



UNIVERSIDADE FEDERAL DE GOIÁS (UFG)
INSTITUTO DE INFORMÁTICA (INF)
PROGRAMA DE PÓS GRADUAÇÃO EM CIÊNCIA DA
COMPUTAÇÃO

GABRIEL MATHEUS FARIA DE ALMEIDA

**Alocação de recursos e posicionamento
de funções virtualizadas em redes de
acesso por rádio desagregadas**

GOIÂNIA
2023



UNIVERSIDADE FEDERAL DE GOIÁS
INSTITUTO DE INFORMÁTICA

TERMO DE CIÊNCIA E DE AUTORIZAÇÃO (TECA) PARA DISPONIBILIZAR VERSÕES ELETRÔNICAS DE TESES

E DISSERTAÇÕES NA BIBLIOTECA DIGITAL DA UFG

Na qualidade de titular dos direitos de autor, autorizo a Universidade Federal de Goiás (UFG) a disponibilizar, gratuitamente, por meio da Biblioteca Digital de Teses e Dissertações (BDTD/UFG), regulamentada pela Resolução CEPEC nº 832/2007, sem ressarcimento dos direitos autorais, de acordo com a [Lei 9.610/98](#), o documento conforme permissões assinaladas abaixo, para fins de leitura, impressão e/ou download, a título de divulgação da produção científica brasileira, a partir desta data.

O conteúdo das Teses e Dissertações disponibilizado na BDTD/UFG é de responsabilidade exclusiva do autor. Ao encaminhar o produto final, o autor(a) e o(a) orientador(a) firmam o compromisso de que o trabalho não contém nenhuma violação de quaisquer direitos autorais ou outro direito de terceiros.

1. Identificação do material bibliográfico

Dissertação Tese Outro*: _____

*No caso de mestrado/doutorado profissional, indique o formato do Trabalho de Conclusão de Curso, permitido no documento de área, correspondente ao programa de pós-graduação, orientado pela legislação vigente da CAPES.

Exemplos: Estudo de caso ou Revisão sistemática ou outros formatos.

2. Nome completo do autor

Gabriel Matheus Faria de Almeida

3. Título do trabalho

Alocação de recursos e posicionamento de funções virtualizadas em redes de acesso por rádio desagregadas

4. Informações de acesso ao documento (este campo deve ser preenchido pelo orientador)

Concorda com a liberação total do documento SIM NÃO¹

[1] Neste caso o documento será embargado por até um ano a partir da data de defesa. Após esse período, a possível disponibilização ocorrerá apenas mediante:

a) consulta ao(à) autor(a) e ao(à) orientador(a);

b) novo Termo de Ciência e de Autorização (TECA) assinado e inserido no arquivo da tese ou dissertação. O documento não será disponibilizado durante o período de embargo.

Casos de embargo:

- Solicitação de registro de patente;
- Submissão de artigo em revista científica;
- Publicação como capítulo de livro;
- Publicação da dissertação/tese em livro.

Obs. Este termo deverá ser assinado no SEI pelo orientador e pelo autor.



Documento assinado eletronicamente por **Kleber Vieira Cardoso, Professor do Magistério Superior**, em 11/09/2023, às 12:52, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do [Decreto nº 10.543, de 13 de novembro de 2020](#).



Documento assinado eletronicamente por **Gabriel Matheus Faria De Almeida, Discente**, em 12/09/2023, às 08:54, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do [Decreto nº 10.543, de 13 de novembro de 2020](#).



A autenticidade deste documento pode ser conferida no site https://sei.ufg.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **4030410** e o código CRC **5BBC28B7**.

GABRIEL MATHEUS FARIA DE ALMEIDA

Alocação de recursos e posicionamento de funções virtualizadas em redes de acesso por rádio desagregadas

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação do Instituto de Informática (INF) da Universidade Federal de Goiás (UFG), como requisito parcial para obtenção do título de Mestre em Ciência da Computação.

Área de concentração: Ciência da Computação.

Orientador: Prof. Dr. Kleber Vieira Cardoso

Co-Orientador: Prof. Dr. Leizer de Lima Pinto

GOIÂNIA
2023

Ficha de identificação da obra elaborada pelo autor, através do
Programa de Geração Automática do Sistema de Bibliotecas da UFG.

Almeida, Gabriel Matheus Faria de
Alocação de recursos e posicionamento de funções virtualizadas em
redes de acesso por rádio desagregadas [manuscrito] / Gabriel
Matheus Faria de Almeida. - .
75 f.

Orientador: Prof. Kleber Vieira Cardoso; co-orientador Leizer de
Lima Pinto.

Dissertação (Mestrado) - Universidade Federal de Goiás, Instituto
de Informática (INF), Programa de Pós-Graduação em Ciência da
Computação, Goiânia, .

Inclui siglas, símbolos, gráfico, tabelas, algoritmos, lista de figuras,
lista de tabelas.

1. redes de acesso por rádio virtualizadas. 2. posicionamento de
funções virtualizadas. 3. alocação de recursos. 4. aprendizado por reforço
profundo. 5. algoritmo genético. I. Cardoso, Kleber Vieira, orient. II.
Título.

CDU 004



UNIVERSIDADE FEDERAL DE GOIÁS

INSTITUTO DE INFORMÁTICA

ATA DE DEFESA DE DISSERTAÇÃO

Ata nº **18** da sessão de Defesa de Dissertação de **Gabriel Matheus Faria de Almeida**, que confere o título de Mestre em Ciência da Computação, na área de concentração em Ciência da Computação.

Aos trinta dias do mês de agosto de dois mil e vinte e três, a partir das catorze horas, na sala 257 do INF, realizou-se a sessão pública de Defesa de Dissertação intitulada **“Alocação de recursos e posicionamento de funções virtualizadas em redes de acesso por rádio desagregadas”**. Os trabalhos foram instalados pelo Orientador, Professor Doutor Kleber Vieira Cardoso (INF/UFG) com a participação dos demais membros da Banca Examinadora: Professor Doutor Leizer de Lima Pinto (INF/UFG), coorientador; Professor Doutor Aldebaro Barreto da Rocha Klautau Júnior (UFPA), membro titular externo; e Professor Doutor Luiz Antonio Pereira Da Silva (Virgina Tech), membro titular externo. A participação dos professores Aldebaro Barreto da Rocha Klautau Júnior e Luiz Antonio Pereira Da Silva ocorreu por meio de videoconferência. O presidente da banca, Prof. Dr. Kleber Vieira Cardoso assina em substituição ao Prof. Dr. Luiz Antonio Pereira Da Silva (Virgina Tech). Durante a arguição os membros da banca não fizeram sugestão de alteração do título do trabalho. A Banca Examinadora reuniu-se em sessão secreta a fim de concluir o julgamento da Dissertação, tendo sido o candidato **aprovado** pelos seus membros. Proclamados os resultados pelo Professor Doutor Kleber Vieira Cardoso, Presidente da Banca Examinadora, foram encerrados os trabalhos e, para constar, lavrou-se a presente ata que é assinada pelos Membros da Banca Examinadora, aos trinta dias do mês de agosto de dois mil e vinte e três.

TÍTULO SUGERIDO PELA BANCA



Documento assinado eletronicamente por **Aldebaro Barreto da Rocha Klautau Junior, Usuário Externo**, em 30/08/2023, às 15:32, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do [Decreto nº 10.543, de 13 de novembro de 2020](#).



Documento assinado eletronicamente por **Kleber Vieira Cardoso, Professor do Magistério Superior**, em 30/08/2023, às 15:43, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do [Decreto nº 10.543, de 13 de novembro de 2020](#).



Documento assinado eletronicamente por **Leizer De Lima Pinto, Professor do Magistério Superior**, em 30/08/2023, às 15:47, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do [Decreto nº 10.543, de 13 de novembro de 2020](#).



Documento assinado eletronicamente por **Gabriel Matheus Faria De Almeida, Discente**, em 30/08/2023, às 15:50, conforme horário oficial de Brasília, com fundamento no § 3º do art. 4º do [Decreto nº 10.543, de 13 de novembro de 2020](#).



A autenticidade deste documento pode ser conferida no site https://sei.ufg.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **3939739** e o código CRC **B9168DE5**.

Referência: Processo nº 23070.043372/2023-05

SEI nº 3939739

Luiz DaSilva

Digitally signed by Luiz

DaSilva

Date: 2023.08.30

15:03:47 -04'00'

Todos os direitos reservados. É proibida a reprodução total ou parcial do trabalho sem autorização da universidade, do autor e do orientador(a).

Gabriel Matheus Faria de Almeida

Bacharel em Ciência da Computação pela Universidade Federal de Goiás. Possui experiência na área de projeto de algoritmos, teoria dos grafos, teoria da computação, redes de computadores, Internet das coisas e redes definidas por software. Desenvolve pesquisa com foco nos seguintes temas: redes móveis sem fio, virtualização, alocação de recurso e avaliação de desempenho.

Dedico este trabalho à minha família, professores e amigos.

Agradecimentos

Agradeço imensamente aos meus orientadores e amigos, Professor Kleber Cardoso e Professor Leizer Pinto. Vocês têm sido uma fonte constante de apoio, guiando minhas decisões, estratégias e moldando minha jornada na formação acadêmica. Agradeço aos professores do Instituto de Informática por todo suporte durante esses anos, seja nos atendimentos individuais ou nas discussões pós-aula. Também não posso deixar de mencionar meus amigos do laboratório de redes de computadores e sistemas distribuídos (LABORA). A colaboração e as conversas de bancada com vocês foram fundamentais.

Manifesto meu sincero agradecimento à Universidade Federal de Goiás e à Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) pelo valioso amparo concedido na forma de bolsa de estudos para a realização deste trabalho. É por meio de instituições públicas de excelência e programas de incentivo à pesquisa como esses que nosso país caminhará à frente.

Agradeço à minha família, uma base inabalável de ajuda ao longo dessa jornada. Primeiramente, agradeço aos meus pais, Valdemar e Marlúcia. Seu apoio e amor incondicional foram pilares fundamentais. À minha irmã, Talita, e ao meu cunhado, David, por todo suporte ao longo de todos esses anos. Mesmo diante das distâncias entre nossas cidades e das limitações, vocês continuam a ser minha força motriz.

Por último, dedico um agradecimento especial à minha companheira de vida, Patrícia. É a você que devo o despertar do desejo de realizar meus sonhos, além de enxergá-los com clareza. Seu amor constante, ternura e companheirismo durante todos esses anos são os motores que me impulsionam a cada novo dia. Juntos, construímos este caminho e sou infinitamente grato por cada passo ao seu lado.

Ciência da computação está tão relacionada aos computadores quanto a astronomia aos telescópios, biologia aos microscópios, ou química aos tubos de ensaio. A ciência não estuda ferramentas. Ela estuda como nós as utilizamos, e o que descobrimos com elas.

Edsger Wybe Dijkstra,
Cientista da computação, Holanda.

Resumo

Almeida, Gabriel Matheus. **Alocação de recursos e posicionamento de funções virtualizadas em redes de acesso por rádio desagregadas**. GOIÂNIA, 2023. 76p. Dissertação de Mestrado. Programa de pós graduação em ciência da computação, Instituto de Informática (INF), Universidade Federal de Goiás (UFG).

A escolha conjunta de divisão funcional e o posicionamento das funções virtualizadas na rede de acesso são essenciais para o uso eficiente dos seus recursos. Esse problema representa um tema de pesquisa atual em redes 5G e Pós-5G, o qual envolve o desafio de escolher simultaneamente o posicionamento das funções virtualizadas, as rotas para o tráfego e o gerenciamento de recursos de computação disponíveis. Neste trabalho, nós apresentamos três abordagens para resolver este problema considerando o cenário de planejamento e duas abordagens considerando o cenário de operação da rede. O primeiro resultado é um modelo de Programação Linear Inteira Mista (MILP), considerando um conjunto genérico de nós de processamento e divisão de carga em múltiplas rotas. A segunda abordagem utiliza conceitos de inteligência artificial e aprendizado de máquina, em que formulamos um agente de aprendizado por reforço profundo. A terceira abordagem utilizada é baseada em meta-heurísticas de busca, via um algoritmo genético. As duas últimas abordagens são formulações de Processos de Decisão de Markov (MDP) que consideram demanda dinâmica nas unidades de rádio. Em todas as formulações, o objetivo é maximizar a centralização de funções da rede enquanto minimiza o custo da solução. A análise das soluções e a comparação de seus resultados mostram que abordagens exatas, tais como nosso MILP, naturalmente, apresentam a melhor solução. Entretanto, em termos de eficiência, o algoritmo genético apresenta o melhor tempo de busca encontrando uma solução de alta qualidade em alguns segundos. O agente de aprendizado por reforço profundo apresenta uma alta convergência, encontrando soluções com alta qualidade para o problema e mostrando capacidade de generalização do problema com topologias diferentes. Por fim, as formulações considerando o cenário de operação da rede com demanda dinâmica apresentam alta complexidade devido ao tamanho do espaço de ações.

Palavras-chave

Redes de acesso por rádio virtualizadas, posicionamento de funções virtualizadas, divisão funcional da pilha de rádio, alocação de recursos, otimização, aprendizado por reforço profundo, algoritmo genético.

Abstract

Almeida, Gabriel Matheus. . GOIÂNIA, 2023. 76p. MSc. Dissertation. Programa de pós graduação em ciência da computação, Instituto de Informática (INF), Universidade Federal de Goiás (UFG).

Jointly choosing a functional split of the protocol stack and placement of network functions in a virtualized RAN is critical to efficiently using the access network resources. This problem represents a current research topic in 5G and Post-5G networks, which involves the challenge of simultaneously choosing the placement of virtualized functions, the routes for traffic and the management of available computing resources. In this work, we present three approaches to solve this problem considering the planning scenario and two approaches considering the network operation scenario. The first result is a Mixed Integer Linear Programming (MILP) model, considering a generic set of processing nodes and multipath routing. The second approach uses artificial intelligence and machine learning concepts, in which we formulate a deep reinforcement learning agent. The third approach used is based on search meta-heuristics, through a genetic algorithm. The last two approaches are Markov Decision Process (MDP) formulations that consider dynamic demand on radio units. In all formulations, the objective is to maximize the network function's centralization while minimizing positioning cost. Analysis of the solutions and comparison of their results show that exact approaches such as MILP naturally provide the best solution. However, in terms of efficiency, the genetic algorithm has the best search time, finding a high quality solution in a few seconds. The deep reinforcement learning agent presents a high convergence, finding high quality solutions for the problem and showing problem generalization capacity with different topologies. Finally, the formulations considering the network operation scenario with dynamic demand are highly complex due to the size of the action space.

Keywords

Virtualized radio access network, virtualized functions placement, radio stack functional splits, resource allocation, optimization, deep reinforcement learning, genetic algorithm.

Sumário

Lista de Figuras	10
Lista de Tabelas	11
1 Introdução	13
2 Fundamentação teórica	18
2.1 Rede de acesso por rádio	18
2.2 Problema de posicionamento das funções da RAN	20
3 Modelo geral de alocação de recurso na vRAN	26
3.1 Introdução	26
3.2 Modelo de sistema	27
3.3 Formulação do problema	28
3.4 Resultados	32
4 Abordagem baseada em aprendizado de máquina	36
4.1 Introdução	36
4.2 Modelo de sistema e formulação do problema	37
4.3 Solução de aprendizado por reforço para solução do problema	39
4.4 Ações, espaço de observação e função de recompensa	39
4.4.1 Ações	39
4.4.2 Espaço de observação e conjunto de ações viáveis	40
4.4.3 Função de recompensa	41
4.4.4 Método de treinamento	42
4.5 Resultados	42
5 Abordagem baseada em meta-heurística	47
5.1 Algoritmo genético	47
5.1.1 Representação dos cromossomos	48
5.1.2 Primeira geração	48
5.1.3 Função de reparo	48
5.1.4 Operadores de variação	50
5.2 Avaliação	51
5.2.1 Avaliação sem decisão de roteamento	51
5.2.2 Avaliação com decisão de roteamento	53

6	Problema de posicionamento de funções da vRAN considerando variação na demanda	55
6.1	Base de dados de variação de demanda	56
6.2	Gerador de carga sintética	58
6.3	Posicionamento dinâmico de funções da vRAN	61
6.3.1	Formulação considerando recursos de processamento gerenciados pela operadora de redes móveis	62
6.3.2	Formulação considerando recursos de processamento terceirizados	63
6.3.3	Análise das formulações apresentadas	64
7	Conclusão, publicações e trabalhos futuros	68
	Referências Bibliográficas	73

Lista de Figuras

1.1	Problema de posicionamento das funções da vRAN.	15
3.1	Conjunto de divisões funcionais consideradas.	28
3.2	Topologias.	33
3.3	Comparando a quantidade de CNs utilizados pelas soluções.	34
3.4	Legenda VNCs.	34
3.5	Comparando as escolhas de VNCs das soluções.	34
3.6	Nível de centralização em função de ϵ .	35
3.7	Tempo de execução em função de ϵ .	35
4.1	VNCs considerados pelo agente DRL.	38
4.2	Topologias utilizadas na avaliação. O ponto vermelho representa o posicionamento da CU, utilizada na formulação de Murti, em azul são representados os CNs.	44
4.3	Nível de centralização das soluções para cada topologia.	44
4.4	VNCs utilizadas em cada instância na avaliação.	45
4.5	Histórico dos sinais de recompensa durante o treinamento.	46
5.1	Uma topologia vRAN e seu cromossomo correspondente.	48
5.2	Nível de centralização no cenário sem roteamento.	52
5.3	Tempo de execução sem considerar roteamento.	52
5.4	Nível de centralização das soluções considerando o roteamento na decisão.	53
5.5	Tempo de execução considerando roteamento.	54
6.1	Quantidade normalizada de usuários conectados a cada período do dia [Chen et al. 2015]. (a) apresenta os dados originais e (b) apresenta os dados filtrados.	56
6.2	CDF da quantidade de RUs e quantidade de leituras na base de dados [Chen et al. 2015].	57
6.3	Exemplo da quantidade de usuários conectados a uma estação base do centro e outra em um bairro periférico na cidade.	57
6.4	Comparando geradores de carga sintética.	61
6.5	Treinamento para topologia com 4 RUs.	66
6.6	Treinamento para topologia com 8 RUs.	67

Lista de Tabelas

2.1	Principais diferenças entre os trabalhos citados.	25
3.1	Parâmetros e variáveis utilizadas na formulação.	29
3.2	Parâmetros utilizados na avaliação dos modelos.	33
4.1	Parâmetros utilizados na avaliação.	43
5.1	Parâmetros utilizados nos testes.	51
7.1	Trabalhos com implementação publicamente disponível.	71

Lista de Abreviaturas

RRH: *Remote Radio Head*
BBU: *Base Band Unit*
C-RAN: *Cloud RAN*
CU: *Centralized Unit*
DU: *Distributed Unit*
RU: *Radio Unit*
3GPP: *3rd Generation Partnership Project*
CN: *Computing Node*
vRAN: *virtualized Radio Access Network*
RRC: *Radio Resource Control*
PDCP: *Packet Data Convergence Protocol*
RLC: *Radio Link Control*
MAC: *Medium Access Control*
PHY: *Physical Layer*
RF: *Radio Frequency*
IA: *Inteligência Artificial*
O-RAN: *Open Radio Access Network*
NG-RAN: *Next Generation Radio Access Network*
TCP/IP: *Transmission Control Protocol/Internet Protocol*
D-RAN: *Distributed Radio Access Network*
VNC: *Viable NG-RAN Configuration*
MMKP: *Multidimensional Multiple-choice Knapsack Problem*
MILP: *Mixed-Integer Linear Programming*
BLP: *Binary Linear Programming*
VM: *Virtual Machine*
ML: *Machine Learning*
DRL: *Deep Reinforcement Learning*
PPO: *Proximal Policy Optimization*
MEC: *Multi-access Edge Computing*
GA: *Genetic Algorithm*

Introdução

Desde o surgimento da tecnologia de redes móveis sem fio, aparelhos celulares estão cada vez mais populares e presentes no cotidiano da sociedade, tornando-se peça importante na evolução da comunicação, relações e comportamentos sociais. O aumento do número de aparelhos celulares conectados às redes móveis nos últimos anos faz com que as operadoras constantemente invistam em sua infraestrutura de rede, a fim de atender a uma demanda cada vez maior. Entretanto, tais melhorias representam custos financeiros às operadoras, que buscam minimizar tais gastos a fim de manter preços competitivos, enquanto oferecem um serviço de alta qualidade aos usuários. Assim, a cada nova geração de redes móveis, as operadoras celulares, em conjunto com órgãos padronizadores e academia, propõem soluções tecnológicas que visam minimizar o custo de operação e infraestrutura da rede, enquanto aumentam a eficiência do serviço ofertado.

Essa proposta para as redes móveis de terceira geração (3G) foi a separação do funcionamento do rádio em dois níveis, conectados via uma interface de transporte chamada *fronthaul*. O primeiro nível é chamado cabeça de rádio remota (RRH), responsável por processar as funções do rádio e fica posicionado próximo às unidades de rádio. O segundo nível é chamado unidade de banda base (BBU) e é responsável pelo processamento de funções de banda base relacionadas ao controle da rede. Já nas redes móveis de quarta geração (4G), introduziu-se a proposta da *cloud-RAN (C-RAN)*, que propõe a centralização das BBUs a fim de minimizar os custos de operação da rede e melhorar a eficiência do gerenciamento de recursos.

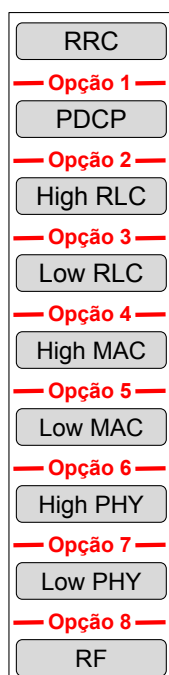
A partir da arquitetura C-RAN, é possível controlar a rede de forma adaptativa conforme a mobilidade dos usuários, o que traz diversos avanços para o gerenciamento de recursos da rede. Alguns destes benefícios são o compartilhamento de recurso de rádio, melhor controle de interferência entre antenas e melhor gestão de transferência de usuários, e estão diretamente relacionados com o aumento da eficiência da rede. Entretanto, o principal problema em relação a essa proposta é a demanda de altas capacidades no *fronthaul*, o que torna a C-RAN uma configuração de posicionamento de alto custo e baixa viabilidade em larga escala, principalmente em redes maiores com centenas ou milhares de unidades de rádio.

Visando minimizar o custo excessivo previsto na C-RAN, mas visando manter seus benefícios, as redes móveis de quinta geração (5G) apresentam uma arquitetura que visa quebrar a pilha de funções de rádio em até três partes (divisões funcionais). Assim, as funções da pilha de rádio podem ser posicionadas de forma distribuída em até três nós da rede chamados: (i) Unidade Central (CU), (ii) Unidade Distribuída (DU) e (iii) Unidade de Rádio (RU). A arquitetura 5G propõe uma rede de transporte chamada *crosshaul*, que conecta os nós da rede de acesso e permite o posicionamento dinâmico de tais funções conforme a demanda em cada região da rede. Além disso, como forma de redução de custos, os órgãos padronizadores das redes 5G [3GPP 2018, ITU-T 2018, 3GPP 2017] sugerem a virtualização da pilha de funções de rádio (vRAN), a fim de abandonar a necessidade de equipamentos específicos para o processamento das funções e permitindo a utilização de máquinas de propósito geral.

A partir desta nova arquitetura proposta para as redes 5G, surge um novo problema de alocação de recursos chamado posicionamento de funções da vRAN. Assim, para cada unidade de rádio na rede, é necessário posicionar um conjunto de funções de rádio virtualizadas em nós de processamento espalhados pela rede e conectados via interface *crosshaul*. As funções podem ser posicionadas de forma distribuída, desde que se cumpram os requisitos de atraso e vazão entre elas. Além disso, a partir da virtualização das funções, é possível utilizar o conceito de centralização definido na arquitetura C-RAN, porém apresentando custos mais baixos. No entanto, o desenvolvimento da arquitetura virtualizada é um problema desafiador, dado que a rede de transporte possui requisitos de rede rigorosos, topologias heterogêneas em diferentes formatos e demanda que varia em relação ao tempo [ITU-T 2018, Sehier et al. 2019].

O problema de posicionamento de funções da vRAN pode ser definido como um problema de otimização [Masdari et al. 2016, Laghrissi e Taleb 2018] baseado na melhor decisão conjunta entre a escolha de divisão funcional e caminho para roteamento para cada unidade de rádio, enquanto satisfaz as restrições de processamento e requisitos de rede de cada divisão funcional. As divisões funcionais representam as oito opções de quebra da pilha de funções da vRAN apresentadas nos padrões. Cada divisão funcional define o posicionamento das funções em nós de processamento genéricos (CNs) que podem executar qualquer conjunto de funções virtualizadas atuando como qualquer elemento da vRAN (vCUs e/ou vDUs). A Figura 1.1 ilustra os elementos do problema de posicionamento das funções da vRAN. À esquerda, são apresentadas as oito divisões funcionais, e à direita, ilustramos um exemplo de topologia de vRAN, representando as RUs como triângulos e os CNs como retângulos. A figura ainda mostra o posicionamento das vCUs e vDUs para ambas as unidades de rádio na rede e destaca as rotas escolhidas para rotar o tráfego de cada uma delas.

