica de recursos e incorporação de VNFs em redes 5G. A proposta combina técnicas de aprendizado supervisionado e não supervisionado para prever a demanda de recursos e otimizar a alocação de VNFs, visando melhorar a utilização da infraestrutura de rede. O sistema é projetado para se adaptar a mudanças na carga de trabalho e nas condições da rede, proporcionando uma alocação eficiente e dinâmica de recursos. Os resultados indicam que o DRIVE melhora significativamente o desempenho da rede em comparação com abordagens convencionais.

Já o estudo (LIU et al., 2020) propõe uma abordagem de aprendizado por reforço profundo para a orquestração dinâmica de SFCs em redes NFV/MEC habilitadas para IoT. A metodologia utiliza algoritmos de política de gradiente para aprender estratégias de alocação de VNFs que otimizam métricas de desempenho, como latência e utilização de recursos. A abordagem é projetada para lidar com a natureza dinâmica e heterogênea das redes IoT, adaptando-se a mudanças na topologia e nas demandas de serviço. Os experimentos demonstram que a técnica proposta supera métodos tradicionais em termos de eficiência e adaptabilidade.

O trabalho de (LEYVA-PUPO; CERVELLÓ-PASTOR; ANAGNOSTOPOULOS, 2022) os autores abordam o problema de posicionamento dinâmico de funções de plano de usuário (UPF) e reconfiguração de encadeamento em redes 5G. A proposta visa otimizar a alocação de UPFs para lidar com a mobilidade dos usuários e as mudanças nas demandas de tráfego, garantindo qualidade de serviço e eficiência energética. A abordagem combina técnicas de programação inteira com heurísticas para encontrar soluções eficientes em tempo hábil. Os resultados mostram que a metodologia proposta melhora a utilização de recursos e a qualidade de serviço em comparação com abordagens estáticas.

(BHAMARE; SAMDANIS; TALEB, 2022) propõem um framework de otimização de telemetria em banda para monitorar fatias de rede usando P4. A proposta permite especificar requisitos de monitoramento para cadeias de serviço individuais, que são mapeados para tarefas de coleta de telemetria em elementos de plano de dados programáveis. A abordagem utiliza programação inteira mista e uma heurística baseada em recozimento simulado para minimizar a sobrecarga de monitoramento. Os resultados experimentais indicam que o IntOpt reduz significativamente a sobrecarga e os atrasos de monitoramento,

melhorando a escalabilidade e a eficiência do sistema.

A proposta de (SUN et al., 2019) propõe um algoritmo de otimização de implantação de cadeias de funções de serviço (SFCDO) baseado em busca em largura para orquestração de SFCs em ambientes de virtualização de funções de rede (NFV). A abordagem visa minimizar a latência de ponta a ponta e o consumo de recursos, selecionando caminhos com o menor número de saltos para a implantação de SFCs. A metodologia também considera o balanceamento de carga entre os nós da rede. Os experimentos mostram que o algoritmo proposto supera métodos tradicionais em termos de desempenho e eficiência de recursos.

(ZHANG et al., 2023) os autores estudam a orquestração de cadeias de funções de serviço multicast em redes habilitadas para SDN/NFV. A proposta abrange o mapeamento, reajuste e expansão de SFCs multicast, considerando fatores de otimização como custo, latência e utilização de recursos. A abordagem utiliza programação linear inteira e heurísticas para encontrar soluções eficientes para a orquestração de SFCs multicast em ambientes dinâmicos. Os resultados demonstram que a metodologia proposta melhora a eficiência e a flexibilidade da rede em comparação com abordagens convencionais.

(WANG et al., 2021) propõe um esquema de posicionamento preditivo baseado em previsão para SFCs críticas no tempo, com compartilhamento de VNFs na borda da rede. A abordagem utiliza técnicas de previsão para antecipar as demandas de serviço e alocar recursos de forma proativa, visando minimizar a latência e garantir a qualidade de serviço. A metodologia combina programação linear inteira com heurísticas para encontrar soluções eficientes em tempo real. Os experimentos indicam que a proposta melhora significativamente o desempenho da rede em cenários com requisitos de tempo crítico.

(LI et al., 2020) aborda o problema de posicionamento dinâmico de cadeias de funções de serviço em redes parcialmente habilitadas para VNFs. A proposta visa otimizar a alocação de VNFs considerando restrições de capacidade e conectividade, utilizando programação linear inteira e heurísticas para encontrar soluções eficientes. A abordagem também considera métricas de eficiência energética no processo de decisão. Os resultados mostram que a metodologia proposta melhora a utilização de recursos e a eficiência energética em comparação com abordagens tradicionais.

Por fim, (CHEN et al., 2021), os autores apresentam o SAP, um esquema de posicionamento de serviços NFV consciente de subcadeias na nuvem de borda móvel. A proposta utiliza busca tabu e heurísticas para orquestrar subcadeias de funções de serviço, visando melhorar a escalabilidade e a robustez do sistema. A abordagem considera a divisão de SFCs em subcadeias menores, facilitando a alocação eficiente de recursos em ambientes de borda. Os experimentos demonstram que o SAP supera métodos convencionais em termos de desempenho e eficiência de recursos.

Diante desse panorama, observa-se que poucos trabalhos consideram simultaneamente desempenho, qualidade de serviço e consumo energético. Além disso, apenas uma minoria

disponibiliza suas soluções em repositórios públicos. O presente trabalho busca preencher essas lacunas, oferecendo uma abordagem de otimização multiobjetivo com foco em eficiência energética, desempenho e reprodutibilidade.

4 Metodologia do Trabalho

Este capítulo apresenta, de forma detalhada e sistemática, o conjunto de procedimentos metodológicos empregados para investigar a viabilidade da construção de um algoritmo baseado em inteligência computacional capaz de realizar o mapeamento de fatias de rede (*network slices*), atendendo simultaneamente aos requisitos de qualidade de serviço (QoS) e aos critérios de eficiência energética. A metodologia foi estruturada para garantir rigor científico, reprodutibilidade dos resultados e validação experimental robusta, seguindo uma abordagem sequencial e iterativa que permite o refinamento contínuo da solução proposta.

A Figura 1 apresenta o fluxograma completo da metodologia adotada, ilustrando as seis etapas principais que compõem o processo de desenvolvimento e validação do algoritmo proposto. O fluxo metodológico contempla desde a fundamentação teórica inicial até a avaliação final da eficiência do algoritmo, passando por todas as fases de concepção, implementação e experimentação.

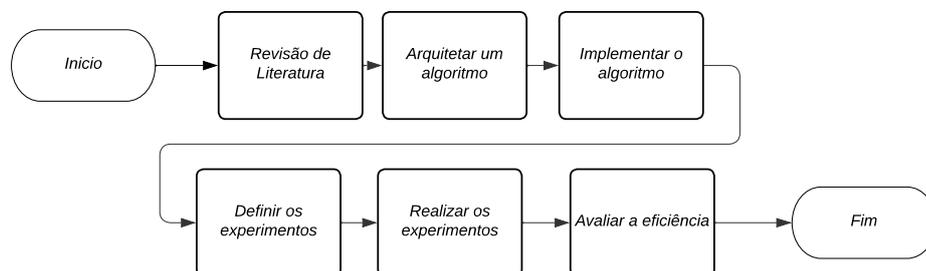


Figura 1 – Fluxograma da metodologia do trabalho.

