UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA ELÉTRICA E DE COMPUTAÇÃO

VINÍCIUS FERREIRA DE SOUZA

Minimização da Ociosidade e da Fragmentação Espectral por Algoritmos Inteligentes num Enlace Óptico Elástico sob Tráfego Intenso

CAMPINAS
2024

VINÍCIUS FERREIRA DE SOUZA

Minimização da Ociosidade e da Fragmentação Espectral por Algoritmos Inteligentes num Enlace Óptico Elástico sob Tráfego Intenso

Tese apresentada à Faculdade de Engenharia Elétrica e de Computação da Universidade Estadual de Campinas como parte dos requisitos exigidos para a obtenção do título de Doutor em Engenharia Elétrica, na Área de Telecomunicações e Telemática.

Orientador: Dr. Prof. Helio Waldman

ESTE TRABALHO CORRESPONDE À VERSÃO FINAL DA TESE DEFENDIDA PELO ALUNO VINÍCIUS FERREIRA DE SOUZA, E ORIENTADA PELO PROF. DR. HELIO WALDMAN.

CAMPINAS

2024

Ficha catalográfica
Universidade Estadual de Campinas (UNICAMP)
Biblioteca da Área de Engenharia e Arquitetura
Rose Meire da Silva - CRB 8/5974

So89m

Souza, Vinícius Ferreira de, 1983-
Minimização da ociosidade e da fragmentação espectral por algoritmos inteligentes num enlace óptico elástico sob tráfego intenso / Vinícius Ferreira de Souza. – Campinas, SP : [s.n.], 2024.

Orientador: Helio Waldman.
Tese (doutorado) – Universidade Estadual de Campinas (UNICAMP), Faculdade de Engenharia Elétrica e de Computação.

1. Redes ópticas elásticas. 2. Alocação de espectro. 3. Fragmentação espectral. 4. Ociosidade. 5. Link de fibra. I. Waldman, Helio, 1944-. II. Universidade Estadual de Campinas (UNICAMP). Faculdade de Engenharia Elétrica e de Computação. III. Título.

Informações complementares

Título em outro idioma: Minimization of idleness and spectral fragmentation by intelligent algorithms in an elastic optical link under heavy traffic
Palavras-chave em inglês:
Elastic optical networks
Spectrum allocation
Spectral fragmentation
Idleness
Fiber link
Área de concentração: Telecomunicações e Telemática
Titulação: Doutor em Engenharia Elétrica
Banca examinadora:
Helio Waldman [Orientador]
Michel Daoud Yacoub
Darli Augusto de Arruda Mello
Luiz Henrique Bonani do Nascimento
Monica de Lacerda Rocha
Data de defesa: 26-11-2024
Programa de Pós-Graduação: Engenharia Elétrica

Identificação e informações acadêmicas do(a) aluno(a)
- ORCID do autor: https://orcid.org/0000-0001-9347-7016
- Currículo Lattes do autor: http://lattes.cnpq.br/8032516510974700

COMISSÃO JULGADORA - TESE DE DOUTORADO

Candidato: Vinícius Ferreira de Souza RA: 191657

Data da Defesa: 26 de novembro de 2024

Título da Tese: "Minimização da Ociosidade e da Fragmentação Espectral por Algoritmos Inteligentes num Enlace Óptico Elástico sob Tráfego Intenso".

Prof. Dr. Helio Waldman

Prof. Dr. Michel Daoud Yacoub

Prof. Dr. Darli Augusto de Arruda Mello

Prof. Dr. Luiz Henrique Bonani do Nascimento

Profa. Dra. Monica de Lacerda Rocha

A ata de defesa, com as respectivas assinaturas dos membros da Comissão Julgadora, encontra-se no SIGA (Sistema de Fluxo de Dissertação/Tese) e na Secretaria de Pós-Graduação da Faculdade de Engenharia Elétrica e de Computação.

AGRADECIMENTOS

Este trabalho é fruto da atuação conjunta de várias pessoas que, de formas diferentes, mas igualmente importantes, contribuíram para que os resultados finais fossem alcançados. A todas elas, manifesto minha gratidão e, de modo particular, agradeço:

A Deus, por tudo;

Ao professor e amigo Helio Waldman, orientador desta tese, pelos ensinamentos, apoio, incentivo, confiança, atenção, paciência e pela oportunidade de desenvolver este trabalho;

Ao professor e amigo Raul Camelo de Andrade Almeida Júnior (UFPE - Recife), coorientador desta tese, pelo apoio, pela parceria nos artigos publicados e pela confiança na realização deste trabalho;

Ao professor e amigo Rodrigo Campos Bortoletto (IFSP - Campus Guarulhos), pelo apoio e pela parceria nos artigos publicados;

Ao professor e amigo Luiz Carlos Branquinho (IFSULDEMINAS - Campus Poços de Caldas), pela atenção e pelas importantes dicas de programação com a linguagem Python;

À secretaria da pós-graduação da FEEC/UNICAMP, pelo excelente atendimento e suporte ao longo do curso;

À Faculdade de Engenharia Elétrica e de Computação (FEEC) e à Universidade Estadual de Campinas (UNICAMP), pelo ensino de qualidade e excelência;

Ao IFSULDEMINAS, Campus Inconfidentes e Campus Machado, pela concessão de flexibilidade de horário e afastamento integral para qualificação, o que me proporcionou dedicação exclusiva ao programa de pós-graduação;

À minha família, especialmente à minha mãe, Nilde, pelo amor e carinho incondicionais. Mãe, a senhora é a minha fortaleza. Obrigado por tudo o que fez pelo meu pai;

À minha querida e amada esposa, Telma, companheira inseparável, pelo amor, carinho, atenção, apoio e paciência nos momentos difíceis durante a realização deste trabalho. Obrigado por estar sempre ao meu lado;

Aos meus amados filhos, Rafael e Gabriel, dois anjos em minha vida, fontes inesgotáveis de amor, carinho e companheirismo;

Por fim, um agradecimento especial: ao meu amado e saudoso pai, Silvio de Souza (*in memoriam*), que me deixou ao longo desta caminhada. Pai, o senhor foi o maior incentivador dos meus estudos. Onde quer que esteja, muito obrigado por tudo o que fez por mim!

EPÍGRAFE

“A necessidade é a mãe da inovação.”

Platão (428/427 a.C. - 348/347 a.C.)

Filósofo Grego

RESUMO

Este trabalho aborda a minimização do tempo ocioso dos recursos espectrais num enlace óptico elástico por meio de algoritmos de alocação espectral para um tráfego com duas classes caracterizadas pelo tamanho das suas requisições. Inicialmente, é discutida a prevenção do impasse (*deadlock*) no tráfego incremental por meio de políticas não vorazes de alocação. Além disso, a ociosidade até a exaustão do espectro é mantida finita e mínima mediante a identificação dos estados de enlace específicos nos quais políticas não vorazes devem ser aplicadas. Em seguida, é considerado o caso do tráfego dinâmico em que, devido à natureza temporária do impasse, evita-se recusar a atribuição buscando um algoritmo voraz com desempenho superior ao do estado da arte, representado pelo algoritmo *first-fit*. O primeiro algoritmo proposto é voraz, mas prioriza a alocação de vazios funcionais, que podem ser preenchidos completamente por uma ou mais combinações de conexões com os tamanhos requisitados pelo tráfego, preterindo sem rejeitar os disfuncionais, que são propensos ao impasse. Esta abordagem visa mitigar as perdas por fragmentação. Simulações realizadas para um tráfego com duas classes, com demandas de 400 Gb/s e 1 Tb/s, sob um tráfego poissoniano, demonstraram que o novo algoritmo pode reduzir essas perdas em quase 50% em comparação ao algoritmo *first-fit*, usando a mesma sequência aleatória de requisições. Ao invés de focar na redução direta da fragmentação por meio do deslocamento espectral de conexões ativas, o algoritmo proposto busca otimizar proativamente o pareamento entre cada requisição e o vazio espectral escolhido para sua alocação, priorizando vazios funcionais. As simulações consideraram diferentes distâncias de transmissão, otimizando o número de slots para distâncias curtas (400 km), intermediárias (1600 km) e longas (8000 km). Em seguida, propõe-se o uso da programação dinâmica para obter um algoritmo ótimo, não necessariamente voraz, de atribuição espectral adjacente à extremidade de um vazio selecionado para minimizar a ociosidade futura dos recursos espectrais. Para intensidades suficientemente altas do tráfego, esse algoritmo preconiza em alguns estados a rejeição de requisições de um determinado tamanho que poderia ser acomodado, mas cuja aceitação prejudicaria a ociosidade a longo prazo. O algoritmo utiliza o conhecimento dos tamanhos das requisições do tráfego e suas respectivas taxas, permitindo calcular, para cada tamanho de vazio e política de alocação, o tempo ocioso esperado até que o vazio seja completamente preenchido ou se una por coalescência a outro vazio devido à terminação de uma conexão adjacente. A minimização da ociosidade é alcançada por meio da comparação entre todas as políticas de alocação possíveis para cada tamanho de vazio, selecionando o vazio que oferece a máxima redução da ociosidade

esperada na presença do tráfego poissoniano utilizado nas simulações. Assim, este estudo sugere novas abordagens para a alocação espectral em enlaces elásticos, enfatizando a minimização da ociosidade dos recursos espectrais e a mitigação das perdas por fragmentação. Os algoritmos propostos demonstram melhorias significativas em ocupação espectral e probabilidade de bloqueio de banda, especialmente quando comparados a abordagens tradicionais como o algoritmo *first-fit*.

Palavras-chave: redes ópticas elásticas; enlace único; fragmentação espectral; minimização da ociosidade; tráfego dinâmico.

ABSTRACT

This work addresses the minimization of idle time of spectral resources in an elastic optical link by means of spectral allocation algorithms for a traffic with two classes characterized by the size of their requests. Initially, a deadlock prevention scheme is discussed under incremental traffic by means of non-greedy allocation policies. Then, the resulting finite idle time is minimized by identifying specific link states where non-greedy policies should be applied. Next, the dynamic traffic scenario is considered, where, due to the transient nature of the deadlock, all feasible assignment requests are accommodated by a new greedy algorithm that prioritizes the allocation on the edge of a functional spectral void, which can be completely filled by one or more combinations of connections with the sizes required by the traffic, without rejecting but just deferring the allocation of dysfunctional gaps until a search for a functional one is proved fruitless. This approach aims to mitigate losses due to fragmentation, and is shown to outperform the first-fit algorithm. Simulations performed for traffic with two classes, with demands of 400 Gb/s and 1 Tb/s, under Poissonian traffic, demonstrated that the new algorithm can reduce these losses by almost 50% compared to the first-fit algorithm, using the same random sequence of requests. Instead of focusing on directly reducing fragmentation through spectral shifting of active connections, the proposed algorithm proactively seeks to optimize the pairing between each request and the spectral gap chosen for its allocation, prioritizing functional gaps. The simulations considered different transmission distances, optimizing the number of slots for short (400 km), intermediate (1600 km), and long (8000 km) distances. Next, the use of dynamic programming is proposed to obtain an optimal, not necessarily greedy, algorithm for adjacent spectral assignment to the edge of a selected gap to minimize the expected future idleness of spectral resources. For sufficiently high traffic intensities, this algorithm in some states implies the rejection of requests of a certain size that could be accommodated but whose acceptance would harm long-term idleness, thus configuring an intelligent non-greedy algorithm which may outperform a hypothetical greedy algorithm with no fragmentation losses. The algorithm uses knowledge of the sizes of traffic requests and their respective rates, allowing the calculation, for each gap size and allocation policy, of the expected idle time until the gap is filled or merges with another gap due to the termination of an adjacent connection. The minimization of idleness is achieved by comparing all possible allocation policies for each gap size, selecting the gap that offers the maximum reduction of the expected idleness in the presence of the Poissonian traffic used in the simulations. Thus, this study suggests new approaches for spectral allocation in elastic links, emphasizing the

minimization of the idleness of spectral resources and the mitigation of fragmentation losses. The proposed algorithms demonstrate significant improvements in spectral occupancy and bandwidth blocking probability, especially when compared to traditional approaches such as the first-fit algorithm.

Keywords: elastic optical networks; single link; spectral fragmentation; idleness minimization; dynamic traffic.

LISTA DE ILUSTRAÇÕES

Fig. 2.1 - CMTD: estados e suas probabilidades de transição. 27
Fig. 2.2 - CMTC: estados e suas taxas de transição. 29
Fig. 2.3 - Representação do modelo de Erlang. 33
Fig. 2.4 - Representação do modelo de Engset. 34
Fig. 3.1 - Redes WDM: espectro óptico subdivido em canais de 50 GHz. 39
Fig. 3.2 - Rede flexível com conexões de diferentes larguras de banda concomitantes. 40
Fig. 3.3 - Alocação espectral para tráfego uniclasse com algoritmo aleatório (a) e *first-fit* (b). 41
Fig. 3.4 - Fragmentação do espectro em redes elásticas em três estados: (a) inicial; (b) intermediário; (c) final. 41
Fig. 3.5 - Algoritmo de alocação *first-fit* em três estados: (a) inicial; (b) intermediário; (c) final. 42
Fig. 3.6 - Fragmentação em uma rede elástica. 45
Fig. 3.7 - Topologia em barramento. 48
Fig. 3.8 - Topologia em anel. 49
Fig. 3.9 - Topologia em árvore. 50
Fig. 3.10 - Topologia em malha completa. 51
Fig. 4.1 - Alocação intermediária em um vazio de tamanho v. 58
Fig. 4.2 - Alocação na extremidade (esquerda e direita) de um vazio de tamanho v. 58
Fig. 4.3 - Árvore de estados para $S = 15$ até a exaustão (azul) ou ao impasse (vermelho). 59
Fig. 4.4 - Árvore de estados para $S = 17$. Em destaque, a árvore livre de impasse (DFT). 60
Fig. 4.5 - Representação das políticas aplicadas para $I = 4,7$ e $S = 28$. 63
Fig. 4.6 - Configurações da DFT para as políticas: (a) voraz; (b) não voraz (7 *slots*); (c) não voraz (4 *slots*). 64
Fig. 4.7 - Exemplo de $s(t)$ para $I = (3,5)$ e $S = 13$. 66
Fig. 4.8 - DFT com um único estado de exaustão. 67
Fig. 4.9 - Tamanhos dos vazios na DFT para $I = (5,11)$ e $S = 160$. 68
Fig. 4.10 - Política ótima para $1/3 \leq \beta = 0{,}5 \leq 1{,}2$. 68
Fig. 4.11 - Política ótima para $1 \leq \beta \leq 1{,}2$. 69
Fig. 4.12 - Política ótima para $\beta \geq 1{,}6$. 70
Fig. 4.13 - Tempo de *slot* ocioso esperado para a DFT voraz. 72
Fig. 4.14 - Tempo de *slot* ocioso esperado para a DFT ótima. 73
Fig. 4.15 - Mapeamento dos conjuntos de congruência do módulo 4 em $F(4,7)$. 76
Fig. 4.16 - Árvore de estados para $I = (5,11)$ e $S = 55$. 77
Fig. 4.17 - Árvore de estados para $I = (5,11)$ e $S = 53$. 78
Fig. 4.18 - Mapeamento dos conjuntos de congruência do módulo 4 em $F(4,7,9)$. 79
Fig. 5.1 - Tamanhos de vazios e números de soluções completas de enchimento. 81
Fig. 5.2 - Árvore de tamanhos de vazios a partir de 84, gerada por atribuições de conexões de 4 e 7 *slots*, colorida de acordo com os membros do conjunto $\sigma_i(4,7)$. 82
Fig. 5.3 - Fluxograma para o algoritmo proposto. 91
Fig. 5.4 - Ocupação espectral média (ρ) × carga (L) para $\beta = 0{,}5$ e $(b_1, b_2) = (5,11)$. 102
Fig. 5.5 - Ocupação espectral média (ρ) × carga (L) para $\beta = 0{,}5$ e $(b_1, b_2) = (5,14)$. 102
Fig. 5.6 - Ocupação espectral média (ρ) × carga (L) para $\beta = 0{,}5$ e $(b_1, b_2) = (7,17)$. 103
Fig. 5.7 - Ocupação espectral média (ρ) × carga (L) para $\beta = 0{,}5$ e $(b_1, b_2) = (13,28)$. 103
Fig. 5.8 - Ocupação espectral média (ρ) × carga (L) para $\beta = 1{,}0$ e $(b_1, b_2) = (5,11)$. 104
Fig. 5.9 - Ocupação espectral média (ρ) × carga (L) para $\beta = 1{,}0$ e $(b_1, b_2) = (5,14)$. 104
Fig. 5.10 - Ocupação espectral média (ρ) × carga (L) para $\beta = 1{,}0$ e $(b_1, b_2) = (7,17)$. 105

Fig. 5.11 - Ocupação espectral média (ρ) × carga (L) para $\beta = 1{,}0$ e $(b_1, b_2) = (13, 28)$.....105
Fig. 5.12 - Ocupação espectral média (ρ) × carga (L) para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 11)$.....106
Fig. 5.13 - Ocupação espectral média (ρ) × carga (L) para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$.....106
Fig. 5.14 - Ocupação espectral média (ρ) × carga (L) para $\beta = 2{,}0$ e $(b_1, b_2) = (7, 17)$.....107
Fig. 5.15 - Ocupação espectral média (ρ) × carga (L) para $\beta = 2{,}0$ e $(b_1, b_2) = (13, 28)$.....107
Fig. 5.16 - Vazão média (τ) em Tb/s × carga (L) para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$. 108
Fig. 5.17 - Probabilidade de bloqueio de banda × carga (L) para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$.....108
Fig. 5.18 - Probabilidade de bloqueio de b_1 × carga (L) para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$. 109
Fig. 5.19 - Probabilidade de bloqueio de b_2 × carga (L) para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$. 109
Fig. 5.20 - Redução das perdas por fragmentação × FF para $\beta = 0{,}5$. 110
Fig. 5.21 - Redução das perdas por fragmentação × FF para $\beta = 1{,}0$. 110
Fig. 5.22 - Redução das perdas por fragmentação × FF para $\beta = 2{,}0$. 110
Fig. 5.23 - Mecanismo de recorrência para $(b_1, b_2) = (3, 5)$ e $S = 10$.....118
Fig. 5.24 - Recompensas D1 × vazios até $v = 50$. 123
Fig. 5.25 - Recompensas D2 × vazios até $v = 50$. 123
Fig. 5.26 - Recompensas D1 × vazios até $v = 160$. 123
Fig. 5.27 - Recompensas D2 × vazios até $v = 160$. 123
Fig. 5.28 - Recompensas D1 × vazios até $v = 320$. 124
Fig. 5.29 - Recompensas D2 × vazios até $v = 320$. 124
Fig. 5.30 - Ocupação espectral média (ρ) × carga (L) para $\beta = 0{,}5$ e $(b_1, b_2) = (5, 11)$.....128
Fig. 5.31 - Ocupação espectral média (ρ) × carga (L) para $\beta = 0{,}5$ e $(b_1, b_2) = (5, 14)$.....128
Fig. 5.32 - Ocupação espectral média (ρ) × carga (L) para $\beta = 0{,}5$ e $(b_1, b_2) = (7, 17)$.....128
Fig. 5.33 - Ocupação espectral média (ρ) × carga (L) para $\beta = 0{,}5$ e $(b_1, b_2) = (13, 28)$.....128
Fig. 5.34 - Ocupação espectral média (ρ) × carga (L) para $\beta = 1{,}0$ e $(b_1, b_2) = (5, 11)$.....130
Fig. 5.35 - Ocupação espectral média (ρ) × carga (L) para $\beta = 1{,}0$ e $(b_1, b_2) = (5, 14)$.....130
Fig. 5.36 - Ocupação espectral média (ρ) × carga (L) para $\beta = 1{,}0$ e $(b_1, b_2) = (7, 17)$.....130
Fig. 5.37 - Ocupação espectral média (ρ) × carga (L) para $\beta = 1{,}0$ e $(b_1, b_2) = (13, 28)$.....130
Fig. 5.38 - Ocupação espectral média (ρ) × carga (L) para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 11)$.....132
Fig. 5.39 - Ocupação espectral média (ρ) × carga (L) para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$.....132
Fig. 5.40 - Ocupação espectral média (ρ) × carga (L) para $\beta = 2{,}0$ e $(b_1, b_2) = (7, 17)$.....132
Fig. 5.41 - Ocupação espectral média (ρ) × carga (L) para $\beta = 2{,}0$ e $(b_1, b_2) = (13, 28)$.....132
Fig. 5.42 - Vazão média (τ) em Tb/s × carga (L) para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$. 134
Fig. 5.43 - Probabilidade de bloqueio de banda × carga (L) para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$.....134
Fig. 5.44 - Probabilidade de bloqueio de b_1 × carga (L) para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$. 134
Fig. 5.45 - Probabilidade de bloqueio de b_2 × carga (L) para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$. 134
Fig. 5.46 - Caso OMa: redução das perdas por fragmentação × FF. 135
Fig. 5.47 - Caso OMa: redução das perdas totais × FF. 135
Fig. 5.48 - Caso OMa: recuperação da ocupação × FF. 136
Fig. 5.49 - Insuficiência de recursos espectrais × Fragmentação. 137
Fig. 6.1 - Caso $L = 4{,}0$: ocupação espectral média (ρ) × carga (L) para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$. 141
Fig. 6.2 - Caso $L = 4{,}0$: probabilidade de bloqueio de banda × carga (L) para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$. 141
Fig. 6.3 - Caso $L = 4{,}0$: probabilidade de bloqueio de b_1 × carga (L) para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$. 142
Fig. 6.4 - Caso $L = 4{,}0$: probabilidade de bloqueio de b_2 × carga (L) para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$. 142

Fig. 6.5 - Caso $L = 4{,}0$: vazão média (τ) em Tb/s × carga (L) para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$.....143
Fig. 6.6 - Caso $L = 4{,}0$: redução das perdas por fragmentação × FF para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$. 143
Fig. 6.7 - Caso $L = 4{,}0$: redução das perdas totais × FF para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$. 143
Fig. 6.8 - Caso $L = 4{,}0$: recuperação da ocupação × FF para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$. 143
Fig. 6.9 - Caso $L = 4{,}0$: ocupação espectral média (ρ) × carga (L) para $\beta = 0{,}5$ e $(b_1, b_2) = (13, 28)$. 144
Fig. 6.10 - Caso $L = 4{,}0$: probabilidade de bloqueio de banda × carga (L) para $\beta = 0{,}5$ e $(b_1, b_2) = (13, 28)$.....144
Fig. 6.11 - Caso $L = 4{,}0$: probabilidade de bloqueio de b_1 × carga (L) para $\beta = 0{,}5$ e $(b_1, b_2) = (13, 28)$.....145
Fig. 6.12 - Caso $L = 4{,}0$: probabilidade de bloqueio de b_2 × carga (L) para $\beta = 0{,}5$ e $(b_1, b_2) = (13, 28)$.....145
Fig. 6.13 - Caso $L = 4{,}0$: vazão média (τ) em Tb/s × carga (L) para $\beta = 0{,}5$ e $(b_1, b_2) = (13, 28)$.146
Fig. 6.14 - Caso $L = 4{,}0$: redução das perdas por fragmentação × FF para $\beta = 0{,}5$ e $(b_1, b_2) = (13, 28)$.....146
Fig. 6.15 - Caso $L = 4{,}0$: redução das perdas totais × FF para $\beta = 0{,}5$ e $(b_1, b_2) = (13, 28)$. 146
Fig. 6.16 - Caso $L = 4{,}0$: recuperação da ocupação × FF para $\beta = 0{,}5$ e $(b_1, b_2) = (13, 28)$. 146
Fig. 6.17 - Caso $L = 10{,}0$: ocupação espectral média (ρ) × carga (L) para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$. 148
Fig. 6.18 - Caso $L = 10{,}0$: probabilidade de bloqueio de banda × carga (L) para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$. 148
Fig. 6.19 - Caso $L = 10{,}0$: probabilidade de bloqueio de b_1 × carga (L) para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$. 148
Fig. 6.20 - Caso $L = 10{,}0$: probabilidade de bloqueio de b_2 × carga (L) para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$. 148
Fig. 6.21 - Caso $L = 10{,}0$: vazão média (τ) em Tb/s × carga (L) para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$.149
Fig. 6.22 - Caso $L = 10{,}0$: redução das perdas por fragmentação × FF para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$. 149
Fig. 6.23 - Caso $L = 10{,}0$: redução das perdas totais × FF para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$. 149
Fig. 6.24 - Caso $L = 10{,}0$: recuperação da ocupação × FF para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$. 149
Fig. 6.25 - Caso $L = 10{,}0$: ocupação espectral média (ρ) × carga (L) para $\beta = 0{,}5$ e $(b_1, b_2) = (13, 28)$.....150
Fig. 6.26 - Caso $L = 10{,}0$: probabilidade de bloqueio de banda × carga (L) para $\beta = 0{,}5$ e $(b_1, b_2) = (13, 28)$.....150
Fig. 6.27 - Caso $L = 10{,}0$: probabilidade de bloqueio de b_1 × carga (L) para $\beta = 0{,}5$ e $(b_1, b_2) = (13, 28)$.....151
Fig. 6.28 - Caso $L = 10{,}0$: probabilidade de bloqueio de b_2 × carga (L) para $\beta = 0{,}5$ e $(b_1, b_2) = (13, 28)$.....151
Fig. 6.29 - Caso $L = 10{,}0$: vazão média (τ) em Tb/s × carga (L) para $\beta = 0{,}5$ e $(b_1, b_2) = (13, 28)$. 151
Fig. 6.30 - Caso $L = 10{,}0$: redução das perdas por fragmentação × FF para $\beta = 0{,}5$ e $(b_1, b_2) = (13, 28)$.....151
Fig. 6.31 - Caso $L = 10{,}0$: redução das perdas totais × FF para $\beta = 0{,}5$ e $(b_1, b_2) = (13, 28)$. 152
Fig. 6.32 - Caso $L = 10{,}0$: recuperação da ocupação × FF para $\beta = 0{,}5$ e $(b_1, b_2) = (13, 28)$. 152

LISTA DE QUADROS

Quadro 5.1 - Número de vagas inflexíveis para todo $v \in (3,5)$. 88
Quadro 5.2 - Níveis de ocupação espectral para $\beta = 0{,}5$ e $(b_1, b_2) = 5,11$. 103
Quadro 5.3 - Níveis de ocupação espectral para $\beta = 0{,}5$ e $(b_1, b_2) = 5,14$. 103
Quadro 5.4 - Níveis de ocupação espectral para $\beta = 0{,}5$ e $(b_1, b_2) = 7,17$. 104
Quadro 5.5 - Níveis de ocupação espectral para $\beta = 0{,}5$ e $(b_1, b_2) = 13,28$. 104
Quadro 5.6 - Níveis de ocupação espectral para $\beta = 1{,}0$ e $(b_1, b_2) = 5,11$. 105
Quadro 5.7 - Níveis de ocupação espectral para $\beta = 1{,}0$ e $(b_1, b_2) = 5,14$. 105
Quadro 5.8 - Níveis de ocupação espectral para $\beta = 1{,}0$ e $(b_1, b_2) = 7,17$. 106
Quadro 5.9 - Níveis de ocupação espectral para $\beta = 1{,}0$ e $(b_1, b_2) = 13,28$. 106
Quadro 5.10 - Níveis de ocupação espectral para $\beta = 2{,}0$ e $(b_1, b_2) = 5,11$. 107
Quadro 5.11 - Níveis de ocupação espectral para $\beta = 2{,}0$ e $(b_1, b_2) = 5,14$. 107
Quadro 5.12 - Níveis de ocupação espectral para $\beta = 2{,}0$ e $(b_1, b_2) = 7,17$. 108
Quadro 5.13 - Níveis de ocupação espectral para $\beta = 2{,}0$ e $(b_1, b_2) = 13,28$. 108
Quadro 5.14 - Caso NA: níveis de ocupação em % para $\beta = 2{,}0$ e $(b_1, b_2) = 5,14$. 109
Quadro 5.15 - Caso NA: níveis de vazão média para $\beta = 2{,}0$ e $(b_1, b_2) = 5,14$. 109
Quadro 5.16 - Ociosidade mínima para $L = 0{,}4$, $\beta = 1{,}0$ e $(b_1, b_2) = 5,11$ 121
Quadro 5.17 - Ociosidade mínima para $L = 1{,}0$, $\beta = 1{,}0$ e $(b_1, b_2) = 5,11$ 122
Quadro 5.18 - Ociosidade mínima para $L = 1{,}9$, $\beta = 1{,}0$ e $(b_1, b_2) = 5,11$ 125
Quadro 5.19 - Níveis de ocupação espectral para $\beta = 0{,}5$ e $(b_1, b_2) = 5,11$. 129
Quadro 5.20 - Níveis de ocupação espectral para $\beta = 0{,}5$ e $(b_1, b_2) = 5,14$. 129
Quadro 5.21 - Níveis de ocupação espectral para $\beta = 0{,}5$ e $(b_1, b_2) = 7,17$. 129
Quadro 5.22 - Níveis de ocupação espectral para $\beta = 0{,}5$ e $(b_1, b_2) = 13,28$. 129
Quadro 5.23 - Níveis de ocupação espectral para $\beta = 1{,}0$ e $(b_1, b_2) = 5,11$. 131
Quadro 5.24 - Níveis de ocupação espectral para $\beta = 1{,}0$ e $(b_1, b_2) = 5,14$. 131
Quadro 5.25 - Níveis de ocupação espectral para $\beta = 1{,}0$ e $(b_1, b_2) = 7,17$. 131
Quadro 5.26 - Níveis de ocupação espectral para $\beta = 1{,}0$ e $(b_1, b_2) = 13,28$. 131
Quadro 5.27 - Níveis de ocupação espectral para $\beta = 2{,}0$ e $(b_1, b_2) = 5,11$. 133
Quadro 5.28 - Níveis de ocupação espectral para $\beta = 2{,}0$ e $(b_1, b_2) = 5,14$. 133
Quadro 5.29 - Níveis de ocupação espectral para $\beta = 2{,}0$ e $(b_1, b_2) = 7,17$. 133
Quadro 5.30 - Níveis de ocupação espectral para $\beta = 2{,}0$ e $(b_1, b_2) = 13,28$. 133
Quadro 5.31 - Caso OMa: níveis de ocupação em % para $\beta = 2{,}0$ e $(b_1, b_2) = 5,14$ 135
Quadro 5.32 - Caso OMa: níveis de vazão média para $\beta = 2{,}0$ e $(b_1, b_2) = 5,14$. 135
Quadro 6.1 - Caso $L = 4{,}0$: níveis de ocupação espectral para $\beta = 2{,}0$ e $(b_1, b_2) = 5,14$. 144
Quadro 6.2 - Caso $L = 4{,}0$: níveis de vazão média para $\beta = 2{,}0$ e $(b_1, b_2) = 5,14$. 144
Quadro 6.3 - Caso $L = 4{,}0$: níveis de ocupação espectral para $\beta = 0{,}5$ e $(b_1, b_2) = 13,28$. 147
Quadro 6.4 - Caso $L = 4{,}0$: níveis de vazão média para $\beta = 0{,}5$ e $(b_1, b_2) = 13,28$. 147
Quadro 6.5 - Caso $L = 10{,}0$: níveis de ocupação espectral para $\beta = 2{,}0$ e $(b_1, b_2) = 5,14$. 150
Quadro 6.6 - Caso $L = 10{,}0$: níveis de vazão média para $\beta = 2{,}0$ e $(b_1, b_2) = 5,14$. 150
Quadro 6.7 - Caso $L = 10{,}0$: níveis de ocupação espectral para $\beta = 0{,}5$ e $(b_1, b_2) = 13,28$. 152
Quadro 6.8 - Caso $L = 10{,}0$: níveis de vazão média para $\beta = 0{,}5$ e $(b_1, b_2) = 13,28$. 152

LISTA DE SIGLAS

EON	Elastic Optical Network
CMTC	Cadeia de Markov em Tempo Contínuo
CMTD	Cadeia de Markov em Tempo Discreto
CS	Complete Sharing
CSMA	Carrier Sense Multiple Access
DF	Desfragmentador
DFT	Deadlock-Free Tree
DSP	Digital Signal Processing
DWDM	Dense Wavelength Division Multiplexing
EF	Exact-Fit
EFF	Exact-First-Fit
FDDI	Fiber Distributed Data Interface
FF	First-Fit
G	Greedy
IR	Idleness Reduction
ITU	International Telecommunications Union
ITU-T	Telecommunication Standardization Sector
LAN	Local Area Network
MGSP	Multi-Graph Shortest Path
MIR	Maximum Idleness Reduction
ML	Machine Learnig
NA	New Algorithm
OMa	Ociosidade Mínima com Alocação Adjacente
PASTA	Poisson Arrivals See Time Averages
RMSA	Routing, Modulation and Spectrum Allocation
ROP	Taxa de Recuperação da Ocupação
RPT	Taxa de Redução das Perdas Totais
RSA	Routing and Spectrum Assignment
RWA	Routing and Wavelength Assignment
SA	Spectral Assignment
SDH	Synchronous Digital Hierarchy
SDM	Space Division Multiplexing
SMF	Single Mode Fiber
SONET	Synchronous Optical Networking
WDM	Wavelength Division Multiplexing

SUMÁRIO

1 INTRODUÇÃO 20
1.2 JUSTIFICATIVA 21
1.3 OBJETIVOS 23
1.3.1 OBJETIVO GERAL 23
1.3.2 OBJETIVOS ESPECÍFICOS 23
1.4 ORGANIZAÇÃO DA TESE 24
2 SISTEMAS MARKOVIANOS 25
2.1 CADEIAS DE MARKOV 25
2.1.1 Cadeia de Markov em Tempo Discreto 25
2.1.2 Cadeia de Markov em Tempo Contínuo 28
2.1.2.1 Reversibilidade 29
2.2 TRÁFEGO POISSONIANO 31
2.2.1 Modelo de Erlang 32
2.2.2 Modelo de Engset 34
3 FRAGMENTAÇÃO ESPECTRAL EM REDES ÓPTICAS ELÁSTICAS 35
3.1 O SISTEMA DESFRAGMENTADO COMO REFERÊNCIA 36
3.2 O SISTEMA FRAGMENTADO 37
3.2.1 Algoritmos RSA 38
3.2.1.1 Roteamento 38
3.2.1.2 Alocação Espectral 39
3.2.2 Fragmentação no Enlace Avulso 44
3.2.3 Topologias com Múltiplos Enlaces 47
3.2.3.1 Topologia de Rede em Barramento 47
3.2.3.2 Topologia de Rede em Anel 48
3.2.3.3 Topologia em Árvore 49
3.2.3.4 Topologia de Rede em Malha 51
3.2.4 Abordagens Sem Desfragmentação em Topologias com Múltiplos Enlaces 52
3.2.4.1 Roteamento Multicaminho 52
3.2.4.2 Multigrafo 53
3.2.4.3 Particionamento do Espectro 54
3.2.4.4 Reencaminhamento 55
4 ALGORITMOS CIENTES DO TRÁFEGO INCREMENTAL 56
4.1 OCORRÊNCIA DE IMPASSE (DEADLOCK) 57

4.1.1 Posicionamento Espectral 57
4.1.2 Ociosidade Residual 59
4.1.3 Minimização da Ociosidade Sensível ao Tráfego 61
4.1.3.1 Único Estado de Exaustão 66
4.1.3.2 Múltiplos Estados de Exaustão 67
4.2 POLÍTICAS NÃO VORAZES 74
5 ALGORITMOS CIENTES DO TRÁFEGO DINÂMICO 80
5.1 DELINEAMENTO DO ESTUDO E TEMÁTICA DA PESQUISA 80
5.1.1 Vazios Espectrais 80
5.1.1.1 Vazios Disfuncionais 82
5.1.1.2 Vazios Inflexíveis 84
5.1.1.3 Vazios Flexíveis 85
5.2 ALGORITMO CIENTE DOS TAMANHOS DAS REQUISIÇÕES 89
5.2.1 Algoritmo Proposto 89
5.3 RESULTADOS NUMÉRICOS 94
5.3.1 O Limite Analítico 95
5.3.2 Simulações Algorítmicas para o Enlace Fragmentado Espectralmente 97
5.3.2.1 Algoritmo First-Fit (FF) 100
5.3.2.2 Algoritmo Exact-First-Fit (EFF) 100
5.3.3 Simulações do NA 100
5.3.3.1 Simulações do NA com $\beta = 0{,}5$ 102
5.3.3.2 Simulações do NA com $\beta = 1{,}0$ 104
5.3.3.3 Simulações do NA com $\beta = 2{,}0$ 106
5.3.3.4 Caso Completo: Simulação do NA com $(b_1, b_2) = 5, 14$ e $\beta = 2{,}0$ 108
5.3.3.5 Discussão dos Resultados para a Simulação do NA 110
5.4 ALGORITMO CIENTE DO PERFIL COMPLETO DO TRÁFEGO 112
5.4.1 Redução Máxima da Ociosidade 112
5.4.2 Tabulação e Simulações 119
5.4.3 Simulações do Algoritmo OMa 126
5.4.3.1 Simulações do OMa com $\beta = 0{,}5$ 127
5.4.3.2 Simulações do OMa com $\beta = 1{,}0$ 129
5.4.3.3 Simulações do OMa com $\beta = 2{,}0$ 132
5.4.3.4 Caso Completo: Simulação do OMa com $(b_1, b_2) = 5, 14$ e $\beta = 2{,}0$ 133
5.3.3.5 Discussão dos Resultados para a Simulação do OMa 136
6 PERSPECTIVAS FUTURAS 139

6.1 SIMULAÇÕES COM A CARGA ATÉ $L = 4{,}0$ 140
6.1.1 Caso $L = 4{,}0$: simulação do OMa com $\beta = 2{,}0$ e $(b_1, b_2) = 5, 14$ 141
6.1.2 Caso $L = 4{,}0$: simulação do OMa com $\beta = 0{,}5$ e $(b_1, b_2) = 13, 28$ 144
6.2 SIMULAÇÕES COM A CARGA ATÉ $L = 10{,}0$ 147
6.2.1 Caso $L = 10{,}0$: simulação do OMa com $\beta = 2{,}0$ e $(b_1, b_2) = 5, 14$ 147
6.2.2 Caso $L = 10{,}0$: simulação do OMa com $\beta = 0{,}5$ e $(b_1, b_2) = 13, 28$ 150
6.3 DISCUSSÃO DOS RESULTADOS 153
6.4 CONCLUSÕES 154
6.5 TRABALHOS FUTUROS 157
REFERÊNCIAS BIBLIOGRÁFICAS 160
APÊNDICE A - TRABALHOS PUBLICADOS 166

1 INTRODUÇÃO

Desde o início do século XXI, quando o tráfego da Internet se tornou predominante nas redes de comunicação de dados, a taxa de crescimento desse tráfego foi acelerada pela introdução de novas tecnologias, tais como *streaming* de vídeo, serviços multiplataforma e Internet das Coisas. Para acompanhar esse crescimento, a infraestrutura óptica da rede tem sido expandida durante as últimas décadas para incorporar o tráfego adicional, principalmente por meio da ativação de cada vez mais comprimentos de onda nas fibras instaladas [1].

A instalação de novas fibras na rede é uma operação onerosa e que tende a ser evitada. O desafio é melhorar o aproveitamento da rede instalada para suportar um maior número de clientes e taxas mais altas. Isso passa por aumentar a sua eficiência espectral, ou seja, aumentar a quantidade de informação transmitida por meio de uma mesma fibra [2].

Para enfrentar este desafio, as Redes Ópticas Elásticas (EONs, do inglês *Elastic Optical Networks*) apostam na flexibilidade como sua principal característica, especialmente na alocação de espectro e na seleção do esquema de modulação mais adequado para cada situação. Nas EONs, o espectro é dividido em *slots* contíguos numa grade flexível e os usuários podem solicitar um número de *slots* adjacentes conforme a sua demanda de largura de banda. Desse modo, o paradigma original das redes WDM (*Wavelength Division Multiplexing*) está sendo substituído por um esquema flexível (ou elástico), no qual a grade de *slots* não segue mais uma tabela fixa, mas sim permite conexões de qualquer tamanho serem alocadas em qualquer região não ocupada do espectro. Além disso, o formato de modulação escolhido depende da distância entre os nós de origem e destino, podendo coexistir diferentes esquemas de modulação [2][3].

As requisições de conexão devem ser atendidas por um algoritmo de alocação espectral que, mediante critérios de seleção, irá posicionar as conexões no espectro dentro dos blocos de *slots* disponíveis. O mesmo bloco deve estar livre em todos os enlaces para atender a requisição, conforme a restrição de continuidade. No modelo de rede óptica elástica, o ITU-T[1] padronizou a divisão do espectro em *slots* de 12,5 GHz. Diferentemente das redes WDM, as conexões nas redes elásticas podem ocupar mais de um *slot*, contanto que os *slots* alocados sejam consecutivos no espectro (restrição de contiguidade) [1].

Durante o funcionamento de uma rede, diversos eventos dinâmicos ocorrem, como a chegada de novas conexões, que podem ser aceitas ou rejeitadas, e a finalização de conexões já estabelecidas. Quando uma conexão é finalizada pelo seu usuário, ela libera *slots* no espectro,

[1] Setor de Padronização de Telecomunicações da ITU (*International Telecommunication Union*).

que ficam disponíveis para serem utilizados por novos usuários. Esse fenômeno, conhecido como fragmentação espectral, resulta na criação de espaços não alocados no espectro. Nos cenários em que usuários possuem requisitos de banda variados, esses espaços podem não ser adequados para atender a novas demandas, levando a perdas por fragmentação que podem comprometer a eficiência da rede óptica elástica [1].

A fim de mitigar essas perdas, foram desenvolvidos diversos esquemas e métricas para gerenciar e minimizar a fragmentação espectral. Essas estratégias buscam otimizar o uso do espectro, garantindo que as conexões possam ser estabelecidas de forma contínua e eficiente, mesmo em ambientes com múltiplos enlaces e diferentes classes de serviço [1].

Duas formas principais de fragmentação são destacadas: a fragmentação horizontal, que ocorre quando os recursos disponíveis ao longo de um caminho de múltiplos enlaces não estão alinhados para garantir a continuidade espectral, e a fragmentação vertical, que surge em um único enlace quando diferentes classes de serviço competem por *slots* de tamanhos variados. Embora algoritmos como o *first-fit* possam impedir a fragmentação em cenários com uma única classe de serviço num enlace avulso, o desafio permanece para situações com múltiplas classes e/ou em redes com topologia em malha [4].

Este projeto de pesquisa é motivado pela possibilidade de criar um alinhamento eficaz entre as requisições de banda e os espaços liberados no espectro, buscando minimizar as perdas por fragmentação em redes com múltiplas classes de tráfego. Para não produzir fragmentação adicional ao atribuir espectro a uma solicitação, assume-se que o algoritmo proposto sempre atribuirá o número solicitado de *slots* para ser contíguo a uma ou duas conexões ativas. Toda a fragmentação é, então, produzida por decisões de terminação independentes tomadas pelos usuários. Dessa forma, cabe ao algoritmo decidir, a cada solicitação de chegada, em qual dos vazios disponíveis e em qual extremidade do vazio o pedido deve ser acomodado. A performance do algoritmo é verificada pela comparação do seu desempenho com um limite analítico e com outros dois algoritmos sugeridos na literatura.