Divisões funcionais



Topologia vRAN

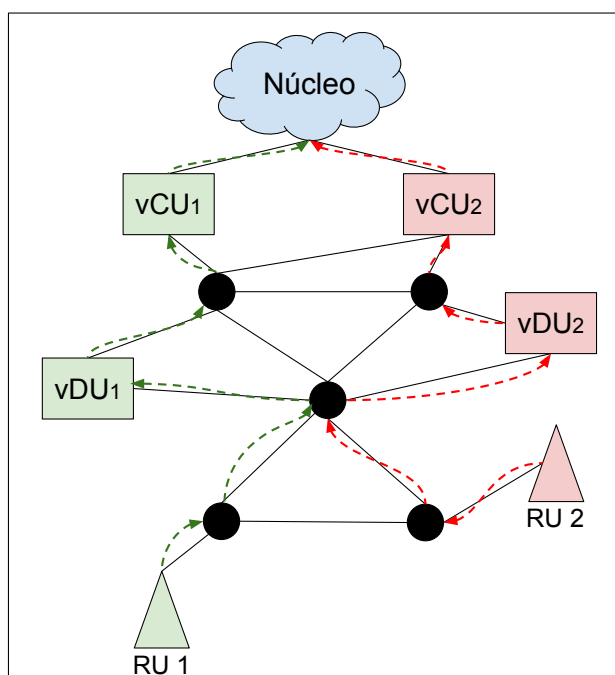


Figura 1.1: Problema de posicionamento das funções da vRAN.

Diversos trabalhos na literatura [[Garcia-Saavedra et al. 2018](#), [Fonseca et al. 2019](#), [Garcia-Saavedra et al. 2018](#), [Murti et al. 2020](#), [Morais et al. 2022](#)] apresentam formulações e soluções para o problema de posicionamento de funções da vRAN. No entanto, esse continua sendo um problema em aberto que vem sendo amplamente explorado na literatura, por meio de análises de soluções e formulações que melhor representam o cenário real. Os trabalhos citados utilizam métodos exatos para a resolução do problema, em que o objetivo é encontrar soluções ótimas de posicionamento das funções da vRAN. Entretanto, tais modelos exatos são bem conhecidos por pertencerem à classe de problemas NP-completos. Isso, consequentemente, faz com que soluções exatas, que realizam buscas ótimas, apresentem um desempenho ruim em cenários reais, com alto tempo de processamento, baixa escalabilidade e alto consumo de recursos computacionais.

Além disso, os trabalhos citados apresentam soluções para o problema de posicionamento de funções da vRAN considerando o seu cenário de planejamento. Neste cenário, é assumido um estado de demanda saturado e estático nas unidades de rádio, ou seja, a demanda de pior caso em que toda RU demanda capacidade máxima de comunicação. Por um lado, as soluções encontradas neste cenário de planejamento permitem encontrar soluções que são viáveis a todo instante da rede, independentemente da demanda nas RUs. Por outro lado, este cenário não representa o estado de operação da rede, em que a demanda nas RUs é dinâmica e varia durante o dia, acompanhando a mobilidade dos usuários na topologia.

Com isso, em [Murti et al. 2023, Alba e Kellerer 2022, Joda et al. 2022, Pamuklu et al. 2021, Gupta et al. 2022], os autores apresentam formulações para o problema considerando o estado de operação da rede, em que a demanda varia ao longo do tempo. Em tais trabalhos, os autores apresentam formulações exatas, resolvidas por meio de arcabouços de otimização, e formulações não exatas, implementadas por soluções de inteligência artificial, avaliando os modelos e apresentando soluções dinâmicas que, ao longo do prazo, reduzem significativamente o custo de operação da rede. Entretanto, para avaliar os modelos formulados, os autores utilizam dados de variação de demanda de redes reais, tais como os dados apresentados em [Chen et al. 2015], que são raramente compartilhados devido a políticas de privacidade dos dados. Isso dificulta a reprodução e comparação dos resultados.

Considerando tais problemas de escalabilidade dos modelos exatos e a necessidade prática de modelos que apresentem soluções de qualidade em tempo eficiente, o foco deste trabalho é um projeto baseado na avaliação de diferentes formulações para o problema de posicionamento de funções da vRAN, considerando o cenário de planejamento e operação da rede. Apresentando soluções não exatas que encontram soluções de qualidade em tempo hábil e uma formulação mais abrangente do problema que considera divisão de carga no roteamento multicaminhos. Este trabalho avança o estado da arte no problema de posicionamento de funções da vRAN, fazendo as seguintes contribuições:

- Formulamos e resolvemos um problema de otimização para a tarefa conjunta de posicionamento de funções da vRAN em nós de processamento de propósito geral e o roteamento do fluxo para cada unidade de rádio. Nessa formulação, consideramos a divisão de carga no roteamento entre diferentes rotas e recursos de rede heterogêneos. O objetivo geral do modelo apresentado é observar o impacto de considerar a divisão de fluxo no roteamento, avaliando (i) a qualidade das soluções encontradas, (ii) a complexidade computacional do problema, (iii) o poder de escalabilidade do modelo e (iv) a eficiência do modelo no processo de busca pela solução ótima.
- Apresentamos um modelo de Inteligência Artificial (IA) utilizando técnicas de aprendizado de máquina para resolver o problema de posicionamento de funções da vRAN. Nesse modelo, consideramos uma formulação do problema que apresentamos em [Morais et al. 2022]. O objetivo geral do modelo é avaliar a utilização de técnicas de aprendizado de máquina para resolver o problema de posicionamento de funções da vRAN, analisando o tempo de treinamento e convergência do agente, a qualidade das soluções encontradas e a capacidade de generalização e abstração do modelo em cenários com recursos heterogêneos.
- Desenvolvemos um algoritmo genético para resolver o problema de posicionamento de funções da vRAN, utilizando métodos evolutivos de busca, e resolvendo o

problema de forma não exata via uma busca heurística no espaço de soluções. O objetivo geral do algoritmo genético é avaliar a utilização de métodos evolutivos para resolver o problema de posicionamento das funções da vRAN, analisando o desempenho da heurística de busca implementada, a qualidade da população inicial através da utilização de uma heurística de criação, a taxa de sobrevivência dos indivíduos durante o processo evolutivo e a qualidade das soluções encontradas.

- Examinamos uma base de dados contendo registros das flutuações na demanda de uma rede real. Realizamos a classificação desses registros e aplicamos filtros criteriosos para selecionar as leituras que serão empregadas na criação de um gerador de carga sintética de alta precisão. Formulamos e implementamos um gerador de carga sintética utilizando conceitos de MDP. Esse gerador desempenha um papel fundamental na geração de conjuntos de dados para treinamento e avaliação de modelos que requerem informações sobre as flutuações na demanda da rede de acesso. Formulamos e implementamos o problema de posicionamento dinâmico de funções da vRAN utilizando conceitos de MDP, considerando a variação de demanda na rede de acesso. Para a formulação desse problema, considerando dois cenários práticos distintos. No primeiro, consideramos a infraestrutura de propriedade da operadora de redes móveis; no segundo, consideramos a terceirização da carga de processamento para um provedor de computação em nuvem. Em ambos os cenários, avaliamos as particularidades do MDP proposto, analisando a complexidade e as características do ambiente. Posteriormente, demonstramos a eficácia da solução proposta por meio da execução de simulações em duas topologias diferentes da vRAN.

O texto a seguir está organizado da seguinte forma: no Capítulo 2, apresentamos uma revisão da literatura. No Capítulo 3, apresentamos o modelo de otimização exato que resolve o problema, considerando divisão de carga no roteamento. Nos Capítulos 4 e 5, apresentamos as formulações não exatas que resolvem o problema de posicionamento de funções da vRAN, ou seja, agente de IA e algoritmo genético, respectivamente. No Capítulo 6, apresentamos os resultados e formulações do problema considerando variação de carga nas RUs. Por fim, no Capítulo 7, apresentamos as considerações finais que englobam os resultados e contribuições alcançados por este estudo. Destacando as publicações e produções que surgiram a partir deste trabalho, bem como as conclusões extraídas das formulações e resultados apresentados. Além disso, apresentamos indicações de possíveis direções para trabalhos futuros para o problema de posicionamento de funções da vRAN.

Fundamentação teórica

Neste capítulo, realizamos uma revisão da literatura a fim de definir o escopo deste trabalho, apresentando conceitos gerais sobre a rede de acesso por rádio, a virtualização dos protocolos de rádio e o problema de posicionamento das funções da vRAN.

2.1 Rede de acesso por rádio

A rede de acesso por rádio é o principal componente em sistemas de redes móveis sem fio, pois é responsável por oferecer, conectar, coordenar e gerenciar o acesso dos equipamentos de usuários. Os equipamentos de usuários podem ser aparelhos celulares, computadores, sensores ou qualquer máquina que esteja conectada à rede sem fio. A rede de acesso domina de forma majoritária os custos de infraestrutura e operação das operadoras de redes móveis, sendo o principal ponto de investimento e manutenção da arquitetura 5G. Desta forma, nos últimos tempos, academia e indústria apresentam esforços na construção de soluções que visam aumentar a eficiência do serviço ofertado enquanto reduzem os custos de operação e infraestrutura da rede de acesso.

Atualmente, tais esforços envolvem abordagens como a padronização de interfaces abertas para promover a interoperabilidade de vários fornecedores, desagregação e virtualização de funções de rede para melhorar a flexibilidade e portabilidade na rede de acesso. Observamos o apoio a essas abordagens nas iniciativas de órgãos de padronização (3GPP, ITU e ETSI) e alianças de empresas (O-RAN, TIP e GSMA), publicando documentos nesse contexto, como a nova arquitetura NG-RAN [ETSI 2019], a evolução da rede de transporte para suportar IMT-2020/5G [ITU-T 2018] e arquitetura de nuvem e cenários de implantação para vRAN [Open RAN Alliance 2020].

A nova arquitetura da rede de acesso por rádio proposta nos padrões [ETSI 2019] é composta por dois elementos principais. O primeiro elemento é um conjunto de unidades de rádio responsáveis por proporcionar acesso sem fio aos equipamentos de usuários, alocando recursos para os usuários conectados e gerenciando a admissão de novos usuários conforme a demanda. O segundo elemento é um conjunto de nós de processamento responsáveis por processar os símbolos e informações provenientes da

comunicação dos equipamentos de usuário conectados, visando transmitir as informações de acordo com seu tipo de acesso. Tais elementos estão conectados via um conjunto de nós de retransmissão e encaminhamento denominado rede de transporte.

A rede de transporte tem o papel de transmitir os dados da comunicação entre as unidades de rádio, nós de processamento e núcleo da rede e pode ser dividida em até três partes. A primeira parte é chamada *fronthaul*, que conecta as unidades de rádio aos nós de processamento. A segunda parte é denominada *midhaul* e conecta nós de processamento a nós de processamento. A terceira parte é denominada *backhaul* e conecta nós de processamento ao núcleo da rede. Entretanto, a nova arquitetura NG-RAN [ETSI 2019] define um novo conceito de rede de transporte chamado *crosshaul*. No *crosshaul*, a separação dos fluxos em *fronthaul*, *midhaul* e *backhaul* é feita de forma lógica e não mais de forma física. Assim, em um determinado ponto da rede de transporte podem ser trafegados dados que antes eram classificados como *fronthaul*, *midhaul* e *backhaul* ao mesmo tempo, abandonando assim a necessidade de segmentação dos fluxos em infraestruturas distintas.

Os principais avanços na tecnologia e melhoria da rede de acesso se dão graças ao processo de desagregação e virtualização das funções de rede. As funções de rede nas redes 5G podem ser representadas como uma pilha de protocolos, similar ao protocolo TCP/IP, em que cada camada processa os dados e encapsula informações de acordo com sua função. Os padrões [ITU-T 2018, Open RAN Alliance 2020] definem a divisão da pilha de protocolos da vRAN em até três conjuntos de funções de rede e os espalham em vários nós: vCU, vDU e RU (ver Fig. 1.1). Essa divisão significa que as operadoras podem escolher o número de divisões funcionais (0, 1 ou 2) e onde cada conjunto de funções da vRAN é executado [ITU-T 2018, Marsch et al. 2018]. Além disso, as funções de rede podem ser virtualizadas e facilmente movidas para diferentes nós de processamento, desde que a latência e a taxa de transferência sejam satisfeitas conforme os requisitos de tal quebra na pilha de protocolos [3GPP 2017].

Desta forma, como discutido no capítulo anterior, a virtualização das funções da rede de acesso nos leva a um problema de decisão sem precedentes e desafiador, chamado problema de posicionamento de funções da vRAN. Atualmente, diversos trabalhos na literatura apresentam soluções que visam resolver esse problema em aberto, a fim de apresentar soluções que maximizem a eficiência da rede enquanto minimizam o custo da solução. Na seção a seguir, iremos discutir tais avanços na literatura e apresentar os resultados e desafios encontrados no estado da arte.

2.2 Problema de posicionamento das funções da RAN

A centralização das funções da vRAN melhora o desempenho por meio da ativação de recursos de rádio avançados, que possibilitam a melhoria no gerenciamento de recursos da rede e do serviço oferecido ao usuário [Larsen et al. 2019]. Desta forma, a centralização é um alvo comum em trabalhos na literatura que visam resolver o problema de posicionamento das funções da rede de acesso. Esse problema, como discutido no Capítulo 1, pode ser definido como um problema de otimização, onde o objetivo é maximizar a eficiência da rede, enquanto minimiza o custo da solução. Nesta seção, iremos apresentar alguns trabalhos na literatura que visam resolver esse problema e discutir as contribuições e estratégias utilizadas.

Em [Garcia-Saavedra et al. 2018], os autores buscam maximizar a centralização de funções da vRAN em uma única CU co-localizada com o núcleo da rede. Os autores utilizaram a centralização em um único ponto (única CU) baseando-se em princípios da C-RAN, que apresenta a melhor eficiência para a rede de acesso em termos de centralização. Entretanto, ao escolherem essa abordagem, o problema formulado pelos autores não explora a possibilidade de posicionamento das funções em mais de um ponto de centralização, o que pode limitar a solução. Além disso, os autores utilizam na formulação divisões funcionais fixas, com valores bem definidos, o que prejudica a capacidade de adaptação do modelo em cenários mais atuais que consideram uma gama maior de possíveis divisões funcionais. Por outro lado, este trabalho apresentou uma formulação pioneira para o problema, fazendo com que a estrutura de sua formulação fosse largamente adotada na literatura em outros trabalhos.

Em [Fonseca et al. 2019], os autores apresentam uma formulação semelhante à formulação apresentada em [Garcia-Saavedra et al. 2018], explorando dois pontos negativos na formulação anterior. O primeiro foi a ausência da divisão funcional D-RAN, que posiciona todas as funções virtualizadas próximas da antena. O segundo ponto foi considerar o posicionamento da CU de forma livre, ou seja, não apenas co-localizada com o núcleo da rede. Por este motivo, os autores conseguem atingir uma centralização maior de funções, aumentando assim a eficiência da rede de acesso. Entretanto, a formulação utilizada é derivada da formulação apresentada em [Garcia-Saavedra et al. 2018], ou seja, também apresenta os problemas de considerar-se apenas um conjunto fixo de divisões funcionais, limitando a adequação do modelo.

Em [Garcia-Saavedra et al. 2018], os autores focam em maximizar a centralização das funções da vRAN em múltiplas CUs posicionadas de forma fixa na rede. Nesse trabalho, os autores apresentam uma melhoria na centralização das funções devido à presença de mais pontos de centralização. Por outro lado, a formulação apresentada assume que as RUs (unidades de rádio) conectadas a uma determinada CU devem utilizar a mesma

quebra da pilha de protocolos, ou seja, a mesma divisão funcional. Desta forma, a formulação não explora a capacidade do *crosshaul* de lidar com fluxos dinâmicos que variam conforme a divisão funcional realizada, limitando assim as possibilidades de centralização e redução de custos.

Todos os trabalhos anteriores apresentados utilizavam em sua formulação um conceito de separação dos protocolos da rede de acesso que posicionavam as funções em até no máximo dois nós de processamento. Entretanto, avanços recentes apresentados pelos padrões [3GPP 2017, 3GPP 2018] definem que a pilha de protocolos da rede de acesso pode ser dividida em até três partes e posicionadas em até três nós de processamento (vCU, vDU e RU). Desta forma, os trabalhos discutidos a seguir apresentam soluções que consideram esse cenário.

Em [Murti et al. 2020], os autores utilizam a nova determinação dos padrões de realizar a quebra da pilha de protocolos em até três partes. Entretanto, os autores utilizam a estrutura apresentada em seu trabalho anterior [Garcia-Saavedra et al. 2018], e acabam apresentando as mesmas desvantagens em relação à centralização, principalmente ao considerarem vDUs fixas e co-localizadas com as unidades de rádio, o que limita fortemente a capacidade de centralização das funções da vRAN.

Em [Morais et al. 2022], nós apresentamos o PlaceRAN, uma formulação para o problema de posicionamento das funções da vRAN que suporta qualquer número de divisões funcionais de forma dinâmica e adaptativa. O objetivo da formulação é maximizar a centralização das funções da vRAN (eficiência) enquanto minimiza o número de nós de processamento utilizados (custo). Diferente dos trabalhos anteriores, nossa formulação assume pontos de centralização (CUs e DUs) livres, ou seja, sem localização fixa. Isso é possível devido à evolução da rede de acesso apresentada na NG-RAN [ETSI 2019], que apresentam os conceitos de vCUs e vDUs e possibilitam seu posicionamento de forma dinâmica em qualquer nó de processamento da rede. Nesse caso, não importa qual nó atue como vCU ou vDU, desde que os requisitos de latência, taxa de transferência e capacidade de processamento sejam atendidos. Assim, neste trabalho, mostramos que nossa formulação consegue atingir altos níveis de centralização, utilizando o menor número de nós de processamento possível, mesmo em cenários com escassez de recursos.

Em [Almeida et al. 2022], nós evoluímos a formulação apresentada no PlaceRAN [Morais et al. 2022] introduzindo um novo modelo de otimização chamado MuStRAN. Nessa nova formulação, nós consideramos um ponto extremamente importante no roteamento da rede de transporte, a divisão de fluxo por múltiplas rotas. Por esse motivo, o modelo apresentou melhorias em termos de centralização e custo da solução quando comparado ao seu modelo precursor (PlaceRAN), principalmente em cenários com maior competição por recursos. É importante destacar que a formulação apresentada no MuStRAN será detalhada no Capítulo 3 por se tratar de uma contribuição derivada deste traba-

lho de mestrado.

Além de apresentarem soluções para o problema de posicionamento de funções da vRAN, outro ponto em comum em todos os trabalhos descritos até o momento é a forma de resolução utilizada para encontrar uma solução para o problema. Todos os trabalhos apresentam uma formulação matemática para o problema e, além disso, apresentam resultados analisando a solução ótima encontrada a partir de métodos exatos de busca. Desta forma, todos os trabalhos descritos apresentam a alta complexidade computacional da busca ótima, que é bem conhecida pertencente à classe dos problemas NP-Completo. Por esse motivo, as soluções apresentam problemas de escalabilidade ao resolver instâncias reais do problema com centenas e milhares de unidades de rádio e nós de processamento.

Uma alternativa para desviar de tais limitações é a utilização de métodos não exatos para encontrar soluções viáveis para o problema que apresentem soluções de qualidade quando comparadas às soluções ótimas conhecidas. Nesse cenário, métodos não exatos como aprendizado de máquinas, inteligência artificial, heurísticas e meta-heurísticas surgem como abordagens interessantes para apresentarem soluções não exatas em tempo hábil para instâncias reais do problema.

Em [Murti et al. 2022, Murti et al. 2022], os autores discutem tais limitações e apresentam um agente de aprendizado por reforço profundo que resolve o problema de posicionamento de funções da vRAN. A formulação do agente apresentada é promissora e apresenta resultados próximos ao ótimo. Entretanto, a formulação utilizada pelos autores é simplista, pois considera uma única CU com posicionamento fixo, DUs com posicionamento fixo e apenas uma rota no roteamento entre tais elementos. Assim, o agente de aprendizado por reforço profundo encontra soluções que não exploram o problema de decisão na totalidade, apresentando soluções com menor centralização e redução de custos quando comparadas a formulações completas do problema.

Visando tais limitações encontradas nos trabalhos anteriores, em [Almeida et al. 2022], nós apresentamos um agente de aprendizado por reforço profundo que também resolve o problema de posicionamento de funções da vRAN. Entretanto, desta vez utilizamos nossa formulação genérica e flexível apresentada em PlaceRAN [Morais et al. 2022]. Neste trabalho, nós comparamos as soluções de nosso agente com as soluções encontradas pelo modelo ótimo PlaceRAN e também com os resultados encontrados pelo modelo de Murti [Murti et al. 2022]. Os detalhes deste trabalho e os resultados encontrados serão discutidos no Capítulo 4, por se tratar de uma publicação fruto deste trabalho de mestrado.

Explorando outras formas não exatas de resolução de problemas, em [Almeida et al. 2023], nós apresentamos um algoritmo genético que resolve o problema de posicionamento de funções da vRAN considerando a formulação apresentada

em PlaceRAN [Morais et al. 2022]. Nesse trabalho, conseguimos analisar nosso algoritmo genético em conjunto com uma heurística de geração e comparar os resultados tanto com nosso modelo ótimo PlaceRAN, mas também com nosso agente de aprendizado por reforço profundo [Almeida et al. 2022] e o modelo apresentado por Murti [Murti et al. 2022]. Os resultados mostram que, mesmo com a ausência de redes neurais em sua formulação, nosso algoritmo genético apresenta resultados promissores em relação aos demais modelos, resolvendo o problema de forma eficiente mesmo em instâncias maiores do problema. Esse resultado também será discutido adiante, no Capítulo 5, por se tratar de uma contribuição deste trabalho de mestrado.

Outros trabalhos na literatura apresentam uma formulação para o problema de posicionamento de funções da vRAN, porém perseguindo outros objetivos específicos. Em [Pires et al. 2022], nós apresentamos uma formulação exata bi-objetiva para a solução do problema, considerando como um dos objetivos a minimização do consumo energético da solução. Em [Fraga et al. 2022], nós apresentamos uma formulação exata e uma heurística de busca para o problema, considerando adicionalmente ao posicionamento das funções da vRAN o posicionamento de aplicações de borda (*Multi-access Edge Computing - MEC*). Em [Lopes et al. 2022], além de aspectos da rede de acesso, nós também consideramos na decisão da centralização de funções da vRAN o posicionamento da antena na topologia, ou seja, a região de sua área de cobertura, calculando soluções que centralizam funções de unidades de rádio vizinhas visando minimizar os efeitos da interferência na comunicação do usuário. Finalmente, em [Morais et al. 2022], nós apresentamos um ambiente prático para simulações de posicionamento de funções virtualizadas, utilizando uma infraestrutura *Kubernetes* para gerenciar as funções e um simulador da rede de acesso (*OpenAirInterface*) para realizar o tráfego das funções de rádio.

A variação dinâmica de demanda nas unidades de rádio (RUs) na topologia da vRAN é um aspecto crucial que pode ter um impacto significativo na eficiência operacional e na qualidade do serviço oferecido pelas redes móveis. Enquanto os trabalhos mencionados anteriormente se concentraram principalmente no cenário de planejamento da rede, no qual a variação de demanda não é considerada, outros estudos apresentam formulações para este problema considerando variação dinâmica de demanda.

Em [Murti et al. 2023, Alba e Kellerer 2022, Joda et al. 2022, Pamuklu et al. 2021, Gupta et al. 2022], os autores apresentam soluções que consideram a variação de demanda dinâmica na rede de acesso, permitindo que o posicionamento das funções seja ajustado para cada instante na rede conforme a demanda atual. Essa análise granular do posicionamento leva a soluções mais adaptadas às condições de tráfego em constante mudança, resultando em custos de operação mais baixos ao longo do tempo. No entanto, é importante observar que a introdução da variação dinâmica de demanda aumenta ainda mais a complexidade do problema, que já é desafiador devido à sua natureza NP-Completa.

Desta forma, enquanto a consideração da variação de demanda dinâmica nas RUs permite a obtenção de soluções mais adaptáveis e eficientes, essa abordagem também introduz um nível mais alto de complexidade computacional, demandando soluções eficazes para o problema.