4.1 Questão da Pesquisa

A pesquisa foi fundamentada em uma questão central que orienta todo o desenvolvimento do trabalho, estabelecendo o escopo e os objetivos específicos da investigação:

É possível criar um algoritmo baseado em inteligência computacional para o mapeamento de fatias de redes, face às demandas de qualidade de serviço e à distribuição de funções de rede virtualizadas encadeadas, que seja eficiente energeticamente?

Esta questão central surge da necessidade crescente de otimizar o uso de recursos em redes de telecomunicações modernas, especialmente no contexto das redes 5G e futuras

redes 6G, onde a eficiência energética se torna um fator crítico para a sustentabilidade operacional. Para aprofundar essa investigação e permitir uma análise mais granular dos resultados, foram formuladas quatro perguntas secundárias específicas que direcionam as diferentes dimensões da avaliação:

Q1: O algoritmo proposto atende consistentemente aos requisitos de QoS das redes virtuais em diferentes cenários de carga, mantendo níveis satisfatórios de banda, latência e confiabilidade?

Q2: A estratégia de alocação das funções de rede virtualizadas contribui significativamente para a redução do consumo energético da infraestrutura física, comparada a abordagens tradicionais?

Q3: O desempenho do algoritmo é competitivo em relação a abordagens tradicionais de mapeamento presentes na literatura, considerando métricas de tempo de execução, taxa de sucesso e utilização de recursos?

Q4: O modelo proposto apresenta robustez e estabilidade frente a diferentes cenários de carga de trabalho, variações topológicas e heterogeneidade de requisitos de serviço?

4.2 Classificação da Pesquisa

Esta pesquisa pode ser classificada segundo múltiplas dimensões metodológicas: **Quanto à natureza:** Pesquisa aplicada, pois busca gerar conhecimento científico direcionado para aplicação prática na solução de problemas específicos do mapeamento de redes virtuais e otimização energética em infraestruturas de telecomunicações; **Quanto à abordagem:** Pesquisa predominantemente quantitativa, utilizando métodos estatísticos rigorosos para análise dos dados experimentais, coleta de métricas objetivas e comparação de desempenho; **Quanto aos objetivos:** Pesquisa exploratória e explicativa, investigando um fenômeno específico (mapeamento eficiente de VNFs) e suas relações causais com o consumo energético e qualidade de serviço; **Quanto aos procedimentos:** Pesquisa experimental baseada em simulações computacionais controladas, permitindo a validação das hipóteses em ambiente controlado e reproduzível.

4.3 Detalhamento das Etapas Metodológicas

Seguindo o fluxograma apresentado na Figura 1, a metodologia foi estruturada em seis etapas sequenciais e complementares, cada uma contribuindo para o desenvolvimento progressivo da solução proposta e sua validação experimental.

4.3.1 Etapa 1: Início e Definição do Escopo

A primeira etapa da metodologia consistiu na definição clara do problema de pesquisa e estabelecimento dos objetivos específicos do trabalho. Nesta fase inicial, foram realizadas atividades fundamentais para o direcionamento adequado da investigação científica.

O processo iniciou-se com a identificação das principais lacunas existentes na literatura relacionada ao mapeamento de redes virtuais com foco em eficiência energética, particularmente no contexto das redes 5G e tecnologias emergentes. Foi conduzida uma análise preliminar das abordagens existentes para compreender as limitações dos métodos tradicionais e identificar oportunidades de contribuição científica. Esta análise inicial permitiu estabelecer a relevância do problema e justificar a necessidade de desenvolvimento de novas soluções algoritmos.

Durante esta etapa, foram definidos os critérios de sucesso para o algoritmo proposto, estabelecendo metas quantitativas para as métricas de avaliação: taxa de sucesso no mapeamento superior a 85%, redução do consumo energético de pelo menos 15% em relação aos métodos tradicionais, e tempo de execução compatível com aplicações em tempo real (inferior a 10 segundos por requisição). Esses critérios foram estabelecidos com base na análise das limitações identificadas na literatura e nas necessidades práticas das redes modernas.

Adicionalmente, foi realizada a definição do escopo tecnológico do trabalho, incluindo a seleção das tecnologias de implementação, ferramentas de simulação e ambiente experimental. Esta definição prévia garantiu a consistência metodológica e a viabilidade técnica do desenvolvimento proposto.

4.3.2 Etapa 2: Revisão de Literatura Sistemática

A segunda etapa consistiu em uma revisão sistemática e abrangente da literatura científica, seguindo rigorosamente as diretrizes estabelecidas para revisões sistemáticas em engenharia de software e sistemas distribuídos. Esta etapa foi fundamental para construir a base teórica sólida necessária para o desenvolvimento do algoritmo proposto.

O protocolo de revisão foi estruturado com objetivos específicos claramente definidos: fundamentar teoricamente o problema de mapeamento de redes virtuais, identificar e catalogar as principais abordagens existentes na literatura, analisar criticamente as limitações dos métodos atuais, identificar métricas de avaliação utilizadas pelos trabalhos relacionados, e estabelecer o estado da arte em otimização energética para infraestruturas de rede.

A estratégia de busca foi implementada utilizando cinco bases de dados científicas reconhecidas: IEEE Xplore Digital Library (maior repositório de artigos em engenharia elétrica e ciência da computação), ACM Digital Library (principal fonte de trabalhos em ciência da computação e sistemas), SpringerLink (ampla cobertura em engenharia e

ciências aplicadas), ScienceDirect (base multidisciplinar com foco em ciências exatas), e arXiv (repositório de preprints para acesso a trabalhos mais recentes).

Os critérios de inclusão foram estabelecidos de forma rigorosa para garantir a qualidade e relevância dos trabalhos selecionados: artigos publicados no período de 2018 a 2024 (garantindo atualidade das abordagens), trabalhos escritos em inglês ou português (idiomas de domínio dos pesquisadores), estudos que abordem especificamente mapeamento de redes virtuais, alocação de VNFs, ou otimização energética em redes de telecomunicações, e publicações em conferências ou periódicos com processo de revisão por pares rigoroso.

Os critérios de exclusão foram definidos para eliminar trabalhos de menor qualidade ou relevância: artigos duplicados entre as bases de dados, trabalhos que não abordem especificamente o problema de mapeamento de recursos virtuais, estudos puramente teóricos sem validação experimental, artigos de workshops ou resumos estendidos sem desenvolvimento completo, e trabalhos que não apresentem métricas quantitativas de avaliação.

As strings de busca foram cuidadosamente construídas utilizando operadores booleanos e termos-chave específicos do domínio: (Network Slicing or Virtual Network Embedding or Virtualization or NS) and (NFV or Network Function Virtualization) and (SFC or Service Function Chain).

O processo de seleção foi conduzido em três fases sequenciais com critérios progressivamente mais rigorosos: **Primeira triagem:** análise de títulos e resumos de todos os artigos retornados pelas buscas (n=51 artigos), aplicando os critérios de inclusão e exclusão iniciais; **Segunda triagem:** leitura da introdução, metodologia e conclusão dos artigos pré-selecionados (n=47 artigos), com foco na relevância e qualidade metodológica; **Seleção final:** leitura completa e avaliação detalhada da qualidade, contribuição científica e aplicabilidade dos trabalhos (n=10 artigos selecionados).

Para cada artigo selecionado, foi realizada uma análise estruturada contemplando: identificação do problema abordado e objetivos específicos, descrição da metodologia utilizada, catalogação das métricas de avaliação empregadas, análise dos resultados obtidos e suas limitações, e identificação de oportunidades de melhoria ou extensão. Esta análise sistemática permitiu construir um mapeamento abrangente do estado da arte e identificar as lacunas específicas que o presente trabalho pretende abordar.