1.2 JUSTIFICATIVA

O dilema da alocação de espectro nas redes WDM se resume à escolha de *slots* que sejam suficientes para acomodar a demanda e que estejam disponíveis nas mesmas posições do espectro em cada enlace do caminho óptico. Os algoritmos que resolvem esse problema são

chamados de RWA (*Routing and Wavelength Assignment*) na rede WDM de grade fixa e de RSA (*Routing and Spectrum Assignment*) na rede elástica [1].

Se vários *slots* forem solicitados, eles precisam ser contíguos, o que torna o problema de roteamento e atribuição espectral mais complicado do que em redes WDM. Assim, a tradicional restrição de continuidade do comprimento de onda do problema RWA torna-se, então, uma restrição de continuidade do espectro no problema RSA [1].

Em redes com atribuição de largura de banda não uniforme, a fragmentação do espectro é um problema recorrente, uma vez que o processo de ativação e término de conexões resulta, em pouco tempo, em um espectro livre fragmentado. Por sua vez, isso resulta em várias lacunas espectrais pequenas e não contínuas que não podem ser usadas para satisfazer novas solicitações de largura de banda maior. Desse modo, a eficiência do espectro adquirida pela flexibilidade na alocação da largura de banda pode ser reduzida devido a essas pequenas faixas de frequências livres e não contíguas espalhadas ao longo do espectro.

As redes ópticas elásticas operando sob condições dinâmicas de tráfego, em que as demandas futuras por largura de banda não podem ser previstas com antecedência, enfrentam um desafio significativo: a fragmentação espectral. Esse fenômeno ocorre quando espaços livres no espectro óptico permanecem ociosos por longos períodos, resultando em uma utilização ineficiente dos recursos disponíveis. A fragmentação torna-se, portanto, um obstáculo inevitável para a maximização da eficiência espectral em EONs.

Neste contexto, o presente trabalho propõe o desenvolvimento de um novo algoritmo de alocação espectral que adota uma abordagem diferenciada para a seleção e alocação dos espaços no espectro. O algoritmo distingue entre dois tipos de vazios funcionais: os “inflexíveis”, que só podem ser preenchidos por combinações específicas de conexões, e os “flexíveis”, que têm a capacidade de suportar múltiplas combinações de requisições. Ao priorizar a alocação dos vazios inflexíveis de acordo com as características de cada requisição, sem favorecer previamente qualquer região do espectro, espera-se aumentar a eficácia na utilização dos recursos e melhorar a flexibilidade no atendimento das demandas dinâmicas.

Essa nova abordagem visa não apenas reduzir a fragmentação espectral, mas também otimizar a alocação de recursos em EONs, contribuindo para uma maior eficiência e adaptabilidade no gerenciamento do tráfego em condições dinâmicas.

Posteriormente, uma segunda proposta discute e apresenta um algoritmo de atribuição espectral com ciência do perfil completo do tráfego, não voraz e inteligente, que visa minimizar a ociosidade espectral de um vazio ao longo do tempo e, consequentemente, reduzir as perdas por fragmentação.

Mediante a comparação com um modelo analítico de um enlace avulso sem perdas por fragmentação e com duas abordagens de algoritmos sugeridos na literatura (*first-fit* e *exact-fit*), espera-se confirmar o pressuposto de que se requisitos inflexíveis são atendidos primeiro, sempre que possível a partir das solicitações de chegada, a flexibilidade é preservada para futuras atribuições sem nenhum custo, o que mitigaria as perdas por fragmentação.

1.3 OBJETIVOS

1.3.1 OBJETIVO GERAL

Contribuir para a evolução das futuras redes ópticas elásticas, mediante a proposta de um novo algoritmo para alocação espectral em cenários de tráfego dinâmico e com múltiplas classes de serviços em um enlace avulso, com o intuito de minimizar as perdas por fragmentação.

1.3.2 OBJETIVOS ESPECÍFICOS

Para o presente trabalho foram definidos os seguintes objetivos específicos:

a) investigar a dinâmica de um enlace avulso sem perdas por fragmentação sob tráfego dinâmico, representado por uma cadeia de Markov de tempo contínuo reversível, por meio de um modelo analítico [5];

b) determinar analiticamente a ocupação espectral e vazão médias de um enlace avulso;

c) propor e implementar um algoritmo de alocação espectral que opere ciente de um tráfego com duas classes;

d) verificar o desempenho do algoritmo proposto para reduzir as perdas por fragmentação, por meio da comparação com um modelo analítico e com outros dois algoritmos sugeridos na literatura.

1.4 ORGANIZAÇÃO DA TESE

Esta tese está dividida em seis capítulos, que estão organizados da seguinte forma:

a) capítulo 1: apresenta uma breve introdução sobre Redes Ópticas Elásticas, seu contexto histórico e a problematização da fragmentação espectral. Além disso, aponta a justificativa e os objetivos do trabalho proposto;

b) capítulo 2: aborda os Sistemas Markovianos, com destaque para os conceitos de processo estocástico e as características das Cadeias de Markov;

c) capítulo 3: trata do problema da fragmentação espectral do enlace óptico e apresenta a definição desse conceito, os principais tipos de fragmentação, suas causas e consequências, os principais algoritmos de alocação espectral e as topologias de redes. O capítulo finaliza destacando as principais abordagens sem desfragmentação para EONs;

d) capítulo 4: discute e analisa matematicamente o problema da ocorrência de impasses e a sua relação com as políticas de alocação espectral para múltiplas classes de serviços sob o tráfego incremental;

e) capítulo 5: exibe de forma detalhada a proposta e simulação de dois algoritmos de alocação espectral sob tráfego dinâmico: um tipo voraz, mas que prioriza um determinado tipo de vazio caracterizado pela sua funcionalidade no preenchimento pelos tamanhos demandados, e um outro não voraz e inteligente, que seleciona o vazio para alocação com base em um critério de recompensa configurado para minimizar a ociosidade espectral do vazio;

f) capítulo 6: avalia as perspectivas futuras para a continuidade desta pesquisa e inclui uma análise da simulação dos algoritmos propostos com a carga estendida. Finalmente, o capítulo é encerrado com a conclusão do trabalho e exibe a lista dos trabalhos publicados.

2 SISTEMAS MARKOVIANOS

Um Processo Estocástico pode ser definido como um conjunto de variáveis aleatórias $X(t)$, indexadas por um índice t, que caracteriza a evolução de um sistema de valores no tempo e $X(t)$ representa uma grandeza mensurável de interesse no tempo t [5].

Os processos estocásticos são utilizados para descrever a operação de uma rede óptica em um determinado período de tempo. Assim, a variável aleatória $X(t)$ indica o estado do sistema no tempo t e, portanto, pode-se afirmar que $X(t)$ é definido em um conjunto denominado Espaço de Estados [5].

Um processo estocástico é classificado como um Processo Markoviano quando satisfaz a Propriedade de Markov, ou seja, a distribuição de probabilidade do próximo estado depende somente do estado atual. Dessa forma, um Sistema Markoviano não apresenta memória dos eventos passados [2]. A próxima seção apresenta uma breve revisão sobre as Cadeias de Markov.

2.1 CADEIAS DE MARKOV

Uma Cadeia de Markov é um caso especial de processo estocástico que satisfaz a propriedade de Markov. Em relação ao tempo como parâmetro, as cadeias de Markov podem ser classificadas como Cadeia de Markov em Tempo Discreto (CMTD), comumente chamada Cadeia de Markov, e Cadeia de Markov em Tempo Contínuo (CMTC), em geral designada como Processo de Markov ou Processo Markoviano [6].

2.1.1 Cadeia de Markov em Tempo Discreto

Considere um processo estocástico discreto no tempo (X_k), em que $k \in \{0, 1, 2, 3, \dots\}$, definido num espaço de estados S, que é denominado Cadeia de Markov em Tempo Discreto se [7]:

$$P\{X(k+1) = x_{k+1} | X(k) = x_k, X(k-1) = x_{k-1}, \dots, X(1) = x_1, X(0) = x_0\} = P\{X(k+1) = x_{k+1} | X(k) = x_k\} \quad (2.1)$$

para $\forall\, 0, 1, 2, \dots, k-1, k, k+1$.

As probabilidades de transição $P\{X(k+1) = x_{k+1} | X(k) = x_k\}$ representam a probabilidade condicional de um estado futuro $X(k+1)$ ser igual a x_{k+1} no instante de tempo $k+1$, dado que o estado presente $X(k)$ é igual a x_k no tempo k e não há qualquer dependência dos estados passados (do tempo 0 ao tempo $k-1$) [7].

Em uma Cadeia de Markov, as probabilidades condicionais são chamadas de Probabilidades de Transição de Passo n. Para simplificar a notação utilizada, seja x_{k+n} igual a j e x_k igual a i, de modo que [7]:

$$p_{ij}^{(n)} = P\{X(k+n) = j | X(k) = i\} \tag{2.2}$$

para $\forall\, n \in \{0, 1, 2, 3, \dots\}$ e $i, j \in S$.

No caso particular quando $n = 1$, por conveniência de notação, a equação anterior resulta em:

$$p_{ij} = P\{X(k+1) = j | X(k) = i\} \tag{2.3}$$

As probabilidades de transição $p_{ij}^{(n)}$ precisam satisfazer as seguintes condições [7]:

$$p_{ij}^{(n)} \geq 0, \qquad \forall\, i, j, n \tag{2.4}$$

$$\sum_{j=0}^{M} p_{ij}^{(n)} = 1, \qquad \forall\, i, n \tag{2.5}$$

Pode-se definir uma matriz $P^{(n)}$ de probabilidades de transição com os elementos $p_{ij}^{(n)}$, num espaço de estados de 0 a M, em que $M = |S|$. A matriz $P^{(n)}$, exibida em (2.6), é denominada Matriz de Transição de Passo n. Quando $n = 1$, a matriz é chamada apenas Matriz de Transição [7].

$$P^{(n)} = \begin{bmatrix} p_{00}^{(n)} & \cdots & p_{0M}^{(n)} \\ \vdots & \ddots & \vdots \\ p_{M0}^{(n)} & \cdots & p_{MM}^{(n)} \end{bmatrix} \tag{2.6}$$

Uma Cadeia de Markov em Tempo Discreto pode ser ilustrada por meio de um grafo direcionado, no qual os círculos correspondem aos estados $X(k)$, e as setas partindo de cada estado indicam as probabilidades de transição de um estado no tempo t para o próximo estado no tempo $t + 1$, conforme ilustra a Figura 2.1 [2].

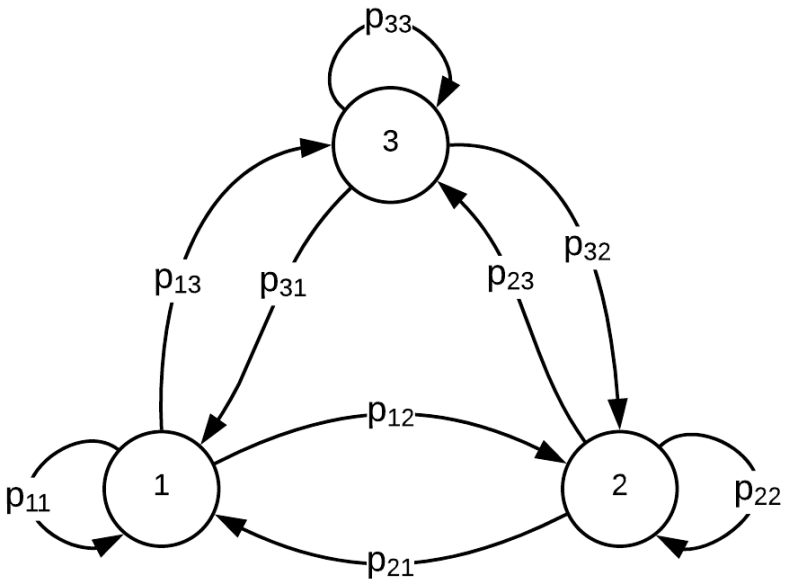


Fig. 2.1 - CMTD: estados e suas probabilidades de transição [2].

Considera-se, neste estudo, que a Cadeia de Markov possui um número finito de estados e seja estável, de modo que suas probabilidades de transição são estacionárias, isto é, não mudam em relação ao tempo. Para garantir tal condição, assume-se que a Cadeia de Markov empregada na representação do modelo analítico deve ser irredutível e recorrente positiva.

Uma Cadeia de Markov é dita irredutível se para todo $i, j \in S$ é possível migrar de qualquer estado i para qualquer estado j e vice-versa, ou seja, todos os estados comunicam entre si e compõem uma única classe. Um estado de uma Cadeia de Markov é recorrente se, partindo desse estado, há certeza (probabilidade 1) de retornar a ele em algum momento, o que implica que o estado será visitado infinitas vezes ao longo do tempo. Por fim, uma Cadeia de Markov é recorrente positiva se, além de ser recorrente, o tempo médio de retorno a cada estado é finito.

Essa característica recorrente da Cadeia de Markov permite uma abordagem matemática e computacional menos onerosa para o estudo e simulação do modelo analítico [2]. O Apêndice D de [5] apresenta essas definições de forma detalhada. No entanto, as redes ópticas operam e são mantidas de forma contínua, ou seja, as mudanças de estado ocorrem de maneira não periódica ao longo do tempo. Dessa forma, a próxima seção apresenta as características da Cadeia de Markov em Tempo Contínuo, necessárias para o embasamento deste estudo.

2.1.2 Cadeia de Markov em Tempo Contínuo

A modelagem analítica dos bloqueios de requisições em redes ópticas elásticas tem sido feita por meio de Cadeias de Markov em Tempo Contínuo. Uma CMTC é um processo estocástico $X(t)$ em que a variável temporal é contínua ($t \geq 0$). Neste ponto, são consideradas somente as CMTC's nas quais $X(t)$ tem valores inteiros e não negativos [4].

O tempo entre as mudanças de estado numa CMTC é distribuído exponencialmente, portanto, o processo considerado permanece constante por um período de tempo exponencial até migrar para o próximo estado. Uma Cadeia de Markov em Tempo Contínuo $X(t)$ é definida num espaço de estados S pela seguinte propriedade [4]:

$$P\{X(t+s) = j | X(v), v \leq t\} = P\{X(t+s) = j | X(t)\} \tag{2.7}$$

para $\forall\, t, s \geq 0$ e $\forall\, j \in S$.

Numa CMTC, de acordo a equação (2.7), a distribuição das probabilidades dos valores futuros do processo, $X(t+s)$, depende somente dos valores presentes $X(t)$, isto é, não está subordinada aos eventos passados $X(v)$, o que atende a Propriedade de Markov.

De forma geral, uma CMTC pode ser definida como um processo estocástico de tempo contínuo e estados discretos em conformidade com as propriedades a seguir [4]:

a) Quando um processo acessa um estado i, ele permanece nesse estado por um período de tempo que é exponencialmente distribuído com uma taxa λ_{ij}, antes de migrar para o próximo estado j;

b) Quando um processo passa diretamente do estado i ao estado j, o faz com probabilidade p_{ij}.

Neste ponto, como a Cadeia de Markov em questão é de tempo contínuo, é importante ressaltar que as probabilidades de transição podem ser expressas pelas taxas de transição $\lambda_{ij}, j \neq i$, que se relacionam com as probabilidades p_{ij} por meio da equação (2.8):

$$p_{ij} = \lambda_{ij} / \sum_{\substack{k=0 \\ k \neq i}}^{M} \lambda_{ik} \tag{2.8}$$

O conjunto das probabilidades de transição p_{ij} deve satisfazer as equações (2.4) e (2.5). A probabilidade de transição $p_{ij}(t)$ pode ser determinada segundo a equação (2.9) [4]:

$$p_{ij}(t) = P\{X(t+s) = j | X(s) = i\} \tag{2.9}$$

para $\forall\, t, s \geq 0$ e $\forall\, i \in S$.

Desse modo, pode-se representar uma matriz de transição $P(t)$, definida num espaço de estados que varia de 0 a M, sendo $M = |S|$, para $t \geq 0$, de acordo com (2.10) [4]:

$$P(t) = \begin{bmatrix} p_{00}(t) & \cdots & p_{0M}(t) \\ \vdots & \ddots & \vdots \\ p_{M0}(t) & \cdots & p_{MM}(t) \end{bmatrix} \tag{2.10}$$

Da mesma forma que a CMTD, é possível representar uma Cadeia de Markov em Tempo Contínuo por meio de um grafo direcionado; no entanto, as setas entre os estados são identificadas com as taxas de transição de um estado para o próximo, como ilustra a Figura 2.2.

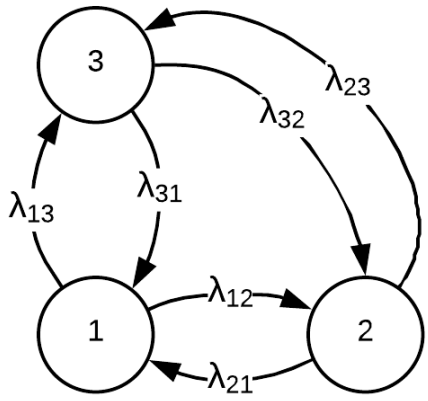


Fig. 2.2 - CMTC: estados e suas taxas de transição [2].

Uma CMTC possui um único vetor de distribuição de probabilidades e, para ser considerada estável, assim como na CMTD, precisa também ser irredutível e recorrente positiva (ou regular, no contexto de tempo contínuo). [2].

2.1.2.1 Reversibilidade

Nesta seção é apresentado o conceito de reversibilidade em processos de Markov.

Considere um processo estocástico de tempo contínuo $X(t)$, $t \in (-\infty, \infty)$. Tal processo será reversível se os valores da distribuição de probabilidades (de dimensão finita) forem idênticos à distribuição de probabilidades do processo reverso $R\{\tilde{X}(t), t \in (-\infty, \infty)\}$, em que $\tilde{X}(t) = X(\tau - t)$. Dessa forma, $X(t)$ e $X(\tau - t)$ devem ser estatisticamente idênticos para qualquer deslocamento arbitrário de tempo $\tau \in R$ [8].

Presuma que $X(t)$, $t \in (-\infty, \infty)$, seja uma Cadeia de Markov em Tempo Contínuo regular e estável, definida num espaço de estados S, e que $P(t)$ denota a matriz de probabilidades de transição para $t \geq 0$. Para esse processo de Markov, define-se ainda que π é uma distribuição estacionária de probabilidades, com $\pi = \pi_i$ para um dado estado i, $i \in S$ [4].

A equação (2.11) relaciona a probabilidade de um evento passado dados, respectivamente, o evento futuro e o evento presente [4]:

$$P\{X(t-\tau) = j | X(s), s \geq t, X(t) = i\} =$$

$$\frac{P\{X(t-\tau)=j\}.P\{X(s),s\geq t,X(t)=i|X(t-\tau)=j\}}{P\{X(s),s\geq t,X(t)=i\}} =$$

$$\frac{P\{X(t-\tau)=j\}.P_{ji}(\tau).P\{X(s),s\geq 0|X(0)=i\}}{P\{X(t)=i\}.P\{X(s),s\geq 0|X(0)=i\}} =$$

$$\frac{\pi_j}{\pi_i} P_{ji}(\tau) \tag{2.11}$$

em que a segunda igualdade aplica a Propriedade de Markov. Desse modo, uma CMTC estacionária e reversível apresenta uma matriz de probabilidades de transição homogêneas no tempo, $P(t)$, na qual as probabilidades são definidas em (2.12) para $t \geq 0$ [4]:

$$\tilde{p}_{ij}(t) = \frac{\pi_j}{\pi_i} p_{ji}(t) \tag{2.12}$$

Ao derivar a equação anterior em $t = 0$, é possível relacionar a matriz das taxas de transição Q da Cadeia de Markov original com a matriz $\tilde{Q}$ da Cadeia de Markov reversa no tempo, conforme (2.13) [4]:

$$\tilde{q}_{ij} = \frac{\pi_j}{\pi_i} q_{ji} \tag{2.13}$$

Assim, uma CMTC regular é reversível se, e somente se, existir uma distribuição de probabilidade π em S para $\forall\, i,j \in S$. A condição anterior exige obter, ou supor, a distribuição estacionária de probabilidades π [2].

O critério de Kolmogorov enuncia que a reversibilidade de uma CMTC depende somente das taxas de transição da Cadeia de Markov considerada. Segundo esse critério, uma CMTC irredutível e recorrente positiva é reversível se, e somente se, o produtório das taxas de transição em um sentido for igual ao produtório das taxas de transição no sentido oposto, para qualquer percurso fechado de estados [2].

Em (2.14), são apresentadas as equações de balanço total empregadas para obter o vetor π de probabilidades estacionárias [4]:

$$\sum_j \pi_j q_{ji} = \pi_i \sum_j q_{ij} \tag{2.14}$$

para $\forall\, i,j \in S$.

Dessa forma, dado que uma CMTC $X(t)$ é reversível, então para os dois estados considerados, $i,j \in S$, os valores de q_{ij} e q_{ji} são, ambos, iguais a zero ou positivos [4].

2.2 TRÁFEGO POISSONIANO

Até o início da Internet, os modelos mais usados na Engenharia de Tráfego eram baseados em processos poissonianos. Tais modelos, chamados modelos Markovianos, determinam que as chegadas de requisições dos clientes satisfazem a um processo de Poisson e o tempo de serviço apresenta uma distribuição exponencial [9].

Por meio dos modelos Markovianos, Agner Krarup Erlang (1878-1929), famoso matemático dinamarquês, elaborou um conjunto de equações que são empregadas para dimensionar as redes de telecomunicações: Erlang-B, usada para calcular a probabilidade de bloqueio das chamadas em um sistema M/M/S/S; e Erlang-C, utilizada em sistemas não bloqueantes para calcular o tempo de espera de uma requisição numa fila antes de ser atendida. Esses modelos, que são usados até hoje no planejamento de capacidade em redes de comutação de circuitos, dependem somente da carga oferecida ao sistema, $\rho = \lambda/\mu$, em que λ é a taxa de chegadas das requisições e $1/\mu$ é o tempo de serviço esperado [9].

Nos sistemas com fio, os recursos da rede precisam ser estimados para atender o tráfego agregado gerado por um determinado número de fontes de maneira que a probabilidade

de bloqueio de uma requisição seja inferior ao estipulado no planejamento da rede. Quando uma fonte está dedicada a uma chamada, ela não gera novas chamadas, o que faz com que o tráfego gerado por essa fonte não seja poissoniano. Entretanto, quando o número de fontes tende ao infinito e a taxa de requisições de cada fonte tende a zero, resultando numa taxa agregada finita, o tráfego se aproxima assintoticamente do modelo poissoniano [10].

Esse pressuposto é o cerne do modelo de Erlang para estimar a probabilidade de bloqueio dos sistemas de comutação de circuitos [5]. A aplicação de tal hipótese era válida nas redes públicas de telefonia comutada do século passado e, com isso, a adoção do modelo de Erlang tornou-se o alicerce do processo de planejamento e dimensionamento das redes com fio na época [10].

Em redes ópticas de comutação de circuitos, as fontes geradoras de tráfego requisitam recursos espectrais das fibras, denominados *slots*, para suportar as chamadas. Dessa forma, nas redes ópticas, os fios de cobre são substituídos por fatias do espectro óptico que, contudo, desempenham a mesma função sistêmica. Apesar disso, o modelo de Erlang permaneceu popular por causa da sua simplicidade. Contudo, as suposições de que há um número muito grande de fontes que se mantêm inativas na maior parte do tempo, fizeram com que o modelo de Erlang passasse a ser questionado [10].

Assim, o modelo elaborado pelo matemático norueguês *Tore Olaus Engset* (1865-1943), que considera um número finito de fontes com atividade frequente, é mais realista e compatível com a abordagem analítica em sistemas ópticos. Esses dois modelos vêm sendo extensamente utilizados e refinados sob diferentes pressupostos referentes ao tráfego oferecido e, no caso do modelo de Engset, às funcionalidades das fontes [10]. As próximas seções detalham ambos os modelos.

2.2.1 Modelo de Erlang

A utilização do modelo de tráfego de Erlang para projetar as redes de comunicação de dados foi muito difundida durante o século passado, período no qual houve grande expansão da infraestrutura física das redes públicas de telefonia comutada. Os terminais telefônicos empregados nesse tipo de rede permaneciam ociosos, em grande número e na maior parte do tempo, conectados a uma central telefônica [10].

Para que esses terminais pudessem se comunicar com telefones conectados em outras centrais, os M telefones atendidos por uma central, ao realizarem uma chamada, deviam

disputar o acesso a um dos k canais de um tronco. Com $k < M$, existia a possibilidade de bloqueio da chamada e, então, era necessário dimensionar k para que a probabilidade desse bloqueio pela rede fosse suficientemente pequena em comparação com a probabilidade de o número chamado estar ocupado [10].

Na rede de telefonia pública, os telefones operam como fontes geradoras de tráfego. Quando uma fonte está ocupada, ela não origina novas chamadas. Apenas quando essa fonte está livre são geradas novas chamadas com uma taxa $\hat{\lambda}$ que, para efeitos de simplificação analítica, é considerada a mesma para todas as fontes. Logo, quando m fontes estão ativas num tronco, ou seja, ocupando m dos k canais, o tráfego originado pelas M fontes, em que M representa o número total de telefones, é dado por $(M - m)\hat{\lambda}$ [10].

Se $M \gg k \geq m$, a expressão $(M - m)\hat{\lambda}$ pode ser aproximada para $M\hat{\lambda}$. Essa aproximação é a sustentação do modelo de Erlang, que caracteriza um sistema no qual a taxa de chegada de novas chamadas é independente do número de canais ativos $m < k$, como mostra a Figura 2.3. A taxa de chegada de novas requisições por canal, representada por λ, é constante, ou seja, não depende do estado do sistema [10].

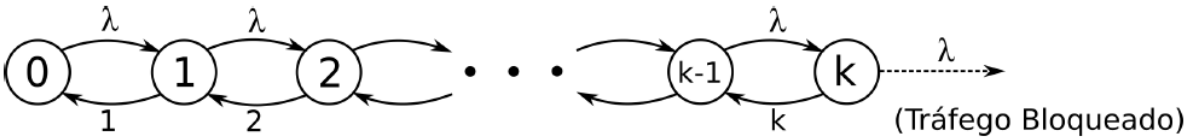


Fig. 2.3 - Representação do modelo de Erlang [10].

O modelo de Erlang despreza o número de fontes M e a taxa de requisições por fonte $\hat{\lambda}$, e considera somente o produto entre elas, que é definido como uma taxa λ independente do estado e identificada com o tráfego oferecido ao sistema [10].

A condição $M \gg k \geq m$ era válida nas antigas redes de telefonia pública, pois o número de fontes era muito maior que o de canais disponíveis para longas distâncias, e o tempo médio entre chamadas por fonte, igual a 1, era muito maior que a duração média de um telefonema (em torno de 3 minutos, pois a tarifa era cobrada por minuto), fazendo com que $\hat{\lambda} \ll 1$ quando expresso em erlangs [11].

Além disso, o modelo de Erlang é mais simples que o de Engset, por não depender do número de fontes M. Tais fatores fizeram com que o modelo de Erlang fosse amplamente empregado no desenvolvimento da Teoria do Tráfego e, principalmente, para o dimensionamento dos recursos físicos da rede, que são necessários para garantir um certo grau de desempenho da mesma [10].

2.2.2 Modelo de Engset

O modelo de Erlang pode ser considerado um limite assintótico do modelo de Engset quando o número de fontes tende ao infinito e o tráfego gerado por cada fonte tende a zero de tal modo que o tráfego oferecido ao sistema se mantém finito. Por causa disso, as expressões que calculam a probabilidade de bloqueio de ambos os modelos compartilham os mesmos problemas numéricos, que são relacionados a ocorrência de fatoriais de números que podem ser grandes em função do tamanho de cada sistema [10]. Para evitar esses problemas, foram derivadas fórmulas recursivas que são discutidas em [12] e usadas em [10] para uma comparação entre os dois modelos.

A Figura 2.4 mostra uma Cadeia de Markov gerada por um tráfego do tipo $(M-m)\hat{\lambda}$, comumente chamado de tráfego "on-off", num sistema com $k < M$ canais. Essa cadeia representa o modelo de Engset do sistema markoviano, no qual a taxa de chegada de novas requisições depende do estado m atual. O estado do sistema é caracterizado apenas pelo número $m \in \{0, 1, 2, \dots, k\}$ de canais ativos [10].

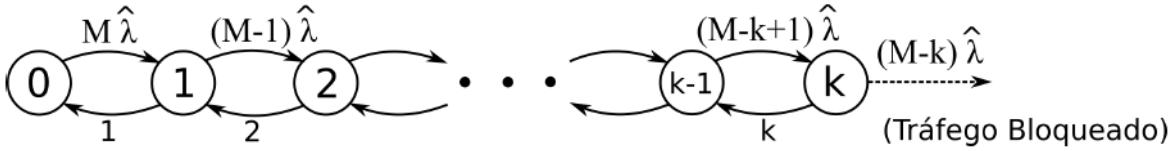


Fig. 2.4 - Representação do modelo de Engset [10].

Quando a análise é transportada para os sistemas modernos, principalmente no contexto da *backbone* da infraestrutura física da Internet, o uso do modelo de Erlang não é válido, pois seus pressupostos podem não podem ser aplicados [10].

Se $M \leq k$, é possível alocar um canal permanentemente para servir cada uma das M fontes, zerando a probabilidade de bloqueio para qualquer taxa $\hat{\lambda}$ de chegada de requisições por fonte. Assim, com cada fonte podendo dispor de um canal privativo para as suas chamadas, não há mais bloqueios.

Com $M > k$, o dimensionamento baseado no modelo de Engset é obtido a partir do sistema markoviano mostrado na Figura 2.4, no qual a intensidade do tráfego depende do estado em que o sistema se encontra. No único estado bloqueante, que continua sendo o estado k, a taxa de chegada de requisições é $\hat{\lambda}(M-k)$. Então, a taxa de bloqueios é dada por $\hat{\lambda}(M-k)p_k$ [10].

3 FRAGMENTAÇÃO ESPECTRAL EM REDES ÓPTICAS ELÁSTICAS

Na operação de uma rede óptica ocorrem diferentes tipos de eventos, como as chegadas de novas requisições, que podem ser aceitas ou rejeitadas, e o término das conexões atualmente alocadas. A liberação de conexões deixa *slots* não alocados, que podem ser reaproveitados por novos usuários da rede. Esse processo de nascimento de "buracos" no espectro é denominado fragmentação espectral [1].

Para a alocação de usuários com requisitos heterogêneos de banda, pode haver a formação de vacâncias que não possam ser ocupadas por uma ou mais classes de acesso, conduzindo a perdas por fragmentação que podem prejudicar grande parte dos ganhos de eficiência espectral promovidos pela rede óptica elástica [1]. Por essa razão, vários esquemas foram propostos para lidar com a fragmentação espectral e as perdas associadas [13][14]. A maioria desses esforços se concentrou em métricas propostas para a fragmentação espectral [15][16] e esquemas para minimizar tais métricas.

As perdas de fragmentação horizontal causam a incapacidade de atender a uma solicitação de conexão em um caminho de vários enlaces nos quais existem recursos disponíveis, mas eles não estão alinhados para fornecer a continuidade do espectro. Tais perdas já estavam presentes nos sistemas WDM na forma da restrição de continuidade do comprimento de onda, mas agora se tornaram mais rigorosas em redes elásticas sob a forma de uma restrição de continuidade do espectro [4].

As perdas de fragmentação vertical estão presentes em um único enlace se o tráfego for composto por duas ou mais classes que solicitam diferentes números de *slots* por requisição. Se apenas uma dessas classes estiver presente, então algoritmos como o *first-fit* [17] são conhecidos por evitar as perdas por fragmentação, mesmo enquanto produzem fragmentação. A explicação desse paradoxo é que os tamanhos dos vazios produzidos pelo algoritmo correspondem exatamente ao tamanho único das requisições ou a um múltiplo dele [4].

É importante destacar que as perdas por fragmentação horizontal já estavam presentes nas redes ópticas baseadas em grade fixa, ao passo que as perdas por fragmentação vertical surgiram no contexto das redes com grade flexível [4].

O cerne da motivação deste trabalho está na possibilidade de promover esse mesmo tipo de casamento entre as requisições e os vazios promovidos pelas terminações, a fim de reduzir as perdas de fragmentação no caso de múltiplas classes de tráfego.

3.1 O SISTEMA DESFRAGMENTADO COMO REFERÊNCIA

Num cenário de rede dinâmica, em que as conexões são estabelecidas e encerradas de forma aleatória por decisão dos usuários, que não estão cientes do estado da rede, os recursos espectrais se tornam altamente fragmentados e "lacunas" são inevitavelmente introduzidas no espectro levando ao problema da fragmentação espectral.

Esse problema é muito semelhante ao da fragmentação de memória na arquitetura de computadores: ao longo do tempo e com a alocação e a liberação de memória, ela torna-se fragmentada em áreas contíguas menores e blocos de dados não podem ser alocados nessas áreas, mesmo que haja memória livre geral suficiente. Da mesma forma, a fragmentação do espectro também leva a uma utilização ineficiente dos recursos disponíveis e à degradação do desempenho da rede [14].

Em um sistema desfragmentado, que neste estudo é considerado como o "*benchmark*" (caso de referência), a fragmentação é inexistente, pois todos os recursos disponíveis estão agrupados de forma contígua. A ocorrência de bloqueios em novas requisições é atribuída, quando ocorre, à insuficiência de recursos espectrais [14].

Considerando que a utilização total dos recursos espectrais é a essência da rede elástica, a fragmentação espectral representa o principal obstáculo para alcançar plenamente esse objetivo. Ao ativar uma nova conexão, a fragmentação espectral pode ser evitada atribuindo seus *slots* de forma contígua aos *slots* ativos de outras conexões. No entanto, ao encerrar uma conexão, pode ser deixado um vazio espectral que contribuirá para tornar o sistema fragmentado [14].

No caso específico da fragmentação espectral, ocorre também um aumento da probabilidade de bloqueio pois os *slots* de frequência não utilizados estão "espalhados" pelo espectro e não há *slots* suficientes disponíveis de forma contígua para atender os novos pedidos de conexões [14].

Com o intuito de reduzir esse problema, são investigados na comunidade científica mecanismos dedicados à desfragmentação de espectro. O propósito desses mecanismos é reorganizar os caminhos de luz existentes, a fim de liberar espaço para demandas bloqueadas. Como o processo de rearranjo, geralmente, envolve o redirecionamento ou a realocação espectral das conexões, as técnicas de desfragmentação podem requerer um tempo significativo para convergir. Portanto, um dos requisitos operacionais fundamentais é evitar interrupções no serviço durante a fase de reconfiguração, ou pelo menos minimizar os seus efeitos [14].

Sendo assim, a desfragmentação constitui-se como um desafio, tanto em termos de complexidade quanto em relação ao tempo de serviço, pois requer o lançamento e a coordenação de diversas operações paralelas para reorganizar o espectro e liberar espaço para a entrada de novos pedidos [14].

A principal estratégia empregada pelos alocadores para gerenciar a fragmentação é a política de posicionamento, que consiste na seleção da localização mais adequada para acomodar uma requisição dentre os vazios disponíveis. Embora seja impossível garantir uma política de posicionamento ideal que acomode todas as requisições sem causar fragmentação futura, os alocadores buscam, de maneira heurística, se aproximar desse ideal [18].

Um exemplo de heurística encontrada na literatura para alocação do espectro, que geralmente promove uma utilização eficiente dos recursos, é a estratégia conhecida como *best-fit*. Essa heurística realiza uma busca em todo o espaço espectral disponível e aloca a requisição de conexão no menor bloco capaz de satisfazer suas demandas [18].

Em redes WDM legadas de grade fixa, as perdas decorrentes da fragmentação espectral não ocorrem em um único enlace e podem ser evitadas em redes *multilink* por meio do uso da conversão de comprimento de onda. No entanto, essa abordagem não é viável em redes ópticas elásticas, que exigem que os *slots* de conexão sejam contíguos. Portanto, é necessário fornecer funcionalidades de desfragmentação em EONs para que possam atingir seu objetivo principal [19].

3.2 O SISTEMA FRAGMENTADO

No âmbito das redes ópticas, a questão fundamental é estabelecer conexões por meio do roteamento e alocação dos recursos espectrais, a fim de atender às demandas de tráfego. Nesse contexto, o problema tradicional da Alocação de Comprimento de Onda (RWA - *Routing and Wavelength Assignment*) utilizado em Redes de Multiplexação por Divisão de Comprimento de Onda (WDM - *Wavelength Division Multiplexing*) é redefinido para as Redes Ópticas Elásticas como o problema do Roteamento e Alocação de Espectro (RSA - *Routing and Spectrum Assignment*). Em redes ópticas elásticas, o espectro é considerado um recurso contínuo, que é subdividido em unidades de frequência de tamanho fixo chamadas *slots*, cuja capacidade de transmissão, atualmente 12,5 GHz, é menor do que a de uma grade típica em redes WDM convencionais, que é de 50 GHz em sistemas considerados densos (DWDM - *Dense Wavelength Division Multiplexing*) [18].

A restrição de continuidade do comprimento de onda (*Wavelength Continuity Constraint*) presente no problema da Alocação de Comprimento de Onda é transformada na restrição de continuidade do espectro (*Spectrum Continuity Constraint*) no problema do Roteamento e Alocação de Espectro. Dessa forma, ao estabelecer uma conexão, um caminho óptico é criado entre um par de nós de origem e destino, utilizando um ou mais *slots* de espectro [18].

Esta seção apresenta uma contextualização teórica sobre roteamento e alocação espectral ou, simplesmente, algoritmos RSA. Além disso, a seção trata do problema da fragmentação espectral no enlace avulso e destaca as topologias com múltiplos enlaces.

3.2.1 Algoritmos RSA

Sob um cenário de tráfego dinâmico, o problema RSA deve ser tratado de forma particionada em abordagens de roteamento e alocação de espectro com resolução de forma sequencial. No roteamento, um dos métodos mais utilizados é pré-calcular os k caminhos mais curtos que serão utilizados na seleção da rota e escolher, a priori, o caminho que oferece a menor taxa de fragmentação conforme uma métrica pré-definida.

Porém, os algoritmos dos k caminhos mais curtos não consideram a funcionalidade dos vazios e o perfil do tráfego. Após o roteamento, a alocação do espectro pode ser realizada com heurísticas. A mais utilizada na literatura, visando minimizar as métricas de fragmentação, é a *first-fit* [20]. Com o emprego de técnicas de Inteligência Artificial e Aprendizagem de Máquina, novas abordagens na literatura investigam o "casamento" dos tamanhos dos vazios disponíveis com o perfil do tráfego.

3.2.1.1 Roteamento

A definição do problema do roteamento é de suma importância no âmbito do gerenciamento de redes e, apesar de não ser o foco dos resultados apresentados neste estudo, pode exercer influência significativa nos desfechos obtidos. Tal como ocorre nas redes WDM tradicionais, nas redes ópticas elásticas, a etapa do roteamento assume a finalidade de determinar um caminho entre o nó de origem e o nó de destino da transmissão, por meio da utilização de uma ou várias subportadoras consecutivas [2].

O problema do roteamento pode ser abordado como o desafio de encontrar o menor caminho em um grafo. Neste contexto, os nós do grafo representam as interfaces de conexão da rede, nas quais o tráfego pode ser multiplexado e demultiplexado para a distribuição em redes de menor escala. Por sua vez, as arestas do grafo correspondem aos enlaces da rede [2]. A literatura já apresentou diversas técnicas para a resolução deste problema, sendo o algoritmo de Dijkstra [21] uma das soluções comumente empregadas, conforme ilustrado por [22].

Outra abordagem mencionada por [23] é a utilização do roteamento de múltiplos caminhos (*multi-path routing*). Neste método, ao lidar com uma requisição de banda B, o algoritmo de roteamento deve encontrar $N \geq 2$ caminhos independentes em enlaces com banda agregada de, no mínimo, B. Dessa maneira, o algoritmo distribui igualmente a banda entre os caminhos encontrados.

3.2.1.2 Alocação Espectral

A alocação espectral estabelece como as requisições dos recursos de frequência são distribuídas ao longo do espectro eletromagnético. Nos sistemas de transmissão óptica convencionais, especificamente nas redes WDM, a faixa operacional do espectro óptico frequentemente está situado na região conhecida como Banda C, com comprimentos de onda entre 1530 nm e 1565 nm, o que corresponde ao intervalo de frequências entre 191,7 e 196 THz [1].

Nesse contexto, o espectro da banda de interesse é subdividido em blocos de tamanho uniforme no domínio da frequência, destinados a serem alocados para estabelecer conexões entre nós na rede. As dimensões desses blocos são estabelecidas conforme diretrizes de padronização. A União Internacional de Telecomunicações (ITU), por intermédio do Setor de Padronização de Telecomunicações (ITU-T), normatizou as redes WDM com canais de transmissão de 50 GHz. Cada um desses canais é associado a um comprimento de onda ou, de outro modo, cada comprimento de onda define um canal, como mostra a Figura 3.1 [1].

Fig. 3.1 - Redes WDM: espectro óptico subdivido em canais de 50 GHz [2].

Assim, as solicitações de conexão precisam ser tratadas por algoritmos de alocação espectral, que determinarão a posição das conexões no espectro dentro dos blocos ainda não alocados para outras conexões. É importante ressaltar que os blocos de frequência alocados devem ser uniformes em todos os enlaces da rede, seguindo um critério de consistência conhecido como restrição de continuidade do espectro [1].

Com a evolução para o paradigma das redes ópticas flexíveis, a ITU-T padronizou a subdivisão do espectro em unidades menores, denominadas *slots*, cada uma com largura de banda de 12,5 GHz. Cada *slot* do espectro é associado a uma ou mais subportadoras no contexto da modulação óptica, permitindo a alocação eficiente dos sinais para transmissão. Ao contrário das redes WDM tradicionais, nas redes ópticas elásticas, as conexões podem ocupar múltiplos *slots*, dependendo da demanda de transmissão, desde que esses *slots* sejam adjacentes no espectro, observando-se a restrição de contiguidade do espectro, conforme a Figura 3.2 [1][2].