Os autores de [Murti et al. 2023] propõem a utilização de um agente DRL para resolver o problema de posicionamento da vRAN considerando a variação de demanda nas RUs. Nesse trabalho, os autores consideram múltiplas CUs e DUs com posicionamento dinâmico, para melhor atender às flutuações na demanda. Entretanto, é destacado um aspecto crítico nesse estudo: a falta de garantias sobre o cumprimento das restrições de posicionamento para todas as soluções encontradas pelo agente DRL. Isso significa que o agente pode chegar a soluções que não são viáveis em termos das restrições operacionais da rede, o que é uma questão significativa dado que a operação eficaz da rede depende da conformidade com essas restrições. A ausência de garantias sobre o respeito às restrições de posicionamento pode comprometer a utilidade prática das soluções propostas.

Similarmente a [Murti et al. 2023], em [Pamuklu et al. 2021] os autores apresentam um agente DRL para o problema de posicionamento considerando variação de demanda. O agente é implementado através dos métodos de *Q-Learning* e *Sarsa*, e considerando como objetivo o custo energético da solução. Entretanto, os autores também não apresentam restrições sobre o posicionamento das funções, fazendo com que o agente apresente o mesmo problema destacado em [Murti et al. 2023], ou seja, encontrar soluções não viáveis. Em [Alba e Kellerer 2022] os autores apresentam um arcabouço para cálculo dos custos de posicionamento, resolvendo o problema considerando demanda variável nas RUs através de um modelo de otimização exato de alta complexidade.

Por fim, em [Joda et al. 2022], os autores formulam o problema considerando variação de demanda apresentando um agente DRL. O agente formulado apresenta em sua função de recompensa um sinal positivo em caso de solução viável e um sinal negativo (penalidade) em caso de solução inviável. Isto faz com que o agente não apresente o problema do trabalho [Murti et al. 2023], minimizando a quantidade de soluções não viáveis encontradas. Por outro lado, em [Gupta et al. 2022] os autores resolvem o problema de posicionamento das funções considerando variação de demanda via uma formulação ótima e exata, ou seja, com alta complexidade e baixa escalabilidade. A Tabela 2.1 apresenta a classificação dos trabalhos apresentados de acordo com suas contribuições e comparando com as contribuições das formulações apresentadas neste trabalho.

Trabalho	Formulação	CUs	Divisões funcionais	Demanda	Aplica restrições
[Garcia-Saavedra et al. 2018]	MILP	Única e fixa	Fixas	Estática	Sim
[Fonseca et al. 2019]	MILP	Única	Fixas	Estática	Sim
[Garcia-Saavedra et al. 2018]	MILP	Fixas	Fixas	Estática	Sim
[Murti et al. 2020]	MILP	Fixas	Fixas	Estática	Sim
[Morais et al. 2022]	BLP	Dinâmicas	Genéricas	Estática	Sim
[Almeida et al. 2022]	MILP	Dinâmicas	Genéricas	Estática	Sim
[Murti et al. 2022]	DRL	Única e fixa	Fixas	Estática	Sim
[Murti et al. 2022]	DRL	Única e fixa	Fixas	Estática	Sim
[Almeida et al. 2022]	DRL	Dinâmicas	Genéricas	Estática	Sim
[Almeida et al. 2023]	GA	Dinâmicas	Genéricas	Estática	Sim
[Murti et al. 2023]	DRL	Dinâmicas	Fixas	Dinâmica	Não
[Alba e Kellerer 2022]	MILP	Única e fixa	Genéricas	Dinâmica	Não
[Joda et al. 2022]	DRL	Única e fixa	Fixas	Dinâmica	Não
[Pamuklu et al. 2021]	DRL	Única e fixa	Fixas	Dinâmica	Não
[Gupta et al. 2022]	MILP	Única e fixa	Fixas	Dinâmica	Não

Tabela 2.1: Principais diferenças entre os trabalhos citados.

Modelo geral de alocação de recurso na vRAN

Neste capítulo, apresentamos uma formulação para o problema de posicionamento de funções da vRAN. O modelo apresentado a seguir é uma evolução do nosso trabalho anterior, o PlaceRAN [Morais et al. 2022], que não permite divisão de carga em múltiplos caminhos. Desta forma, neste capítulo, apresentamos os benefícios e impactos de considerar a divisão de carga no roteamento dos fluxos em um *crosshaul*.

3.1 Introdução

O problema de posicionamento de funções da vRAN é um problema de decisão que pode ser representado como um problema de otimização, por exemplo, programação inteira mista. Tal problema é fortemente investigado na literatura [Garcia-Saavedra et al. 2018, Fonseca et al. 2019, Garcia-Saavedra et al. 2018, Murti et al. 2020], em que autores apresentam uma formulação e resolvem o problema de forma ótima via arcabouços de otimização (por exemplo, CPLEX) que apresentam soluções ótimas do problema.

Em [Morais et al. 2022], nós apresentamos o PlaceRAN, uma formulação genérica que suporta qualquer número de opções de divisões funcionais, com requisitos personalizados e topologias heterogêneas com diferentes quantidades de antenas e nós de processamento. Nossa formulação permite que uma RU seja atendida por qualquer nó de processamento atuando como vDU e/ou vCU. Nesse caso, a alocação é flexível, ou seja, não importa qual CN atue como vDU ou vCU desde que os requisitos de latência e vazão sejam atendidos. No entanto, o modelo de otimização PlaceRAN não permite a divisão de carga dos fluxos em diferentes caminhos. Isso significa que entre qualquer par de nós (envolvendo RU, vDU ou vCU), o tráfego é roteado por um único caminho, o que pode limitar a qualidade da solução, apresentando soluções menos eficientes.

A divisão de fluxo em múltiplos caminhos beneficia a redução dos custos da solução e possibilita maior centralização de funções da vRAN, principalmente quando os enlaces da rede apresentam gargalos, ou seja, cenários com maior competição por recursos. Tais benefícios são possíveis devido à divisão da carga de transporte em várias

partes, otimizando a utilização dos recursos de rede, aumentando a centralização das funções da vRAN e suportando mais funções em um único CN.

Portanto, neste capítulo, propomos um modelo de otimização para resolver o problema de posicionamento das funções da vRAN, considerando a divisão de carga em múltiplos caminhos no roteamento entre as divisões funcionais. Como a formulação utilizada é baseada em nosso trabalho anterior [Morais et al. 2022], o modelo apresentado nesta seção tem como herança as características de flexibilidade e generalização apresentadas pelo PlaceRAN, ou seja, uma RU pode ser atendida por qualquer CN atuando como vDU e/ou vCU. Essa flexibilidade é aprimorada com a possibilidade de divisão de carga em diferentes rotas, porém, ao considerar múltiplas rotas na divisão de carga, a busca pela solução ótima se torna mais complexa, conforme iremos discutir na Seção 3.3. Os resultados também contribuem para ilustrar a vantagem de nossa nova formulação em termos de optimalidade em comparação com o estado da arte, que inclui nossa proposta anterior.

3.2 Modelo de sistema

Para formular o problema, consideramos um NG-RAN composto por um conjunto $\mathcal{B} = \{b_1, b_2, \dots, b_{|\mathcal{B}|}\}$ de RUs, um conjunto $\mathcal{T} = \{t_1, t_2, \dots, t_{|\mathcal{T}|}\}$ de nós de encaminhamento e um conjunto $\mathcal{C} = \{c_1, c_2, \dots, c_{|\mathcal{C}|}\}$ de CNs. Cada nó de encaminhamento $t_k \in \mathcal{T}$ pode ser conectado a RUs, CNs, núcleo da rede ou a outros nós de encaminhamento, e uma capacidade de processamento c_m^{Proc} que caracteriza cada CN $c_m \in \mathcal{C}$. Definimos o grafo $G = (\mathcal{V}, \mathcal{E})$ para representar a topologia da NG-RAN e o núcleo da rede v_0 , com $\mathcal{V} = \{v_0\} \cup \mathcal{B} \cup \mathcal{T} \cup \mathcal{C}$ sendo o conjunto de nós e $\mathcal{E} = \{e_{ij}; v_i, v_j \in \mathcal{V}\}$ representando o conjunto de enlaces da rede que conectam os nós. Cada enlace $e_{ij} \in \mathcal{E}$ tem uma capacidade de transmissão e_{ij}^{Cap} e uma latência e_{ij}^{Lat} .

Cada RU é associada a uma pilha de protocolos com oito opções de divisões funcionais [Larsen et al. 2019]. Esse número de opções corresponde ao padrão O-RAN [Open RAN Alliance 2020], entretanto nosso modelo permite qualquer valor com modificações triviais. Cada opção de divisão funcional separa um conjunto de funções da vRAN e especifica os requisitos de largura de banda e latência para cada configuração. Definimos o conjunto $\mathcal{F} = \{f_1, f_2, f_3, f_4, f_5, f_6, f_7\}$ para representar as funções da vRAN. Devido a questões práticas, nem todas as opções de divisões funcionais são padronizadas [ITU-T 2018]. Desta forma, definimos $\mathcal{D} = \{D_1, D_2, \dots, D_{|\mathcal{D}|}\}$ como o conjunto de configurações NG-RAN viáveis (VNCs) indicadas na literatura. As divisões funcionais consideradas nesta seção são apresentadas na Figura 3.1.

Definimos \mathcal{P}_l como o conjunto de caminhos que conectam o núcleo da rede (origem) a cada RU da topologia (destino) $b_l \in \mathcal{B}$. Cada caminho $p \in \mathcal{P}_l$ é composto por três sub-caminhos: p_{Bh} (*backhaul*, ou seja, $v_0 \leftrightarrow \text{CU}$, ou $v_0 \leftrightarrow \text{CU+DU}$, ou $v_0 \leftrightarrow$

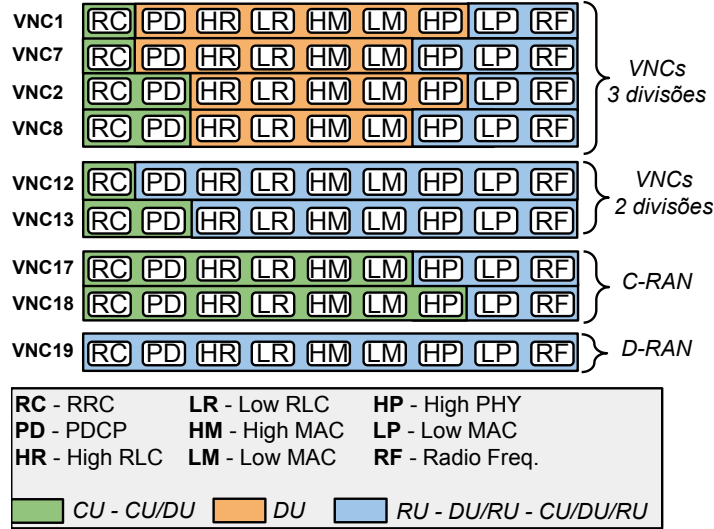


Figura 3.1: Conjunto de divisões funcionais consideradas.

CU+DU+RU), p_{Mh} (*midhaul*, ou seja, CU \leftrightarrow DU, ou CU \leftrightarrow DU+RU), e p_{Fh} (*fronthaul*, ou seja, RU \leftrightarrow DU, ou RU \leftrightarrow CU+DU), em que pelo menos o sub-caminho p_{Bh} não está vazio. Como cada RU $b_l \in \mathcal{B}$ pode utilizar qualquer VNC $D_r \in \mathcal{D}$ que satisfaça os requisitos de largura de banda e latência, assumimos que a rede de acesso é um *crosshaul* capaz de transportar vários VNCs simultaneamente. Finalmente, definimos que cada CN $c_m \in \mathcal{C}$ pode atuar como vCU, vDU ou ambos para uma ou mais RU $b_l \in \mathcal{B}$.

3.3 Formulação do problema

O objetivo da nossa formulação é maximizar a centralização de funções da vRAN enquanto minimiza o número de CNs utilizados. A Tabela 3.1 resume os parâmetros e variáveis usados no modelo do sistema, formulação do problema e solução.

Maximizar a centralização enquanto minimiza o número de CNs utilizados podem ser objetivos conflitantes. Isto ocorre, pois o processamento disponível nos CNs e os recursos de rede disponíveis nos enlaces podem limitar o número de funções da vRAN centralizadas no mesmo CN. Desta forma, a divisão de carga entre múltiplos caminhos no roteamento oferece melhores oportunidades para minimizar a quantidade de CNs utilizados e viabiliza o aumento da centralização das funções. Chamamos esse problema (\mathbb{P}) de *Multi-path functional Split and network function placement in vRAN* (*MuSt-RAN*), que tem a função objetivo formalmente definida a seguir:

$$\text{minimizar} \quad \Phi_1 - \Phi_2, \quad (3-1)$$

em que Φ_1 representa a quantidade de CNs utilizados pela solução,

Tabela 3.1: Parâmetros e variáveis utilizadas na formulação.

Parâmetros	Definição
\mathcal{C}	Conjunto de CNs
\mathcal{B}	Conjunto de RUs
\mathcal{E}	Conjunto de enlaces
\mathcal{P}_l	Conjunto dos k -caminhos mais curtos
\mathcal{F}	Conjunto de funções da vRAN
\mathcal{D}	Conjunto de VNCs
e_{ij}^{Cap}	Capacidade de transmissão do enlace e_{ij}
e_{ij}^{Lat}	Atraso estimado do enlace e_{ij}
u_p^m	Indica se $c_m \in \mathcal{C}$ está em $p \in \mathcal{P}_l$
$M(c_m, f_s, b_l)$	Indica se $c_m \in \mathcal{C}$ executa $f_s \in \mathcal{F}$ do $D_r \in \mathcal{D}$ para $b_l \in \mathcal{B}$
$y_{e_{ij}}^{PBH}$	Indica se e_{ij} é parte do <i>backhaul</i>
$y_{e_{ij}}^{PMH}$	Indica se e_{ij} é parte do <i>midhaul</i>
$y_{e_{ij}}^{PFH}$	Indica se e_{ij} é parte do <i>fronthaul</i>
α_{BH}^r	Demanda de vazão do VNC $D_r \in \mathcal{D}$ no <i>backhaul</i>
α_{MH}^r	Demanda de vazão do VNC $D_r \in \mathcal{D}$ no <i>midhaul</i>
α_{FH}^r	Demanda de vazão do VNC $D_r \in \mathcal{D}$ no <i>fronthaul</i>
β_{BH}^r	Latencia máxima tolerada no <i>backhaul</i> para o VNC $D_r \in \mathcal{D}$
β_{MH}^r	Latencia máxima tolerada no <i>midhaul</i> para o VNC $D_r \in \mathcal{D}$
β_{FH}^r	Latencia máxima tolerada no <i>fronthaul</i> para o VNC $D_r \in \mathcal{D}$
γ_m^s	Demanda de processamento da função $f_s \in \mathcal{F}$
$W(D_r, b_l)$	Indica se o VNC $D_r \in \mathcal{D}$ é utilizado para RU $b_l \in \mathcal{B}$.
c_m^{Proc}	Capacidade de processamento do CN $c_m \in \mathcal{C}$
Variável	Definição
$x_l^{p,r} = \{0, 1\}$	Representa se $p \in \mathcal{P}_l$ e $D_r \in \mathcal{D}$ são utilizados para RU $b_l \in \mathcal{B}$
$y_l^{p,r} = [0, 1]$	Representa a porcentagem de tráfego da RU $b_l \in \mathcal{B}$, usando $D_r \in \mathcal{D}$, no caminho $p \in \mathcal{P}_l$

$$\Phi_1 = \sum_{c_m \in \mathcal{C}} \left[\frac{\sum_{b_l \in \mathcal{B}} \sum_{p \in \mathcal{P}_l} \sum_{D_r \in \mathcal{D}} (x_l^{p,r} u_m^p)}{|\mathcal{P}_l|} \right], \quad (3-2)$$

e Φ_2 representa a quantidade de funções da vRAN centralizadas,

$$\Phi_2 = \sum_{\substack{c_m \in \mathcal{C} \\ f_s \in \mathcal{F}}} \left(\sum_{b_l \in \mathcal{B}} \left[\frac{\sum_{p \in \mathcal{P}_l} \sum_{D_r \in \mathcal{D}} (x_l^{p,r} u_m^p)}{|\mathcal{P}_l|} \right] M(c_m, f_s, b_l) - \left[\frac{\sum_{p \in \mathcal{P}_l} \sum_{D_r \in \mathcal{D}} (x_l^{p,r} u_m^p)}{|\mathcal{P}_l|} \right] \right) \left[\frac{\sum_{b_l \in \mathcal{B}} \left[\frac{\sum_{p \in \mathcal{P}_l} \sum_{D_r \in \mathcal{D}} (x_l^{p,r} u_m^p)}{|\mathcal{P}_l|} \right]}{|\mathcal{B}|} \right] \right). \quad (3-3)$$

Além disso, a variável de decisão $x_l^{p,r} \in \{0, 1\}$ representa a escolha do caminho $p \in \mathcal{P}_l$ e VNC $D_r \in \mathcal{D}$ para servir a RU $b_l \in \mathcal{B}$. A partir dos dados de entrada, $u_m^p \in \{0, 1\}$ indica se $c_m \in \mathcal{C}$ faz parte do caminho $p \in \mathcal{P}_l$. A função teto, tomando $|\mathcal{P}_l|$ como denominador, garante que o fluxo de tráfego servindo a mesma RU $b_l \in \mathcal{B}$ seja contado apenas uma vez, mesmo quando o fluxo é dividido. A outra função teto, tomando $|\mathcal{B}|$ como denominador, tem o mesmo propósito. A função de mapeamento $M(c_m, f_s, b_l) \in \{0, 1\}$ (sobre os dados de entrada) indica se o CN $c_m \in \mathcal{C}$ executa a função $f_s \in \mathcal{F}$ da RU $b_l \in \mathcal{B}$. A seguir, apresentamos as restrições que representam a divisão de fluxo em múltiplos caminhos.

$$\sum_{p \in \mathcal{P}_l} \sum_{D_r \in \mathcal{D}} x_l^{p,r} \geq 1, \quad \forall b_l \in \mathcal{B}. \quad (3-4)$$

$$y_l^{p,r} \geq 0, \quad \forall b_l \in \mathcal{B}, p \in \mathcal{P}_l, D_r \in \mathcal{D}. \quad (3-5)$$

$$\sum_{p \in \mathcal{P}_l} \sum_{D_r \in \mathcal{D}} y_l^{p,r} = 1, \quad \forall b_l \in \mathcal{B}. \quad (3-6)$$

$$\sum_{c_m \in \mathcal{C}} \left[\frac{\sum_{p \in \mathcal{P}_l} \sum_{D_r \in \mathcal{D}} (x_l^{p,r} u_m^p)}{|\mathcal{P}_l|} \right] = F_{CN}(D_r), \quad \forall b_l \in \mathcal{B}. \quad (3-7)$$

$$y_l^{p,r} \leq x_l^{p,r}, \quad \forall p \in \mathcal{P}_l, b_l \in \mathcal{B}, D_r \in \mathcal{D}. \quad (3-8)$$

$$x_l^{p,r} - y_l^{p,r} \leq 1 - \varepsilon, \quad \forall p \in \mathcal{P}_l, b_l \in \mathcal{B}, D_r \in \mathcal{D}. \quad (3-9)$$

A restrição (3-4) garante que todas as RUs de \mathcal{B} sejam atendidas, empregando potencialmente mais de um caminho de \mathcal{P}_l . A variável de decisão $y_l^{p,r} \in [0, 1]$ representa a fração do fluxo da RU $b_l \in \mathcal{B}$, usando VNC $D_r \in \mathcal{D}$, trafegado pelo caminho $p \in \mathcal{P}_l$. A restrição (3-5) proíbe que esse valor seja negativo. A entrega de todo o fluxo de tráfego da pilha de protocolos da RU $b_l \in \mathcal{B}$ é assegurada por restrição (3-6). Segundo a restrição (3-7), o número de CNs empregados para atender a pilha de protocolos da RU $b_l \in \mathcal{B}$ deve ser igual ao número de nós CN do VNC $D_r \in \mathcal{D}$ selecionado (retornado por $F_{CN}(D_r)$). Devido à possibilidade de divisão de fluxo, essa restrição é importante para evitar um efeito “fantasma” no qual VNFs inexistentes são adicionados erroneamente para aumentar o nível de centralização. Para fazer a correlação adequada entre as variáveis de decisão e evitar introduzir não-linearidade no problema, introduzimos restrições (3-8) e (3-9). Essas restrições garantem que se o caminho $p \in \mathcal{P}_l$ for selecionado, ou seja $x_l^{p,r} = 1$, então uma quantidade positiva de tráfego da pilha deve fluir pelo caminho $p \in \mathcal{P}_l$, ou seja, $y_l^{p,r} > 0$. Caso contrário, $x_l^{p,r} = 0$, implica que esse fluxo de tráfego deve ser nulo, ou seja, $y_l^{p,r} = 0$. O parâmetro $0 < \varepsilon \leq 1$ permite controlar a granularidade da divisão de fluxo, afetando o espaço de soluções e o desempenho do MuSt-RAN, conforme ilustramos na seção de avaliação. As seguintes restrições completam a formulação do problema:

$$\sum_{D_r \in \mathcal{D}} W(D_r, b_l) = 1, \quad \forall b_l \in \mathcal{B}. \quad (3-10)$$

$$\sum_{D_r \in \mathcal{D}} \sum_{b_l \in \mathcal{B}} \sum_{p \in \mathcal{P}_l} [y_l^{p,r} (y_{e_{ij}}^{pBh} \alpha_{Bh}^r + y_{e_{ij}}^{pMh} \alpha_{Mh}^r + y_{e_{ij}}^{pFh} \alpha_{Fh}^r)] \leq e_{ij}^{Cap}, \quad \forall e_{ij} \in \mathcal{E}. \quad (3-11)$$

$$\sum_{e_{ij} \in \mathcal{E}} x_l^{p,r} y_{e_{ij}}^{pBh} e_{ij}^{Lat} \leq \beta_{Bh}^r, \quad \forall b_l \in \mathcal{B}, p \in \mathcal{P}_l, D_r \in \mathcal{D}. \quad (3-12)$$

$$\sum_{e_{ij} \in \mathcal{E}} x_l^{p,r} y_{e_{ij}}^{p_{Mh}} e_{ij}^{Lat} \leq \beta_{Mh}^r, \quad \forall b_l \in \mathcal{B}, p \in \mathcal{P}_l, D_r \in \mathcal{D}. \quad (3-13)$$

$$\sum_{e_{ij} \in \mathcal{E}} x_l^{p,r} y_{e_{ij}}^{p_{Fh}} e_{ij}^{Lat} \leq \beta_{Fh}^r, \quad \forall b_l \in \mathcal{B}, p \in \mathcal{P}_l, D_r \in \mathcal{D}. \quad (3-14)$$

$$\sum_{f_s \in \mathcal{F}} \left[\sum_{b_l \in \mathcal{B}} \left[\frac{\sum_{p \in \mathcal{P}_l} \sum_{D_r \in \mathcal{D}} (x_l^{p,r} u_m^p)}{|\mathcal{P}_l|} \right] \times M(c_m, f_s, b_l) \gamma_m^s \right] \leq c_m^{Proc}, \quad \forall c_m \in \mathcal{C}. \quad (3-15)$$

A restrição (3-10) garante que cada RU $b_l \in \mathcal{B}$ esteja relacionada a exatamente um VNC $D_r \in \mathcal{D}$. Conforme a restrição (3-11), a capacidade de transmissão e_{ij}^{Cap} de cada enlace e_{ij} não deve ser excedida, onde $y_{e_{ij}}^{p_{Bh}}$, $y_{e_{ij}}^{p_{Mh}}$ e $y_{e_{ij}}^{p_{Fh}}$ indicam se o enlace e_{ij} faz parte do *backhaul*, *midhaul* e *fronthaul*. Cada VNC $D_r \in \mathcal{D}$ solicita uma demanda de taxa de bits no *backhaul*, *midhaul* e *fronthaul*, representada por α_{Bh}^r , α_{Mh}^r e α_{Fh}^r . Cada VNC $D_r \in \mathcal{D}$ tolera uma latência máxima em cada sub-caminho: β_{Bh}^r (*backhaul*), β_{Mh}^r (*midhaul*) e β_{Fh}^r (*fronthaul*), descrito por restrições (3-12), (3-13) e (3-14). Finalmente, a restrição (3-15), onde γ_m^s representa a demanda computacional da $f_s \in \mathcal{F}$, garante que o recurso de processamento c_m^{Proc} de $c_m \in \mathcal{C}$ não seja excedido.

Baseado nos autores de [Murti et al. 2021], identificamos a complexidade de nossa formulação a partir da redução polinomial do problema da mochila multidimensional de múltipla escolha (MMKP). Como nossa formulação é mais flexível considerando um conjunto genérico de divisões funcionais e CNs de propósito geral, também podemos usar o MMKP para identificar a complexidade de nossa formulação, dado que ambas as formulações possuem a mesma estrutura de decisão.