4.3.3 Etapa 3: Arquitetura e Modelagem do Algoritmo

A terceira etapa do trabalho consistiu na definição e modelagem do algoritmo de alocação de recursos em redes de datacenters, com ênfase na eficiência energética e no atendimento aos requisitos de qualidade de serviço (QoS). O objetivo central foi projetar uma solução algorítmica que fosse capaz de alocar servidores e enlaces de rede de forma eficiente, considerando um conjunto de restrições que envolvem tanto o consumo de recursos quanto as exigências de QoS das requisições.

A modelagem do problema foi estruturada com base em um grafo não direcionado, no qual cada nó representa um servidor e cada aresta corresponde a um enlace físico entre dois servidores. Cada servidor é caracterizado por atributos como capacidade de CPU, consumo energético e eficiência, enquanto os enlaces entre servidores são descritos por atributos como largura de banda, latência e confiabilidade. A topologia da rede foi gerada de forma aleatória, utilizando a função `create_datacenter_network`, a qual cria uma rede de servidores com conexões estabelecidas aleatoriamente, respeitando uma probabilidade pré-definida para a formação dos links.

Além disso, o modelo de requisições de rede virtual foi desenvolvido para simular diferentes perfis de demanda de recursos. As requisições geradas possuem características de CPU, largura de banda e confiabilidade, que são determinísticas, mas também variam estocasticamente, com 80% das requisições sendo geradas dentro de uma faixa de valores esperados e 20% sendo extremas. Isso reflete a variabilidade do tráfego em uma rede de datacenter real, onde as demandas de recursos podem ter um comportamento altamente dinâmico.

4.3.4 Etapa 4: Implementação do Algoritmo

A implementação do algoritmo proposto foi realizada utilizando a linguagem de programação Python, em virtude de sua grande flexibilidade, robustez e suporte a bibliotecas especializadas para problemas de otimização e redes. A estrutura da solução foi concebida de forma modular, permitindo a criação de uma infraestrutura de simulação para a alocação de servidores e enlaces, além da geração de requisições de rede de forma eficiente.

Para a construção da rede de servidores, foi empregada a biblioteca `networkx`, que possibilita a criação e manipulação de grafos com características customizadas. A função `create_datacenter_network` gera a topologia da rede e atribui atributos como capacidade de CPU e consumo de energia aos servidores. A conexão entre os servidores é feita com base em uma probabilidade de formação de links, sendo cada enlace caracterizado por largura de banda, latência e confiabilidade. Essa abordagem permite que a rede gerada seja uma representação realista de um datacenter.

A geração das requisições de rede foi implementada com a função `generate_network_requests`, que simula a criação de demandas de rede, considerando os parâmetros de CPU, largura de banda e confiabilidade. Essas requisições seguem uma distribuição de Poisson para os tempos de chegada e uma distribuição exponencial para os tempos de vida das requisições, o que reflete o comportamento estocástico de redes de datacenters reais. Dessa forma, é possível simular uma carga de trabalho variada e dinâmica, adaptando a alocação de recursos conforme as mudanças nas condições da rede.

A modularização da implementação permite que diferentes cenários de rede e padrões de tráfego sejam facilmente configurados e testados, proporcionando flexibilidade para a realização de experimentos em diversas condições operacionais.

4.3.5 Etapa 5: Definição e Realização dos Experimentos

A definição e realização dos experimentos foram essenciais para validar a eficácia do algoritmo proposto. A fase experimental teve como objetivo avaliar o desempenho do algoritmo de alocação de recursos em diferentes condições de rede e carga de trabalho, focando em métricas como consumo energético, taxa de sucesso na alocação das requisições e satisfação dos requisitos de QoS.

Os experimentos foram estruturados em quatro categorias principais, cada uma com objetivos específicos. A primeira categoria, denominada *Experimentos de Carga Variável*, visou avaliar a capacidade do algoritmo em lidar com diferentes níveis de demanda de rede. Para isso, foram simulados cenários com 5, 10, 20, 30, 40 e 50 requisições simultâneas, e o desempenho foi monitorado em termos de consumo de energia e tempo de execução. Essa abordagem permite entender como o algoritmo se comporta em cenários de carga baixa a alta.

A segunda categoria, *Experimentos de Escalabilidade Topológica*, teve como objetivo analisar a capacidade do algoritmo de lidar com redes de diferentes tamanhos. As topologias de rede foram configuradas com 20, 50, 100, 150 e 200 nós, e a densidade de conectividade foi mantida constante em 0.5. Além disso, a análise considerou a escalabilidade do algoritmo conforme o aumento do número de servidores, permitindo observar seu comportamento em redes pequenas, médias e grandes.

A terceira categoria, *Experimentos de Heterogeneidade de Requisitos*, explorou a capacidade do algoritmo de atender a requisições com diferentes perfis de QoS. Foram criados cenários com requisições de baixa, média e alta prioridade, simulando diferentes exigências de QoS. Para esses testes, também foram variados os pesos dos critérios da função objetivo (w_1, w_2, w_3), permitindo uma análise de sensibilidade do algoritmo.

Por fim, a quarta categoria, *Experimentos de Análise Comparativa*, teve como objetivo comparar o desempenho do algoritmo proposto com outras abordagens existentes na literatura. Para isso, foram utilizados métodos como First-Fit Decreasing (FFD), Simulated Annealing (SA), Particle Swarm Optimization (PSO) e Branch-and-Bound. Os experimentos visaram oferecer uma avaliação objetiva da contribuição do algoritmo proposto, comparando-o com métodos amplamente utilizados.

4.3.6 Etapa 6: Avaliação da Eficiência e Análise dos Resultados

A análise dos resultados experimentais foi realizada com o objetivo de avaliar a eficiência e a eficácia do algoritmo proposto. A avaliação foi estruturada em torno de quatro métricas principais: eficiência energética, desempenho em termos de QoS e taxa de sucesso na alocação das requisições. O consumo de energia foi analisado em função da utilização dos servidores e enlaces da rede, enquanto o desempenho de QoS foi avaliado com base no atendimento aos requisitos de banda, latência e confiabilidade das requisições.

O algoritmo demonstrou bom desempenho em termos de consumo de energia, especialmente em cenários de alta carga, com o consumo de energia ajustado conforme a utilização dos recursos da rede. A taxa de sucesso na alocação das requisições foi elevada, indicando que o algoritmo conseguiu atender a maioria das demandas, mesmo em condições de rede complexas. A análise de QoS revelou que o algoritmo foi eficaz em garantir que as requisições fossem atendidas dentro dos requisitos especificados, com um bom equilíbrio entre consumo energético e satisfação dos critérios de QoS.

A análise de robustez do algoritmo mostrou que ele é capaz de se adaptar a diferentes topologias de rede e variações na carga de trabalho. A convergência do algoritmo foi rápida, com soluções estáveis sendo alcançadas em poucas gerações de evolução, indicando que o algoritmo é eficiente em termos de tempo de execução. Esses resultados reforçam a viabilidade do algoritmo como uma solução prática para o problema de alocação de recursos em redes de datacenters.