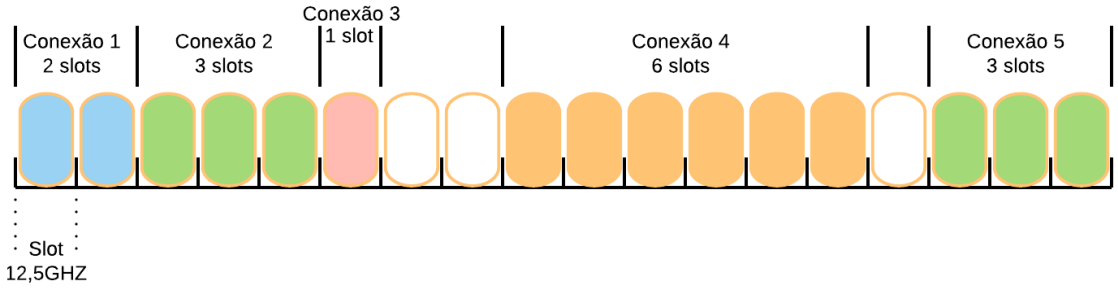
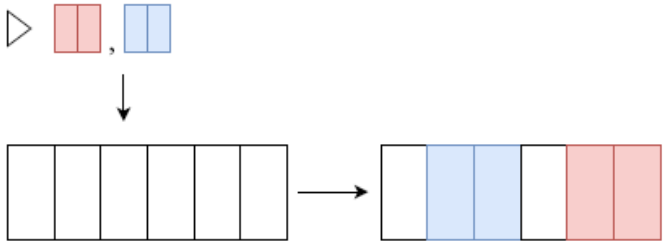
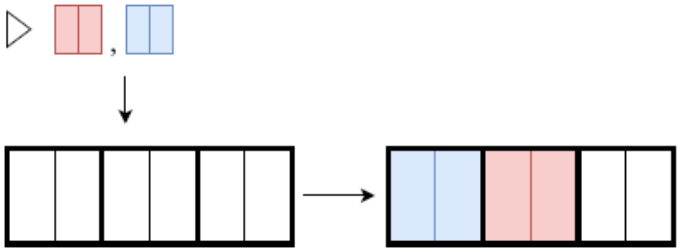


Fig. 3.2 - Rede flexível com conexões de diferentes larguras de banda concomitantes [2].

Na operação de uma rede óptica, observa-se não apenas o estabelecimento de novas conexões, mas também a liberação de recursos anteriormente alocados devido ao término das conexões. A conclusão dessas conexões implica na disponibilização de *slots* no espectro de frequência, que podem ser reutilizados para acomodar novos usuários na rede. Para a alocação de usuários com requisitos de tráfego não uniforme, surge a formação de intervalos que não poderão ser ocupados por nenhuma classe de acesso, conforme ilustrado na Figura 3.3a, gerando assim perdas devido à fragmentação espectral [1].



(a)



(b)

Fig. 3.3 - Alocação espectral para tráfego uniclasse com algoritmo aleatório (a) e *first-fit* (b) [1].

Contudo, em cenários nos quais todos os clientes possuem os mesmos requisitos de ocupação espectral, o que é denominado tráfego uniclasse, e um algoritmo de alocação espectral posiciona essas conexões em localizações predefinidas, como mostrado na Figura 3.3b, não se observam perdas por conta da fragmentação. Na medida em que uma nova conexão for estabelecida, ela sempre pode ocupar o espaço deixado pela liberação de uma conexão anterior. Todavia, a uniformidade entre as distintas classes de acesso (ou seja, diferentes tipos de clientes) pode não ser mantida em ambientes em que a rede opera no paradigma elástico [1].

Em tais circunstâncias, se os clientes da rede não solicitarem quantidades uniformes de *slots* por conexão, o que caracteriza um tráfego multiclasse, a fragmentação pode desencadear situações nas quais a chegada de novas requisições seja bloqueada, culminando na degradação do desempenho devido às perdas por fragmentação, conforme discutido por Waldman [24]. A Figura 3.4 ilustra uma situação de fragmentação resultante num cenário de tráfego multiclasse.

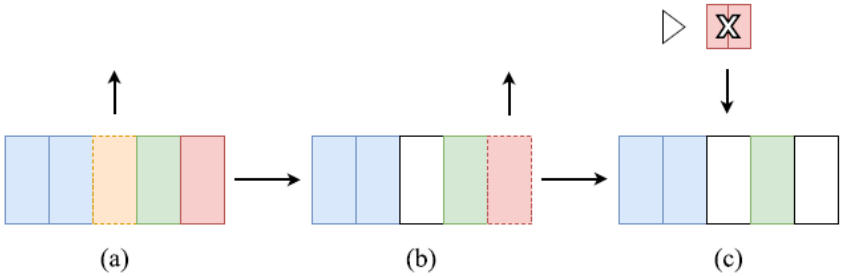


Fig. 3.4 - Fragmentação do espectro em redes elásticas em três estados: (a) inicial; (b) intermediário; (c) final [1].

Inicialmente, o sistema se encontra no estado (a) com quatro conexões ativas, uma que ocupa dois *slots* e três conexões de um slot. Uma das conexões de um *slot* é finalizada, o que conduz o sistema ao estado intermediário (b). Na sequência, outra conexão de um *slot* também é encerrada, resultando no estado final (c). Nesta situação, caso seja solicitado uma conexão que requer dois *slots*, ela será bloqueada devido à inexistência de dois *slots* contíguos disponíveis para a sua acomodação.

Portanto, outra responsabilidade atribuída ao algoritmo de alocação espectral é evitar cenários como o exemplificado na Figura 3.4. Tal propósito pode ser alcançado mediante a adaptação dos algoritmos de alocação, visando minimizar as probabilidades de o sistema encontrar-se em estados bloqueantes devido à fragmentação. Uma segunda abordagem, mais robusta, mas caracterizada por sua complexidade computacional e custo operacional, envolve a desfragmentação do espectro, como já mencionado anteriormente [1].

Diversas técnicas de alocação espectral foram propostas na literatura para atender às demandas das redes elásticas, conforme sumarizado no tutorial apresentado por Tomkos [14]. Em um estudo conduzido por Waldman [24], foi mostrado que, em um enlace único, qualquer algoritmo de alocação baseado em uma estratégia voraz (que maximize a ocupação imediatamente após a alocação) apresenta o mesmo desempenho quando a desfragmentação do espectro é empregada.

Entre vários algoritmos vorazes, um exemplo elementar é o algoritmo conhecido como *first-fit*. Essa metodologia de alocação se fundamenta na disposição das conexões o mais próximo possível do "início" (ou "fim") do espectro disponível, ou seja, nos comprimentos de onda de menor (ou maior) valor, mantendo, assim, as conexões agrupadas em um segmento específico. Por conseguinte, sempre que uma nova conexão é requisitada, ela é alocada nos primeiros *slots* contíguos desocupados, na ordem de frequências crescentes, que apresentem a quantidade adequada para acomodar a conexão da classe solicitante [1]. A Figura 3.5 apresenta um exemplo que demonstra a operação do referido algoritmo.

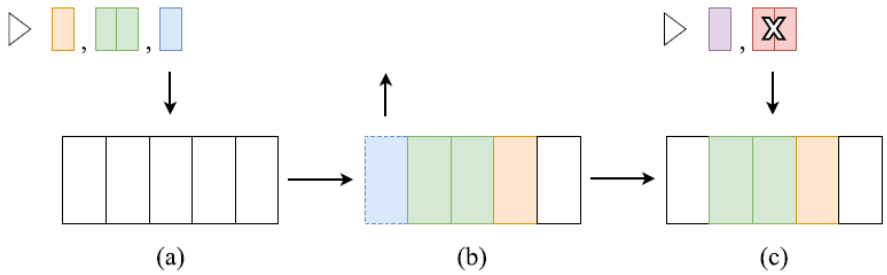


Fig. 3.5 - Algoritmo de alocação *first-fit* em três estados: (a) inicial; (b) intermediário; (c) final [1].

O sistema inicia a sua operação com todo o espectro livre (a). A seguir, o sistema recebe solicitações de requisições provenientes de duas classes de acesso, uma que demanda um *slot* e outra que exige dois. O algoritmo realiza as alocações de acordo com a ordem de chegada, posicionando as conexões nos *slots* de menor índice, conforme indica o estado intermediário (b). Em seguida, a primeira conexão alocada é encerrada, o que resulta na liberação de um *slot* no início do espectro, como mostrado no estado final (c).

No último estado, se chegar uma requisição de dois *slots*, ela será recusada pelo algoritmo, apesar da disponibilidade de *slots* no espectro. Esse impasse se origina do fato de que esses dois *slots* disponíveis não estão adjacentes. Porém, uma requisição para um único *slot* será alocada sem impedimentos.

Esse exemplo ilustra dois desafios intrínsecos à abordagem adotada por esse algoritmo. O primeiro é a ocorrência de perdas por fragmentação, evidenciada no estado (c), o que aumenta o número de requisições bloqueadas e reduz a ocupação efetiva da rede. O segundo é que o algoritmo favorece a menor classe, dado que essa classe é capaz de preencher as lacunas de forma mais conveniente em relação à outra. A essa preferência é atribuído o termo "parcialidade" (*unfairness*), enquanto a ausência de tal viés é denominada "imparcialidade" (*fairness*). A parcialidade da rede não é desejável, pois ela prejudica majoritariamente tráfegos de bandas maiores que, via de regra, utilizam o espectro de forma mais eficiente e são economicamente interessantes [25].

É papel do algoritmo de alocação espectral evitar, ou ao menos minimizar, os bloqueios de requisição. Há vários tipos de algoritmos de alocação que se diferenciam quanto a complexidade, probabilidade de bloqueio e razão de *slots* contíguos disponíveis no espectro em sua operação [1]. Os principais algoritmos de alocação espectral encontrados na literatura são descritos a seguir [15]:

a) *first-fit*: nesta estratégia, como já ilustrado anteriormente, os *slots* são indexados e é mantida uma lista dos índices disponíveis e já utilizados. Quando há uma nova requisição, o algoritmo faz a alocação no menor índice que consiga acomodar a transmissão. Apresenta uma baixa complexidade computacional para ser implementado;

b) *last-fit*: de forma similar ao *first-fit*, este algoritmo aloca a conexão no primeiro espaço disponível a partir da extremidade superior do espectro;

c) *first-last-fit*: atua como uma combinação dos dois algoritmos anteriores. Nesta estratégia é escolhido para a alocação o conjunto de *slots* disponíveis mais próximos de qualquer das duas extremidades do espectro. Assim, espera-se que haja mais *slots* disponíveis contíguos do que o obtido com as estratégias *first-fit* ou *random-fit*;

d) *random-fit*: escolhe de modo aleatório um bloco de *slots* suficientes para atender a conexão requisitada e faz a alocação;

e) *least used*: o algoritmo faz a alocação nos *slots* disponíveis que tenham sido usados no menor número de enlaces da rede. Caso haja um empate, então aloca-se no de índice mais baixo. Esse algoritmo é conveniente se for de interesse distribuir a carga de tráfego por todo o espectro;

f) *most used*: de forma oposta ao algoritmo anterior, a alocação é realizada nos *slots* disponíveis que tenham sido usados no maior número de enlaces da rede. Do mesmo modo, caso haja mais de um bloco que atenda ao requisito, aloca-se no de índice mais baixo. Nesse caso, prioriza-se o máximo reuso do espectro;

g) *exact-fit*: partindo do começo do espectro, este algoritmo procura pelo espaço vazio que atenda exatamente ao número de *slots* requeridos pela conexão. Caso não haja um bloco exato, pode-se usar outro algoritmo para realizar a alocação. Dessa forma, busca-se reduzir a fragmentação do espectro;

h) *min void*: este algoritmo, entre os espaços vazios disponíveis, aloca no menor;

i) *max void*: em oposição ao anterior, aloca no maior espaço disponível.

Os algoritmos de alocação variam em desempenho e funcionamento. É importante, então, levar em consideração sua aplicação para avaliação.

3.2.2 Fragmentação no Enlace Avulso

Em redes ópticas elásticas, o gerenciamento da rede deve estar preparado para enfrentar novos tipos de fragmentação [26]. Nas redes WDM ocorre um determinado tipo de fragmentação em função da necessidade de garantir a continuidade do mesmo comprimento de onda em todos os enlaces que compõem uma rota de uma transmissão. Essa é a chamada Fragmentação Horizontal, na qual conexões já estabelecidas entre os diversos nós da rede impedem que seja originada uma nova conexão, pois o mesmo bloco de espectro pode estar indisponível ao longo

de enlaces sucessivos de um caminho, apesar de cada enlace ter espectro suficiente para alocar a solicitação [2].

Em redes ópticas elásticas, num mesmo ambiente espectral, começam a coexistir transmissões de tamanhos diferentes de modo que o sistema passa a apresentar também fragmentação dentro de um enlace avulso (*single link*). A Fragmentação Vertical cria lacunas no espectro, que apesar de vacantes, estão isoladas e podem não ser suficientes para acomodar um determinado tamanho de conexão. Isso não ocorre em redes WDM porque todas as conexões ocupam a mesma parcela do espectro e, como as faixas são fixas e uniformes, o tráfego tem apenas uma classe. Dessa forma, qualquer requisição pode ser alocada na rede em qualquer faixa disponível [2].

A Figura 3.6 apresenta os dois tipos de fragmentação. Uma rede A-B-C recebe uma requisição de quatro *slots* para conexão entre os nós A e C. O *link* A-B tem quatro *slots* disponíveis, mas eles não estão contíguos em função de uma estratégia ineficiente de ocupação espectral, o que não atende ao critério da restrição de contiguidade do espectro (fragmentação vertical). Já o *link* B-C possui quatro *slots* disponíveis e adjacentes, mas esse bloco não tem correspondência na mesma posição em toda a rota e, portanto, esse cenário não atende ao critério da restrição de continuidade do espectro (fragmentação horizontal).

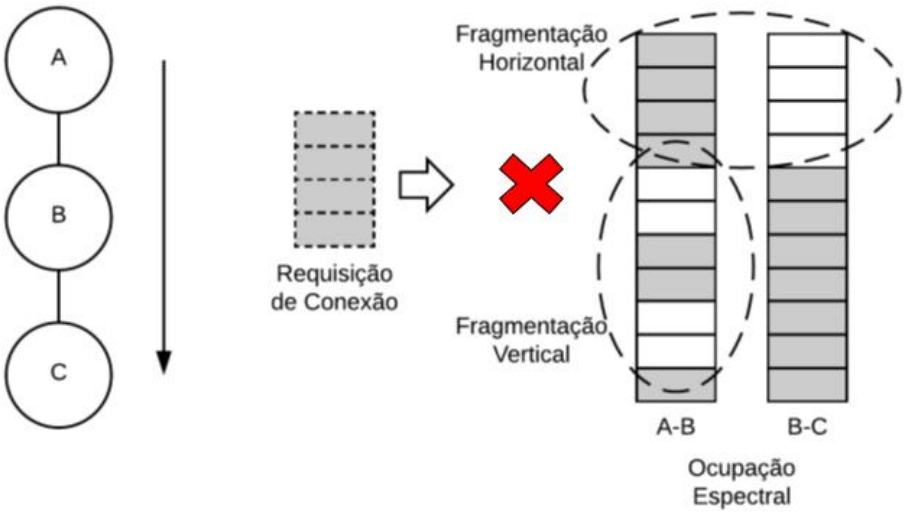


Fig. 3.6 - Fragmentação em uma rede elástica [2].

Duas abordagens podem ser adotadas para tratar o problema da fragmentação: (i) impedir que a fragmentação ocorra ou (ii) corrigir a fragmentação após a sua ocorrência. Os algoritmos RMSA (*Routing, Modulation and Spectrum Allocation*) têm sido aprimorados para otimizar a capacidade da rede, reduzindo assim a fragmentação e a probabilidade de bloqueio.

Diversos mecanismos que corrigem a fragmentação após a sua ocorrência, conhecidos como desfragmentadores, também foram propostos na literatura [2]. O foco deste estudo é uma rede de enlace avulso (*single link*), ou seja, que não apresenta a fragmentação horizontal. Em redes transparentes com topologia em malha (*mesh networks*), as soluções obtidas para o enlace avulso seriam válidas também para uma rota com múltiplos enlaces desde que esta não recebesse interferência de outras rotas. Na prática, porém, essa situação não é usual no ambiente metropolitano porque o compartilhamento de recursos espectrais entre as rotas, o que configura a interferência entre elas, é a própria razão de ser das redes metropolitanas.

Dessa forma, para que os efeitos da fragmentação vertical sejam isolados, é preciso que a rede siga uma topologia de enlace único, ou seja, só tenha dois elementos multiplexadores ligados por uma única fibra. Cada conexão será bidirecional e ocupará uma quantidade de *slots* igual à definida para sua classe de acesso [1].

O bloqueio de novas requisições pode ter dois precedentes: a falta de recursos espectrais disponíveis ou a fragmentação do espectro gerada pelas conexões já alocadas. A primeira causa deve ser evitada com o dimensionamento adequado do espectro oferecido para o padrão de tráfego esperado no sistema [1]. Existem diversas formas de tratar o problema como, por exemplo, a modulação por divisão espacial [27] e os planejamentos multiperíodo [28] e incremental [29].

Já a fragmentação do espectro ocorre com o término de conexões, que podem deixar vacâncias no espectro entre grupos de conexões já alocadas [30]. No caso das redes legadas WDM, essa fragmentação não gera perdas pois todas as conexões ocupam a mesma parcela do espectro (tráfego uniclasse), fazendo com que as lacunas geradas sempre possam ser preenchidas por novas conexões [1].

Com o uso de multiplexação espacial [31], a fragmentação pode acontecer entre modos ou entre fibras, tornando a alocação de conexões um processo ainda mais complexo. Entretanto, nas redes ópticas flexíveis, a fragmentação pode gerar bloqueios, dado que o padrão de tráfego pode conter demandas por conexões de ocupação espectral diferentes (tráfego multiclasse).

Portanto, a proposta deste estudo é atuar sobre a redução das perdas por fragmentação por meio de um algoritmo de alocação espectral que proporcione um uso mais eficiente dos recursos espectrais.

3.2.3 Topologias com Múltiplos Enlaces

As redes ópticas representam uma revolução tecnológica devido à sua capacidade de atender à crescente demanda por largura de banda gerada pela expansão das informações. Essas redes têm se expandido além das infraestruturas de *backbone* da Internet, alcançando áreas urbanas, empresas e, até mesmo, residências. As áreas urbanas desempenham um papel estratégico no estabelecimento de infraestruturas ópticas de última geração, impulsionando, assim, o crescimento econômico. As tecnologias ópticas oferecem uma solução econômica tanto para as atuais necessidades corporativas de largura de banda quanto para as futuras [32].

Uma infraestrutura de transporte de alta capacidade e cobertura abrangente é estabelecida com base em cabos de fibra óptica, micro-ondas digitais e satélites. Contudo, o desenvolvimento dessa infraestrutura de transporte avança de forma mais lenta, pois a demanda por capacidade continua a crescer devido à proliferação de novos serviços de comunicação e tecnologias de suporte [32].

Essas diversas redes de comunicação podem ser configuradas em diferentes topologias, incluindo barramentos com ou sem um tronco principal, redes em estrela e redes em anel, que podem ser redundantes e/ou autorreparáveis, ou uma combinação dessas. A topologia de rede consiste no estudo da disposição ou mapeamento dos elementos (tais como enlaces e nós) em uma rede, englobando tanto as interconexões físicas (reais) quanto as interconexões lógicas (virtuais) entre os nós [32].

Uma rede local (LAN - *Local Area Network*) exemplifica uma rede que exibe uma topologia física e lógica. Cada nó de uma LAN possui um ou mais *links* conectando-o a outros nós da rede, e o mapeamento desses *links* e nós em um gráfico resulta em uma forma geométrica que define a topologia física da rede [32]. A seguir, as principais topologias com múltiplos enlaces são apresentadas e discutidas.

3.2.3.1 Topologia de Rede em Barramento

Uma topologia em barramento, ilustrada na Figura 3.7, representa uma arquitetura de rede na qual um conjunto de nós de comunicação se encontra interligado por meio de uma infraestrutura de comunicação compartilhada denominada “barramento”. Esta topologia é caracterizada por sua facilidade de implementação e instalação, tornando-a particularmente apropriada para redes de escala reduzida ou temporárias que não demandam elevadas taxas de transferência de dados,

resultando em uma configuração de rede de menor latência. Além disso, apresenta a vantagem de permitir a identificação simplificada de falhas no cabo de comunicação [32].

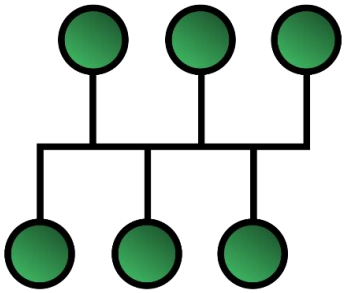


Fig. 3.7 - Topologia em barramento [33].

Entretanto, as redes de barramento frequentemente enfrentam desafios quando múltiplos clientes desejam transmitir dados simultaneamente no mesmo barramento, resultando em colisões de dados. Portanto, sistemas que empregam a topologia de rede de barramento tipicamente incorporam mecanismos de controle de colisões ou prevenção de colisões, como o CSMA (*Carrier Sense Multiple Access*) ou a presença de um mestre de barramento que gerencia o acesso ao recurso compartilhado do barramento [32].

É importante destacar que algumas arquiteturas ativas podem ser categorizadas como “barramento”, uma vez que proporcionam funcionalidades lógicas semelhantes às de um barramento passivo. Por exemplo, uma rede Ethernet comutada pode ser considerada como uma rede de barramento lógico, embora não o seja fisicamente [32].

No entanto, é relevante mencionar algumas limitações inerentes à topologia de rede de barramento. Estas incluem um comprimento de fibra óptica limitado, custos de manutenção mais elevados, já que a falha em qualquer ponto do cabo pode interromper toda a rede, bem como uma degradação de desempenho à medida que mais dispositivos são adicionados à rede ou em situações de tráfego intenso, devido à largura de banda compartilhada [32].

3.2.3.2 Topologia de Rede em Anel

Em uma configuração de topologia de rede em anel, cada nó estabelece conexões diretas com precisamente dois outros nós, resultando na formação de um único circuito fechado para a transmissão contínua de sinais ao longo de um anel. Os dados percorrem um itinerário sequencial passando por cada nó no circuito, sendo sujeitos a processamento em cada ponto de passagem [32].

A topologia de rede em anel apresenta desempenho superior em relação à de barramento, especialmente sob cargas de tráfego intensas. Além disso, não depende de um nó central para gerenciar a conectividade entre os computadores da rede [32]. A Figura 3.8 destaca a configuração da topologia em anel.

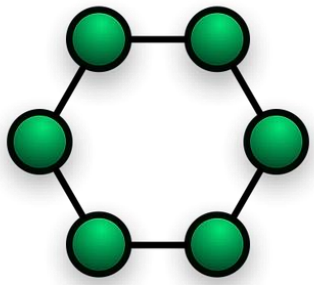


Fig. 3.8 - Topologia em anel [33].

Essa topologia apresenta algumas limitações significativas. Primeiramente, a falha de uma estação de trabalho pode comprometer a integridade de toda a rede, afetando o circuito como um todo. No entanto, em redes com topologia em anel, é possível utilizar a conexão oposta como caminho alternativo, redirecionando o tráfego para reduzir os impactos da falha. Além disso, falhas em nós individuais ou interrupções nos cabos de conexão podem causar o isolamento de todos os nós conectados ao anel. O atraso na comunicação neste tipo de rede tende a aumentar conforme o número de nós presentes. Adicionalmente, a largura de banda disponível é compartilhada entre todos os enlaces que conectam os dispositivos, o que pode limitar a capacidade de transmissão [32].

Para implementar uma rede em anel, é comum a utilização de tecnologias como FDDI (*Fiber Distributed Data Interface*), SONET (*Synchronous Optical Networking*), SDH (*Synchronous Digital Hierarchy*) ou a tecnologia de Token Ring. A topologia de anel é frequentemente empregada em ambientes como edifícios corporativos e campi escolares para fins de conectividade de rede [32].

3.2.3.3 Topologia em Árvore

Uma topologia em árvore, exibida na Figura 3.9, é uma disposição de rede que combina características das topologias de barramento e estrela. Essa configuração consiste em agrupamentos de estações de trabalho organizadas em redes estelares, as quais são interligadas por meio de um cabo de *backbone* configurado como um barramento linear.

A topologia de árvore oferece a vantagem de permitir a expansão de uma rede preexistente, possibilitando que instituições, como escolas, adaptem suas redes de acordo com suas necessidades específicas [32].

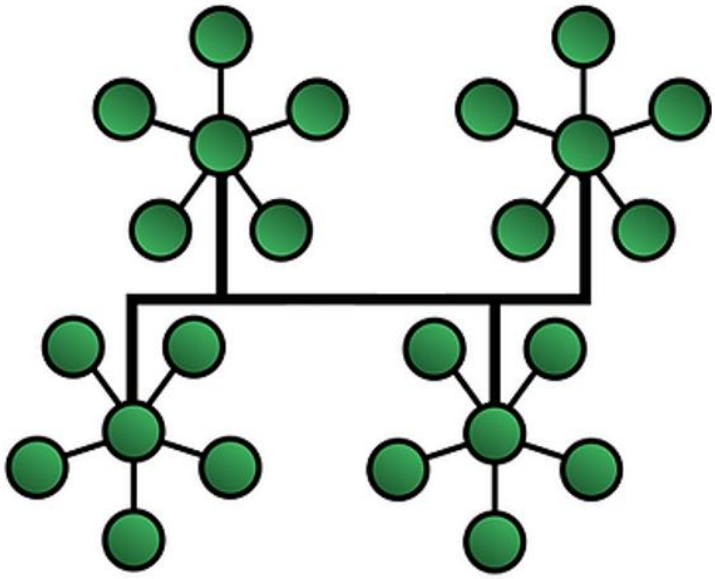


Fig. 3.9 - Topologia em árvore [33].

Essa flexibilidade é obtida por meio da facilidade de inclusão ou remoção de nós individuais em cada rede estelar, dado que a infraestrutura de cabeamento é configurada com conexões ponto a ponto para segmentos individuais. Em uma rede de topologia de árvore, uma eventual falha de cabo em uma das redes estelares resultará no isolamento somente dos nós diretamente conectados ao nó central dessa rede estelar, preservando a conectividade do restante da rede [32].

Entretanto, essa topologia não está isenta de limitações. Se a linha de *backbone*, que constitui o elemento central da topologia, sofrer uma falha, todo o segmento de rede será afetado. Além disso, a configuração e a interconexão de dispositivos em uma topologia de árvore são mais complexas e demandam mais recursos do que em outras topologias de rede [32].

As redes de barramento discutidas em 3.2.3.1 constituem um caso particular das topologias em árvore. Numa conceituação mais geral, as redes em árvore se caracterizam pelo fato de que o rompimento de qualquer enlace divide a rede em duas redes que não se comunicam. Do ponto de vista analítico, essas redes apresentam a importante propriedade da reversibilidade na presença de uma dinâmica markoviana [5], que viabiliza a obtenção de um modelo analítico fechado na ausência de fragmentação espectral. Essa propriedade se estende

a outras redes derivadas de árvores por meio de operações que preservam a reversibilidade [8], como os truncamentos de cadeias de Markov, conforme exemplificado por Waldman [34].

3.2.3.4 Topologia de Rede em Malha

A topologia de rede em malha é uma configuração na qual os nós da rede são extensivamente interconectados, resultando em uma estrutura altamente redundante e resiliente. Nessa topologia, a conectividade direta entre dispositivos desempenha um papel fundamental na garantia da robustez e da disponibilidade da rede. Existem duas variantes principais na topologia em malha: a malha completa (*full-mesh*), representada na Figura 3.10, e a malha parcial (*partial-mesh*) [33].

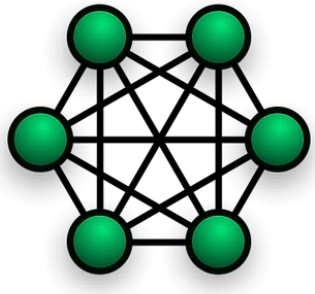


Fig. 3.10 - Topologia em malha completa [33].

Na topologia em malha completa, todos os dispositivos da rede estão interligados diretamente entre si. Isso significa que cada dispositivo possui uma conexão direta com todos os outros dispositivos na rede. Essa abordagem proporciona a máxima redundância e diversidade de caminhos para a transmissão de dados. Em um ambiente em malha completa, a falha de qualquer nó ou conexão específica tem um impacto mínimo na continuidade da comunicação, pois outros caminhos ainda estão disponíveis para a entrega de dados. Essa configuração é especialmente valiosa em redes críticas que exigem alta disponibilidade e tolerância a falhas [35].

Por outro lado, a topologia em malha parcial mantém a ideia de interconexão direta entre a maioria dos dispositivos na rede, mas não exige que todos os dispositivos estejam conectados uns aos outros. Na malha parcial, a maioria dos dispositivos ainda possui várias conexões diretas, criando uma rede robusta com diversos caminhos alternativos para a transmissão de dados. Isso proporciona uma resiliência considerável em comparação com topologias menos interconectadas [35].

Um dos principais benefícios da topologia em malha é a capacidade de utilizar múltiplos caminhos redundantes, o que permite otimizar a entrega de dados, escolher o caminho mais eficiente e garantir maior resiliência a falhas nos componentes da rede. Isso aumenta a eficiência da rede, minimiza os atrasos na comunicação e ajuda a evitar gargalos de tráfego. Além disso, a topologia em malha facilita a identificação rápida e o isolamento de problemas, o que simplifica a manutenção e o diagnóstico de falhas [35].

Em resumo, a topologia de rede em malha, seja na forma em malha completa ou parcial, oferece um alto grau de redundância e flexibilidade na comunicação de dados. Ela é especialmente adequada para ambientes em que a confiabilidade e a tolerância a falhas são essenciais, garantindo que a rede continue operacional mesmo diante de problemas de hardware ou conexão [35].

3.2.4 Abordagens Sem Desfragmentação em Topologias com Múltiplos Enlaces

Na abordagem sem desfragmentação, são tomadas as precauções necessárias para evitar a fragmentação antes do estabelecimento de um caminho óptico. No entanto, nenhuma ação é tomada para caminhos ópticos em serviço, ou seja, o espectro é gerenciado antecipadamente para evitar o efeito da fragmentação. No caso de abordagens com desfragmentação, uma ação necessária é tomada para caminhos ópticos em operação, a fim de suprimir o efeito da fragmentação [36].

Esta seção apresenta as principais abordagens sem desfragmentação para EONs, relatadas na literatura, e destaca seus pontos fortes e fracos. Tais abordagens empregam várias estratégias para melhorar o desempenho da rede e a utilização dos recursos espectrais.

3.2.4.1 Roteamento Multicaminho

Para suprimir o efeito da fragmentação da largura de banda, os autores em [37] apresentaram diferentes algoritmos de provisionamento de serviços online, que incorporam roteamento dinâmico, modulação e alocação espectral com roteamento híbrido de caminho único ou esquemas de roteamento multicaminho. Eles investigaram, principalmente, dois tipos de esquemas de roteamento híbrido de caminho único ou de roteamento de múltiplos caminhos para computação de caminhos on-line e conjuntos de caminhos fixos. Além disso, também analisaram diversas políticas de seleção de caminho em termos de supressão da fragmentação

da largura de banda e maximização do rendimento da rede para otimizar o projeto e avaliaram os algoritmos introduzidos usando um modelo de tráfego de Poisson para duas topologias de rede em malha (*mesh*).

Os autores em [38] introduziram um esquema de roteamento, modulação e alocação de espectro com reconhecimento de fragmentação de múltiplos caminhos para solicitações de reserva antecipadas e imediatas em EONs. Para suprimir a fragmentação e resolver o problema da escassez de recursos, os autores introduziram um esquema de divisão; as solicitações são divididas em partes diferentes e transferidas para um ou mais caminhos ópticos com o uso de transponders de largura de banda variável. Os autores fornecem um modelo matemático para resolver o problema e também apresentam um algoritmo heurístico para a medição da fragmentação.

Para melhorar a utilização do espectro e eliminar a fragmentação da largura de banda em EONs, técnicas de tratamento do tráfego e roteamento multicaminho foram apresentadas em [39]. Em seu trabalho de pesquisa, eles investigaram primeiro os possíveis ganhos ao aplicar conjuntamente os dois métodos com um modelo prático que considera deficiências físicas. Eles atribuíram recursos de espectro e mediram os ganhos na eficácia da utilização do espectro em relação às abordagens existentes. Posteriormente, apresentaram uma heurística para redes grandes e introduziram uma formulação de otimização analítica para redes pequenas.

3.2.4.2 Multigrafo

Em [40], os autores introduziram um algoritmo de caminho mais curto em um multigrafo (MGSP, do inglês *Multi-Graph Shortest Path*) para reduzir a taxa de bloqueio em EONs. As decisões de alocação no algoritmo MGSP são baseadas em funções de custo, que tentam capturar a potencialidade dos fragmentos do espectro de alocação das solicitações recebidas. Os autores observaram que o algoritmo MGSP reduz a taxa de bloqueio de largura de banda quatro vezes em comparação com os algoritmos RSA existentes.

Em [41], uma política de roteamento de caminho mais curto multigrafo com reconhecimento de energia é apresentada considerando diferentes níveis de modulação. Os resultados numéricos do trabalho apresentado indicam que o algoritmo proposto pode economizar até 34% de energia e reduzir significativamente o bloqueio de largura de banda em comparação com os algoritmos existentes.

3.2.4.3 Particionamento do Espectro

Os autores em [42] e [43] introduziram um esquema de partição dedicada que reduz a queda de solicitações, separando os caminhos ópticos disjuntos e não disjuntos em partições diferentes. Esses dois trabalhos particionam todo o espectro antecipadamente para gerenciar os recursos espectrais de forma proficiente, com a restrição de que os caminhos ópticos atribuídos na mesma partição devem ser separados por enlace. A abordagem de partição introduzida fornece *slots* livres alinhados, liberando a alocação de espectro de caminhos ópticos não disjuntos. *Slots* disponíveis contíguos e alinhados entre duas partições são criados usando a política de alocação de ajuste primeiro-último. Como resultado, o esquema introduzido em [42] e [43] reduz o bloqueio de solicitações na rede. Ambos os trabalhos indicam que, embora o número de partições seja minimizado, o desempenho da rede aumenta. Além disso, observam que o esquema de partições dedicadas tem uma desvantagem em termos de probabilidade de bloqueio se o número de partições for grande. Isso ocorre porque o ganho de multiplexação estatística é perdido [44].

Para superar o problema do particionamento dedicado, em [45] os autores fornecem um esquema de pseudo-particionamento que adota a política de alocação de ajuste exato do tipo primeiro-último para EONs, a fim de aumentar o número de *slots* livres alinhados; pequenos *slots* livres contíguos são ignorados. O esquema introduzido em [45] aloca caminhos ópticos disjuntos e não disjuntos separadamente. Caminhos ópticos com caminhos disjuntos são atribuídos usando a política de ajuste exato. Por outro lado, caminhos ópticos com caminhos não disjuntos são atribuídos usando a política de último ajuste exato. Esta separação de caminhos ópticos disjuntos e não disjuntos oferece um maior número de *slots* livres alinhados. A política de ajuste exato oferece um número maior de *slots* gratuitos contíguos. Os *slots* disponíveis alinhados contíguos suprimem a fragmentação da largura de banda e, portanto, a probabilidade de bloqueio é reduzida.

Em [46], os autores introduziram uma abordagem de gerenciamento de fragmentação para EONs que evita a fragmentação da largura de banda sem redirecionar os caminhos ópticos. Se as solicitações de caminhos ópticos com larguras de banda menores e maiores não forem diferenciadas durante a alocação de espectro, as solicitações de caminhos ópticos com largura de banda maior poderão ser rejeitadas. Para evitar essa deficiência, eles investigaram uma política de controle de admissão que captura os desafios distintos impostos por larguras de banda heterogêneas. É adotada uma política preventiva de controle de admissão

que segue a abordagem de particionamento do espectro, a fim de alcançar uma maior eficiência de provisionamento. Isso reduz a parcialidade da atribuição de largura de banda e evita o bloqueio de chamadas.

Em outro trabalho [26], é descrita uma política de controle de admissão de tráfego que foca nos desafios especiais colocados por larguras de banda heterogêneas em EONs. Os autores observaram que o particionamento do espectro é superior ao não particionamento, pois supera as dificuldades de fragmentação e imparcialidade.

3.2.4.4 Reencaminhamento

Em [47] foi apresentado um algoritmo de gerenciamento que reduz a fragmentação da largura de banda em EONs. O algoritmo proposto consegue uma melhor utilização dos recursos por meio da reconfiguração proativa da rede, com o redirecionamento dos caminhos ópticos estabelecidos. Em cada fase, o algoritmo primeiro escolhe alguns caminhos ópticos e depois decide como redirecioná-los com a estratégia de gerenciamento da fragmentação baseada no roteamento e na alocação espectral. Finalmente, o algoritmo realiza o reencaminhamento com melhor esforço para a migração do tráfego, a fim de eliminar a interrupção do tráfego.

O estudo em [48] concentra-se em um esquema de reencaminhamento de caminho óptico minimizado por interrupções que alivia a fragmentação espectral e alcança uma melhor utilização do espectro em EONs. O esquema realiza um reencaminhamento mínimo para evitar a interrupção do serviço, uma vez que as interrupções podem degradar significativamente os serviços de rede. Foi proposto um esquema de reencaminhamento que estabelece rotas alternativas e aloca *slots* de frequência antes de liberar os caminhos ópticos originais, a fim de reduzir a fragmentação da largura de banda na rede.

Os trabalhos [49] e [50] apresentaram duas abordagens de gerenciamento da fragmentação, a saber, (i) um esquema reativo-disruptivo e (ii) um esquema proativo não-disruptivo para minimizar a fragmentação em EONs. A abordagem de reconfiguração reativa-disruptiva aloca solicitações de caminhos ópticos recebidas, e que não podem ser atendidas, reconfigurando alguns caminhos ópticos existentes. A abordagem de reconfiguração proativa e não disruptiva não perturba os caminhos ópticos existentes. Ela permite apenas a reconfiguração desses caminhos ópticos, que podem ser deslocados sem cruzar o espectro de outros caminhos ópticos.

4 ALGORITMOS CIENTES DO TRÁFEGO INCREMENTAL

Apesar do grande salto na utilização da largura de banda proporcionado pelos sistemas WDM desde a sua criação, ainda existem perdas causadas pelo descasamento entre uma demanda heterogênea por largura de banda e a sua oferta sobre uma grade de frequências que favorece uma única granularidade. Mesmo em redes ópticas elásticas, em que o espectro é alocado em blocos de *slots* contíguos de acordo com a necessidade do cliente, e a largura de banda pode ser fornecida em uma ampla gama de granularidades, também podem ocorrer perdas [51].

Conforme discutido em Waldman [52], no entanto, as políticas vorazes de atribuição espectral, nas quais as solicitações são sempre aceitas desde que haja recurso disponível para tal, podem produzir um uso ineficiente do espectro para perfis de tráfego desfavoráveis. No limite, esses perfis de tráfego podem levar a situações em que alguns recursos espectrais fiquem bloqueados permanentemente sob o tráfego incremental.

Um estudo sobre a otimização da atribuição espectral com a ciência de parâmetros do tráfego em um enlace avulso livre de impasses foi apresentado em [52], no qual assume-se uma condição de busca por *slots* únicos. Relaxar essa suposição sob algoritmos vorazes resulta na possibilidade de impasse (em inglês, *deadlock*) na alocação de recursos espectrais [53].

A prevenção do impasse é, então, discutida em [51] para o tráfego incremental. Em [51], os autores mostram a necessidade de usar políticas de atribuição não vorazes em alguns estados da rede para evitar impasses permanentes sob tráfego incremental. Por esta razão, em [53] os autores sugerem o uso de políticas vorazes, porém seletivas, que relutam em atribuir solicitações a vazios disfuncionais em favor de outros vazios que proporcionem uma melhor correspondência com o conjunto dos tamanhos das requisições.

Algoritmos de atribuição espectral que consideram os tamanhos que podem ser solicitados pelo tráfego, apresentam algum conhecimento (em inglês, *awareness*) do tráfego. Em contraste, a maioria das heurísticas utilizadas em redes WDM, como *first-fit* (FF), *least used* ou *most used* [13], não utilizam nenhuma informação sobre o perfil do tráfego. Entre os citados, o algoritmo FF também se abstém de utilizar informações globais sobre o estado da rede e é muito valorizado por usar o mínimo possível de informações, reduzindo assim a carga no plano de controle. Por outro lado, a tendência atual do uso de algoritmos de *Machine Learnig* (ML) implica na captura, o tanto quanto possível, de informações relevantes do ambiente (ou seja, do tráfego) para usá-las na decisão da alocação espectral, contornando assim o plano de controle enquanto fornece inteligência ao algoritmo de atribuição [53].

4.1 OCORRÊNCIA DE IMPASSE (DEADLOCK)

Considere uma rede de um enlace avulso e seja S o número de *slots* contíguos no espectro disponível, e $s(t)$ o número de *slots* disponíveis no tempo t. A chegada de pedidos de novas conexões é considerada como poissoniana, com λ_i, $i = 1, 2, 3, ...$, sendo a taxa de pedidos de conexões com i *slots* contíguos. Uma vez aceitas e ativadas, todas as conexões são permanentes, produzindo um tráfego incremental. O objetivo é garantir que o espectro seja esgotado, sempre que possível, e minimizar o tempo médio necessário para que isso aconteça. Ambas as tarefas são executadas por um algoritmo de atribuição espectral (SA, do inglês *Spectral Assignment*) ciente do tráfego, que recebe cada solicitação e a rejeita ou atribui a ela um posicionamento espectral adequado. A ciência do tráfego significa que o algoritmo SA conhece os valores de λ_i, $i = 1, 2, 3, ...$ [51]. Na prática, esse conhecimento depende de uma avaliação contínua das taxas de chegada, uma vez que elas podem variar com o tempo.

A ociosidade Ω é definida como o valor médio da soma das durações dos tempos de ociosidade de todos os *slots* até a exaustão do espectro a partir do estado vazio inicial, em termos de [(*slots ociosos*) × (*unidade de tempo*)]. A desocupação $s(t)$ gera perda de recursos espectrais devido à não ocupação desses *slots*. Se ocorrer um impasse, os *slots* não ocupados podem gerar uma ociosidade que tende ao infinito com o tempo. Caso haja uma probabilidade não nula de um ou mais *slots* ficarem livres para sempre, tem-se uma ociosidade infinita [4].

A minimização da ociosidade é alcançada por meio de um algoritmo de atribuição espectral ciente do tráfego, que rejeita ou aceita cada requisição, e seleciona uma posição do espectro para a alocação em caso de aceite. A rejeição de uma requisição pode ocorrer devido à insuficiência de recursos disponíveis ou pode ser imposta por uma política seletiva com o objetivo de reduzir a ociosidade espectral [4]. Em [52], esse problema foi tratado para o caso especial quando $\lambda_1 > 0$, condição que garante que o impasse não ocorrerá. Este capítulo apresenta uma discussão sobre a ocorrência de impasses em um cenário de tráfego incremental e múltiplas classes.