MMKP. Seja \mathcal{N} o conjunto de itens distribuídos em \mathcal{M} grupos, $\mathcal{V} = \{v_{11}, v_{12}, \dots, v_{\mathcal{N}\mathcal{M}}\}$ os valores dos itens e $r_{11}, r_{12}, \dots, r_{\mathcal{N}\mathcal{M}}$ o recurso necessário para escolher os itens. A mochila tem um limite de recurso de $\mathcal{R} = (R_1, R_2, \dots, R_{\mathcal{M}})$ onde cada $R_i \in \mathcal{R}$ corresponde ao limite de recurso de um grupo. O objetivo é escolher exatamente um item de cada grupo enquanto maximiza a soma dos valores dos itens escolhidos $v_{ij} \in \mathcal{V}$, sujeito à restrição de limite de recursos da mochila para cada grupo $r_{ij} \leq R_j$.

Encontrar a solução ótima para MMKP é NP-difícil [Akbar et al. 2006]. Dessa forma, provamos que MuSt-RAN, denotado por \mathbb{P} , é NP-difícil reduzindo MMKP a \mathbb{P} em tempo polinomial.

Teorema 3.1 *MMKP pode ser reduzido a \mathbb{P} em tempo polinomial, ou seja, $MMKP \leq_P \mathbb{P}$.*

Prova. Sem perda de generalidade, consideramos que enlaces de capacidade ilimitada e divisões funcionais sem requisitos de atraso garantem que as restrições (3-11) a (3-14) sejam sempre satisfeitas. Desta forma, podemos mapear esse problema para MMKP configurando cada RU como um grupo, onde cada grupo contém as opções de VNCs como itens. Assim, escolher um item de um grupo é o mesmo que escolher um único VNC para uma RU, selecionando o número de funções centralizadas que podemos definir

como os valores dos itens. Adicionalmente, a escolha de um VNC implica na definição do consumo de recursos computacionais de um CN. Mapeamos esse consumo como os recursos necessários para pegar os itens e o recurso computacional do nó como as restrições da mochila. Finalmente, definimos o objetivo como maximizar os valores dos itens, ou seja, maximizar a centralização das funções, sujeitos aos recursos computacionais do nó. Portanto, podemos concluir que se podemos resolver \mathbb{P} em tempo polinomial, também podemos resolver o problema MMKP. \square

Nossa formulação é ainda mais complexa, pois precisamos rotear o tráfego entre as múltiplas RUs, vCUs e vDUs sujeitas às restrições de requisitos de rede das divisões funcionais, considerando a divisão de fluxo em enlaces compartilhados e as restrições de capacidade de processamento dos CNs.

3.4 Resultados

Nós avaliamos nosso modelo (MuSt-RAN) em duas topologias de vRAN: uma tradicional baseada em anel (T1) e uma hierárquica de próxima geração (T2), conforme ilustrado na Figura 3.2. Assim, conseguimos avaliar o impacto da nossa solução na infraestrutura atual (T1) e também na sua evolução do 5G para o 6G (T2). As topologias possuem 25 nós, onde 23 são RUs, sendo todos eles potenciais CNs, permitindo que as soluções determinem quais são efetivamente necessárias para rodar as funções da vRAN. Embora o problema original não seja linear, usamos as funções min e max para linearizá-lo [Morais et al. 2022], transformando o problema em um problema MILP. Assim, o arcabouço de otimização CPLEX recebe um problema MILP como entrada, para o qual retorna a solução ótima.

A Tabela 3.2 apresenta a capacidade de processamento dos CNs (em núcleos) e os parâmetros dos enlaces para as topologias T1 e T2. Semelhante a [Morais et al. 2022], \mathcal{P}_l é preenchido com o conjunto de k -caminhos mais curtos do núcleo da rede para cada RU $b_l \in \mathcal{B}$. Os resultados apresentados pelo MuSt-RAN são comparados com os publicados em PlaceRAN [Morais et al. 2022] e MCU-vRAN [Murti et al. 2020]. Executamos todos os experimentos em uma máquina virtual com 32 vCPUs, 256 GB de RAM e 40 GB de disco virtual. A VM foi hospedada em um servidor HPE ProLiant DL580 G7 com quatro Intel Xeon E7-4830 @ 2,13 GHz, com *Python* 3.6.9 e *docplex* 2.20.204 para implementação do modelo de otimização, e IBM CPLEX 12.8.0 como arcabouço de otimização. Os primeiros resultados empregam $\varepsilon = 10^{-2}$. Detalhes adicionais sobre o ambiente de avaliação e todo o código-fonte estão disponíveis publicamente no repositório *Github*¹.

¹<https://github.com/LABORA-INF-UFG/paper-GLCK-2021>

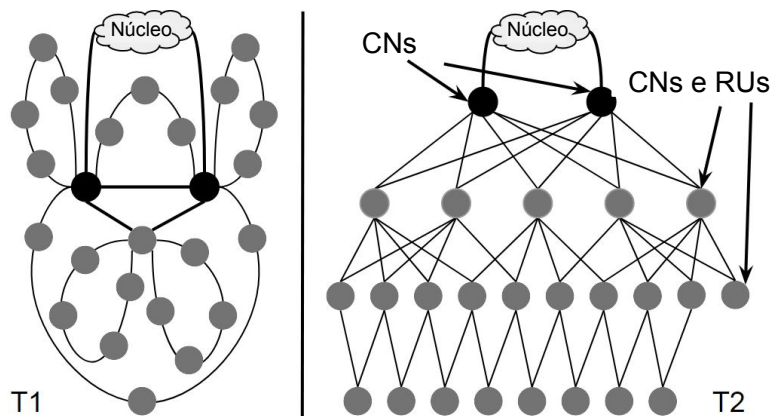


Figura 3.2: Topologias.

Tabela 3.2: Parâmetros utilizados na avaliação dos modelos.

	T1	T2
CNs sem RU (núcleos)	32/32	128/128
CNs com RU (núcleos)	16/8	32/16
Capacidade dos links (Gbps)	50/25/10/5	500/100/50/25/10
Latencia dos links (ms)	0.05/0.000005	0.05/0.000005

A primeira parte da função objetivo apresentada é minimizar a quantidade de CNs utilizados. Soluções que minimizam a quantidade de CNs utilizados apresentam melhor economia de custos, por exemplo, economia de suporte e manutenção, e economia de energia. Tais aspectos são essenciais para redução dos custos de infraestrutura e operação da vRAN [Rahman et al. 2013], além de serem soluções ecológicas.

A Figura 3.3 mostra a proporção de CNs utilizados por cada modelo de otimização. Note que o MCU-vRAN demanda mais nós em ambas as topologias, pois, não emprega o conceito de vDU, limitando as oportunidades de centralização e, conseqüentemente, necessitando de mais CNs como pontos de centralização. Já nossa solução MuSt-RAN exibe o mesmo desempenho que o PlaceRAN na topologia T1 e apresenta melhorias na topologia T2. Dado que MuSt-RAN é uma evolução do PlaceRAN, ele sempre apresenta pelo menos os mesmos resultados que o PlaceRAN. Entretanto, o MuSt-RAN supera o PlaceRAN quando os enlaces da rede apresentam gargalos, ao permitir fluxos divisíveis e otimizar o uso de recursos de rede. Por outro lado, isso implica que o MuSt-RAN também precisa decidir como distribuir cada fluxo por vários caminhos.

A seguir, apresentamos os resultados relacionados à escolha de divisões funcionais, ou seja, escolha de VNCs. A Figura 3.4 mostra os nove VNCs especificados pelo O-RAN, variando da configuração que mais centraliza funções (C-RAN) à esquerda até a configuração que não centraliza funções (D-RAN) à direita. As Figuras 3.5 (a) e 3.5 (b) mostram a distribuição das escolhas de VNCs selecionados por cada modelo de otimização. O MCU-vRAN apresenta o pior desempenho, com D-RAN em ambas as topologias,

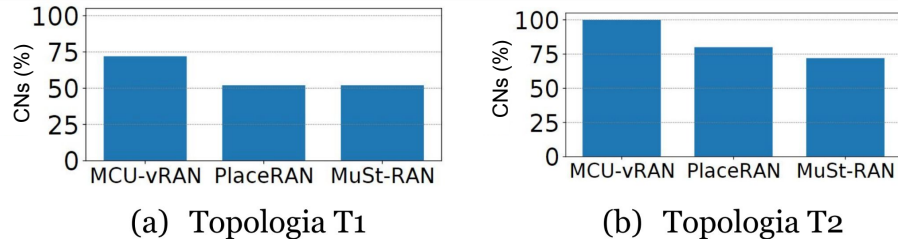


Figura 3.3: Comparando a quantidade de CNs utilizados pelas soluções.

o que está diretamente relacionado à sua limitação de centralização. Já o PlaceRAN apresenta melhores resultados, reduzindo o número de D-RAN e aumentando a quantidade de C-RAN e NG-RAN (3). Por outro lado, o MuSt-RAN supera o PlaceRAN em ambas as topologias, apresentando maior centralização com divisões funcionais NG-RAN (2) graças à divisão de carga em múltiplos caminhos. Além disso, o MuSt-RAN possui a menor ocorrência de D-RAN em ambas as topologias, ilustrando a vantagem de empregar uma abordagem multi-caminhos.

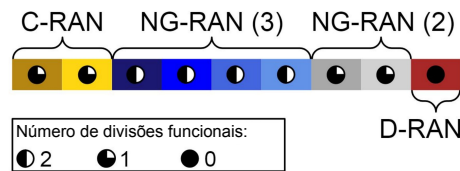


Figura 3.4: Legenda VNCs.

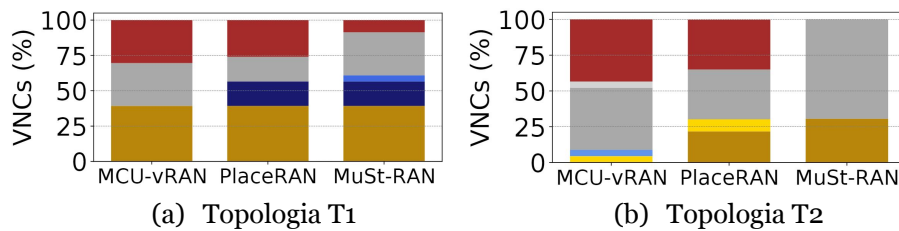


Figura 3.5: Comparando as escolhas de VNCs das soluções.

Também avaliamos o MuSt-RAN em relação à variação do parâmetro ϵ de granularidade. Desta forma, à medida que ϵ diminui, o fluxo entre pares de funções da vRAN pode ser dividido em subfluxos cada vez menores. Portanto, diminuir ϵ implica aumentar o número de subfluxos, sua distribuição ao longo da rede e consequentemente a complexidade do problema. A Figura 3.6 mostra o nível de centralização em função do parâmetro ϵ . A melhor centralização é alcançada com $\epsilon = 0.1$, e mudar ϵ para um valor pequeno aumenta o tempo de computação para encontrar uma solução sem melhorar o nível de centralização. No entanto, diferentes condições de rede podem demandar valores menores para ϵ para alcançar maior centralização, como cenários com maior competição por recursos compartilhados.

Diminuir ϵ implica em aumentar o custo computacional do modelo, conforme ilustrado na Figura 3.7. O tempo de computação é afetado pelo número de nós e pela

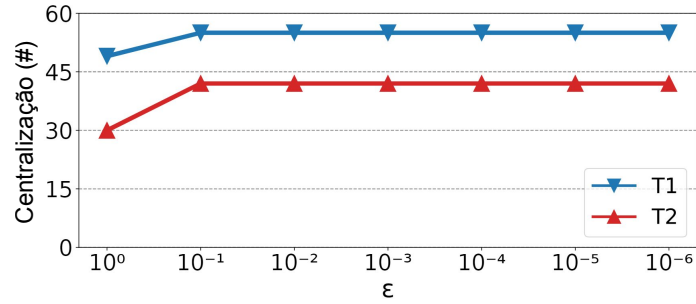


Figura 3.6: Nível de centralização em função de ε .

topologia da vRAN, porém, o crescimento do tempo de computação em função de ε pode ter diferentes inclinações, conforme mostrado na Figura 3.7. Esse comportamento mostra a importância de avaliar diferentes valores de ε , mas a partir de um pequeno conjunto de opções. Por exemplo, poderíamos ter parado a avaliação em $\varepsilon = 10^{-4}$ já que isso representa uma boa performance entre o tempo de computação e a busca por um nível de centralização mais alto. Outra diretriz para limitar o valor mínimo de ε é considerar a maior vazão e o que seria uma fração razoável dela. Por exemplo, em [Morais et al. 2022] – Tabela 1, a opção de divisão funcional oito tem vazão mais alta (≈ 160 Gbps), o que significa que um subfluxo pode ser tão baixo quanto ≈ 16 Mbps. Além disso, as restrições de latência podem limitar os subfluxos úteis efetivos.

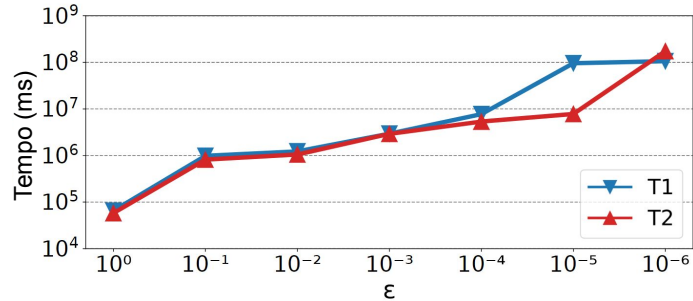


Figura 3.7: Tempo de execução em função de ε .

Finalmente, a formulação do problema apresentada, a qual considera divisão de carga entre múltiplos caminhos no roteamento, proporciona melhorias em relação à minimização de recursos computacionais utilizados e maximização do nível de centralização das funções na vRAN, apresentando maior eficiência e menor custo em comparação com o estado da arte.

Abordagem baseada em aprendizado de máquina

Neste capítulo, apresentamos nossa abordagem baseada em aprendizado de máquina para resolver o problema de posicionamento de funções da vRAN. Para resolver o problema implementamos um agente de aprendizado por reforço profundo utilizando a formulação do problema do PlaceRAN [Morais et al. 2022].

4.1 Introdução

O problema de posicionamento de funções da vRAN é um problema amplamente explorado na literatura, em diversos trabalhos os autores propõem formulações exatas (como nossa formulação apresentada no capítulo anterior) e apresentam soluções ótimas para o problema, visando aumentar a eficiência da rede, minimizando o custo e respeitando as restrições de capacidade e atraso das divisões funcionais. Nos trabalhos [Garcia-Saavedra et al. 2018, Fonseca et al. 2019, Garcia-Saavedra et al. 2018, Murti et al. 2020, Morais et al. 2022], os autores apresentam formulações que buscam um objetivo similar, o de maximizar a centralização de funções da vRAN. Entretanto, outro ponto em comum é a forma de resolver o problema formulado através da utilização de arcabouços de otimização (por exemplo, CPLEX), para encontrar a solução ótima do problema.

É bem conhecido que soluções que visam a busca por optimalidade tendem a ser computacionalmente complexas, demandando um alto tempo de processamento e alto consumo de recursos durante a busca ótima [Hubbs et al. 2020]. Desta forma, abordagens exatas para resolver o problema se tornam soluções ineficientes em cenários reais, devido à larga escala da rede em termos de quantidade de antenas e nós de processamento. Além disso, os órgãos padronizadores (por exemplo, 3GPP, ITU e ETSI) e alianças de empresas (por exemplo, O-RAN, TIP e GSMA) abordaram essa questão em várias frentes, adotando fortemente soluções de Inteligência Artificial (IA) e *Machine Learning* (ML) [Open RAN Alliance 2020], que tem mostrado resultados promissores em

diversas tarefas relacionadas à vRAN, incluindo a resolução de problemas complexos de otimização, que são tradicionalmente abordados por meio de programação matemática clássica, heurísticas ou meta-heurísticas.

Baseados nesses fatos e visando resolver o problema utilizando abordagens mais eficientes, os autores de [Murti et al. 2022, Murti et al. 2022] propõem uma nova abordagem para resolver o problema de posicionamento das funções da vRAN utilizando técnicas de IA e ML. Em [Murti et al. 2022, Murti et al. 2022], os autores apresentam um modelo de aprendizado por reforço profundo (DRL) para resolver o problema, visando maximizar a centralização enquanto reduz o custo total da rede. Os autores mostram que o agente de DRL melhora o tempo computacional em comparação com o modelo ótimo, apresentando soluções de qualidade próxima à solução ótima. No entanto, a formulação do problema utilizada é muito simplista e não representa o problema de forma integral. As principais simplificações são: (i) a consideração de CU fixa e única na topologia, (ii) DUs fixas e previamente posicionadas, (iii) opções limitadas de divisão funcional e (iv) caminhos únicos entre os nós da vRAN, o que simplifica o roteamento do fluxo no *crosshaul*.

Portanto, neste capítulo apresentamos uma abordagem com o uso de técnicas de aprendizado por reforço profundo para resolver o problema de posicionamento de funções da vRAN, utilizando nossa formulação apresentada em [Morais et al. 2022]. Considerando todas as divisões funcionais indicadas pelos padrões, usando nós vRAN flexíveis e CNs genéricos e heterogêneos para posicionar e executar as funções virtualizadas. O objetivo do nosso modelo de DRL é maximizar o nível de centralização de funções na vRAN sujeito aos requisitos de latência e à taxa de transferência das divisões funcionais, as capacidades dos enlaces no *crosshaul* e a capacidade de processamento dos CNs. Em resumo, apresentamos as seguintes contribuições:

- Formulamos um problema detalhado, conforme a abordagem flexível e genérica apresentada em nossos trabalhos anteriores.
- Implementamos um novo agente DRL para resolver o problema de posicionamento de funções da vRAN, usando um arcabouço de código aberto.
- Comparamos as soluções encontradas por nosso agente com outras duas abordagens ótimas na literatura, [Murti et al. 2020, Morais et al. 2022], avaliando a qualidade da solução encontrada.

4.2 Modelo de sistema e formulação do problema

Para representar o problema no cenário utilizando técnicas de DRL, utilizamos a formulação genérica e flexível do problema que apresentamos em [Morais et al. 2022]. Desta forma, como o modelo MuSt-RAN apresentado no capítulo anterior segue o

mesmo modelo de sistema que apresentamos em PlaceRAN, a definição dos conjuntos e parâmetros utilizados para formular o ambiente de aprendizado do agente inteligente são os mesmos definidos na Seção 3.2, com exceção do roteamento em múltiplos caminhos. Entretanto, para fins de comparabilidade entre as soluções encontradas pelo nosso agente e as soluções encontradas pelo agente de [Murti et al. 2020], nesta seção utilizaremos um conjunto diferente de divisões funcionais (ou seja, VNCs) definidos na Figura 4.1.

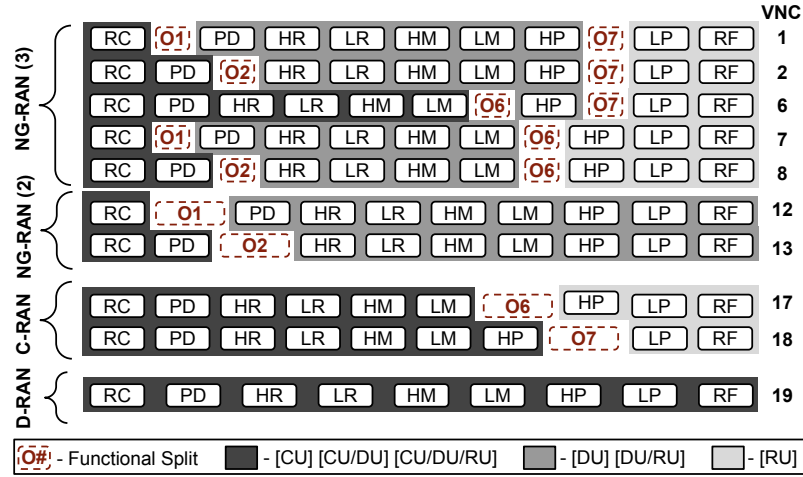


Figura 4.1: VNCs consideradas pelo agente DRL.

A Tabela 3.1 resume os parâmetros e variáveis utilizadas na formulação do problema. A formulação utilizada para implementar o agente de aprendizado por reforço profundo é apresentado a seguir:

$$\text{minimizar} \quad \Phi_1 - \Phi_2. \quad (4-1)$$

Sujeito a:

$$\sum_{p \in \mathcal{P}_l} \sum_{D_r \in \mathcal{D}} x_l^{p,r} = 1, \quad \forall b_l \in \mathcal{B}. \quad (4-2)$$

$$\sum_{D_r \in \mathcal{D}} \sum_{b_l \in \mathcal{B}} \sum_{p \in \mathcal{P}_l} [x_l^{p,r} (y_{e_{ij}}^{pBh} \alpha_{Bh}^r + y_{e_{ij}}^{pMh} \alpha_{Mh}^r + y_{e_{ij}}^{pFh} \alpha_{Fh}^r)] \leq e_{ij}^{Cap}, \quad \forall e_{ij} \in \mathcal{E}. \quad (4-3)$$

$$\sum_{e_{ij} \in \mathcal{E}} x_l^{p,r} y_{e_{ij}}^{pBh} e_{ij}^{Lat} \leq \beta_{Bh}^r, \quad \forall b_l \in \mathcal{B}, p \in \mathcal{P}_l, D_r \in \mathcal{D}. \quad (4-4)$$

$$\sum_{e_{ij} \in \mathcal{E}} x_l^{p,r} y_{e_{ij}}^{pMh} e_{ij}^{Lat} \leq \beta_{Mh}^r, \quad \forall b_l \in \mathcal{B}, p \in \mathcal{P}_l, D_r \in \mathcal{D}. \quad (4-5)$$

$$\sum_{e_{ij} \in \mathcal{E}} x_l^{p,r} y_{e_{ij}}^{pFh} e_{ij}^{Lat} \leq \beta_{Fh}^r, \quad \forall b_l \in \mathcal{B}, p \in \mathcal{P}_l, D_r \in \mathcal{D}. \quad (4-6)$$

$$\sum_{f_s \in \mathcal{F}} \sum_{b_l \in \mathcal{B}} \sum_{D_r \in \mathcal{D}} \sum_{p \in \mathcal{P}_l} x_l^{p,r} u_m^p M(c_m, f_s, b_l) \gamma_m^s \leq c_m^{Proc}, \quad \forall c_m \in \mathcal{C}. \quad (4-7)$$

A Equação (4-1) define a função objetivo. A parte positiva (Φ_1) representa a quantidade de CNs utilizados e a parte negativa (Φ_2) representa o nível de centralização

da vRAN. Como o objetivo é minimizar a Equação (1), nosso modelo persegue dois objetivos: minimizar a parte positiva (quantidade de CRs utilizados) e maximizar a parte negativa (nível de centralização). As Equações (4-2) a (4-7) representam as restrições do problema. A Equação (4-2) representa que apenas uma combinação de caminho e VNC devem ser escolhidos para uma RU. A Equação (4-3) representa a restrição de capacidade do enlace, evitando que a solução extrapole as capacidades dos enlaces. As Equações (4-4) a (4-6) impedem que a solução selecione um VNC inviável devido à latência dos enlaces (*backhaul*, *midhaul* e *fronthaul*, respectivamente). Finalmente, a Equação (4-7) evita que a solução estoure as capacidades computacionais dos CNs.

4.3 Solução de aprendizado por reforço para solução do problema

Nas próximas seções, apresentamos as características e informações utilizadas na implementação do agente de aprendizado por reforço profundo para resolver o problema de posicionamento de funções da vRAN. Desta forma, nosso ambiente de DRL corresponde na escolha conjunta de caminho e VNC para as RUs da topologia, considerando múltiplos caminhos entre o núcleo da rede e as RUs, onde cada caminho corresponde a uma configuração de enlaces, CNs (que são capazes de posicionar vCUs, vDUs ou ambos) e RU. Os VNCs definem quais funções vRAN serão alocadas na vCU ou vDU e os requisitos de recursos de rede para cada RU. O objetivo do nosso agente DRL é selecionar um VNC para cada RU e um caminho para rotear os dados, visando maximizar a centralização das funções da vRAN enquanto minimiza o número de CNs utilizados.

4.4 Ações, espaço de observação e função de recompensa

Nesta seção, definimos as propriedades da formulação de nosso agente DRL, com informações sobre as ações, espaço de ações, espaço de observação e função de recompensa do nosso ambiente de aprendizado por reforço profundo.

4.4.1 Ações

O agente DRL recebe como entrada um conjunto \mathcal{B} de RUs a serem alocadas e um conjunto \mathcal{P}_l de caminhos contendo os k -caminhos mais curtos que vão do núcleo da rede (v_0) para cada RU $b_l \in \mathcal{B}$. Para definir o ambiente, representamos os estados como o conjunto de RUs a serem alocados pelo agente DRL. Consideramos como ações um conjunto de tuplas (p_t, d_t) representando as configurações de alocação possíveis das RUs, onde p_t representa qual caminho é utilizado para rotear o fluxo da RU e d_t representa qual

VNC está alocado para a RU. Em cada etapa, o agente escolhe uma ação $a = (p_t, d_t)$ de um conjunto de ações possíveis. Observe que p_t e d_t representam os índices do caminho escolhido para roteamento e VNC a ser utilizado, implicando que o espaço de ação definido é do tipo discreto.