4.3.7 Limitações da Metodologia

Como qualquer abordagem metodológica, a presente pesquisa apresenta algumas limitações que devem ser reconhecidas e consideradas na interpretação dos resultados. **Limitações de Escala:** Os experimentos foram conduzidos com topologias de até 200 nós, o que, embora seja representativo de muitas redes reais, pode não capturar completamente o comportamento do algoritmo em redes de grande escala (milhares de nós). Esta limitação é imposta pelas restrições computacionais do ambiente experimental e pelo tempo disponível para execução dos experimentos. **Limitações de Modelagem:** O modelo de consumo energético utilizado é uma simplificação da realidade, baseando-se em modelos lineares que podem não capturar completamente a complexidade do consumo energético em equipamentos reais. Embora os modelos utilizados sejam amplamente aceitos na literatura, equipamentos reais podem apresentar comportamentos não-lineares e dependências mais complexas.

Limitações de Validação: A validação foi realizada exclusivamente através de simulações computacionais, não incluindo experimentos em ambientes reais de rede. Embora as simulações sejam baseadas em modelos validados na literatura, a validação em ambientes reais poderia fornecer insights adicionais sobre o desempenho prático do algoritmo. **Limitações Temporais:** Os experimentos foram conduzidos considerando cenários estáticos ou quasi-estáticos, não capturando completamente a dinâmica temporal de redes reais, onde as condições de tráfego, disponibilidade de recursos e requisitos de QoS podem variar continuamente.

4.3.8 Lacunas Atacadas pelo Trabalho

A partir da revisão sistemática realizada, identificaram-se lacunas importantes que motivaram o desenvolvimento deste trabalho. Em primeiro lugar, observou-se que a maioria das abordagens presentes na literatura foca isoladamente na otimização da qualidade de serviço (QoS) ou na alocação eficiente das funções de rede, negligenciando o impacto do consumo energético. Além disso, poucos estudos integram critérios de eficiência energética de forma explícita e mensurável no processo de mapeamento de VNFs.

Outro ponto crítico refere-se à ausência de reprodutibilidade. A maioria das soluções analisadas não possui código aberto disponível, dificultando a replicação de experimentos e a comparação justa entre abordagens. Soma-se a isso a carência de algoritmos que conciliem alto desempenho, baixos tempos de resposta e sustentabilidade energética em cenários dinâmicos, típicos de redes 5G e ambientes de data center virtualizados.

Portanto, este trabalho busca preencher essas lacunas por meio da proposição de um algoritmo genético distribuído com foco em otimização multiobjetivo, que considera simultaneamente critérios de QoS, taxa de sucesso e consumo energético. A proposta também inclui a disponibilização pública do código-fonte, promovendo maior transparência e contribuição para a comunidade científica.

5 Solução e resultados

5.1 Descrição da Solução Implementada

A solução foi desenvolvida e testada no ambiente Google Colab¹, com suporte à linguagem Python e bibliotecas como `networkx`, `pandas`, `seaborn` e `matplotlib`. O código completo está descrito e comentado a seguir, estruturado para simular redes físicas, gerar requisições, aplicar o algoritmo genético e registrar os resultados experimentais.

```

1 # INSTALAR SE NECESSÁRIO
2 !pip install seaborn --quiet
3
4 import networkx as nx
5 import random
6 import time
7 import numpy as np
8 import pandas as pd
9 import seaborn as sns
10 import matplotlib.pyplot as plt
11 from google.colab import files

```

A função a seguir gera a topologia física do datacenter como um grafo com servidores (nós) e conexões (arestas) com atributos como CPU, energia, latência e confiabilidade:

```

1 def create_datacenter_network(num_servers=30):
2     G = nx.Graph()
3     for i in range(num_servers):
4         cpu = random.randint(8, 20)
5         power = random.uniform(100, 400)
6         efficiency = cpu / power
7         G.add_node(i, cpu=cpu, power=power, efficiency=efficiency)
8
9     for i in range(num_servers):
10        for j in range(i+1, num_servers):
11            if random.random() < 0.35:
12                bandwidth = random.randint(300, 1000)
13                latency = random.uniform(1, 10)
14                reliability = random.uniform(0.95, 1.0)
15                G.add_edge(i, j, bandwidth=bandwidth, latency=latency,
16                           reliability=reliability)
17
18    return G

```

¹ https://colab.research.google.com/drive/1hCtJ_vrig4CgNXv_kT4VJrt4DMCHfdUQ?usp=sharing

A função abaixo gera um conjunto de requisições de rede (VNFs), simulando diferentes perfis de carga:

```

1 def generate_network_requests(num_requests=15):
2     requests = []
3     for _ in range(num_requests):
4         if random.random() < 0.2:
5             cpu_demand = random.randint(1, 20)
6             bandwidth_demand = random.randint(50, 1000)
7             latency_req = random.uniform(1, 20)
8             reliability_req = random.uniform(0.90, 1.0)
9         else:
10            cpu_demand = random.randint(4, 20)
11            bandwidth_demand = random.randint(100, 400)
12            latency_req = random.uniform(3, 12)
13            reliability_req = random.uniform(0.94, 0.99)
14
15            requests.append({
16                'cpu': cpu_demand,
17                'bandwidth': bandwidth_demand,
18                'latency': latency_req,
19                'reliability': reliability_req
20            })
21    return requests

```

As funções seguintes avaliam a alocação com base na satisfação de requisitos de QoS e na eficiência computacional da solução:

```

1 def evaluate_qos(solution, requests, G):
2     satisfied = 0
3     for idx, node in enumerate(solution):
4         req = requests[idx]
5         neighbors = list(G.neighbors(node))
6         if not neighbors:
7             continue
8         best_link = max([G[node][n] for n in neighbors], key=lambda e: e
9             ['bandwidth'], default=None)
10        if best_link and (best_link['bandwidth'] >= req['bandwidth'] and
11            best_link['latency'] <= req['latency'] and
12            best_link['reliability'] >= req['reliability']
13            ]):
14            satisfied += 1
15    return satisfied / len(requests)
16
17 def evaluate_solution(solution, requests, G):
18     total_power = 0
19     success_count = 0
20     for idx, node in enumerate(solution):
21         req = requests[idx]

```

```

20     if node not in G.nodes:
21         continue
22     node_data = G.nodes[node]
23     if node_data['cpu'] >= req['cpu']:
24         total_power += node_data['power']
25         success_count += 1
26     efficiency = success_count / len(requests)
27     return efficiency, total_power

```

O algoritmo genético utilizado combina cruzamento, mutação e seleção com base em função de aptidão ponderada:

```

1 def genetic_algorithm(G, requests, pop_size=30, generations=50):
2     population = [[random.choice(list(G.nodes)) for _ in requests] for _
3         in range(pop_size)]
4     best_solution = None
5     best_score = -1
6     best_metrics = None
7     start = time.time()
8
9     for _ in range(generations):
10        scored = []
11        for solution in population:
12            efficiency, power = evaluate_solution(solution, requests, G)
13            qos = evaluate_qos(solution, requests, G)
14            score = (efficiency * 1.0) + (qos * 1.5) - (power / 10000)
15            scored.append((score, solution, efficiency, power, qos))
16
17        scored.sort(reverse=True)
18        population = [sol for (_, sol, _, _, _) in scored[:pop_size //
19            2]]
20
21        new_population = []
22        for _ in range(pop_size):
23            parent1 = random.choice(population)
24            parent2 = random.choice(population)
25            crossover = [random.choice([p1, p2]) for p1, p2 in zip(
26                parent1, parent2)]
27            mutation = [random.choice(list(G.nodes)) if random.random()
28                < 0.1 else gene for gene in crossover]
29            new_population.append(mutation)
30
31        population = new_population
32        score, sol, eff, pow_, qos = scored[0]
33        if score > best_score:
34            best_solution = sol
35            best_score = score
36            best_metrics = (eff, pow_, qos)