4.1.1 Posicionamento Espectral

Sempre que uma solicitação de i *slots* contíguos é acomodada em uma posição intermediária de um vazio espectral, conforme ilustra a Figura 4.1, os recursos disponíveis são fragmentados desnecessariamente, reduzindo assim a capacidade de acomodar solicitações futuras.

Essa alocação resulta em dois novos vazios, com v_1 e v_2 *slots*, de tal forma que $(v_1 + v_2) = (v - i)$, ou seja, o recurso espectral livre em sua totalidade é igual a $(v - i)$ *slots* livres. Porém, o recurso total disponível não está agrupado ao longo do enlace avulso [51].

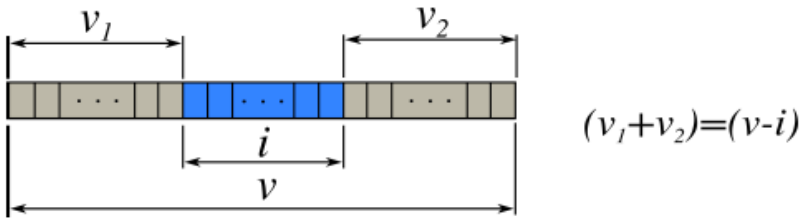


Fig. 4.1 - Alocação intermediária em um vazio de tamanho v [4].

Todas as requisições subsequentes que podem ser alocadas nesses dois vazios também podem ser alocadas em um único vazio. No entanto, há alguns casos em que algumas requisições só podem ser alocadas no cenário em que se tem um único vazio [4].

Sempre que uma requisição por i *slots* contíguos é alocada em um vazio do espectro, dado que o vazio selecionado tem um tamanho $v \geq (i + 1)$ *slots*, ela pode ser acomodada em uma das duas extremidades do vazio, deixando um único vazio restante de tamanho $(v - i)$ *slots*, como mostra a Figura 4.2 [4].

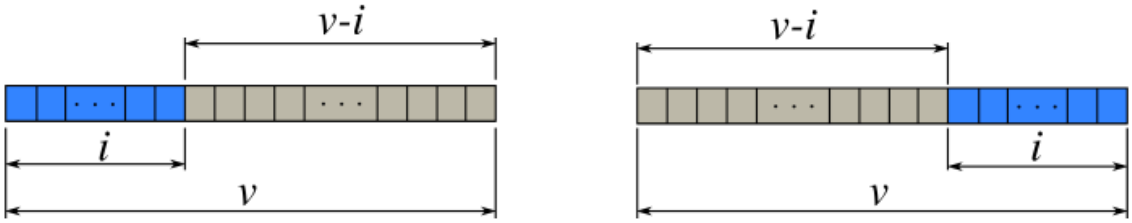


Fig. 4.2 - Alocação na extremidade (esquerda e direita) de um vazio de tamanho v [4].

Por consequência, considera-se que as conexões recém-aceitas são sempre acomodadas em uma posição extrema de um vazio, deixando assim um único vazio residual. Portanto, a partir de um vazio inicial de S *slots*, o enlace sempre apresentará um único vazio para as solicitações futuras até que o espectro disponível seja eventualmente esgotado ou o sistema fique bloqueado, pois o vazio residual não pode acomodar qualquer nova solicitação de entrada [4].

Por esse motivo, uma vez que o algoritmo de atribuição espectral decide aceitar o pedido de uma conexão, nenhuma perda de desempenho ocorre se a requisição for acomodada nas frequências mais baixas ou mais altas do espectro. Dessa maneira, todos os $s(t)$ *slots* disponíveis em um dado tempo t sempre estarão em um único vazio, de modo que a

contiguidade dos *slots* disponíveis é preservada, e a fragmentação espectral é evitada para o caso do tráfego incremental [4].

4.1.2 Ociosidade Residual

À medida que novas conexões são ativadas, $s(t)$ é reduzido a zero ou a um valor residual $s(\infty)$ que, se positivo, fará com que o enlace avulso não possa acomodar nenhuma conexão adicional, gerando uma ociosidade espectral infinita. Assim, após garantir que $s(\infty) = 0$, sobra uma ociosidade residual finita que deve ser minimizada até a exaustão do espectro, conforme explicado em [4] e detalhado a seguir.

Seja S o número de *slots* em um vazio espectral a ser preenchido com solicitações de serviços recebidas. O conjunto dos tamanhos que podem ser solicitados é dado pelo vetor $I = (i_1, i_2, \dots, i_K)$, em que K é o número de classes de tráfego e $0 < i_1 < i_2 < \cdots < i_K$. Observe que o tráfego incremental é o limite assintótico do tráfego dinâmico quando as taxas de solicitações são levadas ao infinito [54].

Considere o caso de $I = (3, 5)$ e $S = 15$. A Figura 4.3 mostra a árvore de todos os estados que o tamanho do vazio pode assumir, pois os pedidos de 3 ou 5 *slots* são acomodados no vazio de forma contígua, a fim de evitar qualquer fragmentação espectral. A Figura 4.3 indica que existem dois estados de exaustão, mostrados em azul, que são definidos como qualquer estado no qual o espectro está esgotado: um é alcançado por 5 conexões de 3 *slots*, e o outro por 3 conexões de 5 *slots*.

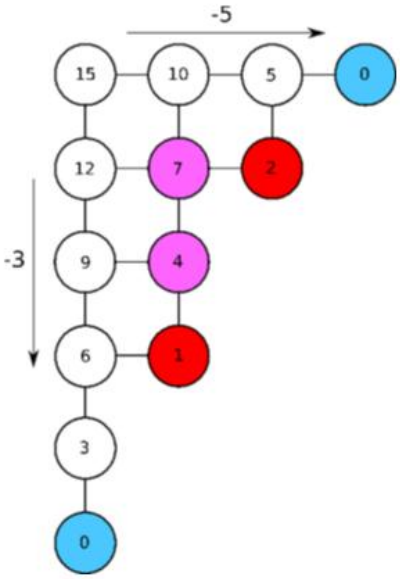


Fig. 4.3 - Árvore de estados para $S = 15$ até a exaustão (azul) ou ao impasse (vermelho) [54].

Note que nenhum estado de exaustão pode ser alcançado a partir dos estados 1, 2, 4 e 7, que incluem os estados de impasse (1 e 2, em vermelho) e os estados predestinados ao impasse (4 e 7, em rosa) [54]. Além disso, pode-se demonstrar que esses são os únicos estados a partir dos quais o impasse, aqui definido como uma condição na qual o preenchimento do vazio se torna impossível, não pode mais ser evitado [51].

O conjunto de todos os estados a partir dos quais o impasse não pode ser evitado é definido como o conjunto de impasse ou *deadlock* $D(I)$. Pode ser mostrado [51] que quando $I = (b_1, b_2)$, com $b_1 < b_2$ e primos entre si, então $D(b_1, b_2)$ é dado pela árvore gerada por solicitações de b_1 ou b_2 *slots* quando eles são acomodados de forma contígua em um vazio com tamanho inicial $[b_1 b_2 - (b_1 + b_2)]$. Se b_1 e b_2 não são primos entre si e $g > 1$ é o seu maior divisor comum, então de acordo com a Equação 4.1 [54]:

$$\mathrm{D}(b_1, b_2) = g \cdot D(b_1/g,\ b_2/g)\ \cup\ \Psi(g) \tag{4.1}$$

em que $\Psi(g)$ é o conjunto de todos os inteiros que não são múltiplos de g. Se $S \notin D(I)$, o impasse só pode ser evitado rejeitando qualquer requisição que leve a um vazio residual em $D(I)$, indicados em rosa e vermelho na Figura 4.4, que mostra a árvore gerada para $I = (3, 5)$ e $S = 17$ [54].

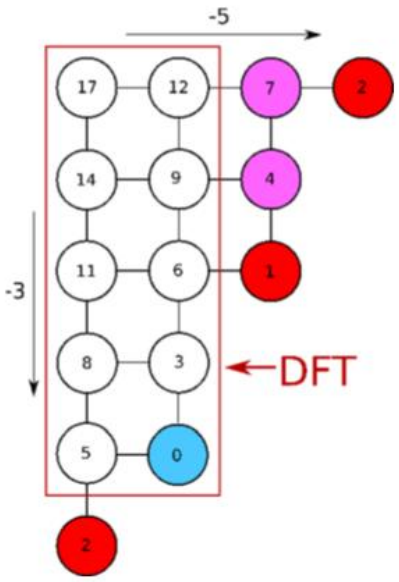


Fig. 4.4 - Árvore de estados para $S = 17$. Em destaque, a árvore livre de impasse (DFT) [54].

Para evitar impasses é necessário que, quando estiver nos estados 5, 6, 9 ou 12, o sistema adote uma política não voraz na qual deve rejeitar algumas solicitações que poderiam

ser acomodadas com os recursos disponíveis. Em seguida, é conveniente remover da árvore original todos os estados que são membros do conjunto de impasse, resultando em uma árvore livre de impasse (DFT, do inglês *Deadlock-Free Tree*), formada pelos estados indicados em branco e azul nas Figuras 4.3 e 4.4 [54].

4.1.3 Minimização da Ociosidade Sensível ao Tráfego

A ociosidade espectral finita $\Omega(S)$ na DFT é definida como o valor médio da integral de tempo do vazio de tamanho $s(t)$, à medida que $s(t)$ desce a DFT de S a zero se $S \notin D(I)$ [54]:

$$\Omega(S) = E\left[\int_0^\infty s(t)dt\right] \tag{4.2}$$

em que $s(0) = S$.

A ideia básica para evitar impasses é garantir que a ociosidade espectral permaneça finita, o que pode ser alcançado com base no conhecimento dos tamanhos das conexões de entrada, independentemente de suas taxas de chegada. Além disso, se as taxas de chegada λ_i de solicitações por i *slots* para todo i são conhecidas ou podem ser estimadas a partir das solicitações de entrada, então a ociosidade finita fornecida pela prevenção de impasses pode ser ainda mais reduzida por políticas dependentes do estado definidas sobre a árvore DFT, que garante a finitude da ociosidade espectral [54].

Exemplo 4.1: considere o caso do estado $s = b_1 b_2$ com $I = (b_1, b_2)$ e $b_1 < b_2$ primos entre si. No caso do tráfego incremental com chegadas poissonianas aqui discutido, o estado s pode sempre ser considerado um recomeço do processo com uma redefinição do estado inicial, uma vez que as requisições já atendidas receberam alocações definitivas e o tráfego poissoniano não tem memória, de modo que as notações s e S são intercambiáveis. Em cada estado da árvore, no máximo três políticas podem ser consideradas candidatas a uma política ótima: a política não voraz N_1, que aceita somente pedidos por b_1 *slots*, a política não voraz N_2, que admite apenas pedidos por b_2 *slots* e a política voraz G (*Greedy*), que recebe qualquer solicitação que possa ser acomodada com os recursos disponíveis [54]. No caso da Figura 4.3, repare que a prevenção de impasses restringe a maioria dos estados a seguir uma das políticas não vorazes, mas ainda há um estado no qual cada uma das três políticas definidas acima pode ser seguida: o estado inicial $S = b_1 b_2 = 15$.

As seguintes equações podem ser escritas a partir de (4.2) para a ociosidade espectral $\Omega_P(s)$ gerada pela política $P \in \{N_1, N_2, G\}$ no estado s [54]:

$$\Omega_{N_1}(s) = \frac{s}{\lambda_1} + \Omega(s - b_1) \tag{4.3}$$

$$\Omega_{N_2}(s) = \frac{s}{\lambda_2} + \Omega(s - b_2) \tag{4.4}$$

$$\Omega_G(s) = \frac{s+\lambda_1\Omega(s-b_1)+\lambda_2\Omega(s-b_2)}{\lambda_1+\lambda_2} \tag{4.5}$$

Nas três equações, o primeiro termo é o produto de s com o tempo médio de espera para o primeiro pedido que é elegível para aceitação conforme cada política. O segundo termo de (4.3) e (4.4) é a ociosidade residual do novo estado, e a soma do segundo e terceiro termos de (4.5) é a ociosidade residual média sobre os dois novos estados possíveis [54].

A política ótima é, então, aquela que produz o valor mínimo sobre as equações acima. Sob cada estado s, quando qualquer $(s - b_1)$ ou $(s - b_2)$ está no conjunto de impasse, tanto a política voraz quanto uma das políticas não vorazes geram ociosidade espectral média infinita, de modo que a política não voraz restante é a ideal, independentemente do perfil do tráfego. O único estado que deve seguir uma política ótima baseada no conhecimento do perfil do tráfego é, então, o estado inicial $s = b_1 b_2$. Para identificar a sua política ótima, é necessário primeiro calcular $\Omega(s - b_1)$ e $\Omega(s - b_2)$. A partir de (4.3) e (4.4), obtém-se [54]:

$$\Omega(b_1 b_2 - b_1) = \frac{(b_2-1)b_1}{\lambda_1} + \frac{(b_2-2)b_1}{\lambda_1} + \cdots + \frac{b_1}{\lambda_1} = \frac{b_2(b_2-1)b_1}{2\lambda_1} \tag{4.6}$$

Analogamente:

$$\Omega(b_1 b_2 - b_2) = \frac{b_1(b_1-1)b_2}{2\lambda_2} \tag{4.7}$$

Substituindo (4.6) e (4.7) em (4.3), (4.4) e (4.5) produz:

$$\Omega_{N_1}(b_1 b_2) = \frac{b_1 b_2(b_2+1)}{2\lambda_1} \tag{4.8}$$

$$\Omega_{N_2}(b_1 b_2) = \frac{b_1 b_2 (b_1+1)}{2\lambda_2} \tag{4.9}$$

$$\Omega_G(b_1 b_2) = \frac{b_1 b_2 (b_1+b_2)}{2(\lambda_1+\lambda_2)} \tag{4.10}$$

Uma comparação cuidadosa entre (4.8), (4.9) e (4.10) sobre todos os valores positivos de λ_2/λ_1 revelará, então, que [54]:

$$\Omega_{min}(b_1 b_2) = \Omega_{N_1}(b_1 b_2) = \frac{b_1 b_2 (b_2+1)}{2\lambda_1} \quad se \quad \frac{\lambda_2}{\lambda_1} \leq \frac{b_1-1}{b_2+1} \tag{4.11}$$

$$\Omega_{min}(b_1 b_2) = \Omega_{N_2}(b_1 b_2) = \frac{b_1 b_2 (b_1+1)}{2\lambda_2} \quad se \quad \frac{\lambda_2}{\lambda_1} \geq \frac{b_1+1}{b_2-1} \tag{4.12}$$

$$\Omega_{min}(b_1 b_2) = \Omega_G(b_1 b_2) = \frac{b_1 b_2 (b_1+b_2)}{2(\lambda_1+\lambda_2)} \quad se \quad \frac{b_1-1}{b_2+1} \leq \frac{\lambda_2}{\lambda_1} \leq \frac{b_1+1}{b_2-1} \tag{4.13}$$

Isso significa que a política ideal será voraz se, e somente se, $(b_1 - 1)/(b_2 + 1) \leq \lambda_2/\lambda_1 \leq (b_1 + 1)/(b_2 - 1)$. Por exemplo, se $(b_1, b_2) = (4, 7)$ e $S = 28$, a política ótima será voraz somente se λ_2/λ_1 estiver entre $3/8$ e $5/6$. Para todos os valores de $\lambda_2/\lambda_1 > 5/6$, incluindo o 1, que é frequentemente usado em simulações, a política ideal é aceitar apenas solicitações por 7 *slots*. Assim, o vazio deve ser preenchido apenas com conexões grandes para alcançar ociosidade mínima.

Já para valores de $\lambda_2/\lambda_1 < 3/8$, a política ótima será não voraz que aceite somente requisições de 4 *slots* [54]. A Figura 4.5 ilustra as políticas aplicáveis para cada um dos intervalos de valores de λ_2/λ_1.

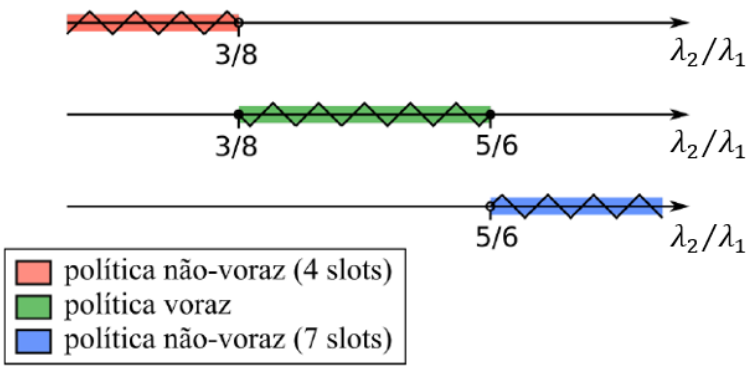
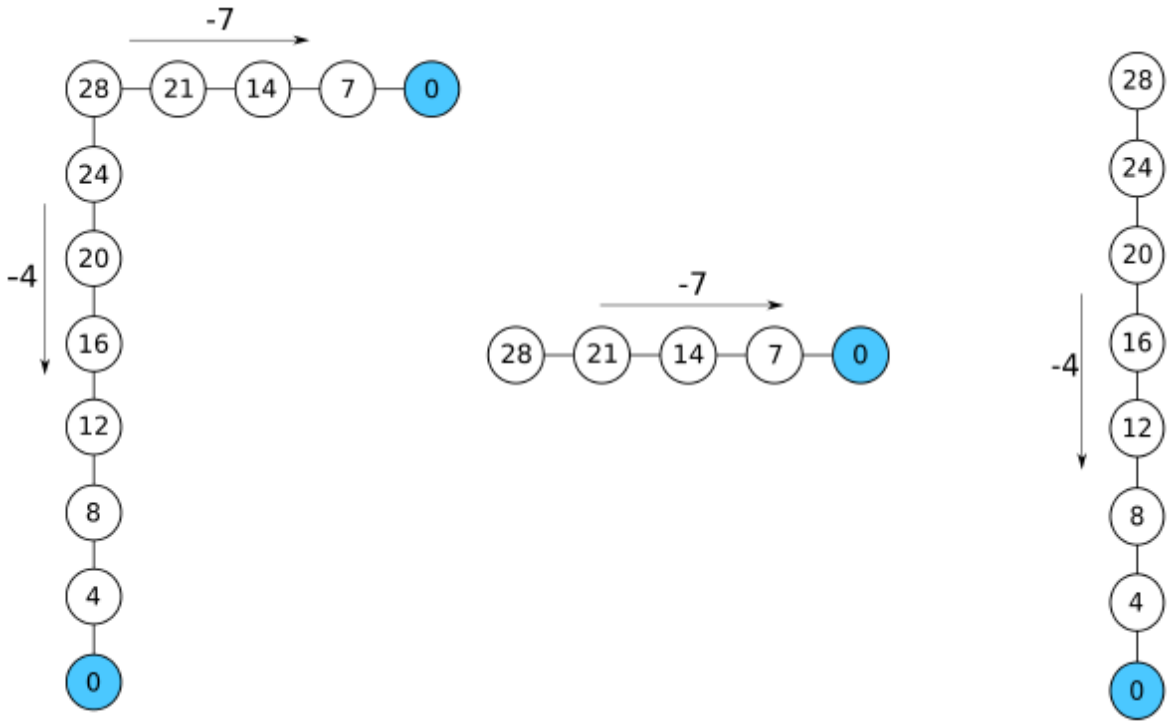


Fig. 4.5 - Representação das políticas aplicadas para $I = (4, 7)$ e $S = 28$ [4].

Observe que, para implementar a política N_2 no estado s, a transição de s para $(s - b_1)$ deve ser proibida, e o mesmo deve acontecer com a transição de s para $(s - b_2)$ a fim de se cumprir a política N_1. Somente a política voraz mantém ambas as transições livres para serem tomadas pelo próximo pedido, levando à escolha do estado final de exaustão. A configuração da DFT ideal deve, então, mudar de acordo com o perfil do tráfego [4]. A Figura 4.6 mostra a configurações ótimas da DFT com $I = (4, 7)$ e $S = 28$ para todos os perfis de tráfego.



(a) (b) (c)

Fig. 4.6 - Configurações da DFT para as políticas: (a) voraz; (b) não voraz (7 *slots*); (c) não voraz (4 *slots*) [4].

Uma DFT em que todos os estados seguem uma política de prevenção de impasses ou a política voraz é chamada de DFT voraz, que pode ou não ser ideal. Estados em que a política ótima é dependente do perfil de tráfego são definidos como estados semafóricos, pois atuam como semáforos virtuais na DFT. O Teorema 4.1 ajuda a identificar os estados semafóricos [54].

Teorema 4.1: se todos os caminhos a partir de um estado s para todos os estados de exaustão alcançáveis tiverem pelo menos uma conexão de tamanho w, a requisição de w *slots* será aceita pela política ideal [54].

Demonstração: se a requisição para uma conexão de tamanho w for rejeitada em algum estado, outra requisição para o mesmo tipo de conexão terá que ser aceita mais tarde em algum ponto da árvore para qualquer um dos estados de exaustão alcançáveis do espectro.

Portanto, a rejeição de uma conexão em um dado estado s a partir do qual essa conexão é necessária para alcançar todos os estados de exaustão da DFT, só adiciona uma quantidade positiva de ociosidade na Equação 4.2, para cada amostra do processo aleatório, aumentando desnecessariamente a ociosidade média [4].

Corolário 4.1: se s não é múltiplo de b_1 nem de b_2 para $K = 2$, então a política ótima no estado s é voraz [54].

Demonstração: se s não for múltiplo de b_1, pelo menos um pedido por b_2 *slots* deve ser aceito para esgotar o espectro com conexões de b_1 *slots* e vice-versa. Então, nenhum pedido por b_1 ou b_2 *slots* pode ser rejeitado no estado s para a obtenção da ociosidade mínima, ou seja, a política ótima é a voraz em s.

Corolário 4.2: se houver apenas um estado de exaustão, a política voraz é ótima em todos os estados que não são adjacentes a um membro do conjunto de impasse [54].

O número de estados de exaustão pode ser qualquer número inteiro positivo, e é uma função do estado inicial S. Em ordem crescente de S, $b_1 b_2$ é o primeiro vazio que possui dois estados de exaustão. No entanto, ainda existem $|D(I)|$ valores de S acima de $b_1 b_2$ que produzem um único estado de exaustão cada, dados pela soma de $b_1 b_2$ com cada elemento de $D(I)$ [54].

Quando $k > 1$, a existência de mais de um estado com o mesmo tamanho de vazio reflete o fato de que os estados da árvore são dados não apenas pelo tamanho do vazio, mas também pelos números de conexões de tamanho b_1 e b_2 cuja acomodação gerou o vazio, em qualquer ordem. Sempre que necessário para não causar ambiguidade, um estado será denotado por $s(\overline{n_1}, \overline{n_2})$, em que s é o tamanho do vazio (zero para estados de exaustão), $\overline{n_1}$ é o número de conexões ativas do tamanho b_1 e $\overline{n_2}$ é o número de conexões ativas do tamanho b_2 já admitidas no sistema em qualquer ordem [54].

4.1.3.1 Único Estado de Exaustão

Se $S \in \sigma_1$, há apenas um estado de exaustão alcançável de qualquer estado da DFT, então o Teorema 4.1 se aplica para todas as transições externas possíveis. Assim, todos os estados devem seguir uma política de prevenção de impasses ou voraz, de maneira que não haverá a necessidade de configurar estados semafóricos, e a DFT voraz é a ideal [54].

O único estado de exaustão denota a ocupação do vazio inicial por uma combinação única de um número n_1 de passos de tamanho b_1 e um número n_2 de passos de tamanho b_2, tal que $S = n_1 b_1 + n_2 b_2$, tomada em qualquer ordem. Se houver apenas um estado de exaustão, essa combinação é a única possível, de modo que a ociosidade espectral média pode ser expressa como a soma das contribuições de cada etapa, conforme expresso na Equação 4.14 [54]:

$$\Omega(s = n_1 b_1 + n_2 b_2) = \frac{n_1(n_1+1)b_1}{2\lambda_1} + \frac{n_2(n_2+1)b_2}{2\lambda_2}, \quad s \in \sigma_1 \tag{4.14}$$

Perceba que (4.14) e a política voraz ótima relacionada são aplicáveis a qualquer estado $s = n_1 b_1 + n_2 b_2$ que possa alcançar um único estado de exaustão, mesmo que a DFT tenha outros estados de exaustão que são inalcançáveis a partir de s [54]. A Figura 4.7 ilustra, para $I = (3, 5)$ e $S = 13$, a ociosidade espectral média por meio da soma das contribuições de cada etapa, dadas pelas expressões em cada retângulo que indicam a sua respectiva área média.

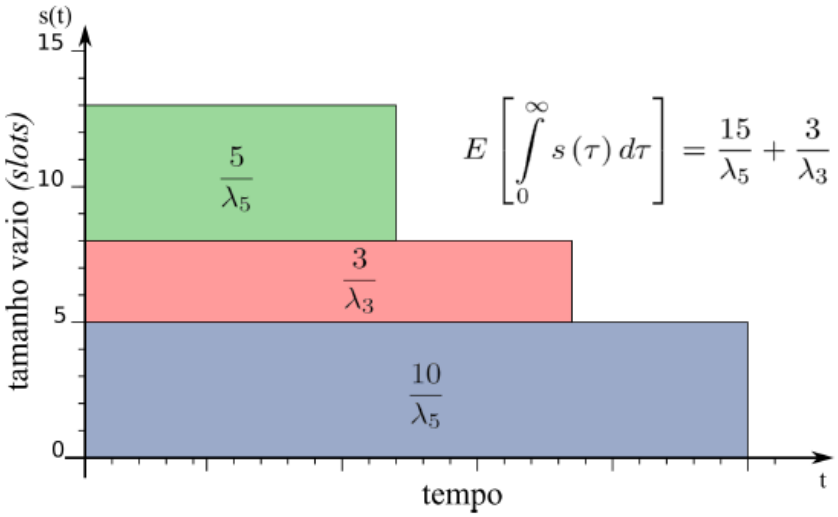


Fig. 4.7 - Exemplo de $s(t)$ para $I = (3, 5)$ e $S = 13$ [4].

A DFT voraz para o caso anterior é apresentada na Figura 4.8. Os estados 5 e 10 seguem políticas não vorazes a fim de evitar impasses. Atingir o único estado de exaustão do estado 8 requer o aceite de uma conexão de tamanho 3 e uma conexão de tamanho 5, em qualquer ordem. A partir do estado 13, duas conexões de tamanho 5 e uma de tamanho 3 são necessárias para a mesma finalidade. Com base no Teorema 4.1, os estados 8 e 13 devem admitir as duas possíveis transições externas de acordo com a próxima chegada, de modo que ambos seguem a política voraz e a Figura 4.8 já é a configuração ideal, sem a necessidade de qualquer poda [54].

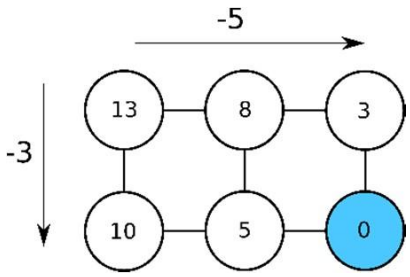


Fig. 4.8 - DFT com um único estado de exaustão [4].

4.1.3.2 Múltiplos Estados de Exaustão

O Exemplo 4.1 e o caso anterior, com $I = (3, 5)$ e $S = 13$, podem ser estendidos ao caso geral de múltiplos estados de exaustão pelo seguinte teorema:

Teorema 4.2: se $s = (b_1 + u).b_2, u = 0, 1, 2, ...$, então a política ótima no estado s mudará de voraz para N_2 quando λ_2/λ_1 exceder $(b_1 + 1 + 2u)/(b_2 - 1)$. Se $s = (b_2 + v)b_1$ com $b_1 \geq 2$, $v = 0, 1, 2, ...$, então a política ótima no estado s mudará de voraz para N_1 quando λ_1/λ_2 exceder $(b_2 + 1 + 2v)/(b_1 - 1)$ [54].

A aplicação recorrente dos melhores resultados entre (4.3), (4.4) e (4.5) em todos os estados da árvore, com o propósito de evitar o impasse ou minimizar a ociosidade finita, pode ser convenientemente programada. As Figuras 4.9, 4.10, 4.11 e 4.12 apresentam algumas planilhas para $I = (5, 11)$ e $S = 160$ com o intuito de ilustrar as implicações do Teorema 4.2. A Figura 4.9 mostra a DFT com cada estado representado apenas pelo seu tamanho vazio, omitindo os ramos por uma questão de legibilidade.

160	149	138	127	116	105	94	83	72	61	50
155	144	133	122	111	100	89	78	67	56	45
150	139	128	117	106	95	84	73	62	51	40
145	134	123	112	101	90	79	68	57	46	35
140	129	118	107	96	85	74	63	52	41	30
135	124	113	102	91	80	69	58	47	36	25
130	119	108	97	86	75	64	53	42	31	20
125	114	103	92	81	70	59	48	37	26	15
120	109	98	87	76	65	54	43	32	21	10
115	104	93	82	71	60	49	38	27	16	5
110	99	88	77	66	55	44	33	22	11	0
105	94	83	72	61	50					
100	89	78	67	56	45					
95	84	73	62	51	40					
90	79	68	57	46	35					
85	74	63	52	41	30					
80	69	58	47	36	25					
75	64	53	42	31	20					
70	59	48	37	26	15					
65	54	43	32	21	10					
60	49	38	27	16	5					
55	44	33	22	11	0					
50										
45										
40										
35										
30										
25										
20										
15										
10										
5										
0										

Fig. 4.9 - Tamanhos dos vazios na DFT para $I = (5, 11)$ e $S = 160$ [54].

As Figuras 4.10, 4.11 e 4.12 mostram as políticas ótimas correspondentes e as configurações ótimas da DFT para três valores de $\beta = \lambda_2/\lambda_1$. As políticas não vorazes N_1 e N_2 são representadas por "↓" e "→→→", respectivamente, enquanto a política voraz é representada por "↓→".

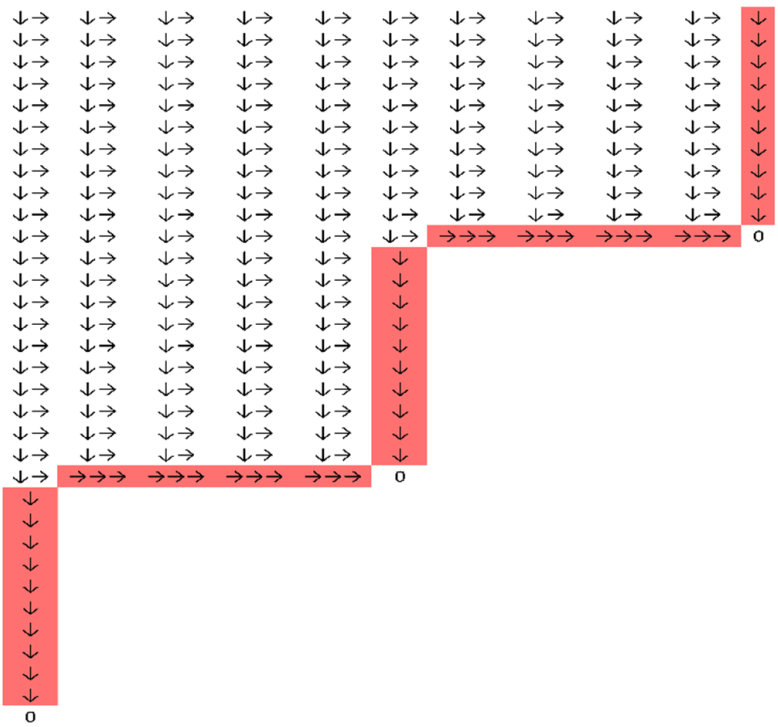


Fig. 4.10 - Política ótima para $1/3 \leq \beta = 0{,}5 \leq 1{,}2$ [54].

A Figura 4.10 é válida para $\beta = 0{,}5$, que está entre $\frac{b_1-1}{b_2+1} = 1/3$ e $\frac{b_1+1}{b_2-1} = 0{,}6$ e, portanto, a política voraz é ótima em todos os estados, exceto os múltiplos de b_1 e b_2 abaixo de $b_1 b_2 = 55$, que são mostrados em vermelho. Quando β atinge 0,6 em ordem crescente, ambos os estados 55 mudam para a política N_2, cortando assim o estado de exaustão $0(32{,}0)$ e todos os estados que o tinham como o único estado de exaustão alcançável. Quando β alcança 0,8, os estados 66 também mudam para a política N_2, assim como os estados 77 quando β atinge 1, resultando na configuração da Figura 4.11, em que os estados que mudaram para uma política não voraz com o propósito de minimizar a ociosidade finita são mostrados em amarelo [54].

Demonstração: o Exemplo 4.1 prova o Teorema 4.2 para $u = v = 0$. A partir de $s = (b_1 + u)b_2$, assume-se que o Teorema 4.2 é verdadeiro para $u = 1, 2, \ldots, k-1$, e a prova é feita para $u = k \geq 1$. Uma vez que o Teorema 4.2 é assumido verdadeiro para todo $u \leq k-1$, então todos os múltiplos de b_2 até $(b_1 + k - 1)b_2$ seguem a política ótima N_2, conforme mostra a Figura 4.11 para $I = (5, 11)$ e $S = 160$, em que o estado s é apresentado (em azul) com a pergunta em investigação aqui [54].

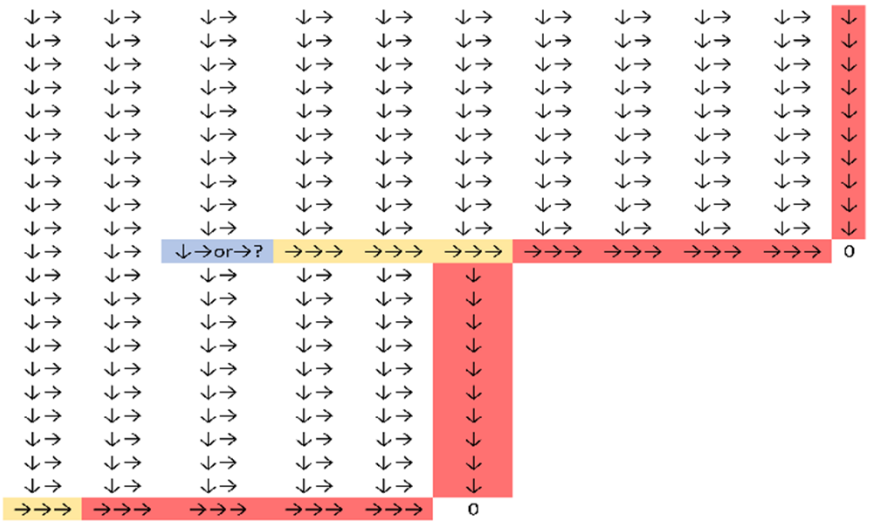


Fig. 4.11 - Política ótima para $1 \leq \beta \leq 1{,}2$ [54].

Finalmente, quando β atinge 1,6, o estado 110 muda para a política N_2, cortando o estado de exaustão $0(21{,}5)$ e todos os estados restantes abaixo de 110, o que resulta na configuração mostrada na Figura 4.12, válida para $\beta \geq 1{,}6$ [54].

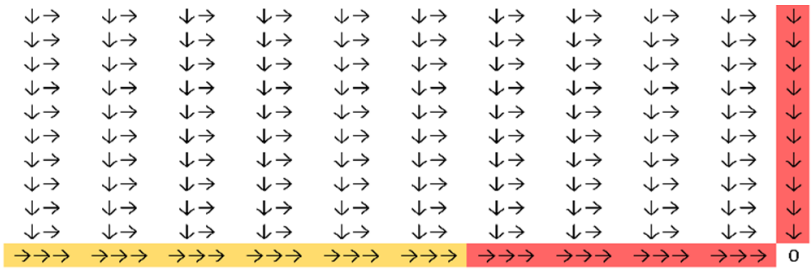


Fig. 4.12 - Política ótima para $\beta \geq 1{,}6$ [54].

Perceba que as Figuras 4.9 a 4.12 também se aplicam ao caso em que $I = (10, 22)$ e $S = 320$. Quando $\beta = \lambda_2/\lambda_1$ ultrapassa o valor de $(b_1 + 1)/(b_2 - 1) = 0{,}6$, os estados sucessivos dados por múltiplos de b_2 acima de $b_1 b_2$ mudarão da política voraz para N_2 em intervalos de $2/(b_2 - 1) = 0{,}2$ de β. Sempre que a política N_2 abranger uma linha completa de múltiplos de b_2 da direita para a esquerda na árvore, o estado de exaustão abaixo dessa linha, assim como todos os estados que o tinham como o único estado alcançável, tornam-se inalcançáveis a partir do estado S, de modo que eles são descartados da DFT [54].

O estado $s = (b_1 + k)b_2$ verá, então, apenas dois estados de exaustão: $0(0, b_1 + k)$ e $0(b_2, k)$, como mostrado na Figura 4.11. Além disso, a transição para $(s - b_2)$ é necessária para atingir um ou outro estado de exaustão e, por isso, não é rejeitada pela política ótima de acordo com o Teorema 4.1. A transição de s para $(s - b_1)$, no entanto, só é necessária para alcançar $0(0, b_1 + k)$, mas não $0(b_2, k)$ e, por isso, pode ou não ser proibida. Portanto, há apenas duas políticas a serem testadas para a otimização: voraz e N_2 [54].

Como o estado $(s - b_1)$ tem apenas um estado de exaustão alcançável dado por $0(b_2, k)$, então a sua política ótima é voraz e sua ociosidade espectral média pode ser derivada de (4.14), resultando em [54]:

$$\Omega_{min}(s - b_1) = \Omega_G(s - b_1) = \frac{b_2(b_2-1)b_1}{2\lambda_1} + \frac{k(k+1)b_2}{2\lambda_2} \tag{4.15}$$

Uma vez que todos os múltiplos de b_2 até $(s - b_2) = (b_1 + k - 1)b_2$ são assumidos como seguindo a política ótima N_2, então ao invocar (4.14) com $n_1 = 0$ e $n_2 = (b_1 + k - 1)$ tem-se [54]:

$$\Omega_{min}(s - b_2) = \Omega_{N_2}(s - b_2) = \frac{(b_1+k-1)(b_1+k)b_2}{2\lambda_2} \tag{4.16}$$

$\Omega_{N_2}(s)$ pode, então, ser derivado de (4.14) ou (4.4) e (4.12), como:

$$\Omega_{N_2}(s) = \frac{(b_1+k)(b_1+k+1)b_2}{2\lambda_2} \tag{4.17}$$

Finalmente, $\Omega_G(s)$ pode ser derivado de (4.5), (4.15) e (4.16):

$$\Omega_G(s) = \frac{(b_1+k+1)(b_1+k)b_2+b_2(b_2-1)b_1+k(k+1)b_2(\lambda_1/\lambda_2)}{2(\lambda_1/\lambda_2)} \tag{4.18}$$

A comparação entre (4.17) e (4.18) mostra que:

$$\Omega_{min}[(b_1+k)b_2] = \Omega_{N_2}[(b_1+k)b_2] \quad se \quad \frac{\lambda_2}{\lambda_1} \geq \frac{b_1+2k+1}{b_2-1} \tag{4.19}$$

Analogamente, para $s = (b_2+k)b_1$, a mesma abordagem pode ser usada para mostrar que:

$$\Omega_{min}[(b_2+k)b_1] = \Omega_{N_1}[(b_2+k)b_1] \quad se \quad \frac{\lambda_1}{\lambda_2} \geq \frac{b_2+2k+1}{b_1-1} \tag{4.20}$$

Além disso, para todos os valores de s múltiplos de b_2 acima de b_1b_2, exceto os múltiplos de b_1b_2:

$$\Omega_{min}[(b_1+k)b_2] = \Omega_G[(b_1+k)b_2] \quad se \quad \frac{\lambda_2}{\lambda_1} \leq \frac{b_1+2k+1}{b_2-1} \tag{4.21}$$

E para todos os valores de s múltiplos de b_1 acima de b_1b_2, exceto os múltiplos de b_1b_2:

$$\Omega_{min}[(b_2+k)b_1] = \Omega_G[(b_2+k)b_1] \quad se \quad \frac{\lambda_1}{\lambda_2} \leq \frac{b_2+2k+1}{b_1-1} \tag{4.22}$$

com o limite inferior do intervalo de otimalidade $\gamma = \beta^{-1}$ para a política voraz sendo zero para $b_1 = 1$, de modo que a política N_1 nunca será ótima sob qualquer perfil de tráfego se $b_1 = 1$. Múltiplos de b_1b_2 são múltiplos de b_1 e b_2, então ambas as equações (4.19) e (4.20) podem ser aplicadas, mas com valores diferentes de k para o mesmo s. Então, referente a ambos os argumentos de Ω_{min} para o mesmo estado $s = qb_1b_2$, obtém-se o intervalo de validade da política voraz em $s = qb_1b_2$, em que q é qualquer inteiro positivo [54]:

$$\Omega_{min}[qb_1b_2] = \Omega_G[qb_1b_2] \quad se \quad \frac{b_1-1}{(2q-1)b_2+1} \leq \frac{\lambda_2}{\lambda_1} \leq \frac{(2q-1)b_1+1}{b_2-1} \tag{4.23}$$

Note que (4.23) é válida também para $b_1 = 1$, uma vez que leva a região de otimalidade da política voraz até $\beta = 0$, não deixando espaço para a otimalidade de N_1. As equações (4.19) a (4.23) provam o Teorema 4.2 por indução finita [54].

As Figuras 4.13 e 4.14 mostram o tempo total de *slot* ocioso esperado em todos os estados remanescentes da DFT, mostrados na Figura 4.9 com $\beta = 1{,}1$, quando as políticas de atribuição são (a) voraz e (b) ótima.

83,1	72,9	63,69	55,6	48,5	42,5	38	33,5	30,5	28,5	27,5
78,3	68,1	58,81	50,6	43,5	37,5	33	28,5	25,5	23,5	22,5
74,1	63,8	54,5	46,2	39,1	33	28	24	21	19	18
70,5	60,2	50,79	42,4	35,2	29	24	20	17	15	14
67,3	57,1	47,69	39,2	31,8	25,6	21	16,5	13,5	11,5	10,5
64,4	54,6	45,24	36,6	29	22,6	18	13,5	10,5	8,5	7,5
61,8	52,4	43,39	34,8	27	20,3	15	11	8	6	5
59	50,5	42,05	33,7	25,7	18,6	13	9	6	4	3
55,9	48,4	40,93	33,3	25,4	17,8	12	7,5	4,5	2,5	1,5
52,1	45,7	39,56	33,2	26,2	18,3	11	6,5	3,5	1,5	0,5
47,5	42,1	37,27	32,7	27,7	21	10	6	3	1	0
42,5	37,5	33,5	30,5	28,5	27,5					
37,5	32,5	28,5	25,5	23,5	22,5					
33	28	24	21	19	18					
29	24	20	17	15	14					
25,6	20,5	16,5	13,5	11,5	10,5					
22,6	17,5	13,5	10,5	8,5	7,5					
20,3	15	11	8	6	5					
18,6	13	9	6	4	3					
17,8	11,5	7,5	4,5	2,5	1,5					
18,3	10,5	6,5	3,5	1,5	0,5					
21	10	6	3	1	0					
27,5										
22,5										
18										
14										
10,5										
7,5										
5										
3										
1,5										
0,5										
0										

Fig. 4.13 - Tempo de *slot* ocioso esperado para a DFT voraz [53].