A formulação do problema utilizada nos leva a um problema combinatório que afeta diretamente o espaço de ações definido, dando ao agente DRL muitas ações possíveis em cada etapa. A dimensão do espaço de ações de nossa formulação pode ser representada como:

$$\kappa = |\mathcal{D}| \sum_{b_l \in \mathcal{B}} (|\mathcal{P}_l|), \quad (4-8)$$

em que $|\mathcal{D}|$ representa a quantidade de VNCs considerados e $|\mathcal{P}_l|$ representa a quantidade de caminhos considerados para a RU $b_l \in \mathcal{B}$. Desta forma, em uma topologia com 10 RUs, com $k = 2$ no algoritmo de k -caminhos mais curtos e 9 VNCs disponíveis, o espaço de ações representará 180 ações possíveis, gerando uma combinação de $(9 \times 2)^{10} \approx 3,5 \times 10^{12}$ soluções possíveis.

4.4.2 Espaço de observação e conjunto de ações viáveis

Problemas combinatórios com espaço de ação esparsos podem se tornar computacionalmente caros, principalmente para treinamento e convergência do agente DRL. Esse problema é abordado em [Murti et al. 2022, Murti et al. 2022], no qual uma política gradiente personalizada e uma abordagem de codificador-decodificador foram consideradas para a redução do espaço de ações. No entanto, em nossa formulação aplicamos uma estratégia de redução do espaço de ações a partir da criação de um subconjunto de ações viáveis no estado observado, no qual o agente pode selecionar apenas ações que não quebrem nenhuma restrição de capacidade da vRAN. Assim, nosso ambiente DRL pode ser facilmente implantado em uma estrutura de código aberto, a fim de testar vários algoritmos de treinamento e escolher o que apresentar melhor desempenho.

Para criar o subconjunto de ações possíveis, definimos o espaço de observação como o estado atual de recursos disponíveis na rede e a quantidade de RUs restantes a serem alocadas no episódio (ou seja, os RUs não alocados). O espaço de observação é uma lista em que cada posição indica a ocupação atual dos enlaces, o processamento disponível nos CNs, e a quantidade de RUs a serem alocadas. Com as informações do espaço de observação, nosso ambiente calcula o conjunto de ações viáveis \mathcal{Z}_t para cada estado t , permitindo ao agente sempre escolher uma combinação viável de caminho e VNC em cada ação. No entanto, em alguns episódios, o agente pode atingir um estado

em que não restam ações viáveis possíveis. Essa situação ocorre em episódios em que o agente escolhe as primeiras ações de forma gulosa, utilizando grande parte do recurso disponível de imediato para as primeiras RUs. Isso significa que nosso conjunto de ações viáveis estará vazio nas próximas etapas, ou seja, $\mathcal{Z}_t = \emptyset$, e esse cenário configura o que chamamos de falha.

4.4.3 Função de recompensa

Cada episódio do nosso ambiente consiste em um conjunto de ações escolhidas que correspondem a uma combinação de caminho e VNC para cada RU na topologia. Desta forma, a cada episódio o agente pode encontrar uma solução viável do problema, exceto em casos de falha ($\mathcal{Z}_t = \emptyset$). Quando ocorre uma situação de falha, significa que o agente esgotou a quantidade de recursos disponíveis e ainda há RUs a serem alocadas, levando o agente a chegar a uma solução inviável do problema. Isso ocorre, pois o agente sempre inicia o treinamento do zero, ou seja, sem uma solução factível como ponto de partida e sem conhecimento prévio do problema. Isso se deve, principalmente, à formulação do problema considerar múltiplos caminhos para o roteamento dos fluxos, pois, para iniciar o treinamento a partir de uma solução viável, o agente precisaria ter os caminhos previamente selecionados, o que não é possível, pois o próprio agente seleciona os caminhos a serem utilizados.

Ao final de cada episódio, o ambiente avalia a precisão das ações selecionadas, dando ao agente um sinal de recompensa, que consiste em uma recompensa positiva em caso de sucesso ou uma penalidade em caso de falha. Esses sinais de recompensa são usados pelo agente para atualizar os parâmetros da rede neural para melhorar o desempenho do modelo. Em caso de sucesso, ou seja, quando o agente selecionou um conjunto de ações viáveis para toda RU na topologia sem quebrar nenhuma restrição, isto é, apresenta uma solução viável do problema, o ambiente retorna ao agente um sinal positivo representando o nível de centralização alcançado no episódio. Em caso de falha, o ambiente penaliza o agente com um sinal de recompensa negativo e termina o episódio prematuramente. O sinal de penalidade é definido como a quantidade de RUs restantes para completar o episódio, ou seja, a quantidade de RUs que não foram alocadas devido à falta de recursos na rede. Os sinais de recompensa positiva e negativa podem ser definidos da seguinte forma:

$$\phi = \sum_{z \in \mathcal{Z}_s} [z_b^{p,v} \lambda(a)], \quad \forall s \in \mathcal{S}, b \in \mathcal{B}, v \in \mathcal{D}, p \in \mathcal{P}_l. \quad (4-9)$$

$$\psi = |\mathcal{B}| - \sum_{z \in \mathcal{Z}_s} z_b^{p,v}, \quad \forall s \in \mathcal{S}, b \in \mathcal{B}, v \in \mathcal{D}, p \in \mathcal{P}_l. \quad (4-10)$$

As Equações (4-9) e (4-10) representam o cálculo da recompensa positiva e negativa respectivamente, onde $z_b^{p,v} = \{0, 1\}$ representam se o caminho p com VNC v foi selecionado para RU b no episódio. O agente atualiza os parâmetros da rede neural guiados pelos sinais de recompensa recebidos durante o treinamento, visando maximizar a recompensa de longo prazo, ou seja, maximizar o nível de centralização da solução.

4.4.4 Método de treinamento

Para treinar nosso agente de aprendizado por reforço profundo, utilizamos o algoritmo *Proximal Policy Optimization* (PPO). O PPO é um método de política gradiente que emprega um objetivo temporário construído baseado no objetivo original e visa encontrar novas políticas que visam tornar o objetivo temporário o mais próximo possível do objetivo original. Escolhemos o algoritmo PPO, pois, em nossos testes, ele apresentou melhores resultados em termos de precisão e convergência quando comparado a outros algoritmos de aprendizado. No entanto, qualquer algoritmo DRL pode ser utilizado para resolver o problema com nossa abordagem, pois na implementação de nossa solução utilizamos um arcabouço de código aberto que permite a modificação e atualização do ambiente conforme o estado da arte.

4.5 Resultados

Para treinar e avaliar nosso agente, utilizamos o arcabouço de código aberto *Stable Baselines 2*, com *Python* versão 3.6.15 e *TensorFlow* versão 1.15.4. Detalhes adicionais sobre o ambiente e todo o código-fonte estão disponíveis publicamente em um repositório do *Github*.¹

Nosso agente DRL é projetado para encontrar uma solução para o problema de posicionamento de funções da vRAN, através de uma formulação genérica que apresentamos em [Morais et al. 2022]. Dessa forma, é natural compararmos os resultados encontrados pelo agente DRL com os resultados obtidos pela abordagem ótima, ou seja, PlaceRAN. Além disso, como nossa estratégia de formular um modelo de aprendizado por reforço profundo para resolver o problema é inspirada em [Murti et al. 2022, Murti et al. 2022], nós também comparamos os resultados obtidos por nosso agente DRL com o modelo ótimo apresentado por Murti [Murti et al. 2022].

Entretanto, a formulação de Murti não considera todos os detalhes apresentados em nossa formulação, ou seja, o problema formulado por Murti [Murti et al. 2022] é um subproblema de nossa formulação. Para fazer uma comparação justa entre os trabalhos,

¹<https://github.com/LABORA-INF-UFG/paper-GVAK-2022>

usaremos uma versão simplificada de nossa formulação que atende todas as limitações de Murti [Murti et al. 2022], permitindo comparar ambos os modelos ótimos com o nosso agente DRL. Para definir o subproblema que corresponde à formulação de Murti, é necessário fazer algumas observações sobre as topologias de entrada.

A primeira observação é sobre os nós CUs e DUs. A formulação de Murti [Murti et al. 2022] define a topologia com uma CU única, fixa e centralizada, o que implica que as topologias utilizadas para comparação das soluções devem ter um CN centralizado (Figura 4.2) que atuará como CU no modelo de Murti. Além disso, Murti recebe como entrada as localizações das DUs fixas da topologia (co-alocadas com RUs), enquanto PlaceRAN e o agente DRL usam o conceito de vDU dinâmico. Para tanto, também definimos nas topologias as localizações das DUs para o modelo de Murti, o que implica que os enlaces entre os nós DUs e RUs suportam os requisitos de divisão funcional O7 (Figura 4.1).

A segunda observação é sobre o roteamento dos fluxos no *crosshaul*. Murti considera um caminho único para cada DU (e RU) conectado a CU fixa, enquanto PlaceRAN e o agente DRL podem lidar com vários caminhos entre os nós vRAN. Desta forma, as entradas do PlaceRAN e as entradas do agente DRL conterão exatamente um caminho entre o núcleo da rede e cada RU da topologia, para atender a essa limitação. Finalmente, consideramos o conjunto de VNCs originalmente apresentado por cada formulação. O agente DRL e PlaceRAN usa os VNCs definidos na Figura 4.1 (exceto VNC 6) e Murti [Murti et al. 2022] usa os VNCs 2, 6, 18 e 19. Observe que os VNCs 2 e 6, na formulação de Murti, são considerados NG-RAN(2), pois os autores consideram DU e RU co-alocados (ou localizados próximos), considerando apenas dois nós vRAN para cada RU.

Para avaliar as soluções, foram consideradas topologias com 8, 16, 32, 64, 128, 256 e 512 nós, onde todas as informações de entrada estão disponíveis e previamente calculadas, por exemplo, caminhos, localização de RUs e opções de VNCs. A Figura 4.2 apresenta as topologias e a Tabela 4.1 apresenta informações sobre os recursos da rede.

Parâmetros	Valores
$ \mathcal{B} $	{5, 10, 19, 35, 80, 122, 213}
$ \mathcal{C} $	{8, 16, 32, 64, 128, 256, 512}
$ \mathcal{D} $	{4, 9}
c_m^{Proc} (núcleos)	{8, 16, 32, 64}
e_{ij}^{Cap} (Gbps)	{200, 400, 800, 1000}
e_{ij}^{lat} (ms)	{0.163, 0.22, 0.29}

Tabela 4.1: Parâmetros utilizados na avaliação.

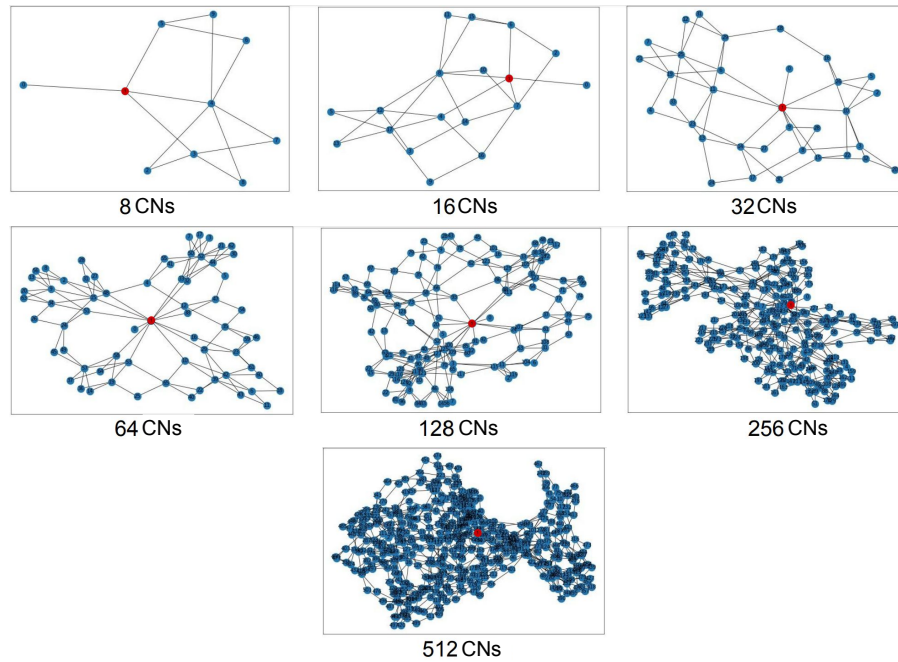


Figura 4.2: Topologias utilizadas na avaliação. O ponto vermelho representa o posicionamento da CU, utilizada na formulação de Murti, em azul são representados os CNs.

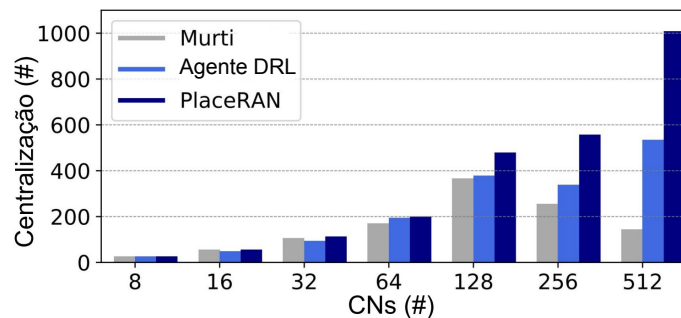


Figura 4.3: Nível de centralização das soluções para cada topologia.

A Figura 4.3 a centralização das funções da vRAN encontrado para cada topologia. Podemos observar que até 32 nós, Murti [Murti et al. 2022] e PlaceRAN encontram soluções semelhantes, com o mesmo nível de centralização. Isso se deve principalmente ao tamanho das topologias. Em topologias com menor quantidade de CNs, os recursos de transporte e processamento da vRAN não representam gargalos, devido ao pequeno número de RUs a serem alocadas. Isso faz com que os modelos exatos apresentem soluções semelhantes, apesar de suas diferenças no posicionamento dos nós vRAN (CU e DUs). Além disso, em cenários com até 32 nós, o agente DRL apresenta soluções que centralizam menos VNFs. Como o agente DRL não utiliza uma abordagem exata (por exemplo, *branch and bound*), é natural que as soluções apresentadas pelo agente DRL centralizem menos VNFs em comparação com o PlaceRAN. Dado que em cenários menores o modelo de Murti tende a empatar com PlaceRAN em termos de nível de centralização, é esperado que o agente DRL também apresente resultados piores que Murti em tais cenários.

No entanto, com 64 nós, PlaceRAN e o agente DRL começam a atingir um nível de centralização mais alto em comparação com Murti [Murti et al. 2022], e esse comportamento é mantido até a maior topologia com 512 nós. Isso se deve à necessidade de um melhor gerenciamento de recursos à medida que o número de RUs aumenta, levando os recursos de transporte e processamento da vRAN a apresentarem gargalos. Desta forma, o PlaceRAN e o agente DRL atingem um maior nível de centralização, pois consideram o posicionamento das vDUs e vCUs de forma dinâmica, enquanto Murti [Murti et al. 2022] as considera fixas. Além disso, o maior número de opções de VNCs também permite que o PlaceRAN e o agente DRL alcancem tais melhorias, visto que o conjunto de divisões funcionais consideradas por Murti é limitado.

Por um lado, o agente DRL encontra soluções que apresentam níveis de centralização cada vez melhores, em comparação ao modelo de Murti [Murti et al. 2022]. Por outro lado, o nível de centralização encontrado pelo agente DRL está cada vez mais distante do nível de centralização encontrado pelo PlaceRAN. Para entender esse comportamento, é necessário analisar as escolhas de VNCs feitas tanto pelos modelos ótimos quanto pelo agente DRL, a fim de entender a diferença no nível de centralização apresentado em topologias com maiores quantidades de CNs.

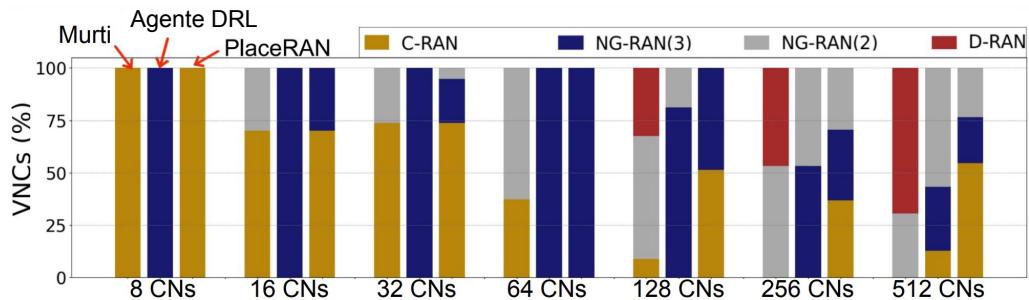


Figura 4.4: VNCs utilizadas em cada instância na avaliação.

A Figura 4.4 apresenta as VNCs escolhidas pela solução de cada modelo. Note que até 16 CNs, as soluções ótimas escolhem a mesma quantidade de C-RANs para resolver o problema, enquanto o agente DRL escolhe NG-RAN(3) (VNC 2). Conforme discutido, isso se deve à menor quantidade de RUs nas topologias menores, ou seja, menor disputa por recursos. Entretanto, no cenário com 32 CNs, os modelos ótimos começam a apresentar escolhas diferentes de VNCs. Enquanto Murti [Murti et al. 2022] continua usando C-RANs, PlaceRAN atinge um maior nível de centralização usando NG-RAN(3), os mesmos VNCs apresentados pelo agente DRL. Isso justifica o aumento do nível de centralização do agente DRL em comparação ao modelo de Murti. Além disso, a partir de 128 CNs, Murti [Murti et al. 2022] passa a selecionar D-RANs (VNC 19) para algumas RUs, enquanto o agente DRL e PlaceRAN não apresentam D-RANs. Isso se deve à necessidade de melhor gerenciamento de recursos em topologias maiores, o que justifica a diferença entre os níveis de centralização apresentados na Figura 4.3.

O modelo de Murti [Murti et al. 2022] segue aumentando a quantidade de D-RANs conforme o tamanho da topologia aumenta, atingindo 60% de D-RANs para a topologia de 512 nós. Isso reforça a conclusão de que em cenários com maior competição por recursos compartilhados, nossa formulação genérica apresenta melhores resultados, por exemplo, na presença de serviços *multi-access edge computing* (MEC). O modelo exato PlaceRAN consegue fazer escolhas de VNCs mais otimizadas em relação ao agente DRL, pois apresenta maior número de C-RANs, ou seja, VNC que centraliza mais funções da vRAN. Esse resultado mostra que para topologias maiores, o agente DRL apresenta soluções que centralizam menos funções da vRAN quando comparado ao ótimo, o que indica um importante ponto de melhoria para nosso agente DRL.

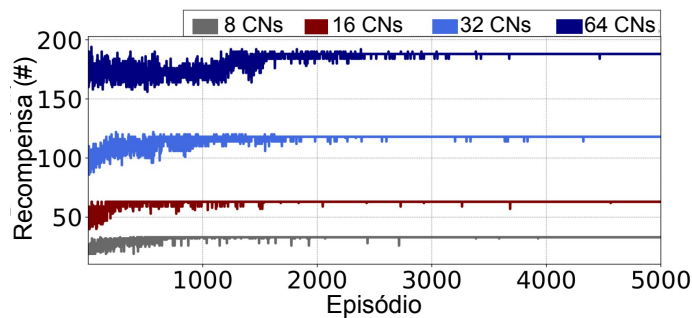


Figura 4.5: Histórico dos sinais de recompensa durante o treinamento.

A Figura 4.5 apresenta o histórico de sinais de recompensa obtido pelo agente DRL a cada episódio durante a fase de treinamento. Podemos observar que nos primeiros episódios, o agente apresenta um comportamento mais exploratório, visando escapar de máximos locais durante os primeiros passos do treinamento. Após 2.000 episódios, o agente começa a agir de forma mais assertiva, enquanto converge para a solução final. O número médio de episódios para convergência está entre 2.000 e 3.000 episódios nos cenários apresentados. No entanto, o ponto de convergência deve variar conforme o tamanho da topologia e a quantidade de recursos disponíveis na vRAN. Cenários com maior competição por recursos compartilhados tendem a ser mais desafiadores.

Finalmente, nosso agente de aprendizado por reforço profundo foi projetado para ser executado em uma estrutura de código aberto e visa maximizar o nível de centralização do vRAN. Para elaborar nosso agente DRL, utilizamos a formulação apresentada em PlaceRAN [Morais et al. 2022], considerando múltiplos caminhos entre os nós e um conjunto de divisões funcionais sugeridas pelos órgãos de padronização. Comparamos os resultados obtidos pelo nosso agente DRL com duas abordagens ótimas da literatura. Os resultados mostram que nossa abordagem escala notavelmente em comparação com soluções exatas e também apresenta resultados satisfatórios em termos de nível de centralização de funções e escolhas de divisões funcionais.

Abordagem baseada em meta-heurística

Neste capítulo, apresentaremos nossa segunda solução não exata para resolver o problema de posicionamento de funções da vRAN. Essa solução é baseada em conceitos de métodos evolutivos, mais precisamente algoritmos genéticos. A escolha dessa abordagem foi motivada por diversos trabalhos da literatura que aplicaram com sucesso métodos evolutivos para solucionar problemas clássicos de tomada de decisão, os quais discutiremos a seguir.

5.1 Algoritmo genético

Conforme discutido em [Morais et al. 2022], o problema de posicionamento de funções da vRAN pertence à classe de problemas NP-difícil. Assim, é inviável resolver tal problema utilizando abordagens exatas que apresentam soluções ótimas em um cenário prático, onde a topologia pode conter centenas e milhares de RUs. Nesse caso, abordagens não exatas surgem como soluções promissoras para resolver o problema em tempo hábil e viável.

Algoritmos Genéticos (GAs) são métodos de metaheurísticas de busca por soluções baseada nos princípios da teoria da evolução natural de Charles Darwin. GAs são muito utilizados na literatura para resolver problemas de decisão de diversas naturezas, por exemplo, o Problema da Mochila Multidimensional [Chu e Beasley 1998], realizando uma busca heurística não exata eficiente, e apresenta resultados promissores, ou seja, próximos ao ótimo. Assim, nesta seção descrevemos as características da nossa solução de GA para resolver o problema de posicionamento de funções da vRAN, visando entregar soluções de qualidade próximas ao ótimo em tempo hábil. Tais características incluem a representação de uma solução como um cromossomo de um indivíduo, a inicialização da população na primeira geração, os aspectos dos operadores de variação (ou seja, cruzamento e mutação) e, finalmente, a seleção dos sobreviventes para gerações futuras.

5.1.1 Representação dos cromossomos

Para resolver o problema de posicionamento de funções da vRAN, devemos escolher uma combinação de caminho e VNC para cada RU na vRAN que minimize a equação 4-1, que representa a função objetivo do nosso modelo de otimização (Place-RAN) e também a função aptidão do nosso Algoritmo Genético. Para representar as decisões de posicionamento, definimos um cromossomo de tamanho $|\mathcal{B}|$, onde cada posição $l \in \{0, 1, \dots, |\mathcal{B}|\}$ representa uma escolha de um caminho $p \in \mathcal{P}_l$ e um VNC $v \in \mathcal{D}$ para uma RU $b_l \in \mathcal{B}$. Assim, cada cromossomo representa uma solução completa (viável ou inviável) com uma combinação de caminho e VNC para cada RU. A Figura 5.1 ilustra uma instância de topologia vRAN e seu cromossomo resultante. Observe que a topologia tem 6 RUs, representadas como triângulos, e o cromossomo tem 6 posições, onde cada posição é mapeada para um par (p, v) para uma determinada RU.

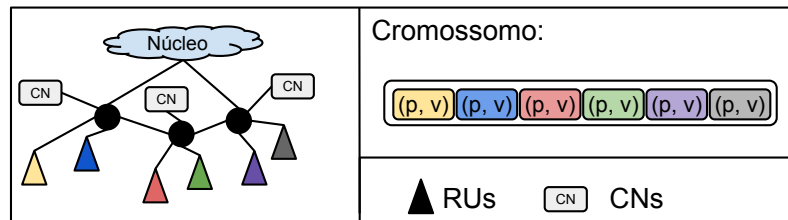


Figura 5.1: Uma topologia vRAN e seu cromossomo correspondente.