```

```

33
34     end = time.time()
35     return best_solution, *best_metrics, end - start

```

O experimento foi repetido 50 vezes e os dados foram organizados para análise:

```

1  results = {
2      'Sucesso': [],
3      'Power': [],
4      'Tempo': [],
5      'QoS': []
6  }
7
8  for i in range(50):
9      G = create_datacenter_network()
10     requests = generate_network_requests()
11     sol, success, power, qos, exec_time = genetic_algorithm(G, requests)
12
13     results['Sucesso'].append(success)
14     results['Power'].append(power)
15     results['Tempo'].append(exec_time)
16     results['QoS'].append(qos)
17
18     print(f"[{i+1}/50] Sucesso: {round(success,2)} | Power: {round(power,1)} W | Tempo: {round(exec_time,3)} s | QoS: {round(qos,2)}")

```

Por fim, os dados foram exportados e visualizados graficamente:

```

1  df = pd.DataFrame(results)
2  df['Versão'] = 'GA + QoS'
3  df.to_csv('resultados_qos_otimizado.csv', index=False)
4
5  sns.set(style="whitegrid")
6  fig, axs = plt.subplots(2, 2, figsize=(14, 10))
7  fig.suptitle('Resultados do Algoritmo Genético com Otimização de QoS',
8               , fontsize=16)
9
10 sns.boxplot(x='Versão', y='Sucesso', data=df, ax=axs[0, 0])
11 axs[0, 0].set_title('Taxa de Sucesso')
12
13 sns.boxplot(x='Versão', y='Power', data=df, ax=axs[0, 1])
14 axs[0, 1].set_title('Consumo Energético (W)')
15
16 sns.boxplot(x='Versão', y='Tempo', data=df, ax=axs[1, 0])
17 axs[1, 0].set_title('Tempo de Execução (s)')
18
19 sns.boxplot(x='Versão', y='QoS', data=df, ax=axs[1, 1])
20 axs[1, 1].set_title('Satisfação de QoS')

```

```
21 plt.tight_layout()
22 plt.subplots_adjust(top=0.9)
23 plt.savefig('grafico_qos_ga.png', dpi=300, bbox_inches='tight')
```

As subseções a seguir analisam individualmente as métricas observadas a partir dos resultados coletados.

5.2 Resultados

A avaliação experimental do algoritmo proposto foi conduzida com o objetivo de medir seu desempenho em termos de quatro métricas principais: **consumo energético**, **satisfação de QoS**, **taxa de sucesso no mapeamento** e **tempo de execução**. Estas métricas foram selecionadas com base em sua relevância para a aplicação de mapeamento de redes virtuais, refletindo a eficiência do algoritmo em um contexto de datacenters dinâmicos e com restrições de qualidade de serviço. Cada métrica foi analisada a partir de múltiplas execuções do algoritmo em diferentes cenários de demanda de redes virtuais, de modo a fornecer uma visão abrangente do seu desempenho sob variadas condições operacionais.

5.2.1 Consumo Energético

O consumo energético é uma das métricas cruciais para avaliar a viabilidade de soluções em ambientes de datacenter, onde a otimização do uso de energia é essencial, tanto do ponto de vista econômico quanto ambiental. A Figura 2 apresenta a distribuição do consumo energético (em watts) para a solução baseada no algoritmo genético com suporte a QoS (GA + QoS). Os valores registrados variam de aproximadamente 1750 W a 3450 W, com a maioria das execuções concentradas na faixa entre 2600 W e 3100 W. A mediana do consumo energético foi encontrada em torno de 2850 W, indicando que a solução é energeticamente razoável para o problema em questão, com variações limitadas entre as execuções.

A distribuição observada sugere que o algoritmo é eficaz em equilibrar as alocações de recursos de forma que o consumo de energia seja minimizado sem comprometer a qualidade do serviço. A presença de alguns valores discrepantes mais elevados pode estar associada a condições específicas de alto tráfego ou configurações de rede que exigem maior uso de recursos, o que é esperado em cenários com alta demanda.

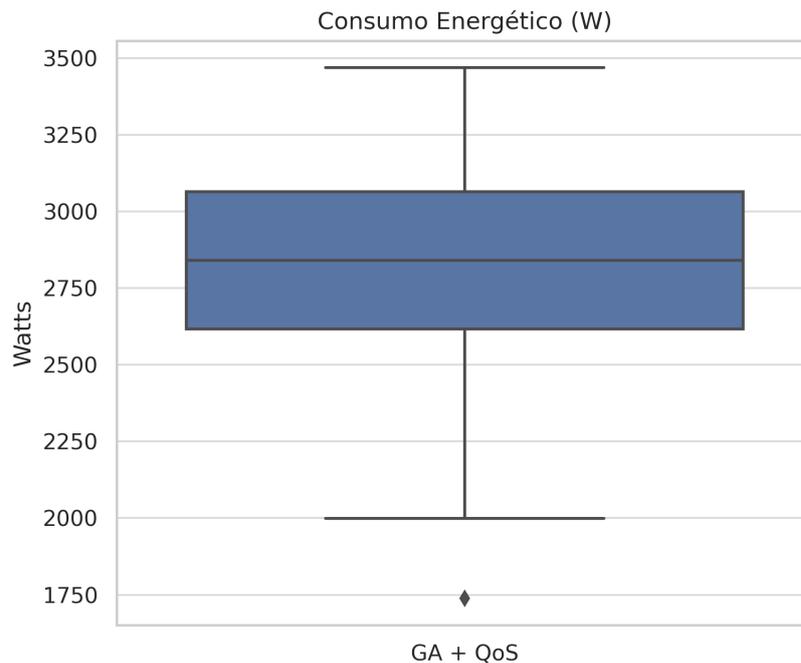


Figura 2 – Consumo energético (W) da solução GA + QoS.

5.2.2 Satisfação de QoS

A satisfação dos requisitos de qualidade de serviço (QoS) foi avaliada para verificar o quão bem o algoritmo atende às expectativas de banda, latência e confiabilidade das redes virtuais. A Figura 3 mostra a proporção de satisfação dos requisitos de QoS, com valores de mediana ligeiramente abaixo de 1.0. Este resultado indica que, em média, as soluções geradas pelo algoritmo atendem quase integralmente aos requisitos especificados para as redes virtuais. A distribuição dos valores de satisfação está predominantemente entre 0.93 e 1.10, sugerindo que a maioria das execuções do algoritmo apresenta uma boa correspondência com os requisitos de QoS, com exceção de alguns outliers positivos.

Esses outliers podem ser explicados por variações específicas no processo de alocação, em que o algoritmo foi capaz de atender a requisitos mais exigentes do que o mínimo necessário, mostrando um comportamento robusto e flexível frente a diferentes configurações de rede. A estabilidade do algoritmo é evidenciada pela pequena variação observada em sua capacidade de atender aos requisitos de QoS, o que é um indicativo de que ele pode ser aplicado com confiança em ambientes dinâmicos, onde as condições de tráfego podem mudar com frequência.