82,5	72,5	63,5	55,5	49	43	38	34	31	29	28
77,4	67,5	58,5	50,5	44	38	33	29	26	24	23
72,9	63	54	46	39	33	28	24	21	19	18
68,8	59	50	42	35	29	24	20	17	15	14
65,2	55,4	46,5	38,5	32	26	21	17	14	12	11
61,9	52,3	43,5	35,5	29	23	18	14	11	8,5	7,5
59	49,7	40,9	33	26	20	15	11	8	6	5
56,2	47,3	38,9	31	24	18	13	9	6	4	3
53,4	45,3	37,2	29,5	23	17	12	7,5	4,5	2,5	1,5
50,4	43,2	35,9	28,5	22	16	11	6,5	3,5	1,5	0,5
46,9	40,8	34,8	28	21	15	10	6	3	1	0
42,5	37,5	33,5	30,5	29	28					
37,5	32,5	28,5	25,5	24	23					
33	28	24	21	19	18					
29	24	20	17	15	14					
25,5	20,5	16,5	13,5	12	11					
22,5	17,5	13,5	10,5	8,5	7,5					
20	15	11	8	6	5					
18	13	9	6	4	3					
16,5	11,5	7,5	4,5	2,5	1,5					
15,5	10,5	6,5	3,5	1,5	0,5					
15	10	6	3	1	0					
27,5										
22,5										
18										
14										
10,5										
7,5										
5										
3										
1,5										
0,5										
0										

Fig. 4.14 - Tempo de *slot* ocioso esperado para a DFT ótima [53].

Algumas observações merecem destaque [53]:

a) em todos os estados que apresentam apenas um estado de exaustão alcançável, as duas políticas produzem o mesmo resultado, conforme implícito em (4.14);

b) em todos os estados restantes, as políticas ótimas geram resultados menores para o tempo de slot ocioso esperado do que a DFT voraz, com a penalidade máxima de 40% no estado 55, conforme destacado em amarelo;

c) os estados $s = 55, 66$ e 77, que são os únicos em que o Teorema 4.2 requer a política N_2 para $1 < \beta < 1{,}2$, também são os únicos nos quais a ociosidade ótima esperada é menor do que no estado $(s - b_1)$, implicando que os números na Figura 4.14 podem ser usados como valores de estado-ação para a minimização da ociosidade dos *slots*, conforme preconizado em [55].

4.2 POLÍTICAS NÃO VORAZES

Ao evitar a transição para os elementos do conjunto de impasse por meio do uso de políticas não vorazes, é possível mitigar a ocorrência permanente de *slots* residuais no espectro, evitando assim que a ociosidade tenda ao infinito. Dessa forma, é importante caracterizar e determinar o conjunto de tamanhos vazios que levam ao impasse [51].

Se $s(\infty) > 0$, a ociosidade total será infinita, produzindo assim um impasse perdulário. Portanto, a minimização da ociosidade deve se concentrar primeiro em evitar impasses, sempre que possível. Se $s_{min}(\infty)$ é o valor mínimo alcançável de $s(\infty)$, então o seguinte lema se aplica [51]:

Lema 4.1: se $\lambda_1 > 0$, então $s(\infty) = 0$ é alcançável para qualquer S.

Isso significa que o sistema estará livre de impasses se houver demanda por *slots* únicos. Na prática, no entanto, essa situação não é provável de ocorrer, devido à necessidade operacional de permitir faixas espectrais de guarda entre conexões vizinhas no espaço espectral, resultando na necessidade de incluir essas faixas de guarda na largura de banda solicitada [51].

Além disso, a demanda pode se concentrar em um conjunto finito de serviços que geram uma demanda por um conjunto finito de larguras de banda discretas. Por esta razão, em [51] os autores investigam os tamanhos de vazio que inevitavelmente levarão o sistema a um impasse, em função do perfil de tráfego de entrada. Para isso, considere que:

$$\lambda_i \begin{cases} > 0 \; para \; i \in \{i_1, i_2, \dots, i_k\}, com \; 1 < i_1 < i_2 < \cdots i_k, \\ = 0, \qquad\qquad\qquad\qquad\qquad\quad caso \; contrário. \end{cases} \tag{4.24}$$

Para manter a notação curta, considere $\alpha_k = \lambda_{i_k} > 0$. Além disso, que $M_k, k \in \{0,1,2, \dots, b_1 - 1\}$ seja o conjunto de todos os inteiros não negativos, em ordem crescente, que são congruentes a $k(mod b_1)$ [51]:

$$M_k = \{k, k + b_1, k + 2b_1, \dots\}, \;\; k = 0, 1, 2, \dots, b_1 - 1. \tag{4.25}$$

Seja $I = \{i|\lambda_i > 0\}$, $K = |I|$, e $g = mdc(I) =$ máximo divisor comum de todos os elementos de I. Dado o conjunto I e o tamanho s de um vazio espectral, deseja-se investigar se

o impasse pode ser evitado no processo de alocação espectral pela rejeição apropriada de algumas requisições que podem levar a um impasse caso sejam aceitas. Os tamanhos dos vazios que, inevitavelmente, levam a um impasse serão chamados de tamanhos proibidos, e $F(I)$ é o conjunto de todos os tamanhos proibidos.

Obviamente, todos os tamanhos em $\{1, 2, \dots, b_1 - 1\}$ são proibidos, com o tamanho mínimo de conexão solicitado pelo tráfego igual a b_1 [51]. Porém, quantos outros tamanhos proibidos existem e quais são? A discussão para responder a essa pergunta é conduzida por meio da análise de alguns casos especiais, apresentados a seguir.

Caso 4.1: $K = 1$ e $b_1 \geq 2$. Nesse caso, b_1 é o único tamanho de conexão exigido pelo tráfego de entrada. Portanto, $s(\infty) = s(mod b_1)$ e um resíduo positivo surgirá, a menos que s seja um múltiplo de b_1. Então, $|F(I)| = \infty$ e $F(I)$ é dado por [51]:

$$F(I) = U_{i=1}^{b_1 - 1} M_i \tag{4.26}$$

Caso 4.2: $K = 2$, com b_1 e b_2 primos entre si. Seja $p_m = mb_2(mod b_1) = mp_1(mod b_1)$, $m = 1, 2, \dots, b_1 - 1$. Então, $mb_2 = \left\lfloor \frac{mb_2}{b_1} \right\rfloor b_1 + p_m$, e o impasse com resíduo p_m ocorrerá se $s = p_m, p_m + b_1, \dots, p_m + \left\{ \left\lfloor \frac{mb_2}{b_1} \right\rfloor - 1 \right\} b_1$, mas é evitável para todos os tamanhos de vazio sucessivos em M_{p_m}, ou seja, para $p_m + \left\{ \left\lfloor \frac{mb_2}{b_1} \right\rfloor \right\} b_1 = mb_2$, $p_m + \left\{ \left\lfloor \frac{mb_2}{b_1} \right\rfloor + 1 \right\} b_1 = mb_2 + b_1$, ... Assim, apenas um número finito de elementos de M_{p_m} causará um impasse inevitável [51].

Uma vez que M_0 não contém nenhum tamanho proibido e $M_0 \cup M_{p_1} \cup M_{p_2} \cup \dots \cup M_{p_{b_1 - 1}}$ abrange todo o conjunto de inteiros não negativos, pode-se concluir que apenas um conjunto finito de tamanhos de vazio causará um bloqueio inevitável sob tráfego incremental.

Observe que $(M_0, M_{p_1}, M_{p_2}, \dots, M_{p_{b_1 - 1}})$ é uma permutação de $(M_0, M_1, M_2, \dots, M_{b_1 - 1})$. Portanto, o processo de preenchimento espectral pode evitar impasses se o tamanho do vazio for pelo menos $d_{max} + 1 = (b_1 - 1)(b_2 - 1)$. Abaixo desse tamanho de vazio, o *deadlock* é inevitável se o tamanho do vazio pertencer a $F(I)$ [51].

Exemplo 4.2: $I = \{4, 7\}$. Se D_i é o conjunto de tamanhos proibidos que pertencem a M_i, então $D_0 = \emptyset$, $D_1 = \{1, 5, 9, 13, 17\}$, $D_2 = \{2, 6, 10\}$ e $D_3 = \{3\}$, então $|F(I)| = 9$, e o impasse pode ser evitado com qualquer vazio que tenha pelo menos $18 = (4 - 1)(7 - 1)$ *slots*. A fim de evitar o impasse, todas as conexões que, se aceitas, reduziriam o vazio a um dos nove tamanhos proibidos listados acima devem ser negadas [51].

A Figura 4.15 mostra um mapeamento desses conjuntos nos conjuntos de inteiros congruentes módulo 4, e sua combinação em $F(4, 7)$. Nela, os pontos marcados em vermelho representam os valores pertencentes ao conjunto de impasse indicado à esquerda da respectiva linha horizontal.

Apenas o conjunto M_0 não apresenta pontos marcados em vermelho, pois todos os seus elementos são múltiplos de $b_1 = 4$, portanto podem ser preenchidos sem deixar resíduo. Nos conjuntos M_1, M_2 e M_3, o primeiro ponto marcado em azul da esquerda para a direita é o primeiro múltiplo de $b_2 = 7$ no respectivo conjunto. Por isso, tanto ele como todos os pontos à sua direita também representam tamanhos de vazios que podem ser preenchidos sem deixar resíduo, ao contrário dos que estão à sua esquerda.

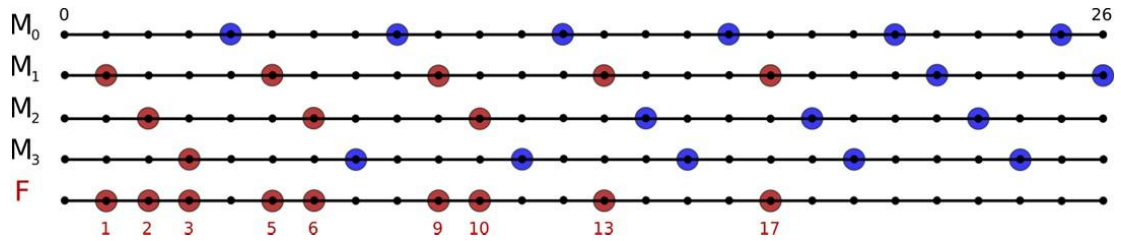


Fig. 4.15 - Mapeamento dos conjuntos de congruência do módulo 4 em $F(4, 7)$ [51].

Outra forma gráfica de visualização, mais conveniente para o caso em que $K = 2$, pode ser gerada a partir da construção de uma árvore de estados. Para exemplificar, considere um enlace avulso de 400 km que recebe requisições por 5 e 11 *slots*, com chegadas poissonianas, e que deve suportar taxas de transferência de 400 Gbps e 1 Tbps, respectivamente [4].

A Figura 4.16 mostra a sequência dos possíveis estados para um enlace avulso e seus respectivos tamanhos de vazio, com $I = (5, 11)$ e $S = 55$. Os estados assinalados em azul escuro representam os estados em impasse, os estados em azul claro indicam os estados predestinados ao impasse, os estados em branco correspondem aos estados que podem levar ao impasse caso seja utilizada uma política voraz de atribuição espectral e os estados em laranja assinalam os estados em que a exaustão do espectro está garantida [4].

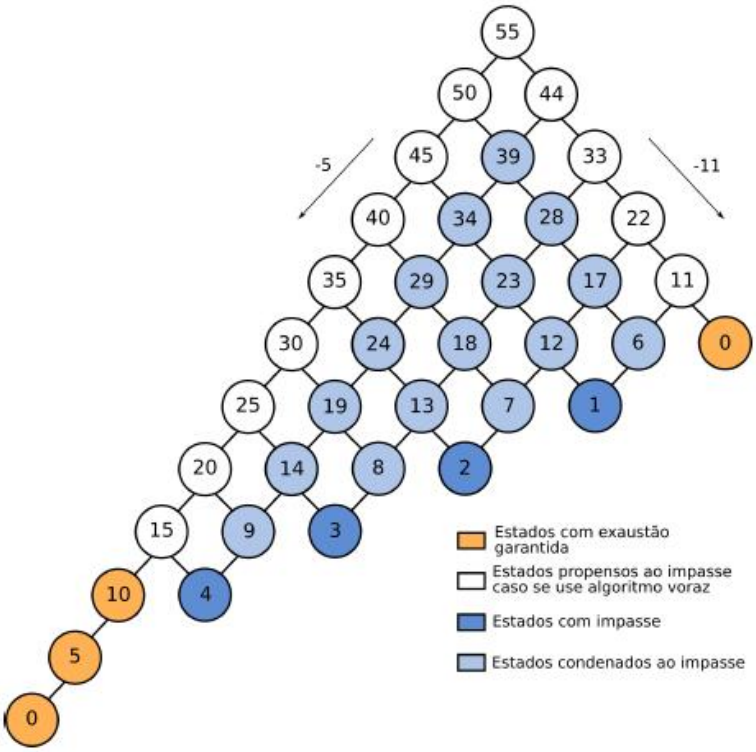


Fig. 4.16 - Árvore de estados para $I = (5, 11)$ e $S = 55$ [4].

Exemplo 4.3: no caso 4.2 observou-se que com $K = 2$, $b_1 = a$ e $b_2 = b$ primos entre si, dado um conjunto de tamanhos $I = (a, b)$ e S o número total de *slots* disponíveis no espectro, o conjunto de estados que levam ao impasse é dado por $d_{max} = (a-1)b - a = ab - (a+b)$ e todos os tamanhos positivos que resultam a partir de d_{max} por meio da subtração de valores múltiplos de a e b. Portanto, o conjunto dos estados nas cores azul claro e azul escuro na Figura 4.16 representam todos os elementos do conjunto de impasse $D(5, 11) = \{39, 34, 29, 28, 24, 23, 19, 18, 17, 14, 13, 12, 9, 8, 7, 6, 4, 3, 2, 1\}$, dado que $d_{max} = 55 - (5 + 11) = 39$ [4].

O exemplo numérico, apresentado na Figura 4.16, sugere que os elementos da árvore gerada por $(ab - a - b)$ por meio de sucessivas subtrações de a e b *slots* pertencem ao conjunto de impasse. O Apêndice A em [4] apresenta a prova matemática deste enunciado, e vai além, pois mostra que a árvore gerada por $(ab - a - b)$ compõe todos os elementos do conjunto de impasse $D(I)$.

Lema 4.2: sendo $K = 2$ e $I = (a, b)$ com a e b primos entre si, o conjunto de impasse é dado por: $D(a, b) = \{n | n = ab - n_a a - n_b b, n \in \mathbb{N}, n_a \in \mathbb{N}, n_b \in \mathbb{N}\}$, sendo que $\mathbb{N}$ representa o conjunto dos números inteiros positivos [4].

Exemplo 4.4: todos os valores restantes de S permitirão a exaustão do espectro, mas para garantir essa exaustão, um algoritmo não voraz, que rejeita todas as transições para elementos do conjunto de impasse, deve ser usado. A Figura 4.17 ilustra esse requisito para $I = (5, 11)$ e $S = 53$. Repare que, neste caso, a prevenção de impasse exige que os pedidos de 5 *slots* sejam rejeitados quando o tamanho do vazio for 33, 22 ou 11, mesmo que existam recursos para acomodá-los; e as solicitações de 11 *slots* também devem ser rejeitadas quando o tamanho do vazio for 15 ou 20 [4].

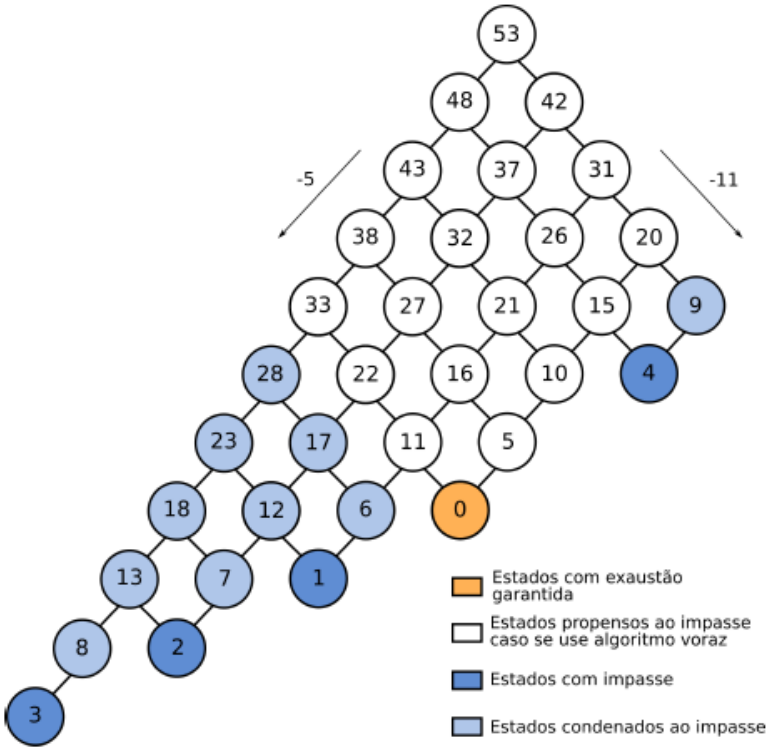


Fig. 4.17 - Árvore de estados para $I = (5, 11)$ e $S = 53$ [4].

Caso 4.3: $K \geq 3$ e todos os elementos de I são primos entre si. Então, todos os tamanhos de vazio que permitem evitar o impasse no caso anterior ainda são capazes de evitá-lo. Além disso, se nenhum elemento de I é um elemento de $F(\{b_1, b_2\})$, então $F(I) = F(\{b_1, b_2\})$. Se um elemento de I é membro de $F(\{b_1, b_2\})$, então $F(I)$ resulta de $F(\{b_1, b_2\})$, por meio da remoção desse elemento e de todos os elementos sucessivos do conjunto M_i ao qual ele pertence. Assim, $(b_1 - 1)(b_2 - 1)$ é um limite superior para o tamanho mínimo do vazio que é maior do que todos os tamanhos proibidos [51].

Exemplo 4.5: $I = \{4, 7, 9\}$. Neste caso, $D_0 = \emptyset$, $D_1 = \{1, 5\}$, $D_2 = \{2, 6, 10\}$ e $D_3 = \{3\}$, então $|F(I)| = 6$, e $11 < 18$ é o tamanho mínimo do vazio que é maior do que todos os tamanhos proibidos. A Figura 4.18 mostra um mapeamento desses conjuntos nos conjuntos de inteiros congruentes do módulo 4, e sua combinação em $F(4, 7, 9)$ [51].

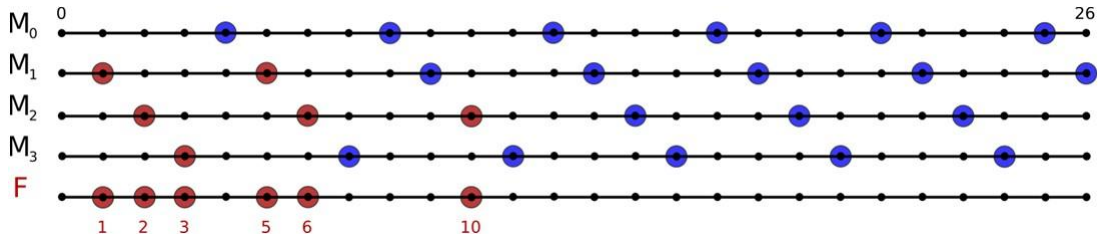


Fig. 4.18 - Mapeamento dos conjuntos de congruência do módulo 4 em $F(4, 7, 9)$ [51].

Caso 4.4: $K \geq 2$, $g > 1$. Neste caso, todas as solicitações são para vários números de *slots* g, portanto, podemos tratar g como o tamanho de um slot macro e ver solicitações de b_i *slots* como solicitações para $y_i = b_i/g$ *slots* macro. Então, todos os tamanhos de vazio que não são múltiplos de g gerarão um impasse. Além disso, se $b_1 = g$, nenhum tamanho vazio múltiplo de g será proibido (Lema 1). Caso contrário, o impasse (*deadlock*) pode ser evitado se s for um múltiplo grande o suficiente de g, com $(b_1 - g)(b_2 - g)/g$ sendo um limite superior para o tamanho mínimo do vazio que é maior do que todos os tamanhos proibidos [51].

Exemplo 4.6: $I = \{8, 14\}$. Neste caso, $g = 2$, e o impasse é evitável a partir de qualquer tamanho de vazio que seja pelo menos $36 = (8 - 2)(14 - 2)/2$ [51].

Exemplo 4.7: $I = \{8, 14, 18\}$. Agora, o impasse é evitável para qualquer tamanho de vazio que seja de pelo menos $22 < 36$ [51].

O impasse é um evento extremo, no qual o tempo médio para preencher o espectro tende ao infinito. Evitar o impasse apenas mantém esse tempo médio finito quando possível. Assim, é necessário estabelecer critérios por meio dos conjuntos de impasse, para o uso das políticas vorazes e não vorazes, com o objetivo de minimizar a ociosidade espectral e otimizar o uso do espectro em um enlace avulso sob tráfego incremental [51].

5 ALGORITMOS CIENTES DO TRÁFEGO DINÂMICO

As EONs sob condições dinâmicas de tráfego, nas quais os futuros pedidos de largura de banda não são conhecidos com antecedência, têm sido amplamente investigadas na literatura. Em condições dinâmicas de tráfego, os términos das conexões criam novos vazios espectrais que não podem ser evitados, de modo que a fragmentação espectral pode causar perdas adicionais. A fragmentação dos recursos espectrais ópticos é um problema importante e inevitável, que faz com que alguns *slots* permaneçam ociosos por um longo tempo, reduzindo assim a eficiência do uso do espectro.

Neste capítulo é apresentado um novo algoritmo de alocação espectral, cujos critérios utilizados para a seleção e alocação do espectro priorizam determinados tipos de vazios, tratados como "inflexíveis", que somente podem ser preenchidos por uma determinada combinação das conexões de entrada. Por outro lado, os vazios que são capazes de suportar duas ou mais dessas combinações, classificados como "flexíveis", fornecem mais flexibilidade para acomodar oscilações na mescla das requisições de chegada. Ao priorizar a alocação das vagas inflexíveis de mesmo tipo de cada requisição ao longo de todo espectro, sem privilegiar a priori nenhuma região do mesmo, é esperado um maior sucesso na exploração da eventual flexibilidade no atendimento das requisições.

5.1 DELINEAMENTO DO ESTUDO E TEMÁTICA DA PESQUISA

5.1.1 Vazios Espectrais

Considere um único enlace com S *slots* que recebe solicitações por conexões de b_1 e b_2 *slots*, com $1 < b_1 < b_2$. A chegada dos pedidos é poissoniana com taxas λ_1 e λ_2, respectivamente. As durações de todas as conexões são independentes e distribuídas de forma idêntica (*i.i.d.*) com uma distribuição exponencial de média 1, assim todas as taxas são expressas em erlangs.

Um vazio espectral, ou apenas vazio por brevidade, é um bloco de *slots* ociosos que é adjacente em ambos os lados a uma conexão ativa ou a uma extremidade do espectro [56]. A fragmentação espectral em um único enlace é caracterizada pela existência de dois ou mais vazios no espectro. A capacidade de um único vazio com tamanho v para acomodar uma sequência de pedidos de b_1 e/ou b_2 *slots* contíguos é, aproximadamente, dependente de seu

tamanho e do perfil de tráfego, mas mais acentuadamente do número i de soluções inteiras não negativas para n_1 e n_2 na seguinte equação [53]:

$$b_1 n_1 + b_2 n_2 = v \tag{5.1}$$

Se b_1 e b_2 são primos entre si, a identidade de Bézout [57] garante a existência de soluções inteiras para n_1 e n_2 em (5.1), mas elas não são necessariamente não negativas, então i pode ser zero em alguns casos. Seja $\sigma_i(b_1, b_2)$ o conjunto de todos os tamanhos vazios para os quais o número de soluções não-negativas para n_1 e n_2 em (5.1) é igual a $i \in \{0, 1, 2, 3, \dots\}$.

A Figura 5.1 mostra todos os pontos $(n_1 b_1, n_2 b_2)$ para $(b_1, b_2) = (4, 7)$ em um plano (x, y) representando as ocupações parciais por conexões de 4 *slots* e 7 *slots*, respectivamente, ao longo dos eixos x e y em um espectro com $S = 84$ *slots*. Os pontos são sobrepostos em linhas retas que representam o *locus* de todos os pontos com a mesma ocupação total v para todos os inteiros v de 0 a 84 [53].

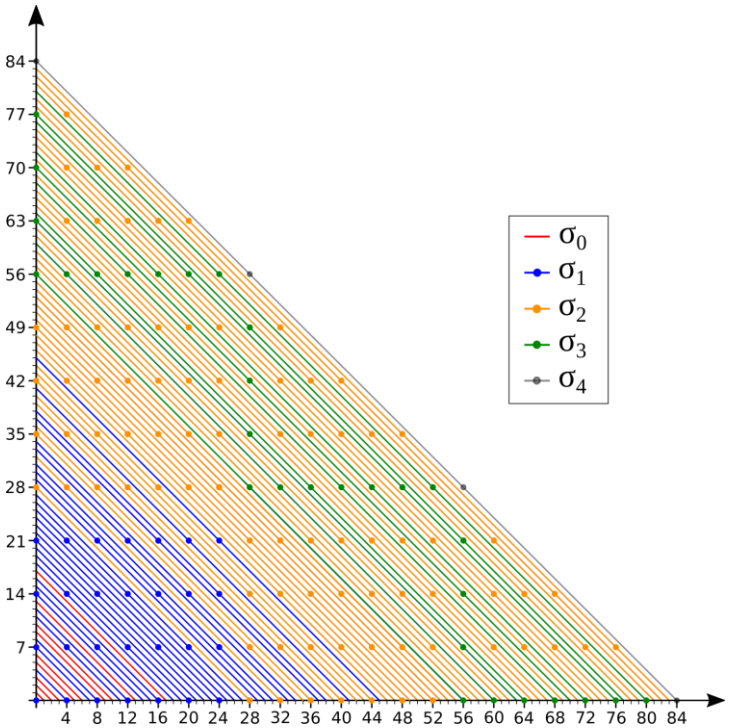


Fig. 5.1 - Tamanhos de vazios e números de soluções completas de enchimento [53].

As linhas vermelhas não contêm nenhum ponto $(n_1 b_1, n_2 b_2)$, por isso correspondem a tamanhos de vazios v que não oferecem nenhuma solução para (5.1), ou seja, vazios que não podem **ser completamente preenchidos** com conexões de b_1 e b_2 *slots* para

$(b_1, b_2) = (4, 7)$. Em outras palavras, as linhas vermelhas correspondem a todos os valores de v em $\sigma_0(4, 7)$ [53].

Da mesma forma, as linhas azuis, que contêm apenas um estado cada, representam todos os tamanhos de vazios em $\sigma_1(4, 7)$; as linhas laranjas, que contêm dois estados cada, expressam todos os tamanhos de vazios em $\sigma_2(4, 7)$; e as linhas verdes indicam alguns tamanhos de vazios em $\sigma_3(4, 7)$ que estão abaixo de 84 [53].

Finalmente, uma única linha cinza representa a ocupação total dos $S = 84 = 3.(4 * 7)$ *slots*, que é o primeiro elemento de $\sigma_4(4, 7)$. Uma configuração mais realista iria até $S = 320$, incluindo assim vazios de $\sigma_0(4, 7)$ até $\sigma_{12}(4, 7)$ [53].

5.1.1.1 Vazios Disfuncionais

Os vazios de $\sigma_0(b_1, b_2)$ são considerados disfuncionais porque não há uma combinação de alocações incrementais de b_1 e b_2 *slots* que possam proporcionar a ocupação total do vazio. A tentativa de preencher um vazio de σ_0 com tais pedidos, conforme eles chegam, irá resultar em um impasse com um resíduo de $1, 2, \ldots$, ou $(b_1 - 1)$ *slots*, como mostrado na Figura 5.2 para $(b_1, b_2) = (4, 7)$ [53].

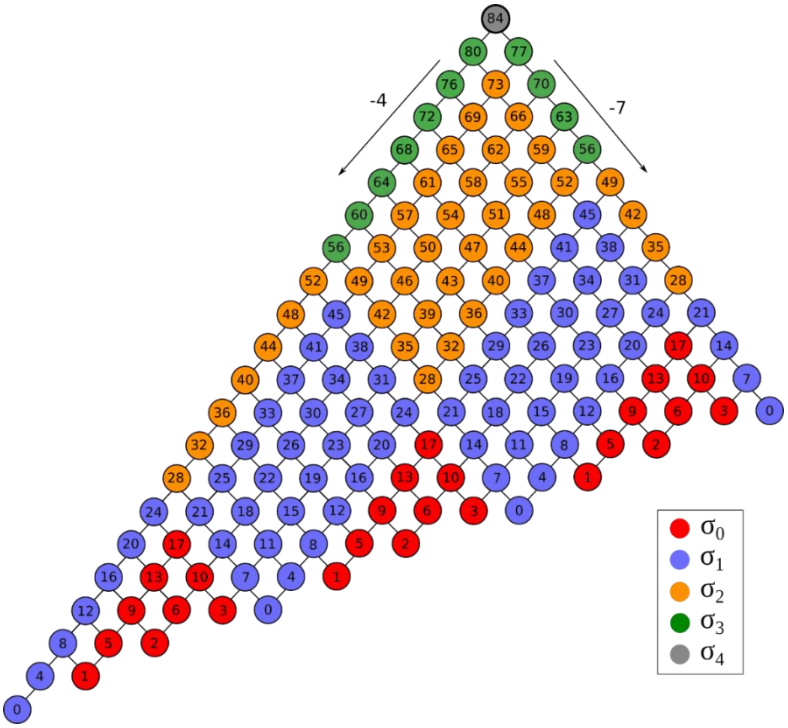


Fig. 5.2 - Árvore de tamanhos de vazios a partir de 84, gerada por atribuições de conexões de 4 e 7 *slots*, colorida de acordo com os membros do conjunto $\sigma_i(4, 7)$ [53].

Esses *slots* residuais permanecerão ociosos até, pelo menos, o término de um número suficiente de conexões adjacentes sucessivas para levar o vazio do conjunto σ_0 para σ_1, o que requer um tempo médio de 1 para cada terminação [53].

Como as taxas de chegada λ_1 e λ_2 normalmente atingirão dezenas de erlangs, a fim de ocupar uma fração considerável dos 320 *slots*, geralmente disponíveis na faixa de ganho dos amplificadores ópticos, os *slots* residuais podem permanecer inutilizáveis por um longo tempo em relação ao tempo entre chegadas, gerando perdas duradouras. Por essa razão, a alocação de qualquer requisição a um vazio de σ_0 deve ser considerada apenas como último recurso [53].

Além disso, a alocação de requisições para qualquer vazio de σ_1 que o reduza a um vazio de σ_0 também deve ser feita apenas como último recurso, ou seja, somente se nenhum outro vazio for capaz de acomodar a solicitação. Como visto na Figura 5.2, isso pode acontecer quando uma requisição de b_1 *slots* for atribuída a vazios que são múltiplos de b_2 inferiores a $b_1 b_2$, ou quando uma solicitação de b_2 *slots* for atribuída a vazios que são múltiplos de b_1 menores que $b_1 b_2$. Os vazios de σ_1 restantes podem acomodar qualquer solicitação sem qualquer restrição [53].

Para a conservação da funcionalidade, os vazios disfuncionais são usados para atribuição apenas quando nenhum outro tamanho de vazio está disponível, de modo que eles possam ter a maior chance possível de recuperar a funcionalidade por meio da coalescência com vazios vizinhos gerados pelas terminações intervenientes [58]. É, então, dada prioridade à atribuição de vazios funcionais [53], mas de acordo com uma determinada hierarquia de prioridades, detalhada na seção 5.2.1.

Por uma questão de brevidade, a notação σ_i é usada para denotar $\sigma_i(b_1, b_2)$ para qualquer i. Uma vez que os vazios de σ_0 devem ser evitados, então é importante descrevê-los explicitamente. Se b_1 e b_2 são primos entre si, o maior tamanho de vazio em σ_0 é $(b_1 b_2 - b_1 - b_2)$, e os tamanhos restantes são derivados desse tamanho maior por subtração de b_1 e b_2 tantas vezes quanto possível enquanto permanecerem positivos [53]:

$$\sigma_0(b_1, b_2) = \{v = b_1 b_2 - k_1 b_1 - k_2 b_2 > 0 | k_1, k_2 \in \mathbb{N}_1\} \tag{5.2}$$

em que $\mathbb{N}_1$ é o conjunto de todos os números naturais excluindo o zero.

Por exemplo, $\sigma_0(4, 7) = \{17, 13, 10, 9, 6, 5, 3, 2, 1\}$, como visto na Figura 5.2. Se b_1 e b_2 não são primos entre si e g é o seu maior fator comum, com $b_1' = b_1/g$, $b_2' = b_2/g$, e $\sigma_0'(b_1, b_2) = g.\sigma_0(b_1', b_2')$, então [53]:

$$\sigma_0(b_1, b_2) = \sigma_0'(b_1, b_2) \cup \left\{ k \in \mathbb{N} \middle| \left({}^{k}/_{g} \right) \notin \mathbb{N} \right\} \quad (5.3)$$

Repare que, neste caso, todos os membros do conjunto σ_0 são de um conjunto finito de múltiplos de g ou qualquer número que não seja um múltiplo de g. No entanto, se o algoritmo sempre atribui múltiplos de g *slots* contíguos a conexões ativas, como ocorre necessariamente, novos vazios serão sempre múltiplos de g, então o algoritmo pode limitar-se a evitar transições para ou de membros de $\sigma_0'(b_1, b_2)$. Por exemplo, $\sigma_0'(8, 14) = \{34, 26, 20, 18, 12, 10, 6, 4, 2\}$ é o conjunto de tamanhos a serem evitados se $(b_1, b_2) = (8, 14)$ [53].

Todos os vazios que não são disfuncionais são considerados funcionais, mas com uma ressalva. Se o tamanho do vazio for um múltiplo de b_1 menor que $b_1 b_2$, então é um vazio funcional porque pode ser preenchido com solicitações de b_1 *slots*, mas levaria a um vazio disfuncional se usado para acomodar uma solicitação de b_2 *slots*, como pode ser verificado na Figura 5.2. Por essa razão, tais vazios são considerados "funcionais para b_1". Analogamente, múltiplos de b_2 menores que $b_1 b_2$ são chamados "funcionais para b_2".

5.1.1.2 Vazios Inflexíveis

Cada vazio em $\sigma_1(b_1, b_2)$ é capaz de suportar exatamente uma combinação de n_1 conexões com b_1 *slots* e n_2 conexões com b_2 *slots* para um único par (n_1, n_2) em qualquer ordem, enquanto os vazios nos conjuntos superiores são capazes de suportar duas ou mais dessas combinações, fornecendo assim mais flexibilidade para acomodar flutuações na mistura das requisições de chegada [53].

Pode-se pensar em vazios de σ_1 como um número fixo de n_1 "vagas" para conexões de b_1 *slots* e n_2 "vagas" para conexões de b_2 *slots*, e em vazios de σ_i, com $i \geq 2$, como sendo capazes de reorganizar suas vagas para fornecer i configurações para melhor se adequar ao tráfego de entrada. Por essa razão, os vazios de σ_1, embora não disfuncionais, são, entretanto, inflexíveis quando comparados com vazios dos conjuntos superiores [53].

Assim sendo, sempre que houver vazios inflexíveis com vagas inflexíveis do mesmo tipo de uma nova requisição, esses vazios devem ser priorizados para atender a essa requisição e para que a flexibilidade dos vazios flexíveis seja preservada para futuras alocações sem nenhum custo. Essa noção sugere que os vazios de σ_1 sejam priorizados sobre todos os outros vazios na atribuição dos pedidos de chegada [53].

Esta proposição confirma a sabedoria de priorizar as chamadas atribuições *best-fit* ou *exact-fit*, que atribuem pedidos de b_1 *slots* para vazios com $v = b_1$ e de b_2 *slots* para $v = b_2$, uma vez que esses são ambos vazios de σ_1 na fronteira de σ_0 (ver Figura 5.2) [53]. Essas atribuições foram recomendadas em [15], mas como as únicas a serem priorizadas.

No entanto, no contexto de um algoritmo com conhecimento do tráfego, que seja ciente de que o tráfego é composto por apenas duas classes de requisições por b_1 e b_2 *slots*, pode-se ampliar essa abordagem para abranger todos os múltiplos do tamanho da requisição de entrada menores que o produto $b_1 b_2$, priorizando assim todos os vazios que são múltiplos do tamanho solicitado e também inferiores a $b_1 b_2$, como sugerido pela Figura 5.2.

Perceba que esses vazios são os mais inflexíveis em σ_1, pois múltiplos de b_1 devem apenas acomodar pedidos de b_1 *slots*, e múltiplos de b_2 devem apenas acomodar pedidos de b_2 *slots* (exceto como último recurso), enquanto os elementos restantes de σ_1 podem acomodar qualquer tipo de solicitação [53].

Portanto, os múltiplos de b_1 e b_2 inferiores a $b_1 b_2$, sendo os mais inflexíveis dentre os elementos de σ_1, devem ser priorizados entre eles, mas apenas para a acomodação de pedidos que não levam ao conjunto σ_0, ou seja, com múltiplos de b_1 sendo priorizados apenas para a atribuição de requisições de b_1 *slots*, e múltiplos de b_2 sendo priorizados apenas para a atribuição de solicitações de b_2 *slots* [53].

O número de tamanhos vazios em qualquer conjunto σ_i com $i \geq 1$ é $b_1 b_2$, mas a diferença entre o maior e o menor elemento é $(2b_1 b_2 - b_1 - b_2) > b_1 b_2$, porque cada σ_i tem alguma intercalação de elementos com σ_{i-1} e σ_{i+1} (ver Figura 5.1). O menor elemento de cada conjunto σ_i com $i \geq 1$ é $(i-1)b_1 b_2$ e o maior é $(i+1)b_1 b_2 - b_1 - b_2$ [53].

Logo, o primeiro elemento de σ_1 é zero e o maior é $(2b_1 b_2 - b_1 - b_2)$. De zero até $(b_1 b_2 - 1)$, σ_1 inclui todos os números exceto os elementos de σ_0, assim intercalando com σ_0. O vazio de tamanho $b_1 b_2$ é o menor elemento de σ_2. De $(b_1 b_2 + 1)$ até $(2b_1 b_2 - b_1 - b_2)$, σ_1 inclui todas as somas de $b_1 b_2$ com elementos de σ_0, assim intercalando com σ_2 [53]:

$$\sigma_1(b_1, b_2) = \{j \in \mathbb{N} | 0 \leq j \leq b_1 b_2 - 1, j \notin \sigma_0\} \cup \{b_1 b_2 + k | \forall k \in \sigma_0\} \qquad (5.4)$$

5.1.1.3 Vazios Flexíveis

Para todo $i \geq 2$, σ_i pode ser expresso como [53]:

$$\sigma_i(b_1, b_2) = \{b_1 b_2 + j | \forall j \in \sigma_{i-1}(b_1, b_2)\}, i = 2, 3, 4, \ldots \quad (5.5)$$

Como mostrado na Figura 5.1, cada vazio de um conjunto σ_i com $i \geq 2$ pode ser preenchido com $i > 1$ combinações de conexões de b_1 *slots* e b_2 *slots*. Isso oferece a oportunidade de combinar o processo de acomodação com o perfil de tráfego e suas flutuações ocasionais até que o vazio seja reduzido ao conjunto σ_1 [53].

Por outro lado, o processo de preenchimento dos vazios é interminável: sob o equilíbrio estocástico, a taxa média de atribuições é igual à taxa média de terminações de conexão, e cada terminação cria por coalescência um novo vazio dado pela soma do tamanho da conexão rescindida com seus dois vazios adjacentes. Se os vazios adjacentes são zero, o que significa que a conexão terminada era contígua com seus vizinhos, o novo vazio é igual a b_1 ou b_2, e nasce no conjunto σ_1 como um vazio priorizado para a atribuição adequada, possivelmente interrompendo o preenchimento de outros vazios maiores [53].

A fim de entender a relação entre os múltiplos estados associados a um determinado tamanho vazio de um conjunto σ superior, considere o exemplo do tamanho $79 \in \sigma_3(4, 7)$. As três combinações de solicitações de 4 *slots* e 7 *slots* que podem preencher as 79 vagas são dadas pelas seguintes igualdades [53]:

$$79 = 18 * 4 + 1 * 7 = 11 * 4 + 5 * 7 = 4 * 4 + 9 * 7 \quad (5.6)$$

Seguindo (5.6) da esquerda para a direita, pode-se ver que cada igualdade difere da anterior, substituindo um grupo de sete conexões de 4 *slots* por um grupo de quatro conexões de 7 *slots*. Isso significa que tanto as solicitações de 4 *slots* quanto de 7 *slots* podem ser atribuídas livremente ao vazio de 79 *slots*, mesmo quando ele é reduzido a um múltiplo de b_1 ou b_2 maior ou igual a $b_1 b_2 = 28$ [53].

Tais atribuições livres podem continuar até que o vazio seja reduzido a um vazio de σ_1, ou perca sua prioridade para novos vazios emergentes criados por terminações de conexão em outro lugar do espectro. Mesmo que essas atribuições livres levem a um vazio de tamanho múltiplo de b_2 (b_1) maior ou igual a $b_1 b_2$, ele ainda pode aceitar um pedido para b_1 (b_2) *slots* sem necessariamente levar a σ_0 [53].

Conclui-se dessa discussão que todos os mecanismos de perda estão associados à existência de vazios do conjunto σ_0, que devem, portanto, ser evitados na medida do possível, exceto talvez como último recurso para não rejeitar um pedido.

Para evitar essas decisões de último recurso, mecanismos flexíveis devem ser usados somente quando todas as possíveis vagas inflexíveis, ou seja, as vagas que podem atender apenas uma das classes de tráfego, já foram utilizadas. Observe que a noção de vaga aqui se refere a um número abstrato de *slots*, associados a um vazio, mas não a qualquer conjunto específico de *slots* nesse vazio até que a vaga seja atribuída contígua a uma das extremidades do vazio [53].

O único mecanismo flexível para a atribuição de vagas sem perdas às requisições que chegam está associado à intercambialidade de atribuir blocos de $b_1 b_2$ *slots* a b_1 solicitações por b_2 *slots* ou a b_2 requisições por b_1 *slots* em vazios dos conjuntos σ_k com $k \geq 2$. Para preservar a intercambialidade dos blocos, o tanto quanto possível para uso futuro, é importante identificar a presença de uma parte inflexível v' de tais vazios, dada por [53]:

$$v' = v - (k-1)b_1 b_2 \tag{5.7}$$

em que $v \in \sigma_k$ e $v' \in \sigma_1$.