5.1.2 Primeira geração

Conforme discutimos anteriormente, o cromossomo de cada indivíduo representa uma solução de posicionamento completa para o problema de posicionamento de funções da vRAN, e cada geração representa um conjunto de soluções viáveis. Em GAs, o processo evolutivo começa com uma população inicial criada com base nos dados de entrada da instância. Como o espaço de busca não é conhecido antecipadamente e para evitar viés, os GAs geralmente geram a população inicial aleatoriamente, criando vários pontos de busca no espaço de soluções. No entanto, essa estratégia não aproveita nenhum conhecimento prévio do problema, que podem auxiliar na geração de indivíduos de maior qualidade. Para isso, nossa GA combina ambas abordagens criando aleatoriamente os indivíduos da primeira geração, mas também incluindo um único indivíduo heurístico gerado a partir de conhecimentos prévios sobre os VNCs e selecionando-os de acordo com sua centralização, ou seja, dando prioridade aos VNCs NG-RAN(3) e C-RAN.

5.1.3 Função de reparo

Na população inicial, o indivíduo heurístico e os gerados aleatoriamente podem representar soluções não viáveis do problema. De fato, como o problema de posiciona-

mento de vRAN possui um espaço de solução fortemente restrito e proibitivo devido às restrições de capacidade, latência e largura de banda, a chance de um indivíduo gerado na primeira geração ser viável é baixa, principalmente em cenários com recursos escassos. Para resolver esse problema, nosso GA adota uma estratégia de reparar indivíduos inviáveis (representando soluções inviáveis) em viáveis. Essa estratégia é amplamente adotada na literatura, principalmente em problemas com espaço de busca restritivo como o problema da mochila, a cobertura de vértices e o problema do caixeiro viajante [Orvosh e Davis 1994].

Um dos desafios na formulação de uma função de reparo é modificar os indivíduos não viáveis sem enviesar a busca, fazendo mudanças pontuais que não modifiquem a região do espaço de busca em que o indivíduo se encontra. Isso significa que uma função de reparo não deve transportar o indivíduo para outra região no espaço de soluções, mas para regiões vizinhas que são viáveis. Assim, em nosso GA, propomos uma função de reparo que realiza mudanças aleatórias em indivíduos inviáveis para torná-los viáveis. O Algoritmo 5.1 representa a função de reparo proposta. Inicialmente, a função de reparo faz uma cópia (\bar{s}) do indivíduo a ser reparado (s). Então, enquanto o indivíduo é inviável, a função de reparo seleciona uma posição aleatória do indivíduo ($RANDOM_POS$) e escolhe uma nova combinação ($NEW_COMBINATION$) de caminho e VNC para a RU representada pela posição selecionada. Se a modificação não tornar o indivíduo viável, o processo se repete. Caso o indivíduo se torne viável após a modificação, o algoritmo retorna o novo indivíduo viável encontrado.

Algoritmo 5.1: Função reparo

Entrada: Indivíduo inviável s com $|\mathcal{B}|$ posições.

Saída: Um indivíduo viável \bar{s} .

```

1  $\bar{s} \leftarrow s$ ;
2  $U \leftarrow True$ ;
3 enquanto  $U$  faça
4    $i \leftarrow RANDOM\_POS(|\mathcal{B}|)$ ;
5    $x \leftarrow NEW\_COMBINATION(i, \bar{s})$ ;
6    $\bar{s}[i] \leftarrow x$ ;
7   se  $IS\_FEASIBLE(\bar{s})$  então
8      $U \leftarrow False$ ;
9   fim
10 fim
11 retorna  $\bar{s}$ ;
```

A função de reparo proposta é simples, altamente aleatória e modifica o cromossomo do indivíduo sequencialmente, buscando soluções viáveis na vizinhança da solução inicial. Porém, a função de reparo descrita apresenta uma alta complexidade computacional em cenários com recursos escassos, ou seja, com poucas soluções viáveis. Em tais

cenários, a função de reparo executa muitas iterações até que o indivíduo se torne viável. Desta forma, visando evitar esse problema, utilizamos a função de reparo apenas nos indivíduos da primeira geração, encontrando diversos pontos iniciais no espaço de buscas e povoar um conjunto de soluções viáveis conhecidas nas primeiras gerações. Esse conjunto de soluções viáveis conhecidas é então usado para implementar uma política de substituição nas gerações subsequentes, em que ao invés de usar a função de reparo, nós realizamos a substituição de uma solução inviável por uma solução viável conhecida no conjunto. A política de substituição também é simples e altamente aleatória, realizada em tempo $O(1)$, e reposiciona a busca para um ponto em que se conhece que existe viabilidade.

5.1.4 Operadores de variação

Um conjunto de soluções viáveis S , herdadas de gerações anteriores ou inicialmente geradas, são conhecidas em cada geração. Esse conjunto é composto por diferentes indivíduos que representam diferentes pontos viáveis no espaço de soluções. A busca no espaço de soluções é baseada em dois aspectos, *exploitation* e *exploration*, ambos presentes em nosso AG como operadores de variação. Para alcançar a *exploration*, usamos a recombinação, ou cruzamento, de indivíduos da mesma geração. Isso é feito selecionando um conjunto de indivíduos pais que irão gerar novos indivíduos filhos para gerações futuras. O conjunto de pais é gerado seguindo uma seleção por torneio amplamente adotada na literatura. Inicialmente, um conjunto com σ indivíduos da geração é selecionado aleatoriamente, e então os dois indivíduos selecionados que apresentarem maior qualidade (função de aptidão) são inseridos no conjunto de pais. Esse mecanismo se repete até que os pais $|S|$ sejam selecionados. Cada par de pais gera dois novos indivíduos a partir do cruzamento em uma posição escolhida aleatoriamente. Por fim, dependendo da taxa de recombinação Θ , os pais podem ou não ser substituídos por seus filhos no conjunto sobrevivente, sendo substituídos pelos filhos nas gerações seguintes.

Para realizar a *exploitation*, utilizamos o conceito de mutação, onde cada posição do indivíduo tem uma chance de mutação uniforme Ω . Se uma posição for selecionada pela mutação, uma nova combinação (p, v) deve ser escolhida para a RU representada pela posição selecionada. Porém, cada novo indivíduo gerado, seja pela recombinação ou mutação, pode representar uma solução inviável para o problema. Nesse caso, a política de substituição é usada para substituir um indivíduo inviável por um indivíduo viável do conjunto de soluções viáveis conhecidas.

Por fim, após as operações de recombinação e mutação ocorrerem, um novo conjunto de indivíduos sobreviventes é selecionado, formado pelos pais sobreviventes, os filhos que substituíram seus pais e os indivíduos que sofreram mutação. No entanto, dado

que o problema de posicionamento do vRAN necessita de apenas uma única solução, e dado que procuramos a melhor solução conhecida durante a busca. Em nossa GA nós utilizamos o conceito de pressão de elitismo [Bhandari, Murthy e Pal 1996], em que o melhor indivíduo conhecido nas gerações é passado de geração em geração, ou seja, o melhor indivíduo conhecido durante toda a busca estará sempre no conjunto de indivíduos sobreviventes. Desta forma, conseguimos a qualquer momento durante a busca conhecer o melhor indivíduo encontrado nas gerações anteriores.

5.2 Avaliação

Para implementar nosso GA utilizamos a formulação apresentada na Seção 3.2, ou seja, a formulação que apresentamos em PlaceRAN [Morais et al. 2022], que também foi o problema resolvido pelo nosso agente DRL apresentado no Capítulo 4 e publicado em [Almeida et al. 2022]. Desta forma, para avaliar nosso GA iremos comparar as soluções encontradas por tais modelos com a solução encontrada por nosso GA, pois todos resolvem o mesmo problema de decisão. Além disso, também comparamos as soluções do nosso GA com o modelo apresentado por Murti [Murti et al. 2022]. No entanto, como discutido no capítulo anterior, a formulação de Murti é, na verdade um subproblema da nossa formulação, pois ele considera apenas uma CU fixa e um conjunto de DUs fixas, além da ausência de decisões de roteamento no *crosshaul*, ou seja, considera apenas uma opção de rota para cada RU. Assim, organizamos a avaliação em duas partes: i) a primeira considerando o subproblema apresentado por Murti que não envolve decisões de roteamento, e ii) a segunda considerando o roteamento no *crosshaul* como parte das decisões. Essa última avaliação envolve os seguintes modelos: PlaceRAN [Morais et al. 2022], DRL agent [Almeida et al. 2022], e nosso GA. Table 5.1 resume os parâmetros do GA e as características da rede empregadas em nossos experimentos.

Parâmetros	Valores
Θ, Ω and $ S $	75%, 2% e 50
σ , gerações	20 e 1000
c_m^{Proc} (núcleos)	{8, 16, 32, 64, 128, 256}
e_{ij}^{Cap} (Gbps)	{50, 100, 200, 400, 1000, 10000}
e_{ij}^{at} (ms)	{0.161, 0.22, 0.29, 0.354, 0.44, 0.5}

Tabela 5.1: Parâmetros utilizados nos testes.

5.2.1 Avaliação sem decisão de roteamento

Avaliamos os modelos usando diferentes tamanhos de instância, ou seja, diferentes números de RUs. Apresentamos duas variantes do GA: GA (1ª solução) que emprega apenas a solução obtida na primeira geração, e GA (melhor solução) que emprega

a melhor solução obtida até o fim da busca. Todas as instâncias usadas na comparação consideram as limitações do subproblema definido por Murti [Murti et al. 2022]. Para o modelo PlaceRAN, o agente DRL e nosso GA, utilizamos os VNCs mostrados na Fig. 4.1 (exceto VNC 6), enquanto o modelo de Murti considera os VNCs 2, 6, 18 e 19.

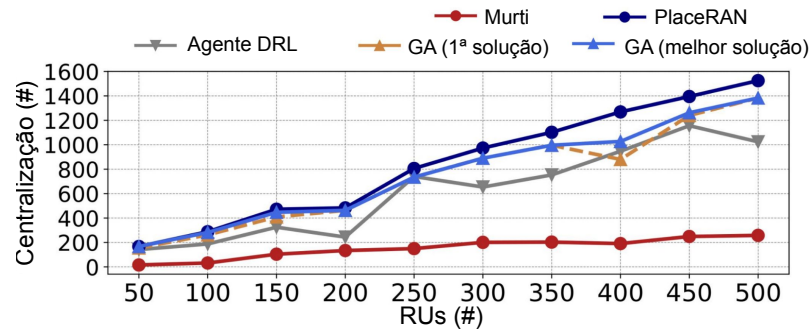


Figura 5.2: Nível de centralização no cenário sem roteamento.

A Figura 5.2 mostra a qualidade das soluções encontradas por cada modelo apresentando o nível de centralização em função do número de RUs. Em todas as instâncias, as soluções encontradas por Murti [Murti et al. 2022] apresentam um nível de centralização bem inferior aos encontrados pelos demais modelos. Isso se deve à diferença entre nossa formulação e a formulação de Murti, onde, mesmo não considerando o roteamento como parte da decisão, o modelo PlaceRAN [Morais et al. 2022], o agente DRL [Almeida et al. 2022] e nosso GA, encontram melhores soluções considerando mais VNCs e posicionamento dinâmico dos nós vCUs e vDUs. Como esperado, o PlaceRAN apresenta o melhor nível de centralização, encontrando uma solução ótima para cada instância do problema. Por outro lado, entre os modelos não exatos, a GA (melhor solução) supera o agente DRL, encontrando soluções com melhores níveis de centralização em diversas instâncias da avaliação. A solução GA (1ª solução) também apresenta um desempenho satisfatório, superando o agente DRL na maioria das instâncias. Isso mostra o sucesso da heurística aplicada na primeira geração para encontrar de forma eficiente nas primeiras gerações um indivíduo de alta qualidade.

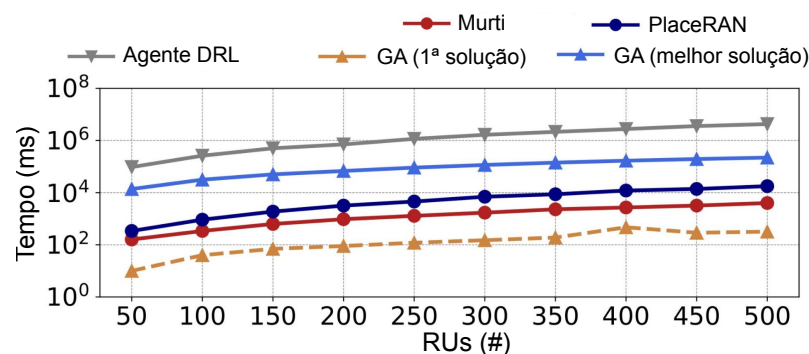


Figura 5.3: Tempo de execução sem considerar roteamento.

A Figura 5.3 apresenta o tempo de execução dos modelos em função do número de RUs. A solução GA (1ª solução) apresenta o melhor desempenho, sendo mais rápida até que o modelo de Murti [Murti et al. 2022] que resolve um problema mais simples. Já o PlaceRAN é o terceiro modelo mais rápido, o que ilustra o quão poderosa é nossa formulação, pois fornece uma solução ótima em tempo eficiente quando não há decisão de roteamento. A GA (melhor solução) não é competitiva neste contexto, o que é esperado uma vez que esse modelo foi projetado para resolver um problema mais complexo. Por fim, o agente DRL apresenta o pior desempenho dado que para cada experimento foi considerado o tempo de treinamento do agente, considerando que nenhuma outra solução apresentada possui etapas de pré-processamento.

5.2.2 Avaliação com decisão de roteamento

Nesta subseção de resultado, nós utilizamos a mesma metodologia de avaliação da subseção anterior, entretanto desta vez consideramos quatro rotas possíveis para cada RU, o que exclui Murti da comparação por não ter decisão de roteamento.

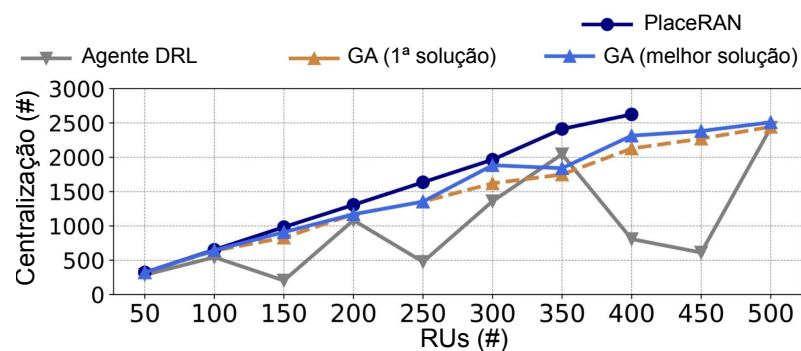


Figura 5.4: Nível de centralização das soluções considerando o roteamento na decisão.

A Figura 5.4 apresenta o nível de centralização em função do número de RUs para cada modelo. Observe que considerar o roteamento na decisão permite que todos os modelos encontrem soluções melhores (ou seja, valores mais altos de centralização) do que as apresentadas anteriormente (na Fig. 5.2), o que corresponde a um melhor gerenciamento de recursos de rede. Naturalmente, o PlaceRAN apresenta o maior nível de centralização, mas agora enfrenta o problema de escalabilidade. Em nossos experimentos, o PlaceRAN não foi capaz de resolver o problema para 450 RUs (ou mais), apesar de executar um período considerável, conforme apresentaremos mais adiante. Por outro lado, modelos não exatos resolvem todas as instâncias do problema, mesmo as maiores. Ambas soluções do GA, ou seja, GA (1ª solução) e GA (melhor solução), superam o agente DRL na maioria das instâncias e também exibem desempenho mais estável, devido ao aumento do número de estados no ambiente DRL.

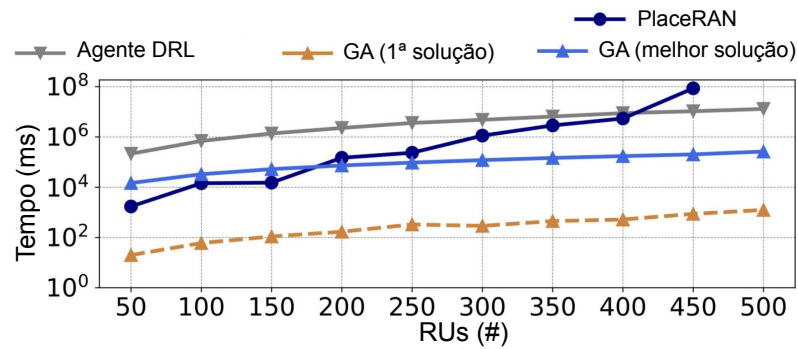


Figura 5.5: Tempo de execução considerando roteamento.

Figura 5.5 apresenta o tempo de execução consumido por cada modelo. O agente DRL apresenta o maior tempo de execução para a maioria das instâncias devido ao tempo de treinamento, entretanto é capaz de resolver todos os tamanhos de instâncias. Já o PlaceRAN cresce exponencialmente e tem sua execução abortada em 450 RUs após rodar por mais de 24 h sem encontrar a solução ótima. Enquanto a GA (melhor solução) já é visivelmente mais rápida que o agente DRL, o desempenho da GA (1ª solução) merece destaque. Dados os resultados satisfatórios mostrados na Fig. 5.4, a solução GA (1ª solução) oferece a melhor compensação com o tempo de execução, consumindo menos de 2 segundos para resolver o problema para 500 RUs. Isto mostra que além de eficiente, nossa heurística utilizada na primeira geração apresenta indivíduos com alta qualidade.

Por fim, podemos concluir que entre os métodos não exatos, nosso GA apresenta o melhor tempo de execução em boa parte das instâncias do problema. Entretanto, por ser um método de busca e não de aprendizado, nosso GA pode ainda enfrentar problemas de escalabilidade, principalmente em cenários com variação de demanda. Entretanto, nossa GA (1ª solução) mostra que nosso GA é um ótimo ponto de partida para cenários em que é necessário soluções de boa qualidade em tempo eficiente. Desta forma, mesmo que os algoritmos genéticos não possuem propriedades de generalização e aprendizado que os algoritmos de DRL apresenta, o mesmo pode ser utilizado em conjunto com tais técnicas para auxiliar na convergência do agente, ou até mesmo como solução emergente para casos de variações abruptas na rede.

Problema de posicionamento de funções da vRAN considerando variação na demanda

O problema de posicionamento de funções da vRAN pode ser representado considerando-se dois cenários práticos, o primeiro de planejamento e o segundo de operação da rede. O cenário de planejamento considera um estado de demanda em que a rede está saturada, ou seja, todas as unidades de rádio estão demandando 100% de capacidade de comunicação. Essa análise é importante para verificar a eficiência da rede de acesso em seu pior caso, permitindo encontrar pontos de gargalo na rede de transporte, analisar a eficiência do posicionamento das funções da vRAN no pior caso e encontrar os melhores pontos para implantação de nós de processamento na topologia. Em contrapartida, devido à mobilidade dos usuários na rede e a demanda variável das diversas aplicações dos usuários, a demanda em todas as RUs flutua amplamente. Desta forma, no cenário de operação, o posicionamento das funções da vRAN é um problema dinâmico que exige adaptação conforme a demanda em cada unidade de rádio da rede.

Os resultados sobre posicionamento de funções da vRAN apresentados nos capítulos anteriores se referem ao problema que considera o cenário estático de saturação da rede. Desta forma, as soluções de posicionamento apresentadas atendem à operação da rede a qualquer tempo. Entretanto, o posicionamento dinâmico das funções da vRAN considerando flutuação de demanda nas RUs permite encontrar soluções mais eficientes e com melhor custo-benefício a longo prazo. Neste capítulo, iremos abordar o problema de posicionamento dinâmico das funções da vRAN apresentando a formulação de um Processo de Decisão de Markov (*Markov Decision Process* – MDP).

Inicialmente, na Seção 6.1, apresentamos uma breve discussão sobre a base de dados utilizada neste trabalho, que apresenta dados sobre variação de demanda. Na Seção 6.2, apresentamos um gerador de carga sintética para criação de dados de treinamento e teste para modelos que resolvem o problema de posicionamento dinâmico de funções da vRAN. Por fim, na Seção 6.3, apresentamos a formulação MDP para o problema de posicionamento dinâmico de funções da vRAN, analisando suas propriedades e discutindo formas de resolver o problema proposto.

6.1 Base de dados de variação de demanda

Em [Chen et al. 2015], os autores disponibilizam uma base de dados que apresenta leituras de variação de demanda em uma rede de acesso real em diferentes horas do dia. As leituras são realizadas periodicamente, de hora em hora, durante oito dias. Os dados coletados são de uma rede celular real e operacional implantada em uma cidade da China, que está anonimizada nos dados, durante o período de 20 a 26 de agosto de 2012. As coletas apresentam informações precisas sobre data, hora e posicionamento das unidades de rádio, indicadas por latitude e longitude. Cada equipamento de usuário é detectado pelo *hash International Mobile Subscriber Identity* (IMSI), e a área de cobertura da rede celular é de cerca de 50 km × 60 km, incluindo aproximadamente 85% da população total da cidade. A base de dados publicada pelos autores é a que melhor atende às nossas demandas em termos de tipo de informações e quantidade de coletas. Por esse motivo, neste trabalho utilizaremos na implementação do nosso gerador de carga sintético a base de dados apresentada em [Chen et al. 2015].

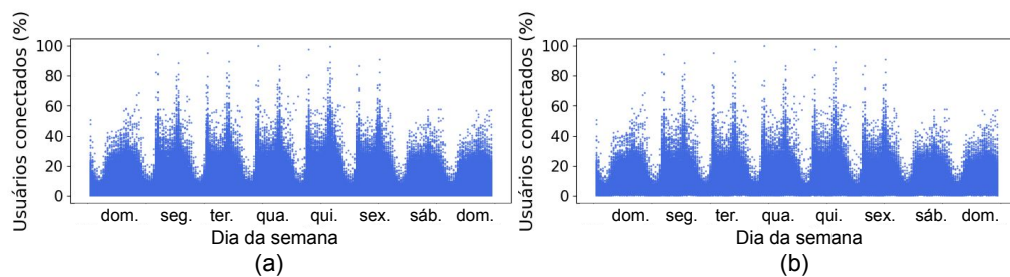


Figura 6.1: Quantidade normalizada de usuários conectados a cada período do dia [Chen et al. 2015]. (a) apresenta os dados originais e (b) apresenta os dados filtrados.

A Figura 6.1(a) apresenta os dados sobre a quantidade de usuários conectados nas RUs em cada dia da semana e sua variação a cada hora. Com isso, para melhor representação dos dados, nós normalizamos os dados tendo como base a maior quantidade de usuários conectados registrado na base de dados. Desta forma, a Figura 6.1(a) ilustra a curva de variação de usuários conectados normalizada, em que cada ponto (x, y) representa a quantidade de usuários conectados a uma RU em uma hora do dia.

A base de dados disponibilizada pelos autores em [Chen et al. 2015] apresenta dados de 13.269 RUs durante oito dias de medições constantes realizadas de hora em hora. Entretanto, algumas RUs da topologia apresentam janelas sem medições, ou seja, leituras incompletas. Enquanto a maioria das RUs apresenta dados de leituras completas durante todos os oito dias, outras RUs apresentam menos de 24 leituras contínuas, ou seja, não completam sequer um dia de dados coletados. A Figura 6.2 apresenta a CDF que relaciona a quantidade de RUs que apresentam leituras contínuas na base de dados. Note que mais de 80% das RUs apresentam pelo menos 24 horas seguidas de medições, enquanto menos de 20% das RUs apresentam dados incompletos.

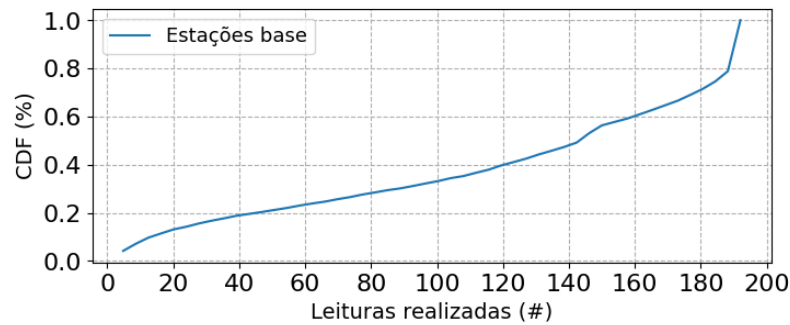


Figura 6.2: CDF da quantidade de RUs e quantidade de leituras na base de dados [Chen et al. 2015].

Desta forma, para adequar os dados das coletas e remover RUs com leituras inconclusivas, realizamos uma filtragem na base de dados, mantendo apenas leituras de RUs com pelo menos um dia completo registrado. A Figura 6.1 apresenta a comparação entre as leituras da base de dados original e a base de dados após a filtragem. Note que os dados filtrados (Figura 6.1(b)) apresentam poucas diferenças em comparação com as leituras originais (Figura 6.1(a)), mostrando que a base de dados filtrada representa fortemente os dados originais. Assim, para a implementação e avaliação do gerador de carga sintética, utilizamos a base de dados filtrada contendo apenas leituras de RUs com pelo menos um dia completo de coletas registradas.