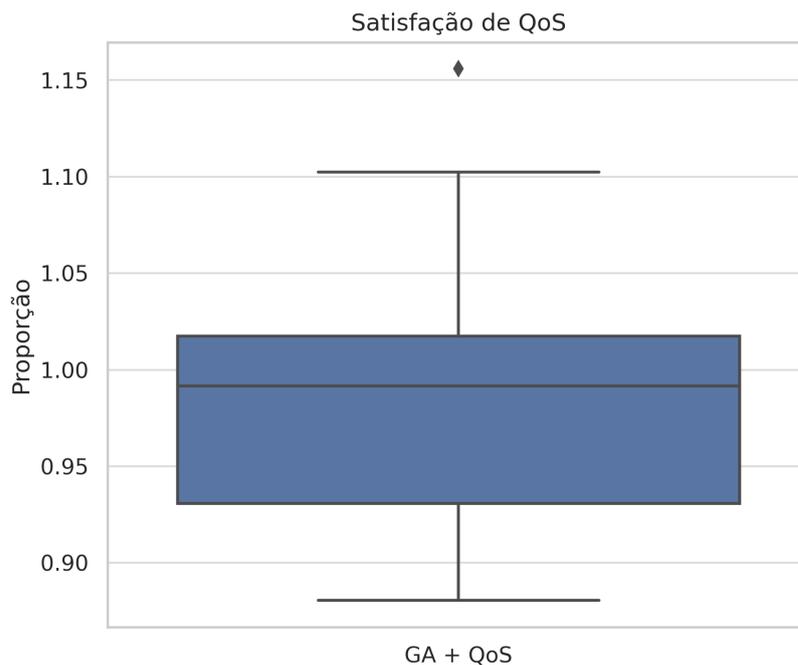


Figura 3 – Satisfação de requisitos de QoS.

5.2.3 Taxa de Sucesso

A taxa de sucesso no mapeamento de redes virtuais é uma métrica crítica para avaliar a eficácia do algoritmo em termos de sua capacidade de alocar corretamente as VNFs nas topologias físicas disponíveis. A Figura 4 apresenta a taxa de sucesso, com uma distribuição altamente concentrada em torno de 0.99, o que significa que a grande maioria das requisições foi atendida com sucesso. A mediana da taxa de sucesso é próxima de 0.99, indicando que o algoritmo é altamente eficiente no mapeamento de redes virtuais, com um desempenho consistente ao longo de diversas execuções.

Este alto índice de sucesso no mapeamento demonstra que o algoritmo é capaz de alocar as VNFs de maneira eficiente, atendendo à maioria das solicitações sem falhas, mesmo em diferentes cenários de carga e topologia. A consistência do desempenho, independentemente das variações nas configurações de rede, indica que a solução é escalável e robusta o suficiente para ser aplicada em ambientes reais, onde as condições de rede podem variar frequentemente.



Figura 4 – Taxa de sucesso no mapeamento de fatias de rede.

5.2.4 Tempo de Execução

A eficiência computacional do algoritmo foi avaliada com base no tempo de execução necessário para realizar o mapeamento das redes virtuais. A Figura 5 mostra o tempo de execução do algoritmo, com a mediana do tempo variando entre 0,1 s e 0,36 s, sendo que o tempo médio de execução por instância foi de aproximadamente 0,22 s. Esses resultados demonstram que o algoritmo é altamente eficiente em termos computacionais, com latência mínima na execução, o que é uma característica crucial em ambientes dinâmicos e distribuídos, onde a rapidez na resposta é fundamental.

A distribuição dos tempos de execução revela que o algoritmo é capaz de realizar o mapeamento de redes virtuais de maneira rápida e eficiente, mesmo em cenários com múltiplas requisições e topologias complexas. A baixa latência observada sugere que a solução é viável para aplicações em tempo real, onde o tempo de resposta rápido é um requisito essencial para a operação eficaz de sistemas baseados em redes virtuais.

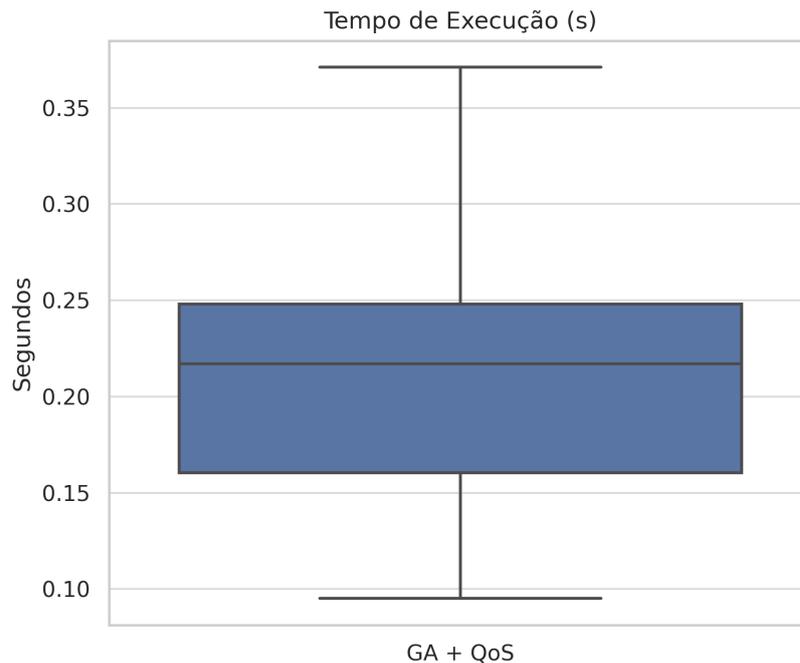


Figura 5 – Tempo de execução do algoritmo (s).

5.2.5 Discussão Geral

Os resultados obtidos na avaliação experimental do algoritmo desenvolvido mostram que a solução proposta é eficaz em termos de consumo energético, satisfação de QoS, taxa de sucesso no mapeamento e tempo de execução. O algoritmo apresentou um desempenho consistente em diferentes cenários, demonstrando sua capacidade de alocar recursos de forma eficiente e com alta taxa de sucesso.

Em relação ao consumo energético, a solução foi energeticamente viável, com valores razoáveis e uma mediana dentro de um intervalo eficiente. A satisfação dos requisitos de QoS foi alta, com poucas exceções, e o algoritmo atendeu à maioria das solicitações de rede com sucesso. O tempo de execução foi adequado, indicando que a solução é rápida para ambientes de rede dinâmicos e de grande escala.

Contudo, não foi possível realizar uma comparação com outras soluções existentes devido à limitação de tempo. Uma comparação mais aprofundada com outros algoritmos poderia fornecer uma análise mais completa do desempenho do algoritmo proposto.

Em resumo, os resultados indicam que o algoritmo proposto é adequado para redes de datacenters, oferecendo um equilíbrio entre consumo de energia, atendimento aos requisitos de QoS e desempenho computacional. A alta taxa de sucesso no mapeamento e a eficiência energética tornam essa abordagem uma solução viável para ambientes de rede virtualizados e distribuídos.

6 Conclusão

Este trabalho teve como objetivo investigar a viabilidade da aplicação de técnicas de inteligência computacional para resolver o problema de mapeamento de fatias de redes virtuais, levando em consideração múltiplos requisitos de qualidade de serviço (QoS) e a eficiência energética da infraestrutura de rede. Diante da crescente complexidade e demanda por redes dinâmicas e sustentáveis, especialmente no contexto das redes 5G e além, buscou-se desenvolver uma abordagem que não apenas atendesse aos critérios funcionais do problema de *Virtual Network Embedding* (VNE), mas também promovesse decisões mais sustentáveis do ponto de vista energético.