No exemplo do vazio $v = 79 \in \sigma_3$ discutido acima, a equação (5.7) produz $v' = 79 - (2 \times 28) = 23$, então o vazio tem uma parte flexível dada por 56 *slots* (composta por dois blocos flexíveis de $b_1 b_2 = 28$ *slots* cada) e uma parte inflexível formada por 23 *slots*. A parte inflexível de 23 *slots* é fornecida por uma única combinação de uma vaga de 7 *slots* e quatro vagas de 4 *slots* e, desse modo, a atribuição dessas vagas deve ser priorizada em detrimento das vagas flexíveis, o que é automático quando o vazio flexível é atribuído com alocação adjacente a uma extremidade do vazio, resultando em uma nova parte inflexível $v' \in \sigma_1$ [53].

Quando $v \in \sigma_1$, então $v' = v$, ou seja, todo o vazio é a sua parte inflexível, por isso é, apropriadamente, chamado de vazio inflexível. Quando v é um múltiplo de $b_1 b_2$, então $v' = 0 \in \sigma_1$, então a parte inflexível é zero, e não há vaga inflexível para ser atribuída, portanto a escolha desse vazio deve ser adiada em favor de outro vazio com uma parte inflexível positiva, se houver. Em todos os vazios restantes, exceto em σ_0, existem vagas inflexíveis para serem atribuídas antes que uma solicitação de b_i *slots* seja atribuída a um bloco flexível de $b_1 b_2$ *slots*, reduzindo-o assim a uma parte inflexível de $(b_j - 1)b_i$ *slots*, com $j \neq i$ e $j \in \{b_1, b_2\}$, para ser atribuída às $(b_j - 1)$ solicitações futuras de b_i *slots* [53].

Porém, quando um pedido de b_i *slots* é atribuído a um vazio flexível que não tem vaga inflexível para b_i *slots*, a solicitação é alocada em um dos blocos flexíveis de $b_1 b_2$ *slots*,

que é então comprometido a ser preenchido com $(b_j - 1)$ pedidos futuros de b_i *slots*, reduzindo assim a flexibilidade do enlace para acomodar futuros surtos de pedidos de b_j *slots* [53].

Já que a parte inflexível de todos os vazios, exceto os disfuncionais, são elementos de σ_1, então é conveniente pré-calcular esses valores em uma tabela com três colunas e $b_1 b_2$ linhas com as soluções não negativas para a identidade de Bézout para todos os elementos de σ_1 [53]. O Quadro 5.1 é um exemplo para $(b_1, b_2) = (3, 5)$. A primeira coluna mostra todos os $b_1 b_2 = 15$ tamanhos de vazios do conjunto σ_1, que são dados por (5.4). A segunda coluna mostra o correspondente número $n_1(v)$ de vagas b_1 inflexíveis, e a terceira coluna o número $n_2(v)$ de vagas b_2 inflexíveis. Esses números são as únicas soluções inteiras não negativas para (n_1, n_2) em (5.1) [53].

Quadro 5.1 - Número de vagas inflexíveis para todo $v \in (3,5)$ [53].

$v \in \sigma_1(3,5)$	$n_1(v)$	$n_2(v)$
0	0	0
3	**1**	**0**
5	**0**	**1**
6	**2**	**0**
8	1	1
9	**3**	**0**
10	**0**	**2**
11	2	1
12	**4**	**0**
13	1	2
14	3	1
16	2	2
17	4	1
19	3	2
22	4	2

Os tamanhos de vazios para os quais apenas uma das soluções da identidade de Bézout é zero estão destacados em amarelo no Quadro 5.1. Esses tamanhos de vazios devem ser priorizados apenas para a atribuição de solicitações da classe de tráfego associada com a solução positiva, mas adiada em função de uma prioridade menor para a atribuição de solicitações da classe de tráfego associada à solução zero, uma vez que essas atribuições são disfuncionais para o tamanho solicitado [53].

5.2 ALGORITMO CIENTE DOS TAMANHOS DAS REQUISIÇÕES

Em um único enlace elástico que recebe um tráfego de solicitações para o mesmo número de *slots* contíguos, alguns algoritmos simples, como é o caso do *first-fit* [17], são conhecidos por não gerarem perdas por fragmentação, mesmo que o espectro se torne fragmentado. A razão para tal é que todos os novos vazios gerados por terminações são dados pelo número exato de *slots* solicitados por conexão ou um múltiplo dele [4].

Em outras palavras, todos os vazios são estritamente funcionais no sentido de que podem ser completamente preenchidos por solicitações futuras. Nesta seção, é apresentada uma proposta de um novo algoritmo que suporta proativamente esta funcionalidade a fim de minimizar as perdas devido à fragmentação sob um tráfego de duas classes, sem necessariamente eliminar a fragmentação em si.

5.2.1 Algoritmo Proposto

À luz da discussão apresentada acima, é proposto um novo algoritmo (NA, do inglês *New Algorithm*) que monitora o estado do enlace após cada evento relevante (chegadas, atribuições e terminações) e implementa um mecanismo hierárquico de múltiplas buscas. Após a chegada de uma **requisição por uma conexão de b_i *slots***, em que $i \in \{1, 2\}$, o algoritmo a atribui aos seguintes tipos de vazios em ordem decrescente de prioridade:

a) múltiplos de b_i que são menores que $b_i b_j$, em que $j \neq i$ e $j \in \{1, 2\}$, priorizando o **menor** tamanho de vazio nesta categoria;

b) outros vazios que não são elementos de σ_0 e nem múltiplos de $b_j \neq b_i$ menores que $b_i b_j$, priorizando o **menor** tamanho de vazio nesta categoria;

c) múltiplos de b_j que são menores que $b_i b_j$, com $j \neq i$, priorizando o **maior** tamanho de vazio nesta categoria;

d) elementos de σ_0, priorizando o **maior** tamanho de vazio nesta categoria.

Observa-se que a preferência dada a tamanhos menores no item 1 endossa a prioridade para atribuições *best-fit* ou *exact-fit*, como sugerido em [15], mas enquadra-a em uma hierarquia mais ampla de prioridades. A preferência por tamanhos menores no item 2 visa focar no preenchimento dos vazios menores, permitindo que os vazios maiores cresçam ainda mais e possivelmente se coalesçam por meio das terminações de suas conexões adjacentes, aproximando-se da configuração ideal de um único vazio, na qual a fragmentação é zero.

A preferência por tamanhos maiores, nos itens 3 e 4, objetiva criar ou manter vazios de σ_0 com o maior tamanho possível, a fim de reduzir o tempo de espera para sua migração a um vazio de σ_1 após a(s) terminação(ões) de conexão(ões) adjacente(s). Se houver um empate em qualquer uma das categorias, o algoritmo escolhe o vazio vinculado com as menores frequências, para que os vazios nas frequências mais altas tenham uma chance maior de crescer e se aglutinar em vazios ainda maiores.

Na inicialização, o algoritmo começa com o cálculo da lista de todos os elementos de $\sigma_0(b_1, b_2)$. Então, na chegada de uma solicitação de b_i *slots*, $i \in \{1, 2\}$, todos os vazios com tamanho $v \geq b_i$ do estado atual do enlace são candidatos à atribuição dessa solicitação. Se, e somente se, não houver tais vazios, o pedido é bloqueado, o que caracteriza um algoritmo voraz.

Os vazios funcionais, para os quais $v \notin \sigma_0$, são priorizados em relação aos vazios disfuncionais, em que $v \in \sigma_0$. Por esse motivo, caso haja pelo menos um vazio funcional no enlace, a busca pela melhor atribuição deve começar no nível de prioridade 1 do fluxograma mostrado na Figura 5.3 e continuar pelos níveis de prioridades 2 e 3 caso o vazio não seja atribuído, com a busca realizada na ordem crescente de frequências, ignorando todos os vazios disfuncionais, se houver. A partir do nível de prioridade 4, que seleciona a última categoria dos vazios funcionais, os múltiplos de b_j menores que $b_1 b_2$, a pesquisa passar a ser executada na ordem decrescente de frequências.

Se, e somente se, não houver vazios funcionais disponíveis no enlace, a busca alcançará os níveis de prioridade 5 até $(3 + b_1)$ em ordem crescente de níveis de prioridade se cada pesquisa for infrutífera. Cabe ressaltar que o nível de prioridade 5 é alcançado somente se nenhum vazio funcional estiver disponível no espectro.

O nível de prioridade 1 é atribuído aos tamanhos de vazios menores que $b_1 b_2$ e múltiplos ou iguais ao tamanho da solicitação. Se a pesquisa não encontra nenhum vazio, então o nível de prioridade 2 escolherá o primeiro vazio com parte inflexível positiva v', que pode ser calculado a partir do tamanho do vazio v com a seguinte equação derivada de (5.5) e (5.7) [53]:

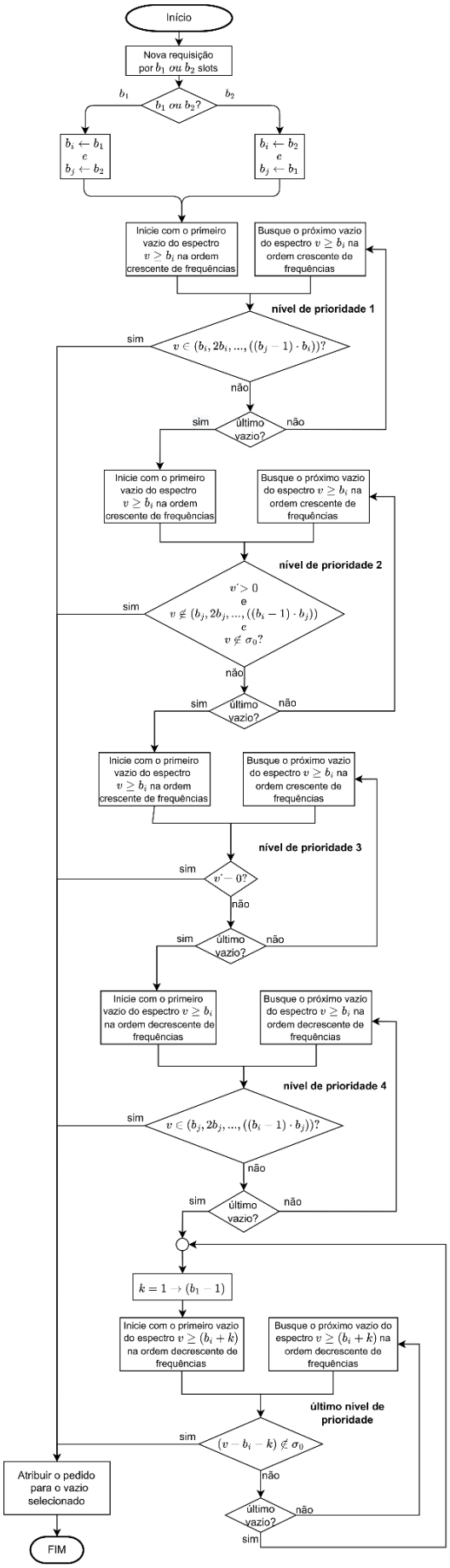


Fig. 5.3 - Fluxograma para o algoritmo proposto.

$$v' = \begin{cases} v \bmod(b_1 b_2) & se \quad v \bmod(b_1 b_2) \notin \sigma_0, \\ b_1 b_2 + v \bmod(b_1 b_2) & se \quad v \bmod(b_1 b_2) \in \sigma_0. \end{cases} \tag{5.8}$$

em que $a \bmod(b)$ denota o resto da divisão de a por b.

Se nenhum vazio funcional for escolhido nas prioridades 1 e 2, então todos os vazios funcionais são múltiplos flexíveis de $b_1 b_2$ e produzem $v' = 0$; ou podem ser múltiplos do outro tamanho diferente do tamanho solicitado b_i, aqui designado como b_j.

O algoritmo prioriza os vazios flexíveis na prioridade 3, se houver algum. Para isso, o algoritmo atribuirá o primeiro vazio com tamanho $b_1 b_2$ se encontrar um; caso contrário, escolhe o primeiro vazio com os tamanhos $2b_1 b_2$, $3b_1 b_2$, e assim por diante. Note que se $b_1 b_2 > S$, como quando $(b_1, b_2) = (13, 28)$ e $S = 320$, não há vazios flexíveis e, desse modo, a busca de prioridade 3 é infrutífera e pode ser ignorada a priori. O adiamento de múltiplos de $b_1 b_2$ para uma terceira prioridade é motivado pelo interesse na preservação da flexibilidade de grandes vazios funcionais.

Se v é um tamanho de vazio funcional, então pode sempre ser expresso como $v = v' + (i-1)b_1 b_2$, em que v' é um tamanho inflexível e i é o número de combinações de conexões de b_1 *slots* e b_2 *slots* que podem ser suportadas por v em qualquer ordem, assim $(i-1)$ é uma métrica da flexibilidade do vazio. Se $i = 1$, não há flexibilidade, então $v = v'$ e ambos são inflexíveis. Quando $v' > 0$, a atribuição de uma requisição de b_i *slots* não reduzirá, necessariamente, a flexibilidade de v porque ela pode apenas reduzir v' para $(v' - b_i)$ e manter a flexibilidade ilesa.

No entanto, se $v' = 0$, como acontece quando v é um múltiplo de $b_1 b_2$, então a flexibilidade é reduzida em 1 se v é atribuído. Por isso, múltiplos de $b_1 b_2$ são diferidos para uma terceira prioridade, membros de $\{b_j, 2b_j, \ldots, (b_i - 1)b_j\}$ para uma quarta prioridade, e membros de σ_0 para uma quinta e última prioridade [58].

Quando os níveis de prioridade 1, 2 e 3 não conseguirem encontrar um vazio elegível para atribuir uma solicitação de b_i *slots*, a única possibilidade restante é que todos os vazios funcionais sejam múltiplos de b_j menores que $b_1 b_2$. Esses tamanhos de vazios são os menos desejáveis entre os tamanhos de vazios funcionais para a atribuição de uma requisição por b_i *slots*, porque atribuí-los leva o vazio resultante ao conjunto de impasse σ_0, ou seja, produz um novo vazio disfuncional. No entanto, uma vez que as simulações em [59] sugeriram que a atribuição deles ainda é melhor do que rejeitar a solicitação sob tráfego dinâmico, então isso é feito no nível de prioridade 4.

Entretanto, novos resultados, apresentados a partir da seção 5.4, mostraram que a rejeição pode ser melhor para alguns vazios, caracterizando um algoritmo não voraz controlado por aprendizagem de máquina, quando a carga se aproxima da sobrecarga ou excede 100% [56]. Essa condição é geralmente desconsiderada na literatura, mas se torna potencialmente importante com o advento dos sistemas de multiplexação espacial, que usam múltiplas fibras ou múltiplos núcleos em fibras emergentes.

No entanto, ao contrário das buscas anteriores por atribuição, a busca de prioridade 4 e as próximas buscas devem ser feitas em ordem decrescente de frequências pelo seguinte motivo: já que as buscas anteriores são realizadas em ordem crescente de frequências, as frequências mais altas, provavelmente, serão mais escassamente ocupadas por conexões ativas, facilitando assim a coalescência de vazios disfuncionais e antecipando a liberação de recursos bloqueados.

Os vazios disfuncionais não podem ser completamente preenchidos com conexões de b_1 *slots* e/ou b_2 *slots*, mas o algoritmo ainda pode tentar minimizar o número de *slots* residuais inutilizáveis, visando a sobra de um único *slot* residual inutilizável. Para isso, o algoritmo rastreia cada vazio $v \in \sigma_0(b_1, b_2)$ do estado atual do enlace em ordem decrescente de frequências e, se $(v - b_1 - 1) \notin \sigma_0(b_1, b_2)$, a requisição de b_i *slots* é atribuída a v. Caso contrário, o algoritmo irá pesquisar vazios com apenas 2 *slots* residuais e repetir este procedimento para $(v - b_1 - 2) \notin \sigma_0(b_1, b_2)$. Se houver um novo insucesso, então a busca é refeita para 3 *slots* residuais com $(v - b_1 - 3) \notin \sigma_0(b_1, b_2)$, e assim por diante [53].

Esse procedimento não assegura de imediato que o resíduo final será k, mas preserva a acessibilidade do resíduo k para pesquisas futuras por ao menos uma sequência de chegadas subsequentes. Como o maior resíduo possível em $\sigma_0(b_1, b_2)$ é $(b_1 - 1)$, um tamanho disfuncional com resíduo dado pelo índice k, no intervalo de 1 a $(b_1 - 1)$, é garantido que a busca realizará no máximo $(b_1 - 1)$ iterações [53], conforme o fluxograma da Figura 5.3.

O algoritmo completo implícito na Figura 5.3 é complexo e pode exigir várias pesquisas em todo o espectro para uma única tarefa, o que acarreta mais tempo e sobrecarga de processamento além do que pode ser aceitável. Essa complexidade ocorre porque o algoritmo completo implementa não apenas uma prioridade, mas uma hierarquia de prioridades: os vazios funcionais são priorizados primeiro em relação aos disfuncionais, depois os vazios inflexíveis são priorizados em vez dos flexíveis, e atribuições seletivas em vez de atribuições não seletivas, enquanto as atribuições geradoras de disfuncionalidade são impedidas por todas as outras atribuições funcionais.

Finalmente, uma hierarquia também é implementada entre as atribuições disfuncionais, visando o menor resíduo possível. Cada prioridade numa hierarquia requer uma pesquisa (possivelmente infrutífera) sobre o espectro, então o algoritmo completo pode se tornar muito mais complexo e demorado do que o algoritmo *first-fit* clássico, exigindo assim uma versão simplificada, caso uma complexidade muito superior à do algoritmo *first-fit* não seja aceitável.

Como alternativa, uma versão simplificada do algoritmo proposto deve, então, ser limitada a uma única pesquisa em ordem crescente de frequências, assim como no algoritmo *first-fit*, mas modificada pelo uso de uma "lista tabu" para tamanhos de vazios a serem ignorados até que a busca chegue ao último vazio viável, que é aceito mesmo que seja membro da lista tabu. A lista tabu deve refletir a distinção mais importante entre os tipos de vazios, ou grupos deles, considerados na hierarquia completa.

Sugere-se uma lista tabu T_i específica para a busca de um vazio para atribuir uma solicitação de b_i *slots*, $i \in \{1, 2\}$, contendo todos os elementos do conjunto de *deadlock* $\sigma_0(b_1, b_2)$ e o conjunto de todos os tamanhos de vazios geradores de disfuncionalidade, dados por $\{b_j, 2b_j, \dots, (b_i - 1)b_j\}$, em que b_j é o tamanho de $\{b_1, b_2\}$ diferente do tamanho b_i sob consulta [53]:

$$T_i = \sigma_0(b_1, b_2) \cup \{b_j, 2b_j, \dots, (b_i - 1)b_j\} \tag{5.9}$$

5.3 RESULTADOS NUMÉRICOS

Esta seção deriva resultados numéricos destinados à determinação das perdas por fragmentação, em termos de ocupação espectral média e vazão média do tráfego, em função da carga oferecida L e de um fator de perfil do tráfego β dado por:

$$\beta = \frac{\lambda_2}{\lambda_1} \tag{5.10}$$

A carga L é expressa em termos fracionários como:

$$L = (b_1\lambda_1 + b_2\lambda_2)/S \tag{5.11}$$

Atente que o numerador em (5.11) é o número de *slots* solicitados por unidade de tempo, ou seja, por tempo médio de serviço. Portanto, $L = 1$ denota uma condição de saturação e $L > 1$ indica sobrecarga, levando a congestionamentos e altas probabilidades de bloqueio. A partir de (5.10) e (5.11), as taxas de chegada λ_i para $i \in \{1, 2\}$ são derivadas como:

$$\lambda_1 = \frac{LS}{b_1 + \beta b_2} \tag{5.12}$$

$$\lambda_2 = \frac{LS\beta}{b_1 + \beta b_2} \tag{5.13}$$

Para obter as perdas por fragmentação, foram realizadas simulações para três algoritmos: *first-fit* (FF); *exact-fit* seguido de *first-fit* (EFF); e o algoritmo ciente do tráfego (NA) proposto. Inicialmente, supõe-se que o algoritmo tem ciência apenas dos tamanhos b_1 e b_2 das requisições. Posteriormente, numa segunda versão do algoritmo, foi acrescentada também a ciência das taxas λ_1 e λ_2. Os resultados da simulação foram, então, comparados com um limite analítico para o enlace desfragmentado (DF) com a política de compartilhamento completo, tomada neste trabalho como referência (*benchmark*).

O limite analítico e as simulações são discutidas nas subseções a seguir para duas classes de serviço, 400 Gb/s e 1 Tb/s, sob a política de compartilhamento completo em três casos [60]: pequenas distâncias (até 1000 km), representadas por $(b_1, b_2) = (5, 11)$ para a maximização da eficiência espectral, e por $(b_1, b_2) = (5, 14)$ para a minimização do número de portadoras e da largura de banda ocupada para reduzir custos; distâncias intermediárias (entre 1000 e 4000 km) representadas por $(b_1, b_2) = (7, 17)$, que é ideal para 1600 km quando o número de portadoras e a largura de banda ocupada são minimizadas; e grandes distâncias (até 8000 km) representadas por $(b_1, b_2) = (13, 28)$, que é ótimo para 8000 km quando o números de *slots* e de portadoras são minimizados; todos em uma largura de banda de ganho de érbio padrão de 320 *slots*.

5.3.1 O Limite Analítico

É sabido que, na ausência da fragmentação espectral, a dinâmica de um enlace avulso elástico é representada por uma Cadeia de Markov em Tempo Contínuo reversível [5]. Para um sistema de duas classes em equilíbrio estocástico, as probabilidades dos estados (n_1, n_2) são dadas por:

$$p(n_1, n_2) = \frac{\left(\frac{\lambda_1^{n_1}}{n_1!}\right)\left(\frac{\lambda_2^{n_2}}{n_2!}\right)}{\sum_{(m_1,m_2)\in\chi}\left(\frac{\lambda_1^{m_1}}{m_1!}\right)\left(\frac{\lambda_2^{m_2}}{m_2!}\right)} = \frac{\left(\frac{\lambda_1^{n_1}}{n_1!}\right)\left(\frac{\lambda_2^{n_2}}{n_2!}\right)}{G(\chi)} \tag{5.14}$$

em que χ é o conjunto de todos os estados permitidos (n_1, n_2) e $G(\chi)$ é a constante de normalização.

Se a única restrição aos estados do sistema for dada pela quantidade finita de *slots* disponíveis S, e as atribuições sempre seguem alguma política voraz, então diz-se que o sistema segue uma política de compartilhamento completo (CS, do inglês *Complete Sharing*). O conjunto χ de estados permitidos é dado por $\chi_{CS} = \{(n_1, n_2) | 0 \leq n_1 b_1 + n_2 b_2 \leq S\}$. Neste caso, uma requisição por b_2 *slots* será bloqueada se, e somente se, $n_2 = \lfloor (S - n_1 b_1)/b_2 \rfloor$ em que $\lfloor . \rfloor$ é a parte inteira do argumento ("piso"). Da mesma forma, uma solicitação de b_1 *slots* será bloqueada se, e somente se, $n_1 = \lfloor (S - n_2 b_2)/b_1 \rfloor$. Portanto, as probabilidades de estado sob a política de compartilhamento completo são dadas por:

$$p_{CS}(n_1, n_2) = \frac{\left(\frac{\lambda_1^{n_1}}{n_1!}\right)\left(\frac{\lambda_2^{n_2}}{n_2!}\right)}{G(\chi_{CS})} \tag{5.15}$$

em que:

$$G(\chi_{CS}) = \sum_{m_1=0}^{\lfloor S/b_1 \rfloor}\left[\left(\frac{\lambda_1^{m_1}}{m_1!}\right)\sum_{m_2=0}^{\lfloor (S-m_1 b_1)/b_2 \rfloor}\left(\frac{\lambda_2^{m_2}}{m_2!}\right)\right] = \sum_{m_2=0}^{\lfloor S/b_2 \rfloor}\left[\left(\frac{\lambda_2^{m_2}}{m_2!}\right)\sum_{m_1=0}^{\lfloor (S-m_2 b_2)/b_1 \rfloor}\left(\frac{\lambda_1^{m_1}}{m_1!}\right)\right] \tag{5.16}$$

é a constante de normalização para a política de compartilhamento completo. Observe que a constante de normalização não é uma função específica de algum estado (n_1, n_2), por isso pode ser calculada apenas uma vez, dados (b_1, b_2) e S.

Uma vez que a probabilidade de bloqueio P_{bi} de uma requisição por i *slots*, $i \in \{1, 2\}$, é dada pela propriedade PASTA (*Poisson Arrivals See Time Averages*) [61] como a soma das probabilidades de todos os estados bloqueantes para solicitações de i *slots*, tem-se:

$$P_{b1} = \sum_{n_2=0}^{\lfloor S/b_2 \rfloor} p(\lfloor (S - n_2 b_2)/b_1 \rfloor, n_2) \tag{5.17}$$

$$P_{b2} = \sum_{n_1=0}^{\lfloor S/b_1 \rfloor} p(n_1, \lfloor (S - n_1 b_1)/b_2 \rfloor) \tag{5.18}$$

Com a ajuda da Lei de *Little* [5][61], essas probabilidades de bloqueio podem, então, ser usadas para determinar o número médio de conexões ativas $\overline{n_1}$ e $\overline{n_2}$ das classes 1 e 2, respectivamente, como:

$$\overline{n_1} = \lambda_1(1 - P_{b1}) \tag{5.19}$$

$$\overline{n_2} = \lambda_2(1 - P_{b2}) \tag{5.20}$$

Finalmente, a ocupação espectral média ρ e a vazão média τ são dadas por:

$$\rho = \overline{n_1}b_1 + \overline{n_2}b_2 \tag{5.21}$$

$$\tau = \overline{n_1}\tau_1 + \overline{n_2}\tau_2 \tag{5.22}$$

em que τ_1 e τ_2 são as taxas de bits das conexões de b_1 *slots* e b_2 *slots*, respectivamente (400 Gb/s e 1 Tb/s no caso sob investigação).

A ocupação espectral média ρ em (5.21) é um limite superior na ocupação espectral média de qualquer algoritmo voraz de atribuição espectral em enlace avulso. No entanto, a vazão média τ não é necessariamente um limite em qualquer algoritmo de vazão se a eficiência espectral dos dois serviços não for a mesma. É o caso aqui, pois o serviço de 1 Tb/s tem uma eficiência de (1000/11) ≅ 90,9 Gb/s/*slot* e o serviço de 400 Gb/s apenas (400/5) = 80 Gb/s/*slot*. Os limites apropriados neste caso são derivados em [62], mas não são tratados nesta tese.

5.3.2 Simulações Algorítmicas para o Enlace Fragmentado Espectralmente

Sob a restrição de contiguidade, as perdas por fragmentação espectral são inevitáveis, e (5.14) não se aplica, a não ser como um limitante superior (*upper bound*) de desempenho sobre todos os algoritmos vorazes. Além disso, o número de estados torna-se extremamente grande, por isso deve-se recorrer a simulações para estimar a ocupação média ρ e a vazão média τ. Para isso, é necessário utilizar uma representação adequada dos estados.

O estado do sistema é representado por um vetor unidimensional no qual os elementos ímpares são inteiros não negativos dados pelo número de *slots* dos vazios em ordem crescente de frequências, incluindo vazios de tamanho zero no caso em que são ladeados por

conexões contíguas; e os elementos pares são retirados de $\{b_1, b_2\}$ e representam o número de *slots* de cada conexão ativa entre seus vazios vizinhos, inclusive os de tamanho zero no caso de contiguidade a outras conexões ativas.

O número de elementos significativos do vetor-estado é, então, dado por:

$$m = [2(n_1 + n_2) + 1] \tag{5.23}$$

a qualquer momento, e sua soma é, invariavelmente, $S = 320$. O valor máximo possível de m é dado por:

$$M = 2\left\lfloor {}^{S}/_{b_1} \right\rfloor + 1 \tag{5.24}$$

então, se for necessário operar com um vetor de tamanho fixo para conveniência de programação, ele deve ter pelo menos M elementos, com os primeiros m elementos sendo significativos enquanto os restantes $(M - m)$ são apenas uma reserva de espaço, sem função. De qualquer forma, a parte significativa do vetor-estado é $v = (v_1, v_2, \dots, v_m)$, com v_{2i} sendo igual ao tamanho (número de *slots*) da i-ésima conexão em ordem crescente de frequências, com v_{2i-1} sendo o tamanho do seu vazio anterior e v_{2i+1} o seu vazio subsequente.

O processo das chegadas de requisições, rejeições, atribuições espectrais e terminações de conexões é produzido pelos seguintes eventos:

a) requisições por conexões de b_i *slots*, que chegam com taxa λ_i, $i \in \{1, 2\}$;

b) rejeição ou atribuição do número necessário de *slots* contíguos de acordo com o algoritmo sob investigação;

c) término de uma conexão ativa, que acontece com taxa 1 para cada uma das conexões ativas atuais.

Em qualquer estado com $n_1$conexões ativas de b_1 *slots* e n_2 conexões ativas de b_2 *slots*, a taxa de ocorrência do próximo evento é, então, dada por $(n_1 + n_2 + \lambda_1 + \lambda_2)$, de modo que o tempo médio de espera para o próximo evento é:

$$\gamma(n_1, n_2) = \frac{1}{(n_1 + n_2 + \lambda_1 + \lambda_2)} \tag{5.25}$$

A probabilidade de que o próximo evento seja a chegada de uma requisição por uma conexão de b_i *slots* é $\lambda_i/(n_1 + n_2 + \lambda_1 + \lambda_2)$ para todos $i \in \{1, 2\}$. A probabilidade de que o próximo evento seja algum término é $(n_1 + n_2)/(n_1 + n_2 + \lambda_1 + \lambda_2)$, com probabilidade $1/(n_1 + n_2 + \lambda_1 + \lambda_2)$ para cada conexão ativa ser terminada.

Se o próximo evento for a chegada de uma solicitação de b_i *slots* e o vetor-estado atual não tiver qualquer elemento ímpar igual ou maior do que b_i, então o pedido é rejeitado, o contador de bloqueio é adicionado a 1, e o vetor-estado não muda. Caso contrário, o algoritmo de atribuição espectral (SA, do inglês *Spectrum Assignment*) é executado e o vazio resultante é atribuído substituindo o elemento ímpar escolhido v_c do vetor-estado pelo trio $(0, b_i, v_c - b_i)$; e 2 é adicionado a m e aos índices de todos os elementos sucessores do vetor-estado. Além disso, 1 deve ser adicionado a n_i, e $\gamma(n_1, n_2)$ atualizado de acordo com (5.25).

Se o próximo evento for o término da k-ésima conexão ativa na ordem crescente das frequências, que é representada pelo elemento v_{2k} no vetor-estado, então 1 deve ser subtraído de n_i se $v_{2k} = b_i$; em seguida, o trio $(v_{2k-1}, v_{2k}, v_{2k+1})$ é substituído por um único elemento ímpar dado por $(v_{2k-1} + v_{2k} + v_{2k+1})$ na $(2k - 1)$-ésima posição; na sequência, 2 deve ser subtraído de m e dos índices de todos os elementos sucessores do vetor-estado a partir de $v_{2k} \leftarrow v_{2k+2}$; e finalmente, $\gamma(n_1, n_2)$ deve ser atualizado conforme (5.25).

Após a atualização de $\gamma(n_1, n_2)$ em todos os casos, o produto $(n_1 b_1 + n_2 b_2) * \gamma(n_1, n_2)$ deve ser acrescentado à estimativa Ω da integral temporal da ocupação, e o produto $(n_1 \tau_1 + n_2 \tau_2) * \gamma(n_1, n_2)$ deve ser somado à estimativa Φ da integral temporal da vazão. Ademais, $\gamma(n_1, n_2)$ deve ser adicionado ao tempo simulado T.

As novas estimativas de ocupação e vazão são, então, dadas por:

$$\hat{\rho} = \frac{\Omega}{T} \tag{5.26}$$

$$\hat{\tau} = \frac{\Phi}{T} \tag{5.27}$$

A simulação termina quando os valores das estimativas não mudarem mais dentro de um determinado alvo de precisão.

5.3.2.1 Algoritmo First-Fit (FF)

No algoritmo FF, o vazio escolhido para atribuição é o primeiro vazio na ordem crescente das frequências, que seja igual ou maior do que o número requisitado de *slots* b_{req}. No vetor-estado, isso corresponde ao elemento ímpar v_c com menor índice c que seja igual ou maior que o número solicitado de *slots*:

$$v_c \geq b_{req} \tag{5.28}$$

Se não for encontrado nenhum vazio para atender (5.28), então a solicitação é bloqueada, de modo que o contador de bloqueios é incrementado por 1, e o vetor-estado não é alterado. Caso contrário, o elemento v_c é substituído no vetor-estado pelo trio $\left(0, b_{req}, v_c - b_{req}\right)$ para denotar o novo vetor-estado. Além disso, 2 é adicionado a m e aos índices de todos os elementos sucessores do vetor-estado. E também, 1 deve ser adicionado a n_i se $b_{req} = b_i$, e $\gamma(n_1, n_2)$ é atualizado de acordo com (5.25).

5.3.2.2 Algoritmo Exact-First-Fit (EFF)

No algoritmo EFF, o vazio escolhido para atribuição é o primeiro vazio na ordem do aumento das frequências, que é igual ao número solicitado de b_{req} *slots*. No vetor-estado, isso corresponde ao elemento ímpar v_c com menor índice igual ao número solicitado de *slots*:

$$v_c = b_{req} \tag{5.29}$$

Se não for encontrado nenhum vazio para atender (5.29), então o procedimento do algoritmo FF é aplicado. Para evitar a repetição da pesquisa, a posição da primeira conformidade com (5.28) pode ser armazenada durante a primeira pesquisa.

5.3.3 Simulações do NA

Nesta seção, são apresentados os resultados obtidos com a simulação do NA, em sua versão completa conforme ilustrado na Figura 5.3, que foi desenvolvido em linguagem *Python*.

A seguir, são listados os parâmetros iniciais configurados na simulação:

a) largura de banda $S = 320$ *slots*;
b) duas classes de tráfego caracterizadas por (b_1, b_2);
c) carga L normalizada variando de 0 até $2{,}0$ com incrementos iguais a $0{,}1$;
d) a chegada dos pedidos é poissoniana, sendo as taxas λ_1 e λ_2 expressas em erlangs;
e) fator de perfil do tráfego $\beta = \lambda_2/\lambda_1$, de maneira que há:
    a. predominância de requisições do tipo 1, quando $\beta = 0{,}5$;
    b. taxa igualitária de chegada, quando $\beta = 1{,}0$;
    c. predominância de requisições do tipo 2, quando $\beta = 2{,}0$.
f) critério de parada da simulação: 500 mil chegadas.

Para cada valor de β considerado acima, a simulação foi realizada para quatro pares de $(b_1, b_2) = (5, 11)$, $(5, 14)$, $(7, 17)$ e $(13, 28)$, referente aos três casos (pequenas, médias e longas distâncias) descritos na seção 5.3, todos sob a política de compartilhamento completo. A simulação compara a saída de quatro algoritmos, todos submetidos ao mesmo tráfego de entrada e sob as mesmas condições simuladas, com o intuito de avaliar o desempenho do algoritmo proposto NA (linha verde), que é um algoritmo voraz, ou seja, somente bloqueia uma requisição por falta de recursos contíguos, com outros dois algoritmos também vorazes disponíveis na literatura, o *exact-first-fit* (linha roxa) e o *first-fit* (linha preta).

Para efeitos de simplificação e padronização da legenda nas curvas apresentadas a seguir, o algoritmo *exact-first-fit* (EFF) é identificado somente como "EF". Como caso de referência, o algoritmo Desfragmentador (DF) representado na linha azul é utilizado como um limitante, sendo o DF considerado ótimo entre os algoritmos vorazes, pois mantém o espectro desfragmentado. A linha vermelha denota a reta $S \times L$, com a linha tracejada vermelha indicando o nível máximo da ocupação espectral, ou seja, $S = 320$ *slots*.

Os resultados da simulação avaliam, em função da carga L, a ocupação espectral média, a vazão média e as probabilidades de bloqueio de banda e das requisições dos tipos b_1 e b_2. A ocupação média resultante é calculada dividindo a integral de tempo da ocupação simulada do enlace pelo tempo de simulação. O mesmo raciocínio é aplicado para a vazão média.

A probabilidade de bloqueio de uma requisição relaciona o número de requisições bloqueadas de um determinado tipo (b_1 ou b_2) e o total de requisições do mesmo tipo (aceitas e bloqueadas).

A probabilidade de bloqueio de banda, ou probabilidade de bloqueio de *slots*, é a relação entre o número de *slots* bloqueados e o número de *slots* requisitados numa simulação. Sendo suficientemente longa, essa relação deverá tender à expressão em (5.30), para duas classes de tráfego num enlace avulso:

$$P_{bb} = \frac{\lambda_1 b_1 P_{b1} + \lambda_2 b_2 P_{b2}}{\lambda_1 b_1 + \lambda_2 b_2} = \frac{P_{b1} + \beta\left(\frac{b_2}{b_1}\right) P_{b2}}{1 + \beta\left(\frac{b_2}{b_1}\right)} \tag{5.30}$$

As próximas seções apresentam as curvas de cada variável analisada para um caso completo selecionado e as curvas da ocupação espectral média para os demais casos. O resultado completo de todas as simulações realizadas para o NA, incluindo todas as curvas e o código fonte do programa, está disponível em https://doi.org/10.25824/redu/HT2MVA.

5.3.3.1 Simulações do NA com $\boldsymbol{\beta = 0,5}$

Com $\beta = 0{,}5$, há uma predominância na chegada de requisições do tipo b_1. As Figuras 5.4 a 5.7 apresentam as curvas da ocupação espectral média em função da carga (L), respectivamente, para os pares $(b_1, b_2) = (5, 11)$, $(5, 14)$, $(7, 17)$ e $(13, 28)$.

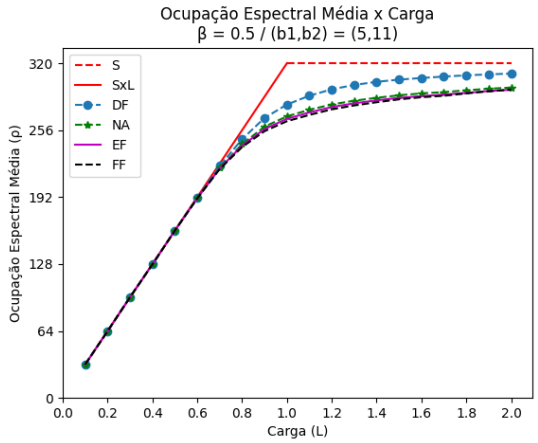


Fig. 5.4 - Ocupação espectral média (ρ) × carga (L) para $\beta = 0{,}5$ e $(b_1, b_2) = (5, 11)$.

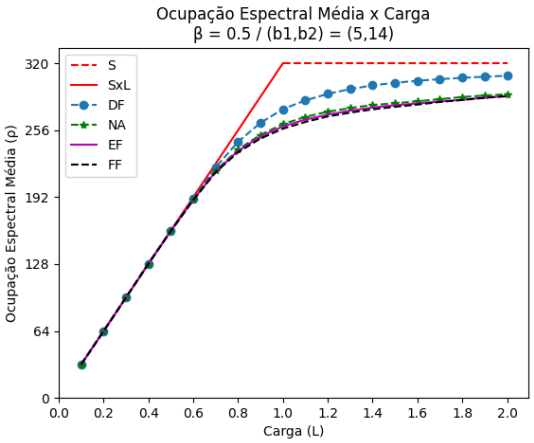


Fig. 5.5 - Ocupação espectral média (ρ) × carga (L) para $\beta = 0{,}5$ e $(b_1, b_2) = (5, 14)$.

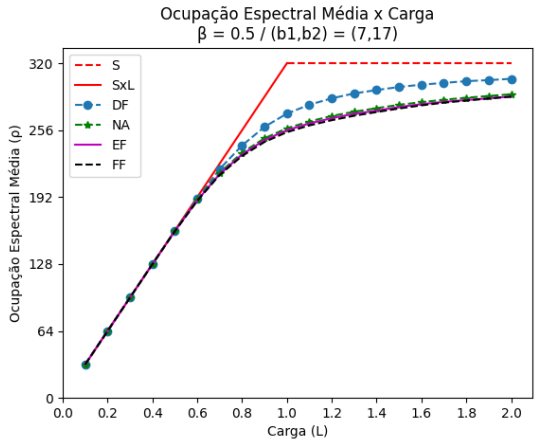


Fig. 5.6 - Ocupação espectral média (ρ) × carga (L) para $\beta = 0{,}5$ e $(b_1, b_2) = (7, 17)$.

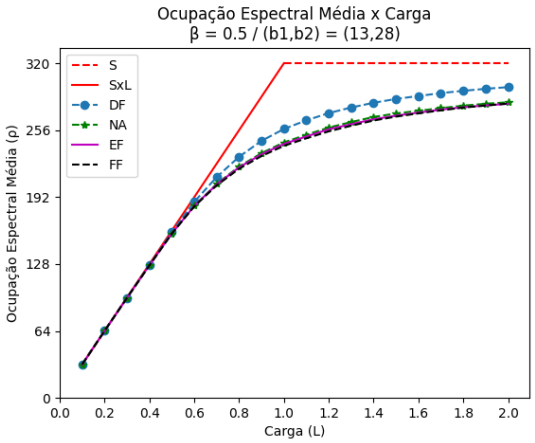


Fig. 5.7 - Ocupação espectral média (ρ) × carga (L) para $\beta = 0{,}5$ e $(b_1, b_2) = (13, 28)$.

A seguir, os Quadros 5.2 a 5.5 apresentam os resultados numéricos dos níveis de ocupação espectral produzidos pelos quatro algoritmos comparados com $\beta = 0{,}5$ para cada ponto de carga (L). Em destaque (amarelo), a faixa de carga na qual o NA ultrapassa os algoritmos EF e FF, permanecendo superior a ambos em todos os pontos até a carga $L = 2{,}0$, e empatado com os demais para as cargas inferiores.

Quadro 5.2 - Níveis de ocupação espectral para $\beta = 0{,}5$ e $(b_1, b_2) = (5, 11)$.