Os dados estão ordenados cronologicamente de domingo a domingo. Na figura, é possível identificar diferenças no padrão de variação da quantidade de usuários conectados em dias de semana (segunda a sexta-feira), em que o pico de usuários conectados é maior, e finais de semana (sábado e domingo), em que o pico de usuários conectados é menor. A diferença na quantidade de usuários conectados entre dias de semana e finais de semana se dá principalmente nas RUs do centro da cidade, o que ilustra o padrão de mobilidade dos usuários que tendem a ir ao centro em menor número durante o final de semana. Nas RUs localizadas em bairros periféricos, a variação da quantidade de usuários conectados é menor do que nas RUs do centro, atingindo pico de demanda no fim da tarde ou início da noite.

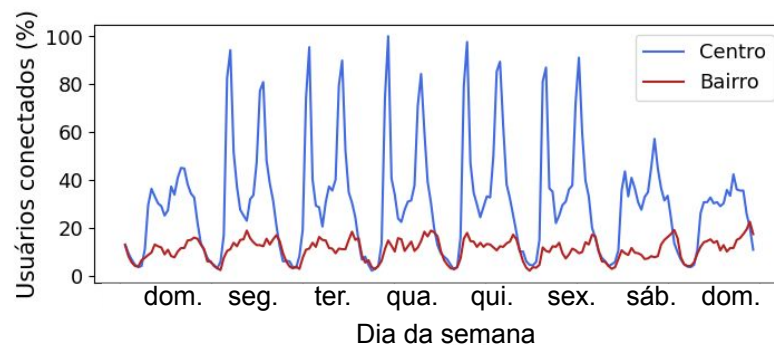


Figura 6.3: Exemplo da quantidade de usuários conectados a uma estação base do centro e outra em um bairro periférico na cidade.

A Figura 6.3 apresenta a comparação da quantidade de usuários conectados em duas RUs localizadas em diferentes pontos da topologia. Em azul é apresentada a variação da quantidade de usuários conectados a uma RU no centro e em vermelho de uma RU em um bairro afastado. Os dados estão ordenados cronologicamente durante os oito dias de leitura (de domingo a domingo). Enquanto a RU no centro apresenta picos de 90% durante o horário comercial nos dias de semana, entre 8:00 e 11:00 e novamente entre 13:00 e 16:00, nos finais de semana o pico de usuários conectados é de no máximo 60%. Por outro lado, na RU localizada no bairro afastado, a quantidade média de usuários conectados é de 10% durante todos os dias da semana, com picos de aproximadamente 20% após as 17:00. Este comportamento ilustra o padrão de mobilidade dos usuários na topologia em que, durante os dias de semana, os usuários se deslocam até o centro da cidade durante o horário comercial e voltam para casa ao fim do expediente. Nos finais de semana as RUs apresentam picos menores e menor variação de usuários conectados, pois os usuários se movem menos na topologia.

6.2 Gerador de carga sintética

Dados sobre variação de demanda na rede de acesso são importantes para a implementação, formulação e avaliação de modelos que resolvem o problema de posicionamento dinâmico de funções da vRAN. Entretanto, encontrar dados reais, atuais e em abundância de forma aberta e livre é uma tarefa difícil, dado que as operadoras de redes móveis não publicam tais informações. Em contrapartida, modelos que resolvem problemas dinâmicos conforme a variação de demanda necessitam de muitos dados para treinamento e teste de desempenho. Neste cenário, diversos trabalhos na literatura [Guo e Haenggi 2013, Francesco et al. 2015, Ghazzai et al. 2016, Malandrino et al. 2018] apresentam formulações e implementações de geradores de carga sintética que produzem dados artificiais fundamentados em base de dados com coletas parciais de redes reais ou coletas de dados simulados, a fim de produzir uma larga quantidade de dados de variação de demanda.

Em [Malandrino et al. 2018], os autores apresentam a formulação de um gerador de carga sintética utilizando o conceito de Cadeia de Markov (MC). Na formulação os autores consideram que o posicionamento das RUs é conhecido, porém, a variação de demanda é desconhecida. Para a representar o problema, os autores definem um conjunto de estados S no qual cada estado é representado como

$$s = (A(b), d, h, \delta), \quad (6-1)$$

sendo $A(b)$ o tipo de área que se encontra a RU $b \in B$ (B representa o conjunto de

todas as RUs), por exemplo, zona rural, zona urbana, centro ou periferia. $d \in \{\text{seg., ter., qua., qui., sex., sáb., dom.}\}$ representa o dia da semana em que o dado foi coletado, $h \in \{0h, 1h, \dots, 23h\}$ representa a hora da leitura, e δ representa o tráfego normalizado daquele período.

Desta forma, a solução baseada em MC seleciona uma demanda δ_i para cada período h_i do dia, seguindo uma função de probabilidade de transição de estados representada como:

$$p = (A(b), d, h_i, h_j, \delta_i, \delta_j), \quad (6-2)$$

que calcula a probabilidade de uma RU na região $A(b)$ no dia d e instante h_i com demanda de tráfego δ_i apresentar na janela de tempo seguinte h_j a demanda de tráfego δ_j . A função de probabilidade de transição, por sua vez, é calculada a partir das leituras de uma base de dados real ou simulada dada como entrada do modelo. Os autores assumem que a base de dados deve apresentar coletas de variação de demanda a cada hora, durante todos os dias da semana, durante pelo menos duas semanas.

A formulação apresentada pelos autores é eficiente, uma vez que o conjunto de estados e a função de probabilidade de transição apresentada conseguem representar múltiplas RUs na mesma região ao mesmo tempo. Entretanto, os autores assumem receber como entrada a informação do tipo de região de cada RU na topologia. Desta forma, considerando um cenário em que tal informação é desconhecida, seria necessário classificar cada região da topologia antes de realizar a geração de dados sintéticos, o que é um problema NP-completo [Rybalov 2020]. Além disso, para calcular a função de transição, os autores demandam uma grande base de dados com pelo menos 14 dias completos de leituras contínuas, pois, para cada dia da semana, é calculada uma função de probabilidade de transição diferente.

Neste trabalho, apresentamos duas formulações para a geração de carga sintética considerando dois cenários diferentes. No primeiro cenário, assumimos que a classificação do tipo de região da topologia é dada como entrada do modelo. No segundo cenário, consideramos que o tipo de região das RUs é desconhecido, avaliando o perfil de tráfego de cada RU de forma independente. Em ambos os modelos, o objetivo é gerar dados sintéticos sobre a quantidade de usuários conectados em cada RU da topologia durante cada período do dia. Para isso, vamos classificar os dias como dias de semana e finais de semana, minimizando assim a quantidade de dados necessários para a execução do modelo.

Cenário com regiões da topologia classificadas: Neste cenário, em que assumimos que o tipo de região de cada RU da topologia é conhecido, representamos o conjunto de estados como S , onde cada estado $s \in S$ pode ser descrito como

$$s = (A(b), DS, h, \delta), \quad (6-3)$$

em que $A(b)$ representa o tipo de região da RU $b \in B$, $DS \in \{\text{dia de semana, final de semana}\}$ representa o tipo do dia e δ representa a quantidade normalizada de usuários conectados na RU no período $h \in \{0h, 1h, \dots, 23h\}$. Em nossa formulação classificamos os dias como dias de semana e finais de semana, devido à uniformidade na variação de usuários conectados que analisamos na Seção 6.1, a fim de maximizar eficiência do gerador de carga sintética. A cada instante h_j , nossa formulação baseada em MC, seleciona uma demanda δ_j para uma RU da topologia seguindo uma função de probabilidade de transição de estados representada como

$$p = (A(b), DS, h_j, \delta_j, h_j, \delta_j), \quad (6-4)$$

que calcula a probabilidade de uma estação base da região $A(b)$ no dia DS com δ_j usuários conectados no período h_j apresentar na janela de tempo seguinte h_j uma demanda δ_j . Dado que neste cenário consideramos a classificação das RUs em regiões e a classificação dos dias da semana, nossa formulação possibilita representar múltiplas RUs e múltiplos dias da semana em uma única função de transição, maximizando a eficiência do gerador de carga sintética.

Cenário em que as regiões da topologia não estão classificadas: Neste cenário, consideramos cada RU na topologia de forma independente, ou seja, não classificamos as RU, apenas os dias da semana. Desta forma, consideramos um conjunto de estado S em que cada estado

$$s = (b, DS, h, \delta), \quad (6-5)$$

representa que a RU $b \in B$ no dia de tipo $DS \in \{\text{dia de semana, final de semana}\}$ e instante $h \in \{0h, 1h, \dots, 23h\}$ apresenta a quantidade δ de usuários conectados. A cada instante h_j , nossa formulação baseada em MC, seleciona uma demanda δ_j para uma RU da topologia seguindo uma função de probabilidade de transição de estados representada como

$$p = (b, DS, h_j, \delta_j, h_j, \delta_j), \quad (6-6)$$

que representa a probabilidade de transição do estado $s_i = (b, DS, h_j, \delta_j)$ para o estado $s_j = (b, DS, h_j, \delta_j)$. Note que dado que as RUs não estão classificadas, neste cenário consideramos que cada RU possui duas funções de probabilidade de transição própria, uma para cada tipo de dia da semana.

Na Figura 6.4 apresentamos os dados sintéticos gerados por cada uma das formulações apresentadas. A Figura 6.4(a) apresenta os dados gerados considerando a

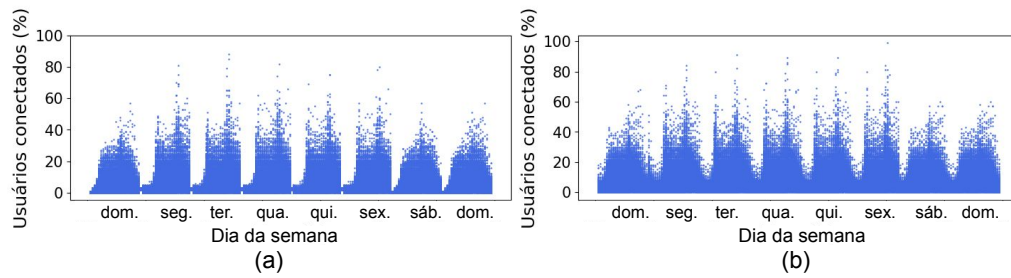


Figura 6.4: Comparando geradores de carga sintética.

informação dos tipos de região da topologia e a Figura 6.4(b) apresenta os dados gerados quando esta informação é desconhecida. Note que, em ambos cenários, os dados sintéticos mantiveram o comportamento da base de dados original em que os picos de demanda nos dias de semana e finais de semana são diferentes. Entretanto, devido às diferenças de cada formulação, o resultado de cada geração possui características diferentes.

Enquanto a formulação que considera as RUs independentes (Figura 6.4(b)) apresenta uma geração de carga sintética melhor distribuída e mais próxima dos dados reais, o modelo que considera o tipo de região das RUs (Figura 6.4(a)) apresenta menor variedade de curvas. Isso ocorre porque, no cenário com os tipos de região, existe uma menor quantidade de funções de probabilidade de transição a serem calculadas e uma menor quantidade de estados, o que consequentemente impacta em sua generalização. Por outro lado, a formulação que considera os tipos de região é mais eficiente. Para testar os geradores de carga sintética, consideramos uma topologia com 13 mil RUs e geramos dados para 8 dias da semana (domingo a domingo) de hora em hora, resultando em 192 ações por RU. Neste cenário, a geração de dados da Figura 6.4(a) levou em média 6 minutos, enquanto a geração de dados da Figura 6.4(b) levou em média 12 minutos devido ao seu maior número de estados e, consequentemente, de funções de probabilidade de transição.

6.3 Posicionamento dinâmico de funções da vRAN

Neste trabalho, apresentamos duas formulações MDP para o problema de posicionamento dinâmico de funções da vRAN considerando dois cenários práticos diferentes. No primeiro cenário, assumimos que a operadora de redes móveis possui seu próprio conjunto de unidades centralizadas (CUs) e unidades distribuídas (DUs) espalhadas pela rede, e no segundo cenário consideramos a terceirização da carga de processamento das CUs e DUs para uma provedora de computação em nuvem (por exemplo, Amazon Web Services, Google Cloud e Microsoft Azure), responsável por gerenciar e disponibilizar os recursos de processamento em forma de máquinas virtuais. Em ambos cenários, o conjunto de ações considerados na formulação MDP é o mesmo, porém, cada cenário possui

sua própria definição de conjunto de estados e funções de recompensa diferentes.

6.3.1 Formulação considerando recursos de processamento gerenciados pela operadora de redes móveis

Nesta seção, apresentamos a formulação MDP do problema de posicionamento de funções da vRAN considerando que a operadora de redes móveis gerencia seu próprio conjunto de CUs e DUs espalhados na rede. Desta forma, para formular o conjunto de ações consideramos o vetor $\mathbf{R}^t = [R_{b_1}^t, \dots, R_{b_n}^t]$, em que $R_{b_i}^t$ indica o caminho utilizado para roteamento do fluxo de *midhaul* da RU $b_i \in \mathcal{B}$ no instante t . O vetor $\mathbf{S}^t = [S_{b_1}^t, \dots, S_{b_n}^t]$, em que $S_{b_i}^t$ indica a divisão funcional escolhida para a RU $b_i \in \mathcal{B}$ no instante t . O vetor $\mathbf{P}^t = [P_{b_1}^{t,vCU}, \dots, P_{b_n}^{t,vCU}]$, em que $P_{b_i}^{t,vCU}$ indica o posicionamento da vCU da RU $b_i \in \mathcal{B}$ no instante t . Por fim, o vetor $\mathbf{I}^t = [I_{b_1}^{t,vCU}, \dots, I_{b_n}^{t,vCU}]$, em que $I_{b_i}^{t,vCU}$ indica a quantidade de recurso alocado para a vCU da RU $b_i \in \mathcal{B}$ no instante t .

Desta forma, podemos formular o conjunto de ações do MDP como

$$a^t = \{\mathbf{R}^t, \mathbf{S}^t, \mathbf{P}^t, \mathbf{I}^t\}, \quad (6-7)$$

que indica a rota utilizada, a divisão funcional escolhida, o posicionamento da vCU e o recurso alocado na vCU para cada RU na topologia no instante de tempo t . Neste sentido, cada ação tomada pelo agente representa uma solução completa de posicionamento de funções da vRAN que pode ser viável ou inviável. Uma ação viável representa uma solução de posicionamento que não quebra nenhuma restrição do problema, ou seja, as rotas escolhidas e a carga de processamento nas CUs e DUs não excedem sua capacidade. Por outro lado, uma ação inviável representa uma solução que quebra alguma dessas restrições. Desta forma, para permitir ao agente conhecimento da capacidade de processamento e transporte disponível na rede, o posicionamento das funções e a demanda atual das RUs, consideramos o conjunto de estados representados como

$$s^t = \{\mathbf{D}^t, \mathbf{P}^t, \mathbf{C}^t, \mathbf{S}^t, \mathbf{E}^t\}, \quad (6-8)$$

em que $\mathbf{D}^t = [D_{b_1}^t, \dots, D_{b_n}^t]$ é o vetor que representa a demanda das RUs no instante t , $\mathbf{C}^t = [C_C^t U_1, \dots, C_C^t U_n]$ é o vetor que representa a capacidade de processamento disponível nas CUs no instante t e $\mathbf{E}^t = [E_{\rho_1}^t, \dots, E_{\rho_n}^t]$ é o vetor que representa a quantidade de recurso disponível em cada rota da topologia. Representando assim, a demanda atual de cada RU, a quantidade de recurso de processamento e transporte disponível na rede e a posição das funções de cada RU no instante de tempo t , fazendo com que o agente tenha informações para analisar o impacto de transição entre um estado e outro.

Para analisar as ações tomadas pelo agente, é necessário avaliar a transição entre os estados, ou seja, o estado de posicionamento das funções atual e o estado do posicionamento futuro definido pela ação tomada. Para isso, consideramos a função de recompensa do agente como o custo total da solução de posicionamento calculado a partir de quatro tipos de custo: custo de instanciação $\gamma_{b_i}^{Inst}$ das funções da RU $b_i \in \mathcal{B}$, custo de processamento $\gamma_{b_i}^{Proc}$ das funções da RU $b_i \in \mathcal{B}$, o custo de roteamento $\gamma_{b_i}^{Rout}$ dos fluxos da RU $b_i \in \mathcal{B}$ e custo de migração $\gamma_{b_i}^{Migr}$ das funções da RU $b_i \in \mathcal{B}$. Desta forma, podemos representar a função de recompensa para cada instante de tempo t como

$$rw^t = \sum_{b_i \in \mathcal{B}} (\gamma_{b_i}^{Inst} + \gamma_{b_i}^{Proc} + \gamma_{b_i}^{Rout} + \gamma_{b_i}^{Migr}), \quad (6-9)$$

que avalia a transição entre a solução de posicionamento anterior e a solução de posicionamento atual. Note que neste cenário é possível que o agente escolha não modificar o posicionamento das funções, ou seja, manter o posicionamento atual. Com isso, nossa formulação permite o agente avaliar a qualidade do reposicionamento das funções para atender uma nova demanda, mas também avaliar quando é de fato necessário modificar o posicionamento das funções, permitindo uma análise de custo a curto e longo prazo.

Devido à complexidade da formulação apresentada, analisada na Seção 6.3.3, neste trabalho não apresentamos a implementação do modelo proposto. Desta forma, a implementação da formulação MDP, a forma de resolução e a avaliação do agente desta formulação são listados como trabalhos futuros no Capítulo 7. A seguir, apresentamos a formulação MDP considerando a terceirização da carga de processamento para provedoras de computação em nuvem.

6.3.2 Formulação considerando recursos de processamento terceirizados

Nesta seção, apresentamos a formulação MDP para o problema de posicionamento da vRAN considerando que a carga de processamento das CUs e DUs é terceirizada para um provedor de computação em nuvem. Com isso, consideramos que os nós de processamento (*hosts*) e as máquinas virtuais (VMs) contratadas estarão sempre disponíveis para uso. Além disso, consideramos que o provedor de computação em nuvem oferece serviços do tipo comprimento de onda (*wavelength services*), em que enlaces de roteamento dedicados são contratados pela operadora para uso exclusivo.

Nesta formulação, o conjunto de ações considerado para tomada de decisão do agente é o mesmo apresentado na Equação (6-7). Desta forma, consideramos que a provedora de computação em nuvem oferece à operadora de redes móveis um conjunto de rotas para cada RU, um conjunto de *hosts* para posicionamento das funções e um conjunto

de tipos de VMs, onde cada tipo possui uma quantidade de recursos diferente. Com isso, visando minimizar a complexidade da formulação MDP, consideramos os estados como

$$s^t = \{\mathbf{D}^t\}, \quad (6-10)$$

ou seja, contendo apenas informações sobre a demanda das RUs no instante t . Isso é possível graças a terceirização dos nós de processamento, que transfere para a provedora de computação em nuvem a tarefa de gerenciamento dos recursos. Além disso, a operadora de redes móveis não contabiliza o custo total da solução considerando os quatro tipos descritos na Equação (6-9). Neste cenário, o custo total da solução é representado pelo somatório dos custos das VMs utilizadas e contratadas pela operadora. Desta forma, a função de recompensa no segundo cenário é descrita como

$$rw^t = \mathbf{M} - \sum_{b \in \mathcal{B}} Cost(I_{b_i}^{t,vCU} + I_{b_i}^{t,vDU}), \quad (6-11)$$

em que $I_{b_i}^{t,vCU}$ e $I_{b_i}^{t,vDU}$ determinam, respectivamente, o tipo de VM contratada para as vCUs e vDUs de cada RU $b_i \in \mathcal{B}$, e \mathbf{M} é um fator de correção da solução definido como

$$\mathbf{M} = \begin{cases} 0, & \text{se ação selecionada é viável} \\ \max Cost(I_{b_i}^{t,vCU} + I_{b_i}^{t,vDU}), & \text{caso contrário.} \end{cases} \quad (6-12)$$

em que $\max Cost(I_{b_i}^{t,vCU} + I_{b_i}^{t,vDU})$ representa o custo da solução viável mais cara, ou seja, o custo da pior solução viável.

O fator de correção \mathbf{M} é calculado a partir dos dados de entrada do problema e é utilizado para que o agente MDP receba uma penalização (número positivo) quando selecionar uma ação inviável e uma recompensa (número negativo) baseada no custo da ação tomada. Dado que o agente MDP maximiza a função de recompensa de longo prazo, em nossa formulação o agente minimizará o custo da solução representado como a parte negativa na equação (6-11) e, devido o fator de correção \mathbf{M} , evitará cometer erros encontrando soluções inviáveis.

6.3.3 Análise das formulações apresentadas

As formulações MDP apresentadas consideram dois cenários práticos distintos de recursos computacionais. Para representar os diferentes problemas consideramos em cada formulação um conjunto próprio de estados e uma função de recompensa específica. Entretanto, ambas formulações consideram o mesmo conjunto de ações, onde cada ação representa uma solução completa de posicionamento das funções da vRAN. Por um lado,

o tipo de ação formulada permite representar todas as decisões sobre o posicionamento de funções da vRAN em uma única estrutura. Por outro lado, esta capacidade de representação impacta fortemente o tamanho do espaço de ações. Considerando a formulação das ações apresentada na Equação (6-7) podemos calcular o tamanho do espaço de ações como

$$\Theta(|\mathcal{C}||\mathcal{S}||\mathcal{F}||\mathcal{P}|^{|\mathcal{B}|}), \quad (6-13)$$

em que $|\mathcal{C}|$ representa a quantidade de CUs na topologia, $|\mathcal{S}|$ representa a quantidade de opções de divisões funcionais consideradas, $|\mathcal{F}|$ considera o tipo de *flavors* ou VMs disponíveis, $|\mathcal{P}|$ representa a quantidade de rotas para roteamento na topologia e $|\mathcal{B}|$ representa a quantidade de RUs na topologia. Note que o espaço de ações da formulação apresentada é exponencial, ou seja, em um cenário considerando 2 CUs, 4 divisões funcionais, 4 tipos de *flavors* ou VMs, 2 caminhos por RU e 10 RUs, o conjunto de ações apresenta $1,15 \times 10^{18}$ ações possíveis.

Enquanto o tamanho do espaço de ações é exponencial e apresenta uma complexidade alta, as demais propriedades das formulações MDP apresentadas são de complexidade polinomial. O tamanho do espaço de observação é $\Theta(|\mathcal{B}| + |\mathcal{P}|)$ no primeiro cenário e $\Theta(|\mathcal{B}|)$ no segundo cenário, e a complexidade do cálculo da função de recompensa é $\Theta(\mathcal{B}^2)$ no primeiro cenário e $\Theta(\mathcal{B})$ no segundo cenário. Desta forma, para resolver as formulações MDP propostas, indicamos o uso de técnicas de aprendizado de máquinas, mais especificamente de aprendizado não supervisionado.

Técnicas de aprendizado supervisionado demandam dados sobre soluções de posicionamento de funções da vRAN de boa qualidade, em diferentes topologias com diferente quantidade de RUs e recursos computacionais, o que é um problema NP-Completo. Desta forma, como a construção de tais bases de dados é inviável, o uso de aprendizado não supervisionado, por exemplo, aprendizado por reforço profundo, pode minimizar o impacto da complexidade do problema e possibilitar o agente a encontrar soluções de boa qualidade, em tempo hábil e sem a necessidade de base de dados de soluções para o treinamento. Entretanto, devido ao tamanho exponencial do espaço de ações, mesmo utilizando um agente de aprendizado por reforço profundo, o treinamento pode demorar muito tempo para convergir, sendo necessário o uso de técnicas de redução do espaço de ações do MDP formulado.

Para verificar se o método indicado para resolução das formulações MDP é eficiente, neste trabalho, implementamos a formulação do segundo cenário, ou seja, o cenário que considera a terceirização da carga de processamento, via um agente por aprendizado por reforço profundo utilizando o método de treinamento *actor critic*. Para implementação, utilizamos o arcabouço Stable Baselines 3 na versão 2.1.0a4 e Python versão 3.8.10, em um computador equipado com um Intel(R) Core(TM) i7-10700F com 16 núcleos, 32 GB de memória RAM e 1 TB de SSD.

Para avaliação do agente, consideramos duas topologias vRAN com 4 e 8 RUs, em que, para cada RU, a provedora de computação em nuvem oferece 3 tipos de VMs com 4, 8 e 16 vCPUs e 8, 16 e 32 *Gigabytes* de memória. Para transporte, assumimos que a operadora disponibiliza duas rotas por RU, uma com serviço de comprimento de onda e 250 microssegundos de atraso máximo, e outra rota dedicada com atraso de 1 milissegundo. Para os preços das VMs consideramos as informações disponibilizadas no catálogo da Amazon Web Services, onde cada VM com serviço de comprimento de onda custa, respectivamente, 100, 193 e 526 dólares/mês e sem o serviço de comprimento de onda, 67, 117 e 380 dólares/mês.