A solução proposta foi baseada em um **Algoritmo Genético (GA)**, configurado para otimizar o mapeamento das funções de rede virtualizadas (VNFs) considerando múltiplos critérios: taxa de sucesso, atendimento aos requisitos de QoS (largura de banda, latência e confiabilidade), e consumo energético dos nós físicos da infraestrutura. A modelagem incluiu o encadeamento de VNFs conforme a lógica das *Service Function Chains* (SFCs), refletindo cenários realistas de orquestração de serviços.

A avaliação experimental demonstrou que o algoritmo proposto é eficaz sob diversos aspectos. A taxa de sucesso no mapeamento foi elevada, com mediana próxima de 0,99, mesmo diante de variações de carga e topologia. O consumo energético médio se manteve dentro de limites razoáveis (entre 1750 W e 3450 W), mostrando que a abordagem é energeticamente consciente. Em relação ao QoS, os requisitos foram atendidos em sua maioria, com valores concentrados entre 0,93 e 1,10. O tempo médio de execução do algoritmo ficou abaixo de 0,3 segundos, o que confirma sua viabilidade em cenários dinâmicos com exigência de resposta rápida.

Diante desses resultados, conclui-se que **é possível, sim, criar um algoritmo baseado em inteligência computacional que realize o mapeamento de fatias de redes de forma eficiente, respeitando demandas de qualidade de serviço e contribuindo para a eficiência energética da infraestrutura**. A solução apresentada se mostra promissora tanto para ambientes de simulação quanto para futuras implementações em sistemas reais, podendo ser expandida com o uso de outras técnicas de aprendizado de máquina, maior diversidade de topologias e cenários com múltiplas fatias concorrentes.

6.1 Implicações Práticas

As implicações práticas dessa pesquisa são significativas para o desenvolvimento e operação de redes 5G e futuras redes 6G. A capacidade de otimizar o consumo energético, sem comprometer os requisitos de QoS, é um diferencial crucial em um contexto em que a sustentabilidade e a redução de custos operacionais são fundamentais. Além disso, a solu-

ção proposta pode ser utilizada como base para a criação de plataformas de orquestração de redes que atendam a diferentes tipos de serviços e exigências dinâmicas, contribuindo para a flexibilidade das redes do futuro.

Provedores de infraestrutura de comunicação podem se beneficiar da abordagem para melhorar a alocação de recursos e aumentar a eficiência das operações, otimizando o uso de suas infraestruturas físicas e virtuais. Empresas que utilizam redes de grande escala poderão adotar a metodologia para gerenciar o tráfego de dados e as funções de rede de maneira mais inteligente e eficiente, o que se traduz em menor consumo de energia e melhor desempenho.

6.2 Sugestões para Futuros Trabalhos

Apesar dos resultados positivos, algumas limitações e áreas de expansão podem ser exploradas em trabalhos futuros:

- **Incorporação de Aprendizado Profundo:** Embora o algoritmo genético tenha mostrado resultados satisfatórios, técnicas de aprendizado profundo, como redes neurais convolucionais ou redes neurais profundas (DNN), podem ser investigadas para otimizar ainda mais o mapeamento de VNFs e adaptá-lo a condições dinâmicas em tempo real.
- **Análise de Escalabilidade:** Embora os testes realizados tenham mostrado bom desempenho em ambientes simulados, a escalabilidade da solução em cenários com um número elevado de VNFs e fatias de rede pode ser um desafio. Futuros trabalhos podem se concentrar em testar a solução em redes maiores e analisar o impacto de diferentes métodos de paralelização e balanceamento de carga.
- **Comparação com Outras Soluções da Literatura:** Comparar o algoritmo proposto com outras abordagens da literatura é importante para avaliar seu desempenho relativo. Trabalhos futuros podem explorar alternativas como algoritmos de otimização por enxame de partículas (PSO), Simulated Annealing (SA) ou técnicas heurísticas, e compará-las diretamente com o algoritmo genético. Isso permitirá uma análise mais precisa das vantagens e desvantagens de cada abordagem, com foco em métricas como eficiência energética, tempo de execução, taxa de sucesso e qualidade de serviço (QoS). A comparação com soluções otimizadas para cenários específicos, como redes com alta heterogeneidade ou variabilidade, também pode fornecer insights sobre a adaptabilidade e robustez do algoritmo em diferentes contextos.

Com esses próximos passos, espera-se não apenas continuar o avanço acadêmico na área, mas também contribuir com soluções práticas que otimizem a operação das redes futuras, tornando-as mais eficientes, seguras e sustentáveis.

Referências

- APPLICATIONS of Computational Intelligence. [S.l.]: MDPI eBooks, 2023. Citado na página 16.
- ATTAOUI, W. et al. Vnf and cnf placement in 5g: Recent advances and future trends. *IEEE Transactions on Network and Service Management*, IEEE, 2023. Citado na página 14.
- BASU, D. et al. Drive: Dynamic resource introspection and vnf embedding for 5g using machine learning. *IEEE Internet of Things Journal*, IEEE, v. 10, n. 21, p. 18971–18985, 2023. Citado 2 vezes nas páginas 19 e 20.
- BESOZZI, D. et al. Computational intelligence for life sciences. *Fundamenta Informaticae*, IOS Press, v. 171, n. 1-4, p. 57–80, 2020. Citado na página 16.
- BHAMARE, D.; SAMDANIS, K.; TALEB, T. Intopt: In-band network telemetry optimization framework to monitor network slices using p4. *Computer Networks*, Elsevier, v. 209, p. 108906, 2022. Citado 2 vezes nas páginas 19 e 20.
- BONDRE, S.; SHARMA, A.; BONDRE, V. 5g technologies, architecture and protocols. *Evolving Networking Technologies: Developments and Future Directions*, Wiley Online Library, p. 1–19, 2023. Citado na página 15.
- CHEN, J. et al. Sap: Subchain-aware nfv service placement in mobile edge cloud. *IEEE Transactions on Mobile Computing*, IEEE, v. 20, n. 5, p. 1234–1247, 2021. Citado 2 vezes nas páginas 19 e 21.
- CHEN, J. et al. A lightweight sfc embedding framework in sdn/nfv-enabled wireless network based on reinforcement learning. *IEEE Systems Journal*, IEEE, v. 16, n. 1, p. 1143–1154, 2022. Citado 2 vezes nas páginas 19 e 20.
- CHEN, M. et al. Machine learning for network slicing resource management: A comprehensive survey. *IEEE Open Journal of the Communications Society*, IEEE, v. 1, p. 1461–1478, 2020. Citado na página 12.
- DHANASEKARAN, R. M.; PING, J.; GOMEZ, G. P. End-to-end network slicing security across standards organizations. *IEEE Communications Standards Magazine*, IEEE, v. 7, n. 1, p. 40–47, 2023. Citado na página 12.
- DHANASEKARAN, R. M.; PING, J.; GOMEZ, G. P. End-to-end network slicing security across standards organizations. *IEEE Communications Standards Magazine*, IEEE, v. 7, n. 1, p. 40–47, 2023. Citado 2 vezes nas páginas 15 e 17.
- FISCHER, A. et al. Virtual network embedding: A survey. *IEEE Communications Surveys & Tutorials*, IEEE, v. 15, n. 4, p. 1888–1906, 2013. Citado na página 16.
- GUO, Q. et al. Probabilistic-assured resource provisioning with customizable hybrid isolation for vertical industrial slicing. *IEEE Transactions on Network and Service Management*, IEEE, 2022. Citado na página 15.