(L)	DF	NA	EF	FF
0,1	32,0131	32,0131	32,0131	32,0131
0,2	63,8489	63,8489	63,8489	63,8489
0,3	96,3187	96,3187	96,3187	96,3187
0,4	127,7149	127,7149	127,7149	127,7149
0,5	159,9571	**159,9420**	159,9399	159,9301
0,6	191,3803	**191,0267**	190,9311	190,8123
0,7	221,9259	**220,0466**	219,4104	218,8014
0,8	247,3836	**242,6040**	241,1835	239,8543
0,9	267,6489	**259,3435**	257,0450	255,0689
1,0	280,4873	**269,1633**	266,7981	264,5610
1,1	288,9294	**275,5005**	272,9654	270,5763
1,2	295,0573	**280,5697**	278,0765	275,9225
1,3	299,2127	**284,1430**	281,6487	279,6336
1,4	302,2123	**286,9431**	284,5117	282,6964
1,5	304,3189	**289,3201**	286,8965	285,4739
1,6	305,8654	**291,1510**	288,5199	287,4923
1,7	307,2414	**292,3895**	290,0164	288,8889
1,8	308,4095	**294,0958**	291,6585	291,1467
1,9	309,3225	**295,6148**	293,3616	293,1593
2,0	310,1303	**296,6789**	294,6557	294,7314

Quadro 5.3 - Níveis de ocupação espectral para $\beta = 0{,}5$ e $(b_1, b_2) = (5, 14)$.

(L)	DF	NA	EF	FF
0,1	32,0277	32,0277	32,0277	32,0277
0,2	63,7996	63,7996	63,7996	63,7996
0,3	96,0163	96,0163	96,0163	96,0163
0,4	128,3973	**128,3972**	128,3910	128,3914
0,5	159,7285	**159,6505**	159,5901	159,5537
0,6	190,4537	**189,7431**	189,4116	189,1995
0,7	220,6110	**217,2616**	216,4969	215,8159
0,8	244,8922	**237,4997**	236,1490	234,9015
0,9	263,0435	**251,5785**	250,0399	248,0293
1,0	276,1070	**261,7731**	259,5580	257,4210
1,1	284,4032	**268,6844**	266,1289	263,8529
1,2	290,7164	**273,6807**	270,8625	269,1241
1,3	295,2435	**277,1216**	274,6556	272,7866
1,4	298,7761	**280,0402**	277,6068	275,9031
1,5	301,3087	**281,6552**	279,8276	278,3284
1,6	303,2058	**283,6406**	281,5181	280,4490
1,7	304,7709	**285,5951**	283,3834	282,9549
1,8	306,0523	**287,6672**	285,0335	284,8847
1,9	307,1188	**289,1050**	287,2427	286,9281
2,0	308,0407	**290,3096**	288,4757	288,5468

Quadro 5.4 - Níveis de ocupação espectral para $\beta = 0{,}5$ e $(b_1, b_2) = (7, 17)$.

(L)	DF	NA	EF	FF
0,1	32,0169	32,0169	32,0169	32,0169
0,2	63,9760	63,9760	63,9760	63,9760
0,3	95,9493	**95,9493**	95,9489	95,9490
0,4	127,9345	**127,9171**	127,9108	127,9032
0,5	159,7543	**159,5351**	159,4133	159,3652
0,6	190,4797	**189,3143**	188,8972	188,4779
0,7	218,8834	**215,3321**	214,0918	213,2602
0,8	241,3861	**234,4223**	232,7509	231,3324
0,9	259,1469	**248,2354**	246,5502	244,7481
1,0	271,8466	**257,8154**	255,9841	254,1087
1,1	280,3104	**264,2997**	262,4179	260,5423
1,2	286,4425	**269,5879**	267,4650	265,5426
1,3	291,1050	**273,5952**	271,6765	269,8081
1,4	294,3527	**277,0140**	274,6510	273,3312
1,5	297,2729	**280,2053**	278,1533	276,7169
1,6	299,4754	**282,7341**	280,8135	279,7311
1,7	301,1972	**285,0983**	283,0482	282,2898
1,8	302,8394	**286,9819**	284,8221	284,2961
1,9	303,8455	**288,6713**	286,4178	286,2442
2,0	305,0915	**290,2808**	288,2140	288,0258

Quadro 5.5 - Níveis de ocupação espectral para $\beta = 0{,}5$ e $(b_1, b_2) = (13, 28)$.

(L)	DF	NA	EF	FF
0,1	32,0572	32,0572	32,0572	32,0572
0,2	64,1521	64,1516	64,1516	64,1516
0,3	95,7522	95,7287	95,7312	95,7275
0,4	127,0709	126,7929	126,8214	126,7809
0,5	158,7043	**157,3034**	157,2897	157,1166
0,6	187,4325	**183,7597**	183,5680	183,1150
0,7	211,1653	**204,5647**	204,0906	203,1880
0,8	230,5832	**221,2564**	220,4598	219,1557
0,9	245,7498	**233,9916**	232,9403	231,2605
1,0	257,1334	**244,3101**	242,8627	241,0533
1,1	265,3722	**251,5796**	249,9750	248,2040
1,2	272,3220	**258,4783**	256,8215	254,8749
1,3	277,6548	**263,7502**	261,7027	260,2460
1,4	282,1822	**268,6002**	266,4293	265,2250
1,5	285,8504	**271,9229**	269,8596	268,9606
1,6	288,6634	**274,5278**	272,7635	272,0727
1,7	291,4218	**277,0309**	275,6302	274,9486
1,8	293,6049	**279,4232**	277,9026	277,5929
1,9	295,5471	**281,2211**	279,6298	279,5539
2,0	297,1213	**282,6895**	281,1840	281,4187

5.3.3.2 Simulações do NA com $\boldsymbol{\beta = 1,0}$

No caso de $\beta = 1{,}0$, a taxa de chegada é igualitária entre as requisições do tipo b_1 e do tipo b_2. Na sequência, as Figuras 5.8 a 5.11 mostram os níveis da ocupação espectral média versus a carga (L), respectivamente, para os pares $(b_1, b_2) = (5, 11)$, $(5, 14)$, $(7, 17)$ e $(13, 28)$.

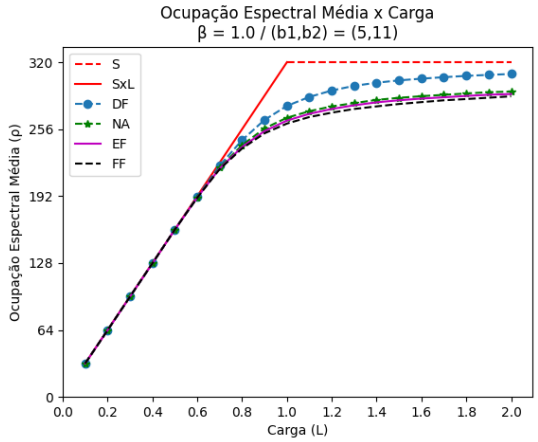


Fig. 5.8 - Ocupação espectral média (ρ) × carga (L) para $\beta = 1{,}0$ e $(b_1, b_2) = (5, 11)$.

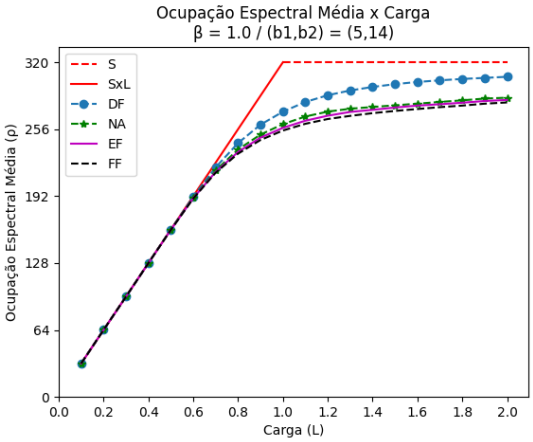


Fig. 5.9 - Ocupação espectral média (ρ) × carga (L) para $\beta = 1{,}0$ e $(b_1, b_2) = (5, 14)$.

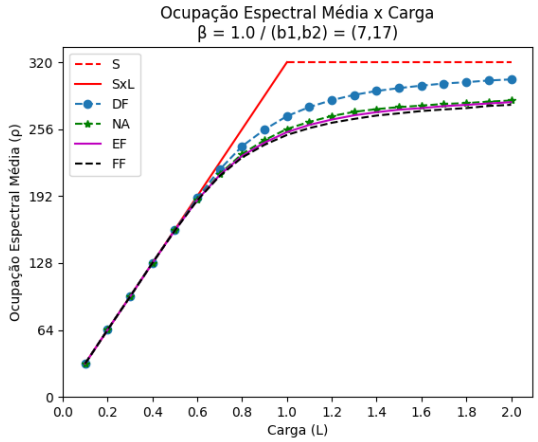


Fig. 5.10 - Ocupação espectral média (ρ) × carga (L) para $\beta = 1{,}0$ e $(b_1, b_2) = (7, 17)$.

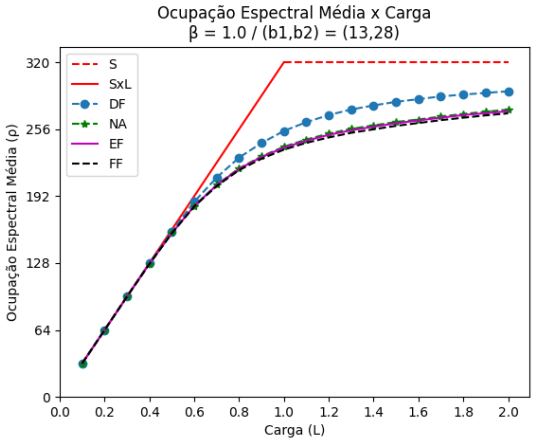


Fig. 5.11 - Ocupação espectral média (ρ) × carga (L) para $\beta = 1{,}0$ e $(b_1, b_2) = (13, 28)$.

Na sequência, os Quadros 5.6 a 5.9 detalham os resultados numéricos dos níveis de ocupação espectral gerados pelos quatro algoritmos comparados com $\beta = 1{,}0$ para cada ponto de carga (L).

Quadro 5.6 - Níveis de ocupação espectral para $\beta = 1{,}0$ e $(b_1, b_2) = (5, 11)$.

(L)	DF	NA	EF	FF
0,1	32,0366	32,0366	32,0366	32,0366
0,2	63,7913	63,7913	63,7913	63,7913
0,3	95,8663	95,8663	95,8663	95,8663
0,4	127,7241	127,7209	127,7212	127,7212
0,5	160,0609	**160,0285**	160,0083	160,0102
0,6	191,2358	**190,7821**	190,6269	190,4529
0,7	221,2262	**219,2479**	218,4385	217,7810
0,8	245,5465	**240,6910**	238,9517	237,5279
0,9	264,8336	**256,7299**	254,1588	252,0517
1,0	278,2354	**266,9218**	264,1975	261,3417
1,1	286,8296	**273,1631**	270,6725	267,6005
1,2	292,9468	**277,7742**	275,3134	271,7970
1,3	297,4232	**280,9203**	278,8136	275,3107
1,4	300,3898	**283,9653**	281,6290	277,7757
1,5	302,6599	**285,9917**	283,6274	280,0573
1,6	304,2646	**287,5149**	285,2442	281,8411
1,7	305,7020	**288,8281**	286,4757	283,5869
1,8	306,9352	**290,0958**	287,7741	284,7885
1,9	307,8965	**291,3798**	288,8929	286,0787
2,0	308,7478	**291,9122**	289,6025	287,3891

Quadro 5.7 - Níveis de ocupação espectral para $\beta = 1{,}0$ e $(b_1, b_2) = (5, 14)$.

(L)	DF	NA	EF	FF
0,1	32,0132	32,0132	32,0132	32,0132
0,2	64,0752	64,0752	64,0752	64,0752
0,3	95,8800	95,8800	95,8800	95,8800
0,4	127,9692	**127,9646**	127,9623	127,9602
0,5	160,0142	**159,8847**	159,8019	159,7501
0,6	191,5004	**190,5400**	190,0654	189,7184
0,7	219,5105	**216,4705**	215,1783	214,3881
0,8	242,7308	**236,5810**	234,2097	232,6322
0,9	260,2906	**250,8213**	247,9290	245,6611
1,0	272,6874	**260,7717**	257,2515	254,7096
1,1	282,0000	**268,0904**	264,0777	261,1374
1,2	288,4394	**272,7582**	268,8642	265,6966
1,3	292,9836	**275,5267**	272,4783	268,7045
1,4	296,3510	**277,1775**	274,4675	271,2961
1,5	299,0351	**278,8102**	276,6715	273,4379
1,6	301,1315	**280,3325**	278,5278	275,2587
1,7	302,6866	**282,0237**	279,8177	277,0494
1,8	304,0097	**283,5573**	281,2755	278,5266
1,9	305,0253	**285,2620**	282,9290	280,3522
2,0	306,0668	**286,0762**	283,9380	281,5253

Quadro 5.8 - Níveis de ocupação espectral para $\beta = 1{,}0$ e $(b_1, b_2) = (7, 17)$.

(L)	DF	NA	EF	FF
0,1	31,8720	31,8720	31,8720	31,8720
0,2	64,1517	64,1517	64,1517	64,1517
0,3	95,7927	95,7925	95,7918	95,7916
0,4	128,0593	**128,0358**	128,0265	128,0213
0,5	159,6132	**159,2916**	159,1811	159,0955
0,6	190,3988	**188,7467**	188,2884	187,8888
0,7	217,3802	**213,1804**	212,1183	211,2465
0,8	239,7202	**232,3000**	230,5141	228,9256
0,9	255,6101	**245,5212**	243,1641	241,0438
1,0	268,4620	**256,1600**	253,1421	250,4749
1,1	277,1594	**262,9999**	260,0874	257,0289
1,2	283,7339	**268,5756**	265,1885	262,1017
1,3	288,7944	**272,5399**	269,2560	265,8116
1,4	292,5325	**275,1782**	272,2189	269,0177
1,5	295,2666	**277,0898**	274,6164	271,1420
1,6	297,6052	**278,4984**	276,0910	273,3468
1,7	299,6593	**280,1316**	278,0348	274,8642
1,8	300,9253	**280,8716**	279,2076	276,1474
1,9	302,5471	**282,3728**	280,5692	278,1988
2,0	303,5725	**283,4079**	281,8015	279,2645

Quadro 5.9 - Níveis de ocupação espectral para $\beta = 1{,}0$ e $(b_1, b_2) = (13, 28)$.

(L)	DF	NA	EF	FF
0,1	32,0787	32,0787	32,0787	32,0787
0,2	63,9332	63,9318	63,9328	63,9327
0,3	96,0135	95,9817	95,9842	95,9772
0,4	127,5973	**127,2511**	127,2432	127,2034
0,5	158,1693	**156,6553**	156,6206	156,4886
0,6	186,5803	**182,7087**	182,5710	182,1310
0,7	209,7627	**202,8020**	202,3799	201,5884
0,8	228,8685	**218,5470**	218,0051	216,8622
0,9	242,7461	**229,9859**	229,3213	227,7213
1,0	254,3236	**239,1760**	238,3845	236,4470
1,1	262,6870	**246,1029**	244,9314	242,9810
1,2	269,5147	**251,7560**	250,6025	248,0595
1,3	274,8050	**256,2852**	254,8826	252,5341
1,4	278,6951	**259,5843**	258,3753	255,8706
1,5	282,1253	**262,8510**	261,6036	259,1593
1,6	284,7079	**265,2248**	264,3441	261,8901
1,7	287,3274	**268,0539**	266,9433	264,6046
1,8	289,0633	**270,5227**	269,4824	266,9514
1,9	290,6547	**272,6436**	271,2805	269,1826
2,0	292,1938	**274,6915**	273,1387	271,1494

5.3.3.3 Simulações do NA com $\boldsymbol{\beta = 2{,}0}$

Na situação de $\beta = 2{,}0$, há uma predominância na chegada de requisições do tipo b_2, ou seja, prevalecem os tamanhos maiores. As Figuras 5.12 a 5.15 exibem as curvas da ocupação espectral média em função da carga (L) para os quatros pares (b_1, b_2) analisados.

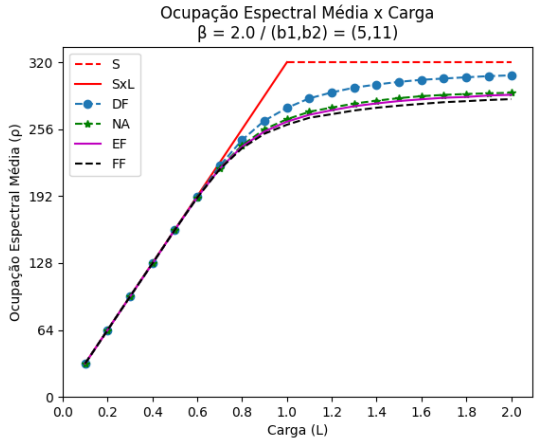


Fig. 5.12 - Ocupação espectral média (ρ) × carga (L) para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 11)$.

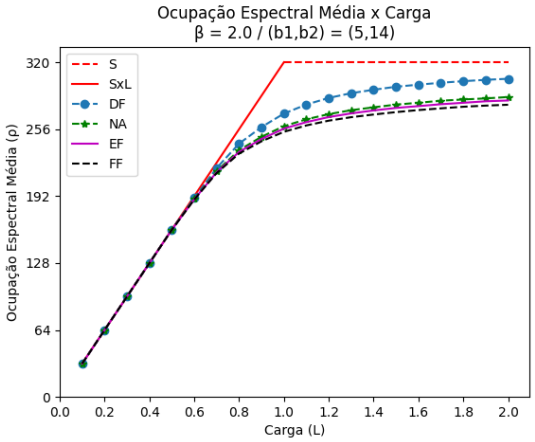


Fig. 5.13 - Ocupação espectral média (ρ) × carga (L) para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$.

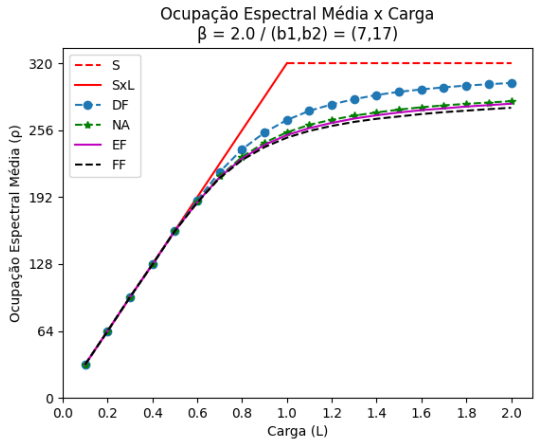


Fig. 5.14 - Ocupação espectral média (ρ) × carga (L) para $\beta = 2{,}0$ e $(b_1, b_2) = (7, 17)$.

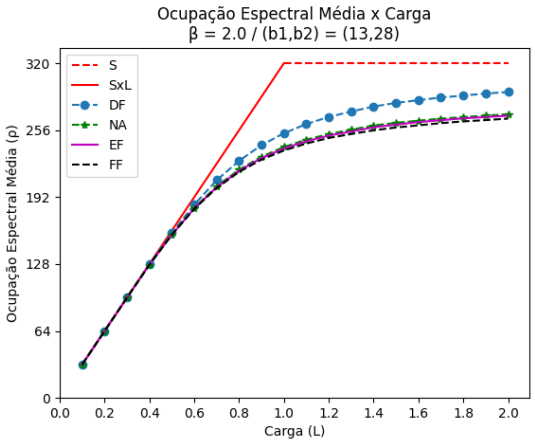


Fig. 5.15 - Ocupação espectral média (ρ) × carga (L) para $\beta = 2{,}0$ e $(b_1, b_2) = (13, 28)$.

Os Quadros 5.10 a 5.13, exibidos a seguir, especificam os valores numéricos dos níveis de ocupação espectral atingidos pelos quatro algoritmos comparados com $\beta = 2{,}0$ para cada ponto de carga (L).

Quadro 5.10 - Níveis de ocupação espectral para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 11)$.

(L)	DF	NA	EF	FF
0,1	31,9473	31,9473	31,9473	31,9473
0,2	63,9300	63,9300	63,9300	63,9300
0,3	95,9172	95,9172	95,9172	95,9172
0,4	127,5593	**127,5588**	127,5587	127,5583
0,5	159,8114	**159,7765**	159,7594	159,7371
0,6	191,3989	**190,9148**	190,6917	190,5243
0,7	221,0076	**218,7921**	218,1948	217,6025
0,8	245,6589	**240,5530**	239,3333	237,8146
0,9	264,1781	**255,9243**	253,8246	251,6780
1,0	276,3268	**265,4876**	263,1738	260,2485
1,1	285,4674	**272,6166**	269,7654	266,8752
1,2	291,1369	**276,6554**	274,1119	270,3225
1,3	295,7145	**280,4709**	277,9386	273,8756
1,4	298,6899	**283,2513**	280,6221	276,4828
1,5	301,2400	**285,6800**	282,8863	278,4897
1,6	303,1808	**287,5275**	284,5473	280,1837
1,7	304,4210	**288,7643**	286,0466	281,7846
1,8	305,6100	**289,6071**	286,7913	282,7645
1,9	306,7076	**290,2876**	288,1898	283,9073
2,0	307,4524	**290,7725**	288,8456	284,7011

Quadro 5.11 - Níveis de ocupação espectral para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$.

(L)	DF	NA	EF	FF
0,1	31,8585	31,8585	31,8585	31,8585
0,2	63,9625	63,9625	63,9625	63,9625
0,3	96,0338	96,0338	96,0338	96,0338
0,4	127,9416	**127,9379**	127,9332	127,9304
0,5	160,0728	**159,9587**	159,8564	159,8221
0,6	190,2152	**189,3664**	188,9579	188,7087
0,7	218,8842	**215,8944**	214,8888	214,1027
0,8	242,1674	**235,9194**	234,2654	232,6756
0,9	257,9888	**248,7148**	246,6130	244,4122
1,0	270,9253	**258,8643**	256,5273	253,4682
1,1	279,5267	**265,4267**	262,7419	259,5711
1,2	285,8817	**270,3955**	267,5998	264,1293
1,3	290,4059	**274,1259**	271,0912	267,4568
1,4	293,6768	**276,8942**	273,9036	270,0058
1,5	296,5637	**279,4765**	276,3088	272,3063
1,6	298,5308	**281,2319**	278,0230	274,2629
1,7	300,3755	**283,1918**	279,8338	275,9004
1,8	301,8847	**284,3866**	281,2066	277,4255
1,9	303,1719	**285,4247**	282,6486	278,5266
2,0	304,1873	**286,7179**	283,4270	279,3448

Quadro 5.12 - Níveis de ocupação espectral para $\beta = 2{,}0$ e $(b_1, b_2) = (7, 17)$.

(L)	DF	NA	EF	FF
0,1	31,9777	31,9777	31,9777	31,9777
0,2	63,8875	63,8875	63,8875	63,8875
0,3	96,4496	**96,4495**	96,4494	96,4494
0,4	127,5565	**127,5190**	127,5203	127,5089
0,5	160,0986	**159,7002**	159,5950	159,5498
0,6	189,0794	**187,3603**	187,0336	186,7501
0,7	216,1094	**211,8355**	211,0555	210,2873
0,8	237,6699	**230,4693**	228,8885	227,4775
0,9	253,7740	**244,0601**	241,9609	239,7787
1,0	265,7088	**253,9385**	251,3109	248,8503
1,1	274,4891	**261,1104**	257,9616	255,3430
1,2	280,6225	**265,9789**	262,9460	260,0470
1,3	285,6166	**269,8630**	267,1037	263,9213
1,4	289,6192	**272,9460**	270,2323	266,8692
1,5	292,6540	**275,6610**	272,9004	269,0982
1,6	294,9542	**277,8916**	275,1398	271,3950
1,7	296,9207	**279,7089**	276,7774	273,2442
1,8	298,6515	**281,1371**	278,5667	274,6522
1,9	299,8809	**282,4332**	279,9134	276,1006
2,0	301,1780	**283,7023**	281,2743	277,4185

Quadro 5.13 - Níveis de ocupação espectral para $\beta = 2{,}0$ e $(b_1, b_2) = (13, 28)$.

(L)	DF	NA	EF	FF
0,1	32,0079	32,0079	32,0079	32,0079
0,2	63,9516	63,9504	63,9513	63,9513
0,3	95,9604	**95,9258**	95,9167	95,9094
0,4	127,7780	**127,4355**	127,3982	127,3459
0,5	157,6605	**156,3649**	156,1163	155,9359
0,6	184,8289	**181,7122**	181,2068	180,7870
0,7	208,1795	**202,4018**	201,4897	200,7091
0,8	226,7102	**218,2238**	217,1272	216,1095
0,9	241,7036	**230,6403**	229,3245	227,8754
1,0	253,0588	**239,9087**	238,5069	236,5990
1,1	261,8343	**247,0153**	245,3699	243,2630
1,2	268,6255	**252,1971**	250,8393	248,5419
1,3	273,8053	**256,5183**	254,6919	252,3084
1,4	278,5936	**260,4249**	258,8014	255,8644
1,5	282,0757	**263,0136**	261,3060	258,5721
1,6	284,6689	**265,0631**	263,7339	260,6315
1,7	287,2099	**266,9652**	265,5265	262,7172
1,8	289,0460	**268,5090**	267,3170	264,3783
1,9	291,0069	**269,9008**	268,5178	265,6242
2,0	292,4650	**271,2614**	269,8835	267,0651

5.3.3.4 Caso Completo: Simulação do NA com $(\boldsymbol{b_1, b_2}) = (\mathbf{5, 14})$ e $\boldsymbol{\beta = 2{,}0}$

O caso completo avaliado apresenta para $(b_1, b_2) = (5, 14)$ e $\beta = 2{,}0$, além da ocupação espectral média, as curvas da vazão média, da probabilidade de bloqueio de banda e das probabilidades de bloqueio para requisições de b_1 e b_2 *slots*, respectivamente, por meio das Figuras 5.16 a 5.19.

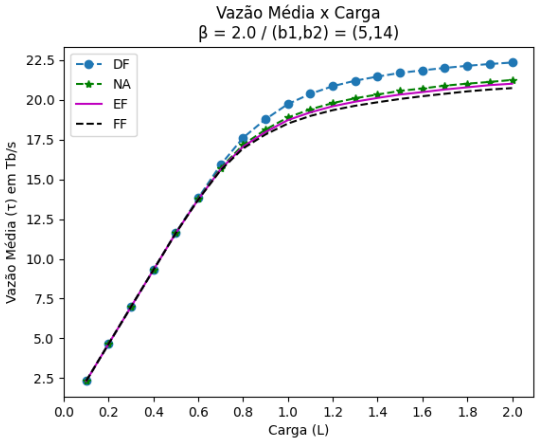


Fig. 5.16 - Vazão média (τ) em Tb/s × carga (L) para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$.

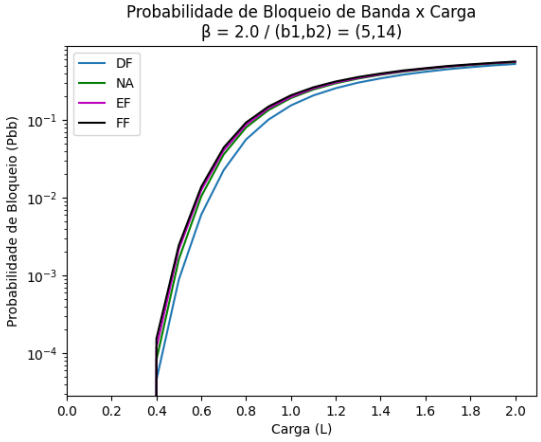


Fig. 5.17 - Probabilidade de bloqueio de banda × carga (L) para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$.

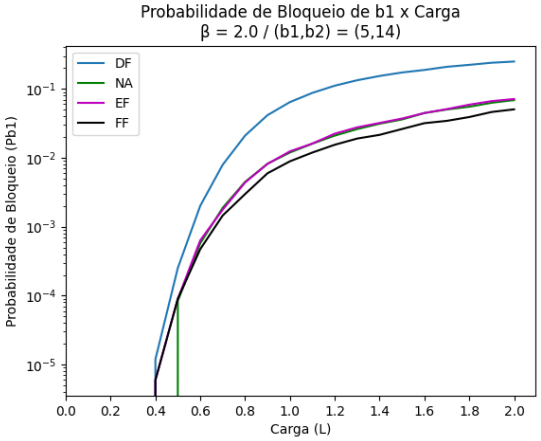


Fig. 5.18 - Probabilidade de bloqueio de b_1 × carga (L) para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$.

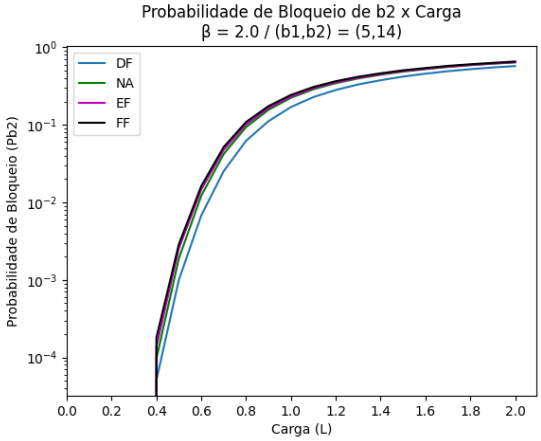


Fig. 5.19 - Probabilidade de bloqueio de b_2 × carga (L) para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$.

A probabilidade de bloqueio aumenta com o aumento da carga. O NA apresentou menor probabilidade de bloqueio de banda e de requisições para b_2 *slots*, em comparação aos algoritmos EF e FF na simulação com $\beta = 2{,}0$, ou seja, com a predominância do tamanho maior, neste caso $b_2 = 14$. No caso da probabilidade de bloqueio para requisições de $b_1 = 5$ *slots*, o NA teve um desempenho melhor que o EF, mas ficou abaixo do FF.

Os Quadros 5.14 e 5.15 listam os resultados numéricos dos quatro algoritmos avaliados para os níveis de ocupação (em %) e a vazão média. Novamente, a faixa de carga (L) na qual o NA supera o EF e o FF é destacada em amarelo e, assim como na ocupação, o NA permanece superior a ambos em todos os pontos até $L = 2{,}0$ também na vazão média.

Quadro 5.14 - Caso NA: níveis de ocupação em % para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$.

(L)	DF	NA	EF	FF
0,1	9,96%	9,96%	9,96%	9,96%
0,2	19,99%	19,99%	19,99%	19,99%
0,3	30,01%	30,01%	30,01%	30,01%
0,4	39,98%	**39,98%**	39,98%	39,98%
0,5	50,02%	**49,99%**	49,96%	49,94%
0,6	59,44%	**59,18%**	59,05%	58,97%
0,7	68,40%	**67,47%**	67,15%	66,91%
0,8	75,68%	**73,72%**	73,21%	72,71%
0,9	80,62%	**77,72%**	77,07%	76,38%
1,0	84,66%	**80,90%**	80,16%	79,21%
1,1	87,35%	**82,95%**	82,11%	81,12%
1,2	89,34%	**84,50%**	83,62%	82,54%
1,3	90,75%	**85,66%**	84,72%	83,58%
1,4	91,77%	**86,53%**	85,59%	84,38%
1,5	92,68%	**87,34%**	86,35%	85,10%
1,6	93,29%	**87,88%**	86,88%	85,71%
1,7	93,87%	**88,50%**	87,45%	86,22%
1,8	94,34%	**88,87%**	87,88%	86,70%
1,9	94,74%	**89,20%**	88,33%	87,04%
2,0	95,06%	**89,60%**	88,57%	87,30%

Quadro 5.15 - Caso NA: níveis de vazão média para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$.

(L)	DF	NA	EF	FF
0,1	2,3170	2,3170	2,3170	2,3170
0,2	4,6521	4,6521	4,6521	4,6521
0,3	6,9848	6,9848	6,9848	6,9848
0,4	9,3057	**9,3054**	9,3051	9,3049
0,5	11,6413	**11,6332**	11,6259	11,6234
0,6	13,8358	**13,7755**	13,7463	13,7285
0,7	15,9226	**15,7108**	15,6390	15,5829
0,8	17,6220	**17,1814**	17,0632	16,9502
0,9	18,7855	**18,1357**	17,9855	17,8291
1,0	19,7396	**18,8995**	18,7325	18,5154
1,1	20,3838	**19,4091**	19,2173	18,9927
1,2	20,8638	**19,8023**	19,6019	19,3573
1,3	21,2112	**20,1056**	19,8877	19,6328
1,4	21,4686	**20,3396**	20,1259	19,8534
1,5	21,7001	**20,5656**	20,3390	20,0594
1,6	21,8629	**20,7227**	20,4930	20,2335
1,7	22,0130	**20,8968**	20,6567	20,3875
1,8	22,1459	**21,0232**	20,7933	20,5384
1,9	22,2600	**21,1324**	20,9318	20,6537
2,0	22,3526	**21,2560**	21,0189	20,7447

5.3.3.5 Discussão dos Resultados para a Simulação do NA

Em todas as simulações, o algoritmo FF foi superado pelo EF, e ambos foram batidos pelo NA em quase todos os pontos de carga, com raras exceções em casos com $L < 0{,}5$. A redução porcentual das perdas por fragmentação em relação ao FF alcançada pelo NA é expressa como:

$$\Re_{NA} = \left[\frac{(B_{FF} - B_{NA})}{(B_{FF} - B_{DF})}\right] \times 100\% \tag{5.31}$$

em que B_A é o número de *slots* bloqueados pelo algoritmo A na mesma simulação que os outros algoritmos da Equação (5.31). A redução porcentual da fragmentação $\Re_{NA}$ é mostrada nas Figuras 5.20 a 5.22 para os quatro pares (b_1, b_2) considerados, respectivamente para $\beta = 0{,}5$, 1,0 e 2,0. Somente os pontos de carga nos quais a simulação alcançou pelo menos 100 bloqueios em todos os algoritmos são exibidos em cada curva.

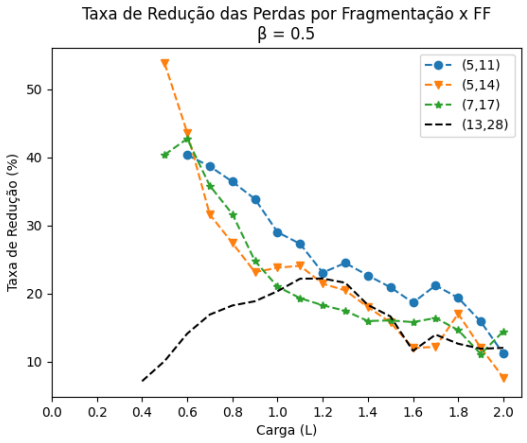


Fig. 5.20 - Redução das perdas por fragmentação × FF para $\beta = 0{,}5$.

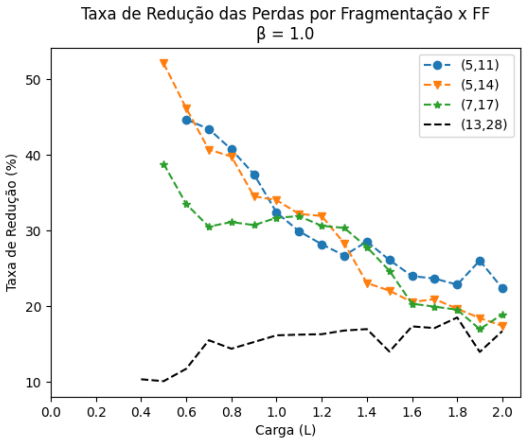


Fig. 5.21 - Redução das perdas por fragmentação × FF para $\beta = 1{,}0$.

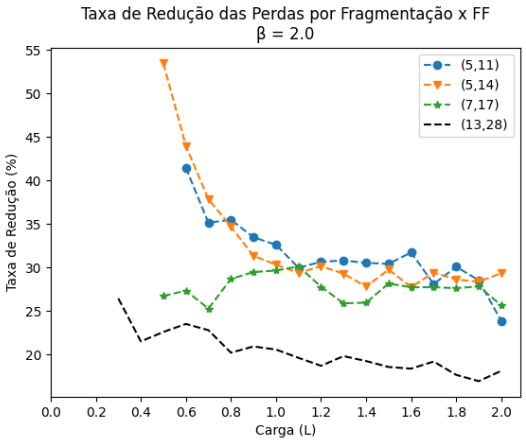


Fig. 5.22 - Redução das perdas por fragmentação × FF para $\beta = 2{,}0$.

São necessárias probabilidades de bloqueio de pelo menos 10^{-4} para atingir uma média de 100 bloqueios com apenas um milhão de chegadas. Os cálculos de $\Re_{NA}$ para probabilidades de bloqueio menores que 10^{-6} podem exigir mais de 100 milhões de requisições para atingir 100 bloqueios, o que inviabiliza o tempo de simulação.

Os resultados sugerem que o NA é capaz de reduzir as perdas por fragmentação cerca de 40 a 50% em relação ao FF e sob condições de tráfego intenso, o que é caracterizado por $L \cong 0{,}5$ para distâncias curtas ($\sim$ 400 km), em que $b_1 = 5$ é ótimo. Para longas distâncias ($\sim$ 8000 km), em que $(b_1, b_2) = (13, 28)$ é ideal, no entanto, o pico de desempenho em $L \cong 0{,}5$ não apareceu. As distâncias intermediárias, otimizadas pelo par $(b_1, b_2) = (7, 17)$, parecem se comportar como distâncias curtas para $\beta \leq 1{,}0$, mas seguem o comportamento de longas distâncias para $\beta = 2{,}0$.

Apenas alguns pontos puderam ser calculados para cargas normalizadas abaixo de $L = 0{,}5$. Eles sugerem uma rápida deterioração do desempenho do NA para grandes distâncias com cargas baixas, levando em alguns casos, a um desempenho inferior em relação ao FF. Isso reflete o fato bem conhecido de que preencher um espaço grande com objetos grandes de tamanhos diferentes é mais difícil do que com objetos pequenos devido à perda de flexibilidade, o que reduz o número de soluções de encaixe. Assim, uma opção seria buscar abordagens proativas alternativas para cargas baixas normalizadas, com a ressalva de que as perdas totais já são muito pequenas sob cargas baixas e, portanto, o esforço para reduzi-las pode não ser justificado.

Em [63], os autores comentam que "a maior eficiência do espectro não implica necessariamente mais desfragmentação". Tal percepção inicial levou à simulação do desempenho de um algoritmo voraz, porém seletivo, que busca um vazio que melhor corresponda ao tamanho solicitado ao longo do processo de atribuição espectral.

Desse modo, tal propósito é baseado na derivação matemática de um algoritmo ótimo para o tráfego incremental que foi discutido em [54], e que representa um limite assintótico do tráfego dinâmico quando a carga é levada ao infinito. Por esta razão, é natural que o NA esteja apto a reduzir as perdas por fragmentação, especialmente sob tráfego intenso. Os resultados confirmam essa expectativa, mas apenas para tamanhos solicitados que sejam pequenos o suficiente para serem acomodados de forma flexível no enlace. Se $b_1 b_2 > S$, como quando $(b_1, b_2) = (13, 28)$ e $S = 320$, não há vazios flexíveis e, portanto, o pico da redução em $L = 0{,}5$ não acontece.

É importante notar que as perdas por fragmentação tendem a zero com o algoritmo FF e outros algoritmos vorazes, tanto para cargas muito leves quanto para condições extremamente sobrecarregadas. No primeiro caso, porque a probabilidade de bloqueio tende a zero, e no segundo, porque o enlace tende a um estado assintótico naturalmente desfragmentado no qual o espectro é completamente tomado por conexões contíguas de b_1 *slots*.

Por essa razão, deve haver um pico em algum lugar na variação da redução das perdas por fragmentação com carga. Os resultados das simulações realizadas sugerem que este pico está próximo de $L = 0{,}5$ ou abaixo, o que pode motivar a extensão desses resultados para cargas mais baixas.

5.4 ALGORITMO CIENTE DO PERFIL COMPLETO DO TRÁFEGO

Esta seção apresenta uma discussão analítica sobre a minimização da ociosidade de recursos em EONs. Os recursos a serem atribuídos são os *slots* espectrais em um único enlace óptico. Um tráfego de duas classes de requisições para conexões é assumido, com cada classe solicitando conexões para um determinado número de *slots* contíguos. A chegada dos pedidos é poissoniana com uma taxa conhecida para cada classe. O tráfego é dinâmico, com cada conexão liberada após um tempo exponencialmente distribuído independente com média unitária.

A segunda proposta se concentra no desempenho de um único enlace óptico elástico e mostra que a sua otimização depende do conhecimento do perfil de tráfego e, por isso, foi considerado um algoritmo de atribuição espectral com ciência do tráfego. Assim, as próximas seções analisam a minimização do número médio de *slots* ociosos sob tráfego dinâmico.

5.4.1 Redução Máxima da Ociosidade

Seja S o número de *slots* contíguos disponíveis no espectro de um único enlace e $s(t)$ o número de *slots* ociosos no instante de tempo t, com $s(0) = S$. A chegada dos pedidos para novas conexões é assumida poissoniana, em que λ_i, $i = 1, 2, 3, \ldots, K$, são as taxas de solicitações de conexões com b_i *slots* contíguos, $b_1 < b_2 < b_3 < \cdots < b_K \leq S$, e K é o número de classes de tráfego. As durações das conexões são variáveis exponenciais independentes com média 1, sendo as taxas de chegada de requisições λ_i expressas em erlangs [56].

Enquanto as atribuições espectrais de conexão são feitas sob controle algorítmico pela rede após a chegada de novas solicitações, as terminações de conexão são decididas unicamente por seus usuários, de modo que o algoritmo de atribuição permanece como um observador das terminações, limitando-se a estar ciente delas e atualizar o estado do sistema em conformidade com as decisões dos usuários.

Somente após a chegada de novas solicitações de conexão, o algoritmo decidirá quais *slots* ociosos contíguos, se houver, serão atribuídos à nova conexão (se for o caso, a depender da política: não voraz ótima ou voraz) com o propósito de minimizar a ociosidade esperada Ω, definida como [56]:

$$\Omega(S) = \lim_{T\to\infty} \frac{E\left[\int_0^T s(\tau)d\tau\right]}{T} \tag{5.32}$$

em que $E[.]$ é a expectativa do argumento. A minimização é executada por um algoritmo de atribuição espectral com conhecimento do tráfego (*traffic awareness*) que toma cada solicitação e a rejeita ou a atribui a um dos vazios espectrais que são grandes o suficiente para acomodá-la. O conhecimento do tráfego significa que o algoritmo conhece os valores de λ_i e b_i, $i = 1, 2, 3, \dots, K$, e usa essas informações para minimizar $\Omega(S)$. Para este propósito, o processo contínuo é dividido em vazios episódicos sem interromper a dinâmica geral da população de *slots* ociosos [56].

Após a chegada de uma requisição por b_i *slots*, o algoritmo deve procurar todos os vazios com tamanho $v \geq b_i$ no sistema. Se tal vazio não existir, o pedido deve ser rejeitado. Caso contrário, o algoritmo deve decidir em qual vazio atualmente elegível, se houver, a solicitação deve ser atribuída (ou não, em função da política utilizada) para minimizar (5.32), considerando que [56]:

$$s(t) = \sum_{m=1}^{V(t)} v_m \tag{5.33}$$

em que $V(t)$ é o número de vazios no tempo t e v_m é o tamanho do m-ésimo vazio. Para fazer isso, o algoritmo precisa ser baseado em políticas, ou seja, ele deve ser capaz de decidir se aceita ou não a solicitação, e em qual vazio ela será atribuída (se for o caso).