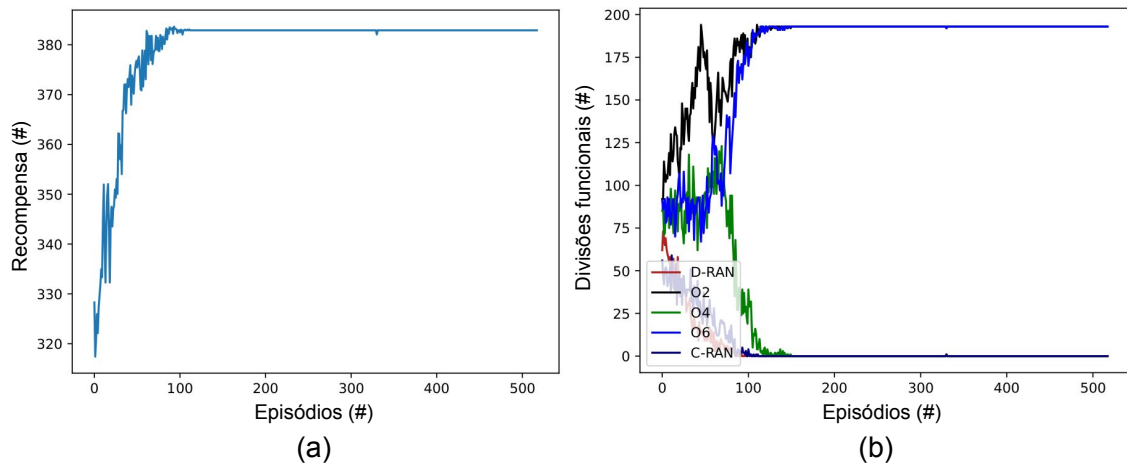


Figura 6.5: Treinamento para topologia com 4 RUs.

A Figura 6.5 apresenta os resultados do treinamento do agente de aprendizado por reforço profundo para a topologia com 4 RUs. A Figura 6.5(a) apresenta a média de recompensa dos episódios durante o treinamento. Neste cenário, utilizamos um conjunto de dados de variação de demanda sintéticos contendo oito dias seguidos de medições. Em cada episódio o agente seleciona 24 ações, cada ação representa a solução de posicionamento das funções para uma determinada hora do dia. A Figura 6.5(b) apresenta a quantidade escolhida de cada divisão funcional durante um episódio. Note que, neste cenário, o agente apresenta comportamento exploratório nos primeiros 100 episódios, encontrando soluções com alto custo e apresentando diversas escolhas de divisões funcionais. Entretanto, a partir de 100 episódios o agente começa a apresentar convergência, priorizando divisões funcionais de melhor custo-benefício. Neste caso, o agente prioriza a escolha das divisões funcionais O2 e O6 que correspondem aos VNCs 2 e 6 ilustrados na Figura 4.1.

Por outro lado, no cenário com 8 RUs, o agente apresenta um comportamento ruim durante o treinamento, em que não existe a convergência para nenhum ponto no espaço de soluções. A Figura 6.6 apresenta os dados de treinamento durante este cenário, a Figura 6.6(a) apresenta a média de recompensa dos episódios e a Figura 6.6(b) apresenta as escolhas de divisões funcionais de cada episódio durante o treinamento. Note que

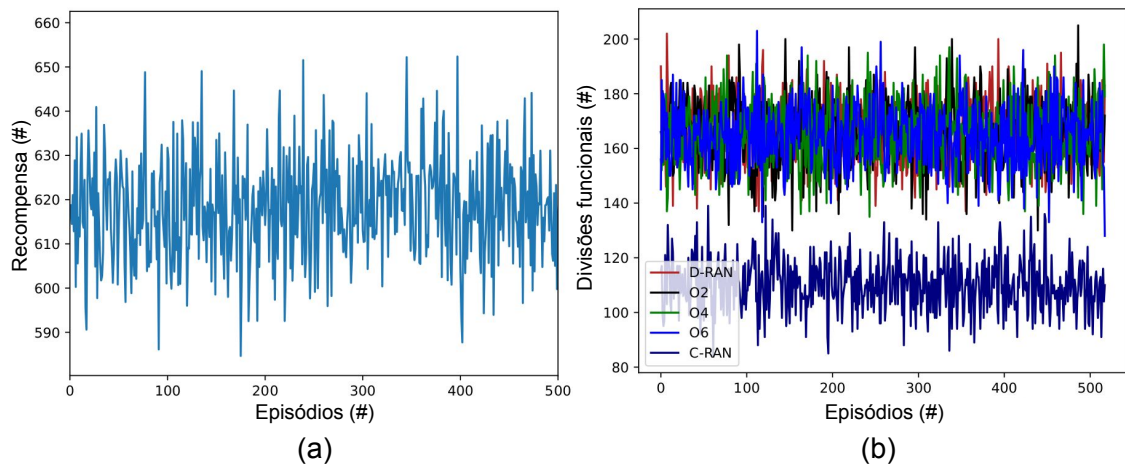


Figura 6.6: Treinamento para topologia com 8 RUs.

em nenhum momento o agente apresenta convergência, apresentando grande variação nas escolhas de divisões funcionais e, conseqüentemente, no custo da solução. Este comportamento ocorre devido à complexidade desta instância, em que o espaço de ações é muito grande, com $2,8 \times 10^{14}$ ações possíveis. Este espaço de ações demanda muito tempo de treinamento até que o agente apresente, se possível, algum grau de convergência. Este resultado exemplifica a necessidade de uma estratégia de redução do espaço de ações no ambiente de aprendizado proposto. Esta tarefa e a implementação e avaliação do agente em topologias diferentes estão listadas como trabalhos futuros no capítulo a seguir.

Conclusão, publicações e trabalhos futuros

Conforme descrito anteriormente, o objetivo deste trabalho foi formular, resolver e avaliar soluções para o problema de posicionamento de funções na rede de acesso por rádio. Neste documento, inicialmente, apresentamos um modelo de otimização linear inteira que resolve o problema de forma genérica e flexível, ou seja, sem limitação no posicionamento das funções virtualizadas em nós específicos e fixos na rede. Além disso, nossa formulação possibilita a divisão de carga em múltiplos fluxos no roteamento do tráfego no *crosshaul*. Isto faz com que nossa formulação apresente maiores ganhos em relação ao modelo original (PlaceRAN [Morais et al. 2022]) em termos de centralização e recursos computacionais utilizados, ou seja, redução de custo da solução. Além disso, avaliamos o parâmetro de granularidade dos fluxos a serem divididos, mostrando que nosso modelo pode ser adaptado a cenários com baixa taxa de tráfego, mas também a cenários futuros com maior demanda.

Entretanto, nossa formulação exata possui baixa escalabilidade devido à complexidade computacional do problema, que pertence à classe dos problemas NP-Completo. Desta forma, visando resolver o problema em cenários com maior quantidade de nós e mais próximos a cenários reais, neste trabalho, também apresentamos duas formulações não exatas para resolver o problema de posicionamento de funções da vRAN. A primeira formulação é a implementação de um agente de aprendizado por reforço profundo que resolve o problema de posicionamento de funções da vRAN. Para implementar o agente, utilizamos a formulação utilizada em PlaceRAN a fim de obter comparabilidade de soluções em cenários menores. Na avaliação do agente, podemos observar que ele apresenta um bom comportamento em resolver o problema, encontrando soluções que apresentam alto nível de centralização em topologias de diferentes tamanhos. Assim, podemos observar que o nosso agente consegue aprender o problema e apresentar soluções de alta qualidade, superando até mesmo outro modelo exato da literatura. Por outro lado, as soluções encontradas pelo agente, principalmente em cenários com maior quantidade de nós, ficam distantes do nível de centralização ótimo, o que nos indica um importante ponto de melhoria para nossa formulação. Além disso, outro resultado importante foi o tempo de convergência do agente, que apresenta uma média de convergência a partir de 2.000

episódios no treinamento.

A segunda formulação não exata apresentada deste trabalho foi um algoritmo genético que resolve o problema de posicionamento de funções da vRAN. Em sua formulação, utilizamos técnicas avançadas de computação evolutiva a fim de melhorar a busca heurística realizada pelo algoritmo. Dentre essas técnicas, é importante destacar o uso de uma função de reparo adaptativa que utiliza uma estratégia híbrida combinando o reparo aleatório de um indivíduo inviável com a técnica de substituição de indivíduo, a presença de elitismo entre as gerações para exercer pressão na seleção e operadores de variação adaptativos, ou seja, utilizando aleatoriedade e substituição. Além disso, outro ponto interessante da nossa formulação foi a utilização de uma heurística de geração que insere na primeira população do algoritmo um indivíduo heurísticamente criado a partir de conhecimentos específicos do problema, como o nível de centralização das divisões funcionais. Essa heurística possibilitou a nosso algoritmo genético encontrar uma solução de qualidade próxima ao ótimo em poucos segundos de busca, independentemente do tamanho da topologia. Além disso, este indivíduo criado a partir da heurística apresenta um alto nível de sobrevivência por apresentar uma qualidade próxima à ótima em todos os cenários avaliados. Entretanto, em diversas topologias, o algoritmo genético conseguiu melhorar o indivíduo criado heurísticamente e, assim, encontrar soluções melhores que a inicialmente encontrada. É importante ressaltar que o algoritmo genético apresentado conseguiu encontrar soluções em diferentes cenários utilizando o menor tempo entre as formulações avaliadas, sendo mais rápido que o modelo de otimização exato e agente de aprendizado por reforço profundo.

Também apresentamos duas formulações para o problema de posicionamento de funções da vRAN considerando o estado de operação da rede, em que as unidades de rádio apresentam demanda dinâmica no tempo. Ambas formulações consideram o todos os aspectos do problema de posicionamento de funções, incluindo as características dos nós de processamento e as restrições do posicionamento. Avaliamos as formulações MDP propostas indicando a complexidade para resolução e implementação dos modelos, analisando o tamanho do espaço de ações e das demais características do ambiente. Por fim, apresentamos resultados de simulações para exercitar e ilustrar a complexidade do problema considerando topologias diferentes de redes de acesso virtualizadas. Nos resultados conseguimos observar que na topologia menor o agente consegue encontrar soluções que utilizam divisões funcionais de melhor custo benefício. Já na topologia maior, devido ao tamanho do espaço de ações, mostramos que o agente apresenta desempenho pior, não apresentando convergência para um ponto do espaço de soluções. Este comportamento mostra a necessidade de explorar e combinar alternativas para redução do espaço de ações do agente, a fim de otimizar o treinamento e manter soluções de boa qualidade disponíveis.

A partir das investigações realizadas no contexto desta dissertação foram publicados os seguintes trabalhos:

- G. M. Almeida, L. Pinto, C. B. Both and K. V. Cardoso, “Optimal Joint Functional Split and Network Function Placement in Virtualized RAN With Splittable Flows” in *IEEE Wireless Communications Letters*, vol. 11, no. 8, pp. 1684-1688, Aug. 2022, doi: 10.1109/LWC.2022.3172878.
- G. M. Almeida, V. H. Lopes, A. Klautau and K. V. Cardoso, “Deep reinforcement learning for joint functional split and network function placement in vRAN,” *GLOBECOM 2022 - 2022 IEEE Global Communications Conference*, Rio de Janeiro, Brazil, 2022, pp. 1229-1234, doi: 10.1109/GLOBECOM48099.2022.10001282.
- G. M. Almeida, C. Camilo-Junior, S. Correa, K. V. Cardoso, “A genetic algorithm for efficiently solving the virtualized radio access network placement problem,” in *ICC 2023 - IEEE International Conference on Communications*, Rome, Italy, 2023.
- G. M. Almeida, G. Z. Bruno, Huff A., Hiltunen M., Duarte Jr E. P., Both C. B., and K. V. Cardoso, “RIC-O: Efficient placement of a disaggregated and distributed RAN Intelligent Controller with dynamic clustering of radio nodes,” in arXiv preprint arXiv:2301.02760.

Os três primeiros trabalhos tratam sobre o problema de posicionamento de funções da vRAN, enquanto o quarto trabalho aborda um problema de alocação de recursos e posicionamento de funções virtualizadas na rede de acesso considerando a virtualização das funções de um novo elemento na rede de acesso, proposto pela O-RAN [IEEE 2021], o *Near-Real Time RAN Intelligent Controller* (Near-RT RIC). O Near-RT RIC é um controlador inteligente da rede de acesso cujo objetivo é otimizar o gerenciamento da vRAN através de aplicações chamadas xApps, as quais tomam decisões sobre admissão de usuários, *traffic steering*, alocação de blocos de recursos, dentre outras. Neste trabalho, propomos a desagregação das funções do Near-RT RIC permitindo seu posicionamento distribuído na rede, formulando e resolvendo este problema de posicionamento baseado no conhecimento prévio sobre alocação de recursos na rede de acesso dos trabalhos anteriores. Esse último artigo foi aceito para publicação no *IEEE Journal on Selected Areas in Communications (JSAC)*.

Além disso, o presente estudante também vem contribuindo em diversos outros trabalhos durante o seu mestrado, aparecendo como co-autor nas seguintes publicações:

- F. Z. Morais et al., “PlaceRAN: optimal placement of virtualized network functions in Beyond 5G radio access networks,” in *IEEE Transactions on Mobile Computing*, doi: 10.1109/TMC.2022.3171525.

- F. Z. Morais et al., “OPlaceRAN -a Placement Orchestrator for Virtualized Next-Generation of Radio Access Network,” in IEEE Transactions on Network and Service Management, 2022, doi: 10.1109/TNSM.2022.3222298.
- V. H. L. Lopes, G. M. Almeida, A. Klautau and K. Cardoso, “A Coverage-Aware VNF Placement and Resource Allocation Approach for Disaggregated vRANs,” GLOBECOM 2022 - 2022 IEEE Global Communications Conference, Rio de Janeiro, Brazil, 2022, pp. 185-190, doi: 10.1109/GLOBECOM48099.2022.10000776.
- L. S. Fraga, G. M. Almeida, S. Correa, C. Both, L. Pinto and K. Cardoso, “Efficient allocation of disaggregated RAN functions and Multi-access Edge Computing services,” GLOBECOM 2022 - 2022 IEEE Global Communications Conference, Rio de Janeiro, Brazil, 2022, pp. 191-196, doi: 10.1109/GLOBECOM48099.2022.10001347.
- W. Pires, G. de Almeida, S. Correa, C. Both, L. Pinto and K. Cardoso, “Bi-objective Optimization for Energy Efficiency and Centralization Level in Virtualized RAN,” ICC 2022 - IEEE International Conference on Communications, Seoul, Korea, Republic of, 2022, pp. 1034-1039, doi: 10.1109/ICC45855.2022.9838898.

Com o intuito de contribuir com a comunidade científica da área de redes no sentido de melhorar a reprodutibilidade dos trabalhos acadêmicos, todos os artigos publicados contam com repositório público de software e dados. A Tabela 7.1 lista os repositórios e a quais artigos estão associados.

Publicação	Endereço
[Almeida et al. 2022]	github.com/LABORA-INF-UFG/paper-GLCK-2021
[Almeida et al. 2022]	github.com/LABORA-INF-UFG/paper-GVAK-2022
[Almeida et al. 2023]	github.com/LABORA-INF-UFG/paper-GCSK-2022
[Almeida et al. 2023]	github.com/LABORA-INF-UFG/paper-GGAMECK-2023
[Morais et al. 2022]	github.com/LABORA-INF-UFG/paper-FGLKLRC-2021
[Morais et al. 2022]	github.com/my5G/OPlaceRAN
[Lopes et al. 2022]	github.com/LABORA-INF-UFG/paper-VGAK
[Fraga et al. 2022]	github.com/LABORA-INF-UFG/paper-LGSCLK-2022
[Pires et al. 2022]	github.com/LABORA-INF-UFG/EEPRAN_ICC_2022

Tabela 7.1: Trabalhos com implementação publicamente disponível.

Por fim, sugerimos pontos que julgamos interessantes para trabalhos futuros no contexto de problemas de alocação de recursos na vRAN. Um problema conhecido sobre formulações MDP de problemas dinâmicos, discutido no Capítulo 6, é o tamanho do espaço de ações. Desta forma, a aplicação de técnicas para redução do tamanho do espaço de ações é relevante, pois impacta diretamente na qualidade das soluções encontradas e no desempenho do treinamento do agente. Outro ponto importante para evolução da formulação apresentada no Capítulo 6 é a implementação e avaliação do agente proposto

em topologias diferentes, com configurações de nós de processamento diferentes e enlaces heterogêneos. Além disso, problemas de decisão que levam em consideração aspectos dinâmicos nem sempre podem ser formulados através equações fechadas, por exemplo, qualidade de sinal dos usuários, interferência entre dispositivos e mobilidade dos usuários. Tais aspectos impactam diretamente na qualidade do serviço entregue aos usuários, e conseqüentemente, na alocação de recursos na vRAN. Nesse contexto, listamos duas questões para trabalhos futuros. A primeira é a utilização de mecanismos de predição sobre aspectos da rede, por exemplo, demanda das RUs, atraso nas rotas e mobilidade dos usuários, a fim de utilizar a previsão para determinar a alocação de recursos e evitar interrupção de serviços ou não atendimento de requisitos. A segunda é a utilização de conceitos de otimização estocástica, dado que há incerteza no comportamento da rede. A partir da formulação estocástica é possível realizar decisões baseadas no padrão de comportamento estatístico da rede, ou seja, sem ter a informação exata a priori.

Referências Bibliográficas

- [3GPP 2017]3GPP. *Study on New Radio Access Technology; Radio Access Architecture and Interfaces (Release 14)*. [S.l.], 2017.
- [3GPP 2018]3GPP. *System Architecture for the 5G (Release 15)*. [S.l.], 2018.
- [Akbar et al. 2006]AKBAR, M. M. et al. Solving the multidimensional multiple-choice knapsack problem by constructing convex hulls. *Computers & operations research*, Elsevier, v. 33, n. 5, p. 1259–1273, 2006.
- [Alba e Kellerer 2022]ALBA, A. M.; KELLERER, W. Dynamic functional split adaptation in next-generation radio access networks. *IEEE Transactions on Network and Service Management*, v. 19, n. 3, p. 3239–3263, 2022.
- [Almeida et al. 2023]ALMEIDA, G. et al. A genetic algorithm for efficiently solving the virtualized radio access network placement problem. In: *ICC 2023 - IEEE International Conference on Communications*. [S.l.: s.n.], 2023. p. 1–1.
- [Almeida et al. 2023]ALMEIDA, G. M. et al. *RIC-O: Efficient placement of a disaggregated and distributed RAN Intelligent Controller with dynamic clustering of radio nodes*. 2023.
- [Almeida et al. 2022]ALMEIDA, G. M. et al. Deep reinforcement learning for joint functional split and network function placement in vran. In: *GLOBECOM 2022 - 2022 IEEE Global Communications Conference*. [S.l.: s.n.], 2022. p. 1229–1234.
- [Almeida et al. 2022]ALMEIDA, G. M. et al. Optimal joint functional split and network function placement in virtualized ran with splittable flows. *IEEE Wireless Communications Letters*, v. 11, n. 8, p. 1684–1688, 2022.
- [Bhandari, Murthy e Pal 1996]BHANDARI, D.; MURTHY, C.; PAL, S. K. Genetic algorithm with elitist model and its convergence. *International journal of pattern recognition and artificial intelligence*, World Scientific, v. 10, n. 06, p. 731–747, 1996.
- [Chen et al. 2015]CHEN, X. et al. Analyzing and modeling spatio-temporal dependence of cellular traffic at city scale. In: *2015 IEEE International Conference on Communications (ICC)*. [S.l.: s.n.], 2015. p. 3585–3591.

- [Chu e Beasley 1998]CHU, P.; BEASLEY, J. E. A genetic algorithm for the multidimensional knapsack problem. *Journal of heuristics*, Springer, p. 63–86, 1998.
- [ETSI 2019]ETSI. *ETSI TS 138 401 V15.5.0 (2019-05) – 5G; NG-RAN; Architecture description (3GPP TS 38.401 version 15.5.0 Release 15)*. 2019.
- [Fonseca et al. 2019]FONSECA et al. Optimizing allocation and positioning in a disaggregated radio access network aware of paths through the core infrastructure. In: *Anais do XXXVII Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos*. [S.l.: s.n.], 2019. p. 791–804.
- [Fraga et al. 2022]FRAGA, L. d. S. et al. Efficient allocation of disaggregated ran functions and multi-access edge computing services. In: *GLOBECOM 2022 - 2022 IEEE Global Communications Conference*. [S.l.: s.n.], 2022. p. 191–196.
- [Francesco et al. 2015]FRANCESCO, P. D. et al. A Sharing- and Competition-Aware Framework for Cellular Network Evolution Planning. *IEEE Transactions on Cognitive Communications and Networking*, v. 1, n. 2, p. 230–243, 2015.
- [Garcia-Saavedra et al. 2018]GARCIA-SAAVEDRA et al. FluidRAN: Optimized vRAN/MEC Orchestration. In: *IEEE INFOCOM 2018 - IEEE Conference on Computer Communications*. [S.l.: s.n.], 2018. p. 2366–2374.
- [Garcia-Saavedra et al. 2018]GARCIA-SAAVEDRA, A. et al. WizHaul: On the Centralization Degree of Cloud RAN Next Generation Fronthaul. *IEEE Transactions on Mobile Computing*, v. 17, n. 10, p. 2452–2466, 2018.
- [Ghazzai et al. 2016]GHAZZAI, H. et al. Optimized lte cell planning with varying spatial and temporal user densities. *IEEE Transactions on Vehicular Technology*, v. 65, n. 3, p. 1575–1589, 2016.
- [Guo e Haenggi 2013]GUO, A.; HAENGGI, M. Spatial stochastic models and metrics for the structure of base stations in cellular networks. *IEEE Transactions on Wireless Communications*, v. 12, n. 11, p. 5800–5812, 2013.
- [Gupta et al. 2022]GUPTA, H. et al. *Traffic-Aware Dynamic Functional Split for 5G Cloud Radio Access Networks*. 2022.
- [Hubbs et al. 2020]HUBBS, C. D. et al. Or-gym: A reinforcement learning library for operations research problems. *arXiv preprint arXiv:2008.06319*, 2020.
- [IEEE 2021]IEEE. *O-RAN ALLIANCE Introduces 48 New Specifications Released Since July 2021*. 2021. Disponível em: <<https://www.o-ran.org/blog/o-ran-alliance-introduces-48-new-specifications-released-since-july-2021>>.

- [ITU-T 2018]ITU-T. *Transport network support of IMT 2020/5G*. 2018.
- [Joda et al. 2022]JODA, R. et al. Deep reinforcement learning-based joint user association and cu–du placement in o-ran. *IEEE Transactions on Network and Service Management*, v. 19, n. 4, p. 4097–4110, 2022.
- [Laghrissi e Taleb 2018]LAGHRISSI, A.; TALEB, T. A survey on the placement of virtual resources and virtual network functions. *IEEE Communications Surveys & Tutorials*, v. 21, n. 2, p. 1409–1434, 2018.
- [Larsen et al. 2019]Larsen, L. M. P. et al. A Survey of the Functional Splits Proposed for 5G Mobile Crosshaul Networks. *IEEE Communications Surveys & Tutorials*, v. 21, n. 1, p. 146–172, 2019.
- [Lopes et al. 2022]LOPES, V. H. L. et al. A coverage-aware vnf placement and resource allocation approach for disaggregated vrans. In: *GLOBECOM 2022 - 2022 IEEE Global Communications Conference*. [S.l.: s.n.], 2022. p. 185–190.
- [Malandrino et al. 2018]MALANDRINO, F. et al. Cellular network traces towards 5g: Usage, analysis and generation. *IEEE Transactions on Mobile Computing*, v. 17, n. 3, p. 529–542, 2018.
- [Marsch et al. 2018]MARSCH, P. et al. *5G system design: architectural and functional considerations and long term research*. [S.l.]: John Wiley & Sons, 2018.
- [Masdari et al. 2016]MASDARI, M. et al. An overview of virtual machine placement schemes in cloud computing. *Journal of Network and Computer Applications*, Elsevier, v. 66, p. 106–127, 2016.
- [Morais et al. 2022]MORAIS, F. Z. et al. Placeran: optimal placement of virtualized network functions in beyond 5g radio access networks. *IEEE Transactions on Mobile Computing*, p. 1–1, 2022.
- [Morais et al. 2022]MORAIS, F. Z. et al. Oplaceran -a placement orchestrator for virtualized next-generation of radio access network. *IEEE Transactions on Network and Service Management*, p. 1–1, 2022.
- [Murti et al. 2023]MURTI, F. W. et al. Deep reinforcement learning for orchestrating cost-aware reconfigurations of vrans. *IEEE Transactions on Network and Service Management*, Institute of Electrical and Electronics Engineers (IEEE), p. 1–1, 2023. Disponível em: <<https://doi.org/10.1109%2Ftnsm.2023.3292713>>.