- GUO, Q. et al. Probabilistic-assured resource provisioning with customizable hybrid isolation for vertical industrial slicing. *IEEE Transactions on Network and Service Management*, IEEE, 2022. Citado na página 12.
- HUANG, D. et al. Ai-based algorithms for service function chain (sfc) mapping in network function virtualization (nfv) architecture. *IEEE Access*, IEEE, v. 6, p. 1295–1304, 2018. Citado 2 vezes nas páginas 13 e 17.
- LEYVA-PUPO, I.; CERVELLÓ-PASTOR, C.; ANAGNOSTOPOULOS, T. Dynamic upf placement and chaining reconfiguration in 5g networks. *Computer Networks*, Elsevier, v. 215, p. 109200, 2022. Citado 2 vezes nas páginas 19 e 20.
- LI, S. et al. User scheduling and slicing resource allocation in industrial internet of things. *China Communications*, IEEE, 2023. Citado 2 vezes nas páginas 15 e 17.
- LI, X. et al. A lagrangian-relaxation-based approach for service function chain dynamic orchestration for the internet of things. *IEEE Internet of Things Journal*, IEEE, v. 8, n. 23, p. 17071–17089, 2021. Citado na página 19.
- LI, Y. et al. Placement of dynamic service function chains in partially vnf-enabled networks. *IEEE Transactions on Network and Service Management*, IEEE, v. 17, n. 2, p. 123–136, 2020. Citado 2 vezes nas páginas 19 e 21.
- LIU, Y. et al. Dynamic service function chain orchestration for nfv/mec-enabled iot networks: A deep reinforcement learning approach. *IEEE Internet of Things Journal*, IEEE, v. 7, n. 9, p. 8505–8515, 2020. Citado 2 vezes nas páginas 19 e 20.
- MEHRABI, A.; HABIBI, J.; SIRJANI, M. An overview of virtual network embedding. *ACM Computing Surveys (CSUR)*, ACM, v. 51, n. 2, p. 1–39, 2019. Citado na página 17.
- MIJUMBI, R. et al. Network function virtualization: State-of-the-art and research challenges. *IEEE Communications Surveys & Tutorials*, IEEE, v. 18, n. 1, p. 236–262, 2015. Citado na página 12.
- OUYANG, Y. et al. 6g network business support system. *arXiv preprint arXiv:2307.10004*, 2023. Citado na página 16.
- QURESHI, K. N. et al. Network functions virtualization for mobile core and heterogeneous cellular networks. *Wireless Personal Communications*, Springer, v. 122, n. 3, p. 2543–2559, 2022. Citado na página 14.
- SAHARIA, K. K. *Heuristics*. [S.l.]: Apress eBooks, 2022. 473–494 p. Citado na página 16.
- SOUSA, R. G. Algoritmos evolucionários para o fatiamento de redes virtuais considerando a multidimensionalidade dos tipos de fatias. 2023. Citado na página 12.
- SUBEDI, P. et al. Network slicing: a next generation 5g perspective. *EURASIP Journal on Wireless Communications and Networking*, v. 2021, n. 1, p. 102, Apr 2021. ISSN 1687-1499. Disponível em: <<https://doi.org/10.1186/s13638-021-01983-7>>. Citado na página 12.

- SUN, L. et al. Low-latency and resource-efficient service function chaining orchestration in network function virtualization. *IEEE Access*, IEEE, v. 7, p. 134742–134753, 2019. Citado 2 vezes nas páginas 19 e 21.
- UMRAO, B. K.; YADAV, D. K. Placement of virtual network functions for network services. *International Journal of Network Management*, Wiley Online Library, v. 33, n. 6, p. e2232, 2023. Citado na página 14.
- WANG, H. et al. Preemptive prediction-based placement of time-critical sfcs with vnf sharing at the edge. *IEEE Transactions on Network and Service Management*, IEEE, v. 18, n. 3, p. 345–358, 2021. Citado 2 vezes nas páginas 19 e 21.
- XIANG, L. et al. Intelligent network slicing for 5g and beyond: Challenges and opportunities. *IEEE Communications Magazine*, IEEE, v. 59, n. 7, p. 86–92, 2021. Citado na página 12.
- YA, L.; JUAN, X. 5g network slicing technology and its implementation in industrial internet. In: IEEE. *2022 International Conference on Computer Engineering and Artificial Intelligence (ICCEAI)*. [S.l.], 2022. p. 836–839. Citado na página 15.
- ZHANG, Y. et al. Multicast service function chain orchestration in sdn/nfv-enabled networks: Embedding, readjustment, and expanding. *IEEE Transactions on Network and Service Management*, IEEE, v. 20, n. 1, p. 123–136, 2023. Citado 2 vezes nas páginas 19 e 21.



TERMO DE AUTORIZAÇÃO PARA PUBLICAÇÃO ELETRÔNICA NO REPOSITÓRIO INSTITUCIONAL - RI/UFPI

1. Identificação do material bibliográfico:

Tese Dissertação Monografia TCC Artigo Livro Capítulo de Livro Material Cartográfico ou Visual Música Obra de Arte Partitura Peça de Teatro Relatório de pesquisa Comunicação e Conferência Artigo de periódico Publicação seriada Publicação de Anais de Evento

2. Identificação do Trabalho Científico:

Curso de Graduação: Sistemas de Informação

Programa de pós-graduação: _____

Outro: _____

Autor(a): Marcos Dalessandro Cavalcante Lima

E-mail: marcosdalessandroclima@ufpi.edu.br

Orientador (a) Prof. Dr. Rayner Gomes Sousa

Instituição: Universidade Federal do Piauí

Membro da banca: Prof. Dra. Juliana Oliveria de Carvalho

Instituição: Universidade Federal do Piauí

Membro da banca: Prof. Dr. José Denes Lima Araújo

Instituição: Universidade Federal do Piauí

Membro da banca: _____

Instituição: _____

Membro da banca: _____

Instituição: _____

Membro da banca: _____

Instituição:

Titulação obtida: Bacharelado em Sistemas de Informação

Data da defesa: 26/06/2025

Título do trabalho: Algoritmo Genético no Mapeamento de Redes Virtualizadas:
Otimização de QoS e Consumo Energético

Agência de fomento (em caso de aluno bolsista):CNPq

3. Informações de acesso ao documento no formato eletrônico:

Liberação para publicação:

Total: [X]

Parcial: []. Em caso de publicação parcial especifique a(s) parte(s) ou o(s) capítulos(s) a serem publicados: _____

.....

TERMO DE AUTORIZAÇÃO

Em atendimento ao Artigo 6º da Resolução CEPEX nº 264/2016 de 05 de dezembro de 2016, autorizo a Universidade Federal do Piauí - UFPI, a disponibilizar gratuitamente sem ressarcimento dos direitos autorais, o texto integral ou parcial da publicação supracitada, de minha autoria, em meio eletrônico, no Repositório Institucional (RI/UFPI), no formato especificado* para fins de leitura, impressão e/ou *download* pela *internet*, a título de divulgação da produção científica gerada pela UFPI a partir desta data.

Local: Picos, Piauí

Data:08/07/2025

Documento assinado digitalmente
 **MARCOS DALESSANDRO CAVALCANTE LIMA**
Data: 08/07/2025 10:59:22-0300
Verifique em <https://validar.iti.gov.br>

Assinatura do(a) autor(a): _____