Em um algoritmo genérico baseado em políticas, as políticas são definidas por variáveis binárias p_i^v, em que $v = 1, 2, 3, \dots, S$, com $p_i^v = 1$ se o algoritmo aceitar uma solicitação de b_i *slots* contíguos quando o vazio tem tamanho $v \geq b_i$, e $p_i^v = 0$ se o algoritmo

a rejeitar. Um valor ideal de p_i^v é denotado por q_i^v. O vetor $\mathbf{P}^v = (p_1^v, p_2^v, \dots, p_K^v)$ representa a política seguida pelo algoritmo $\mathbf{P}$ quando v *slots* estão disponíveis, e $\mathbf{Q}^v = (q_1^v, q_2^v, \dots, q_K^v)$ representa a política ótima para o vazio de tamanho v [56].

Políticas especiais, e dignas de nota, incluem a política voraz, dada por $(1, 1, \dots, 1)$; e a política inativa, descrita por $(0, 0, \dots, 0)$. O algoritmo $\mathbf{P}$ é definido como o conjunto de todas as políticas $(\mathbf{P}^1, \mathbf{P}^2, \dots, \mathbf{P}^S)$, e $\mathbf{Q} = (\mathbf{Q}^1, \mathbf{Q}^2, \dots, \mathbf{Q}^S)$ denota um algoritmo ótimo. Diz-se que uma política é ativa se $p_i^v = 1$ para pelo menos um valor de $i \in [1, K]$ para todos os vazios $v \geq b_1$ [56].

Por convenção, quaisquer duas conexões contíguas são separadas por um vazio de zero *slot*, ou quando uma conexão é contígua a uma extremidade do espectro disponível. Então, o número de vazios é simplesmente 1 mais o número de conexões ativas no enlace. Quando uma conexão de b_i *slots* é atribuída a um vazio v, a nova conexão será ladeada por dois vazios cujos tamanhos somam $(v - b_i)$. Se a atribuição for adjacente a uma das extremidades do vazio, um dos novos vazios será zero e o outro $(v - b_i)$. Neste ponto, é importante destacar que apenas as atribuições adjacentes são consideradas a seguir na análise de otimização em questão [56].

Por outro lado, quando uma conexão é encerrada, seus *slots* recém-liberados se aglutinam em um novo vazio com os *slots* ociosos de seus dois antigos vazios vizinhos (que podem ser zero ou positivos), substituindo assim os dois vazios antigos, que também são efetivamente "terminados". Portanto, cada vazio individual $v(t)$ é um processo episódico no sentido de que tem uma duração média finita, mesmo que a soma de todos os vazios $s(t)$ seja um processo contínuo e interminável alimentado pela renovação dos vazios associados com terminações de conexão [56].

Cada novo vazio é, então, gerado pelo término de uma conexão com um tamanho inicial que é pelo menos o tamanho da conexão terminada. Durante a sua vida, ele pode ou não ser sucedido por vazios menores quando é designado para acomodar solicitações. O término de um vazio acontece quando ele é substituído por um novo vazio gerado pelo término de uma conexão vizinha por seu usuário. A maioria dos vazios no enlace tem duas conexões vizinhas, cada uma terminando independentemente com taxa 1, de modo que a taxa de terminação do vazio é 2 e sua duração média é de $1/2$. É verdade que um vazio pode atingir o tamanho zero antes que uma conexão vizinha seja encerrada, mas isso apenas acrescenta dois vazios de tamanho zero, cuja contribuição para (5.32) será nula [56].

Devido à sua natureza episódica, a ociosidade esperada de um vazio $v_k(t)$ a partir do instante t_{0k} e terminando no tempo t_{fk} é definida como [56]:

$$\omega_k = \int_{t_{0k}}^{t_{fk}} v(t) \cdot dt \tag{5.34}$$

de modo que, de (5.32) e (5.33):

$$\Omega(S) = \lim_{T\to\infty} \frac{\sum v_k}{T}, \qquad \forall v_k | 0 \leq t_{0k} \leq T \tag{5.35}$$

Uma vez que as conexões têm duração exponencial com média 1, então a ociosidade esperada de um vazio v com $0 < v < b_1$ é $v/2$ se o vazio ficar ladeado por duas conexões, mas simplesmente v se o vazio for adjacente a uma extremidade do espectro ou a uma conexão permanente, se permitido, pois não há espaço para acomodar nenhuma nova requisição. Na sequência, discute-se apenas o caso de vazios que são ladeados por duas conexões com duração média igual a 1.

Se $v < b_1$, o vazio não pode acomodar nenhuma solicitação, de modo que o único evento possível é o término de uma conexão vizinha e do próprio vazio após um tempo médio de espera de $1/2$, o que leva a uma ociosidade esperada total de $v/2$ [56].

Se $v \geq b_1$ e $K = 2$, então apenas $K + 1 = 3$ tipos de "próximos eventos" podem resultar num vazio em qualquer intervalo de tempo infinitesimal dt quando levado a zero [56]:

a) terminação por coalescência, com probabilidade $2 \cdot dt$;

b) ou redução a $(v - b_i)$, com probabilidade $p_i^v \lambda_i \cdot dt$ para algum $b_i \leq v$, $i = 1, 2, 3, \ldots, K$, com probabilidade $\lambda^{\mathbf{P}}(v) \cdot dt$, em que:

$$\lambda^{\mathbf{P}} = \sum_{\{i | b_i \leq v\}} p_i^v \lambda_i \tag{5.36}$$

c) ou nada, com probabilidade $\left[1 - \left(2 + \lambda^{\mathbf{P}}(v)\right) dt\right]$.

Para que um vazio termine por coalescência entre t e $(t + dt)$, é necessário que os seguintes dois eventos independentes ocorram:

a) que nada tenha ocorrido entre os instantes 0 e t, cuja probabilidade é:

$$\lim_{dt\to 0}\left[1-\left(2+\lambda^{\mathbf{P}}(v)\right)dt\right]^{\frac{t}{dt}}=e^{-\left(2+\lambda^{\mathbf{P}}(v)\right)t} \tag{5.37}$$

b) que uma coalescência aconteça entre t e $(t+dt)$, cuja probabilidade é $2\cdot dt$.

Sendo dois eventos independentes, conclui-se que a probabilidade de um término por coalescência ocorrer entre t e $(t+dt)$ é o produto $2e^{-\left(2+\lambda^{\mathbf{P}}(v)\right)t}dt$. Portanto, a probabilidade τ de que o primeiro evento na vida do vazio seja um fim por coalescência é [56]:

$$\tau=\int_0^{\infty}2\cdot e^{-[2+\lambda^{\mathbf{P}}(s)]t}\cdot dt=\frac{2}{2+\lambda^{\mathbf{P}}(s)} \tag{5.38}$$

Da mesma forma, o tempo médio de espera $\bar{t}$ para que uma terminação por coalescência venha a ocorrer antes da primeira alocação no vazio é dado por [56]:

$$\bar{t}=\int_0^{\infty}t\cdot(2\cdot dt)\cdot e^{-[2+\lambda^{\mathbf{P}}(s)]t}=\frac{2}{[2+\lambda^{\mathbf{P}}(s)]^2} \tag{5.39}$$

Com probabilidade τ, a ociosidade do vazio será dada pela área de um retângulo de altura v e base de comprimento médio $\bar{t}$, resultando numa ociosidade média $\bar{t}v$. Com probabilidade $(1-\tau)$, o próximo evento será a chegada bem-sucedida de uma requisição por b_j *slots* contíguos, sendo $p_j^v=1$ na política $\mathbf{P}^v$ do algoritmo **P** em análise. Nessas condições, supondo que a alocação seja adjacente a uma extremidade do vazio, o tamanho efetivo deste será reduzido de v para $\left(v-b_j\right)$ com probabilidade $\lambda_j/\lambda^{\mathbf{P}}(v)$, sendo $b_j\leq v$.

Assim sendo, no caso do primeiro evento ser uma alocação bem-sucedida, a ociosidade do vazio será dada pela soma de duas partes:

a) a ociosidade gerada pelo tempo médio de espera pela chegada de uma requisição aprovada pela política $\mathbf{P}^v$, dada por $v/\lambda^{\mathbf{P}}(v)$ e;

b) o valor médio da ociosidade do novo vazio residual gerado pela alocação, considerando todas as alocações permitidas pela política $\mathbf{P}^v$.

Então, considerando alocações sempre adjacentes a uma extremidade do vazio, a ociosidade esperada produzida por uma política **P** em qualquer vazio v ladeado por duas conexões em um único enlace pode ser obtida a partir da seguinte **equação de recorrência** [56]:

$$\omega(v,\mathbf{P}^v) = \tau\bar{t}v + (1-\tau)\left[\frac{v}{\lambda^{\mathbf{P}}(v)} + \sum_{\{j|b_j\le v\}} p_j^v \left(\frac{\lambda_j}{\lambda^{\mathbf{P}}(v)}\right)\omega_{min}(v-b_j)\right]$$

$$= \left(\frac{2}{2+\lambda^{\mathbf{P}}(v)}\right)\left(\frac{2}{[2+\lambda^{\mathbf{P}}(v)]^2}\right)v + \left(\frac{\lambda^{\mathbf{P}}(v)}{2+\lambda^{\mathbf{P}}(v)}\right)\left[\frac{v}{\lambda^{\mathbf{P}}(v)} + \sum_{\{j|b_j\le v\}} p_j^v \left(\frac{\lambda_j}{\lambda^{\mathbf{P}}(v)}\right)\omega_{min}(v-b_j)\right]$$

$$= \frac{4v}{[2+\lambda^{\mathbf{P}}(v)]^3} + \left(\frac{\lambda^{\mathbf{P}}(v)}{2+\lambda^{\mathbf{P}}(v)}\right)\left[\frac{v}{\lambda^{\mathbf{P}}(v)} + \frac{\sum_{\{j|b_j\le v\}} p_j^v \lambda_j \omega_{min}(v-b_j)}{\lambda^{\mathbf{P}}(v)}\right] \tag{5.40}$$

$$= \frac{4v}{[2+\lambda^{\mathbf{P}}(v)]^3} + \left[\frac{v + \sum_{\{j|b_j\le v\}} p_j^v \lambda_j \omega_{min}(v-b_j)}{2+\lambda^{\mathbf{P}}(v)}\right] \quad se \quad \lambda^{\mathbf{P}}(v) > 0,$$

$$= v/2 \quad se \quad \lambda^{\mathbf{P}}(v) = 0.$$

A Figura 5.23, na qual $< x >$ denota a média de uma variável aleatória com média x, ilustra as oito possíveis "histórias de vida" de um vazio para um sistema de duas classes e suas respectivas probabilidades, com a área em branco em cada gráfico representando os valores sucessivos da ociosidade do vazio sob uma política voraz de forma recorrente, dados $(b_1, b_2) = (3, 5)$ e um tamanho inicial do vazio $S = 10$.

A equação (5.40) calcula as médias das três possibilidades em cada ciclo de recorrência e expressa a ociosidade em função de v e dos valores de ociosidade $\omega(v-b_1)$, $\omega(v-b_2)$, ..., $\omega(v-b_K)$, que devem, então, ser previamente calculados [56].

Com a aplicação recorrente de (5.40) para v crescente, a ociosidade esperada associada a cada tamanho de vazio pode então ser calculada e armazenada em uma tabela para todas as políticas **P**. A inicialização desse processo é dada por $\omega(v) = v/2$ para $v \in \{0, 1, 2, \ldots, (b_1 - 1)\}$ [56].

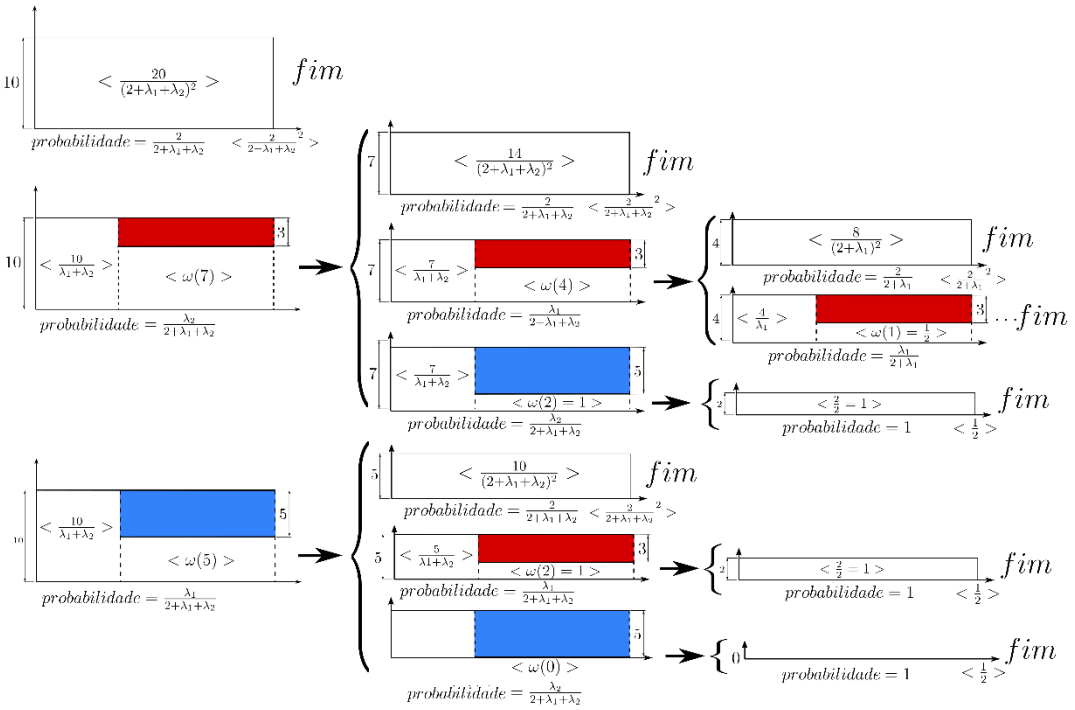


Fig. 5.23 - Mecanismo de recorrência para $(b_1, b_2) = (3, 5)$ e $S = 10$ [56].

Se $v \in \{b_1, (b_1 + 1), (b_1 + 2), \dots, (b_2 - 1)\}$, então $\lambda^{\mathbf{P}}(v) = p_1^v \lambda_1$ e existem apenas duas políticas, dadas por $p_1^v = 0$, que é inativa e somente será ótima quando for a única permitida, e $p_1^v = 1$. Então, considerando somente as políticas ativas, tem-se [56]:

$$\omega_{min}(v) = \frac{4v}{(2+\lambda_1)^3} + \frac{v+\lambda_1 \omega_{min}(v-b_1)}{2+\lambda_1} \tag{5.41}$$

Se $v \in \{b_2, (b_2 + 1), (b_2 + 2), \dots, (b_3 - 1)\}$, então $\lambda^{\mathbf{P}}(v) = p_1^v \lambda_1 + p_2^v \lambda_2$ e existem 4 políticas candidatas para a otimidade, dadas por $(p_1^v, p_2^v) = (0,0), (0,1), (1,0)$ e (1,1). Em seguida, é necessário conectar esses 4 vetores em (5.40) para encontrar a política ótima que produz o valor mínimo para $\omega(v, \mathbf{P}^v)$ dado por [56]:

$$\omega_{min}(v) = \min\left(\frac{v}{2}, \frac{4v}{(2+\lambda_1)^3} + \frac{v + \lambda_1 \omega_{min}(v - b_1)}{2 + \lambda_1}, \frac{4v}{(2+\lambda_2)^3} + \right. \tag{5.42}$$

$$\left. + \frac{v + \lambda_2 \omega_{min}(v - b_2)}{2 + \lambda_2}, \frac{4v}{(2+\lambda_1+\lambda_2)^3} + \frac{v + \lambda_1 \omega_{min}(v - b_1) + \lambda_2 \omega_{min}(v - b_2)}{2 + \lambda_1 + \lambda_2}\right)$$

Seguindo o mesmo padrão, se $b_3 \leq v \leq b_4 - 1$ então $\lambda^{\mathbf{P}}(v) = p_1^v\lambda_1 + p_2^v\lambda_2 + p_3^v\lambda_3$ e há $2^3 - 1 = 7$ políticas ativas a serem comparadas para a minimização da ociosidade esperada um vazio v, e assim por diante. Para todo $v \geq b_k$ há, então, $2^K - 1$ políticas ativas a serem comparadas para cada $v \leq S$. Isso significa que há mais de um milhão de políticas que devem ser comparadas para cada v se $K = 20$, mais de 1 bilhão de políticas se $K = 30$, um trilhão se $K = 40$, e assim por diante, construindo, então, um claro problema de escalabilidade para a minimização efetiva da ociosidade esperada [56].

Uma vez que os valores da ociosidade mínima esperada $\omega(v, \mathbf{Q})$ para cada v de 1 a S são calculados e armazenados em uma tabela juntamente com suas respectivas políticas ótimas $\{q_1^s, q_2^s, \dots, q_K^s\}$, o melhor vazio espectral para acomodar cada solicitação de conexão é escolhido comparando a redução da ociosidade obtida pela atribuição em cada vazio disponível e selecionando o que oferece a maior recompensa em termos de redução da ociosidade esperada. Havendo mais de um vazio nessa condição, escolhe-se o de menor frequência [56].

Após a chegada de um pedido de b_j *slots*, o algoritmo analisa o estado do enlace e identifica todos vazios V_j com pelo menos b_j *slots* que podem ser atribuídos para acomodar a solicitação. Sejam $v_1, v_2, \dots, v_{V_j}$ os tamanhos desses vazios. Em seguida, a esperada redução da ociosidade (IR, do inglês *Idleness Reduction*) produzida pela atribuição do vazio v_i é [56]:

$$(\Delta\omega)_i^j = \omega(v_i, \mathbf{Q}) - \omega\left(v_i - q_j^{v_i} b_j, \mathbf{Q}\right) \tag{5.43}$$

e a redução máxima da ociosidade é igual a:

$$(\Delta\omega)_{MIR} = \max_i (\Delta\omega)_i^j \tag{5.44}$$

Finalmente, o algoritmo atribui os b_j *slots* requisitados para qualquer vazio com $(\Delta\Omega)_i^j = (\Delta\Omega)_{MIR}$. No caso de um empate entre dois ou mais vazios ótimos, escolhe-se aquele com frequências mínimas.

5.4.2 Tabulação e Simulações

O Quadro 5.16 mostra o cálculo da ociosidade mínima associada com tamanhos de vazios que variam de 0 até 50, bem como as recompensas correspondentes (reduções de

ociosidade, identificadas como Δ_1 e Δ_2) das atribuições de b_1 *slots* e b_2 *slots*, dados $\beta = 1{,}0$, $(b_1, b_2) = (5, 11)$, com destaque em verde, e $(\lambda_1, \lambda_2) = (8, 8)$, que produzem uma carga normalizada de $0{,}4$ num espectro de 320 *slots*, caracterizando um tráfego leve.

Neste primeiro cenário, a política voraz $(1, 1)$ é viável e ótima para todos $v \geq 11$ e a política $(1, 0)$ é a única possível para $5 \leq v < 11$. Com um tráfego leve, todas as políticas ótimas são vorazes, o que sugere um papel para políticas de atribuição vorazes na minimização da ociosidade. Porém, sob tráfego intenso, as políticas não vorazes inteligentes tornam-se ótimas para alguns tamanhos de vazios, quando só existem vazios que aumentam a ociosidade para o aceite de um determinado tipo de requisição. As rejeições forçadas, que se devem ao tamanho insuficiente do vazio para acomodar a solicitação, são destacadas em amarelo.

Os valores mínimos de ociosidade são calculados para cada política ativa e destacados em azul quando são ótimos sobre todas as políticas ativas. No intervalo de vazios exibido no Quadro 5.16, a maior recompensa para atribuições de 5 *slots* é obtida para o tamanho vazio 47, seguido por 42, 48, 37 e 50, nesta ordem (destaque em laranja). A maior recompensa para atribuições de 11 *slots* é obtida para o tamanho vazio 49, seguido por 50, 48, 44 e 45, nesta ordem (destaque em cinza).

O Quadro 5.17 apresenta, novamente, o cálculo da ociosidade mínima associada a tamanhos de vazios que variam de 0 até 50, assim como as respectivas recompensas, porém agora para $\beta = 1{,}0$, $(b_1, b_2) = (5, 11)$ e $(\lambda_1, \lambda_2) = (20, 20)$, que geram uma carga normalizada de $1{,}0$ para um espectro de 320 *slots*, o que caracteriza um tráfego pesado no limite da sobrecarga. No segundo cenário analisado, a política voraz $(1, 1)$ é ótima para a maioria dos tamanhos de vazios, mas não todos: as exceções são a política $(0, 1)$ que rejeita solicitações de 5 *slots* para $v = 11, 12, 13$ e 14; e a política $(1, 0)$ que rejeita solicitações de 11 *slots* para $v = 15$ e 20, que são múltiplos de $b_1 = 5$. Com o aumento do tráfego, aparecem políticas não vorazes como ótimas para a redução da ociosidade em alguns tamanhos de vazios.

Conforme mostrado no Quadro 5.17, calculado com $L = 1{,}0$, a maior recompensa para atribuições de 5 *slots* é alcançada novamente para o tamanho de vazio 47, seguido por 42, 20, 31 e 36, nessa ordem. A maior recompensa para atribuições de 11 *slots* é alcançada novamente para o tamanho vazio 49, seguido por 50, 44, 45 e 48, nessa ordem. Recompensas negativas (em vermelho) estão associadas a rejeições prescritas por políticas ótimas não vorazes, que assim evitam o aumento da ociosidade esperada na eventual ausência de vazios com recompensas positivas. O mesmo código de cores usado no Quadro 5.16 é válido para os Quadros 5.17 e 5.18.

Quadro 5.16 - Ociosidade mínima para $L = 0{,}4$, $\beta = 1{,}0$ e $(b_1, b_2) = (5, 11)$.

	Política (0,1)		Política (1,1)		Política (1,1) (voraz)		Ociosidade Mínima e Política Ótima				Recompensas	
	0	1	1	0	1	1	Tx Chegada	Ociosidade	Políticas Selecionadas		D1	D2
v	$\lambda(v,P)$	$\omega(v,P)$	$\lambda(v,P)$	$\omega(v,P)$	$\lambda(v,P)$	$\omega(v,P)$	$\lambda(v,Q)$	$\omega(v,Q)$	q_1	q_2		
0	0	0	0	0	0	0	0	0	0	0		
1	0	0,5	0	0,5	0	0,5	0	0,5	0	0		
2	0	1	0	1	0	1	0	1	0	0		
3	0	1,5	0	1,5	0	1,5	0	1,5	0	0		
4	0	2	0	2	0	2	0	2	0	0		
5	0	2,5	8	0,52	8	0,52	8	0,52	1	0	0,52	
6	0	3	8	1,024	8	1,024	8	1,024	1	0	0,524	
7	0	3,5	8	1,528	8	1,528	8	1,528	1	0	0,528	
8	0	4	8	2,032	8	2,032	8	2,032	1	0	0,532	
9	0	4,5	8	2,536	8	2,536	8	2,536	1	0	0,536	
10	0	5	8	1,456	8	1,456	8	1,456	1	0	0,936	
11	8	1,144	8	1,9632	16	1,0737668	16	1,0737668	1	1	0,0497668	1,0737668
12	8	1,648	8	2,4704	16	1,5762305	16	1,5762305	1	1	0,0482305	1,0762305
13	8	2,152	8	2,9776	16	2,0786941	16	2,0786941	1	1	0,0466941	1,0786941
14	8	2,656	8	3,4848	16	2,5811578	16	2,5811578	1	1	0,0451578	1,0811578
15	8	3,16	8	2,7248	16	2,3796214	16	2,3796214	1	1	0,9236214	0,3796214
16	8	2,08	8	2,5230134	16	1,6082036	16	1,6082036	1	1	0,5344368	1,0882036
17	8	2,5872	8	3,0289844	16	2,1117622	16	2,1117622	1	1	0,5355318	1,0877622
18	8	3,0944	8	3,5349553	16	2,6153208	16	2,6153208	1	1	0,5366267	1,0873208
19	8	3,6016	8	4,0409262	16	3,1188794	16	3,1188794	1	1	0,5377217	1,0868794
20	8	4,1088	8	3,9836971	16	3,3095492	16	3,3095492	1	1	0,9299278	0,7735492
21	8	3,3488	8	3,4705629	16	2,5429382	16	2,5429382	1	1	0,9347346	1,0869382
22	8	3,1470134	8	3,9774098	16	2,6531021	16	2,6531021	1	1	0,5413398	1,5793353
23	8	3,6529844	8	4,4842567	16	3,1564645	16	3,1564645	1	1	0,5411437	1,580234
24	8	4,1589553	8	4,9911036	16	3,6598269	16	3,6598269	1	1	0,5409475	1,5811328
25	8	4,6649262	8	5,2476393	16	4,0241276	16	4,0241276	1	1	0,7145785	1,4429699
26	8	4,6076971	8	4,7383506	16	3,6500814	16	3,6500814	1	1	1,1071431	1,27046
27	8	4,0945629	8	4,9304817	16	3,4124322	16	3,4124322	1	1	0,7593301	1,8042285
28	8	4,6014098	8	5,4371716	16	3,916194	16	3,916194	1	1	0,7597296	1,8044318
29	8	5,1082567	8	5,9438615	16	4,4199559	16	4,4199559	1	1	0,760129	1,8046351
30	8	5,6151036	8	6,3393021	16	4,8619126	16	4,8619126	1	1	0,837785	1,7430332
31	8	5,8716393	8	6,1440651	16	4,8366533	16	4,8366533	1	1	1,186572	1,5271042
32	8	5,3623506	8	6,0579457	16	4,4465569	16	4,4465569	1	1	1,0341248	1,9036187
33	8	5,5544817	8	6,5649552	16	4,7756542	16	4,7756542	1	1	0,8594602	2,1225522
34	8	6,0611716	8	7,0719647	16	5,2795065	16	5,2795065	1	1	0,8595505	2,123042
35	8	6,5678615	8	7,5295301	16	5,7558897	16	5,7558897	1	1	0,8939771	2,0960628
36	8	6,9633021	8	7,6133227	16	5,9628162	16	5,9628162	1	1	1,1261629	1,9386886
37	8	6,7680651	8	7,4052456	16	5,6794387	16	5,6794387	1	1	**1,2328818**	2,0293573
38	8	6,6819457	8	7,7725234	16	5,7763237	16	5,7763237	1	1	1,0006695	2,3638916
39	8	7,1889552	8	8,2796052	16	6,2803936	16	6,2803936	1	1	1,0008872	2,3641996
40	8	7,6959647	8	8,7647118	16	6,7722551	16	6,7722551	1	1	1,0163654	2,3522992
41	8	8,1535301	8	9,034253	16	7,1168891	16	7,1168891	1	1	1,1540729	2,2549765
42	8	8,2373227	8	8,911551	16	7,0359586	16	7,0359586	1	1	**1,3565199**	2,1993053
43	8	8,0292456	8	9,093059	16	6,9618839	16	6,9618839	1	1	1,1855601	2,5153269
44	8	8,3965234	8	9,6003149	16	7,3884218	16	7,3884218	1	1	1,1080282	**2,6127676**
45	8	8,9036052	8	10,097804	16	7,8872027	16	7,8872027	1	1	1,1149476	**2,6076962**
46	8	9,3887118	8	10,477511	16	8,3083406	16	8,3083406	1	1	1,1914516	2,5524509
47	8	9,658253	8	10,516767	16	8,4205803	16	8,4205803	1	1	**1,3846217**	2,4577641
48	8	9,535551	8	10,561507	16	8,3179541	16	8,3179541	1	1	**1,3560702**	**2,6385154**
49	8	9,717059	8	11,006737	16	8,6068279	16	8,6068279	1	1	1,2184061	**2,8305042**
50	8	10,224315	8	11,509762	16	9,1087808	16	9,1087808	1	1	**1,2215781**	**2,8283872**

Quadro 5.17 - Ociosidade mínima para $L = 1{,}0$, $\beta = 1{,}0$ e $(b_1, b_2) = (5, 11)$.

	Política (0,1)		Política (1,0)		Política (1,1) (voraz)		Ociosidade Mínima e Política Ótima				Recompensas	
	0	1	1	0	1	1	Tx Chegada	Ociosidade	Políticas Selecionadas		D1	D2
v	$\lambda(v,P)$	$\omega(v,P)$	$\lambda(v,P)$	$\omega(v,P)$	$\lambda(v,P)$	$\omega(v,P)$	$\lambda(v,Q)$	$\omega(v,Q)$	q_1	q_2		
0	0	0	0	0	0	0	0	0	0	0		
1	0	0,5	0	0,5	0	0,5	0	0,5	0	0		
2	0	1	0	1	0	1	0	1	0	0		
3	0	1,5	0	1,5	0	1,5	0	1,5	0	0		
4	0	2	0	2	0	2	0	2	0	0		
5	0	2,5	20	0,229151	20	0,229151	20	0,229151	1	0	0,229151	
6	0	3	20	0,7295267	20	0,7295267	20	0,7295267	1	0	0,2295267	
7	0	3,5	20	1,2299023	20	1,2299023	20	1,2299023	1	0	0,2299023	
8	0	4	20	1,730278	20	1,730278	20	1,730278	1	0	0,230278	
9	0	4,5	20	2,2306536	20	2,2306536	20	2,2306536	1	0	0,2306536	
10	0	5	20	0,6666211	20	0,6666211	20	0,6666211	1	0	0,4374701	
11	20	0,5041322	20	1,1673383	40	0,6098923	20	0,5041322	0	1	-0,2253944	0,5041322
12	20	1,0045079	20	1,6680555	40	1,1101252	20	1,0045079	0	1	-0,2253944	0,5045079
13	20	1,5048835	20	2,1687726	40	1,6103581	20	1,5048835	0	1	-0,2253944	0,5048835
14	20	2,0052592	20	2,6694898	40	2,1105909	20	2,0052592	0	1	-0,2253944	0,5052592
15	20	2,5056349	20	1,2934723	40	1,6277723	20	1,2934723	1	0	0,6268511	-0,7065277
16	20	0,9416023	20	1,1915853	40	0,7309987	40	0,7309987	1	1	0,2268665	0,5018477
17	20	1,4423195	20	1,6923024	40	1,2314105	40	1,2314105	1	1	0,2269026	0,5018838
18	20	1,9430367	20	2,1930196	40	1,7318222	40	1,7318222	1	1	0,2269387	0,5019199
19	20	2,4437538	20	2,6937368	40	2,232234	40	2,232234	1	1	0,2269748	0,501956
20	20	2,944471	20	2,0924879	40	2,1554255	20	2,0924879	1	0	**0,7990157**	-0,1381657
21	20	1,5684535	20	1,6269785	40	1,166667	40	1,166667	1	1	0,4356683	0,5000459
22	20	1,4665665	20	2,1277285	40	1,3514462	40	1,3514462	1	1	0,1200357	0,847314
23	20	1,9672837	20	2,6284785	40	1,8518752	40	1,8518752	1	1	0,1200529	0,8473673
24	20	2,4680008	20	3,1292285	40	2,3523041	40	2,3523041	1	1	0,1200701	0,8474206
25	20	2,968718	20	3,0480168	40	2,547896	40	2,547896	1	1	0,4554081	0,5426368
26	20	2,3674691	20	2,2521917	40	1,7919463	40	1,7919463	1	1	0,6252792	0,498474
27	20	1,9019598	20	2,4660029	40	1,6359553	40	1,6359553	1	1	0,2845091	0,9049566
28	20	2,4027097	20	2,9667686	40	2,1364096	40	2,1364096	1	1	0,2845345	0,9049992
29	20	2,9034597	20	3,4675342	40	2,636864	40	2,636864	1	1	0,2845599	0,9050417
30	20	3,4042097	20	3,6911752	40	2,9921578	40	2,9921578	1	1	0,4442618	0,7599238
31	20	3,322998	20	3,0497783	40	2,5894995	40	2,5894995	1	1	**0,7975532**	0,4970116
32	20	2,5271729	20	2,9537986	40	2,0982145	40	2,0982145	1	1	0,4622592	0,9315475
33	20	2,7409842	20	3,4545873	40	2,4483797	40	2,4483797	1	1	0,31197	1,0969335
34	20	3,2417498	20	3,955376	40	2,9488543	40	2,9488543	1	1	0,3119903	1,0969791
35	20	3,7425154	20	4,3242005	40	3,3802048	40	3,3802048	1	1	0,388047	1,0279007
36	20	3,9661564	20	4,0039777	40	3,3054653	40	3,3054653	1	1	**0,7159658**	0,7575693
37	20	3,3247596	20	3,6031852	40	2,7354075	40	2,7354075	1	1	0,637193	0,9434613
38	20	3,2287798	20	3,9673474	40	2,8517349	40	2,8517349	1	1	0,4033553	1,2157796
39	20	3,7295685	20	4,4681545	40	3,3522313	40	3,3522313	1	1	0,403377	1,2158216
40	20	4,2303572	20	4,9061216	40	3,8198114	40	3,8198114	1	1	0,4396066	1,1829474
41	20	4,5991818	20	4,8840068	40	3,9772722	40	3,9772722	1	1	0,6718069	0,9851144
42	20	4,278959	20	4,4116026	40	3,5379376	40	3,5379376	1	1	**0,8025301**	0,9484381
43	20	3,8781665	20	4,563185	40	3,3832499	40	3,3832499	1	1	0,5315149	1,2850354
44	20	4,2423286	20	5,0640119	40	3,8121903	40	3,8121903	1	1	0,459959	**1,3638106**
45	20	4,7431358	20	5,5349149	40	4,2970323	40	4,2970323	1	1	0,4772208	**1,348178**
46	20	5,1811028	20	5,7238913	40	4,6012821	40	4,6012821	1	1	0,6240099	1,2210773
47	20	5,158988	20	5,3703264	40	4,3803484	40	4,3803484	1	1	**0,8424108**	1,0748831
48	20	4,6865838	20	5,2755314	40	4,059095	40	4,059095	1	1	0,6758452	**1,3236875**
49	20	4,8381663	20	5,7113075	40	4,3426099	40	4,3426099	1	1	0,5304196	**1,490875**
50	20	5,3389931	20	6,1979031	40	4,8356821	40	4,8356821	1	1	0,5386499	**1,4834508**

As Figuras 5.24 e 5.25 ilustram graficamente a relação entre as recompensas (D1 e D2) e os respectivos tamanhos de vazios até $v = 50$, conforme o caso apresentado no Quadro 5.17. Por meio dessas figuras, é possível observar a existência de máximos locais em tamanhos quasi-periódicos.

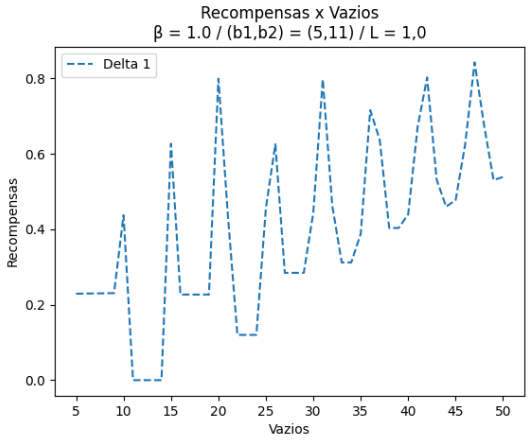


Fig. 5.24 - Recompensas D1 × vazios até $v = 50$.

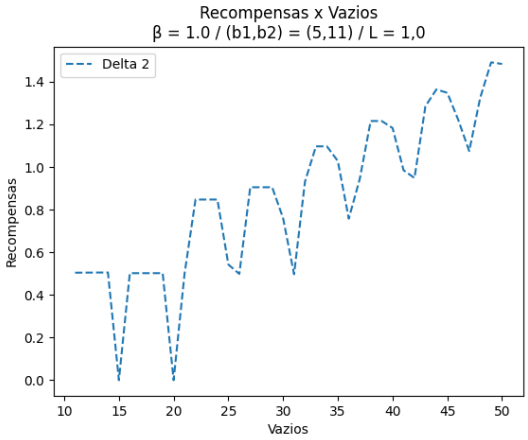


Fig. 5.25 - Recompensas D2 × vazios até $v = 50$.

Na sequência, as Figuras 5.26 e 5.27 estendem a visualização dos resultados obtidos para o caso anterior, com $L = 1{,}0$, até $v = 160$. Verifica-se um crescimento médio dos valores de recompensa à medida que aumenta o tamanho do vazio, o que confirma a percepção de que a preferência pelos vazios maiores, em geral, é recompensada por sua flexibilidade na acomodação das requisições por b_1 e b_2 *slots*.

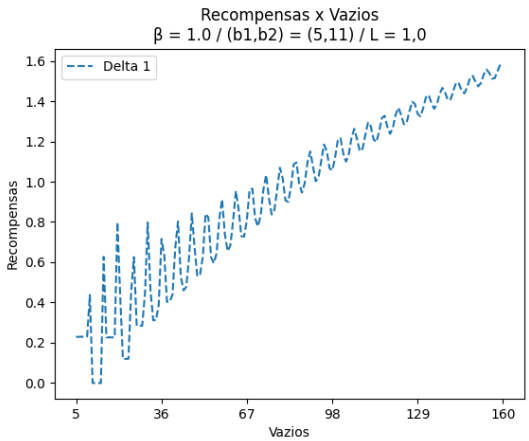


Fig. 5.26 - Recompensas D1 × vazios até $v = 160$.

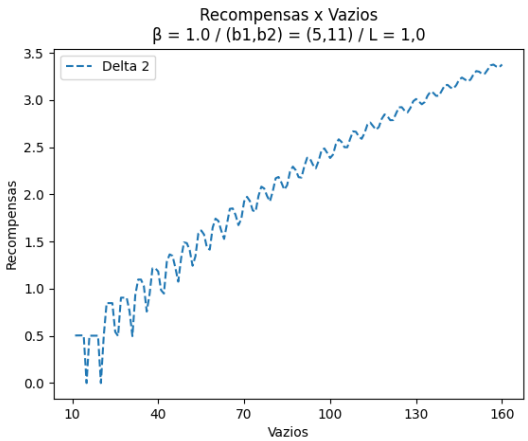


Fig. 5.27 - Recompensas D2 × vazios até $v = 160$.

As Figuras 5.28 e 5.29 apresentam o resultado completo da correspondência entre as recompensas (D1 e D2) e os tamanhos de vazios até $v = 320$, para o caso analisado no Quadro 5.17.

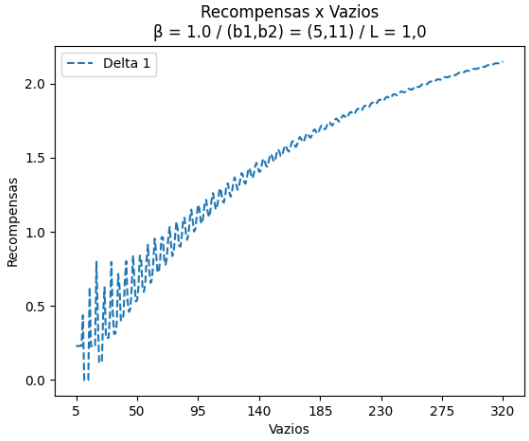


Fig. 5.28 - Recompensas D1 × vazios até $v = 320$.

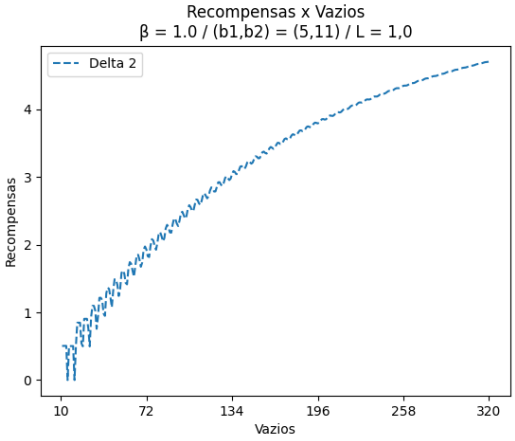


Fig. 5.29 - Recompensas D2 × vazios até $v = 320$.

Além da elevação no nível médio das recompensas quanto maior o vazio escolhido, observa-se que, com a exceção de alguns tamanhos de vazios, em geral a recompensa para a alocação do tamanho maior (b_2) é mais elevada, para um mesmo tamanho de vazio, do que para a alocação do tamanho menor (b_1).

O próximo quadro, 5.18, traz o cálculo da ociosidade mínima outra vez com tamanhos de vazios que variam de 0 até 50, com $\beta = 1{,}0$, $(b_1, b_2) = (5, 11)$ e $(\lambda_1, \lambda_2) = (38, 38)$, que geram uma carga normalizada de 1,9 (sobrecarga de 90%) para um espectro de 320 *slots*, simulando uma condição de sobrecarga do tráfego.

Neste terceiro cenário investigado, a política voraz $(1, 1)$ continua ótima para a maioria dos tamanhos de vazios, mas o número de exceções aumenta: a política $(0, 1)$ rejeita solicitações de 5 *slots* para $v = 11, 12, 13, 14, 22, 23$ e 24; e a política $(1, 0)$ rejeita solicitações de 11 *slots* para $v = 15, 20$ e 25, todos múltiplos de $b_1 = 5$. Com o tráfego operando na região de sobrecarga, as políticas não vorazes são ótimas na redução da ociosidade para um número ainda maior de vazios. Tal comportamento sinaliza que o conjunto de vazios no tráfego dinâmico para os quais a política não voraz é ótima tende a se aproximar do conjunto de vazios não vorazes do tráfego incremental.

De acordo com o exposto no Quadro 5.18, recalculado com $L = 1{,}9$, a maior recompensa para atribuições de 5 *slots* é atingida com o tamanho vazio 25, seguido por 36, 47, 41 e 20, nesta ordem. A maior recompensa para atribuições de 11 *slots* é alcançada novamente para o tamanho vazio 49, seguido por 50, 44, 45 e 48, nesta ordem. Essa sequência de vazios com a maior recompensa para atribuições de 11 *slots*, obtida com $L = 1{,}9$, é a mesma verificada com $L = 1{,}0$ para alocações de 11 *slots*, e se mantém a mesma em cargas altas e em toda a faixa de sobrecarga.

Quadro 5.18 - Ociosidade mínima para $L = 1{,}9$, $\beta = 1{,}0$ e $(b_1, b_2) = (5, 11)$.

	Política (0,1)		Política (1,0)		Política (1,1) (voraz)		Ociosidade Mínima e Política Ótima				Recompensas	
	0	1	1	0	1	1	Tx Chegada	Ociosidade	Políticas Selecionadas		D1	D2
v	$\lambda(v,P)$	$\omega(v,P)$	$\lambda(v,P)$	$\omega(v,P)$	$\lambda(v,P)$	$\omega(v,P)$	$\lambda(v,Q)$	$\omega(v,Q)$	q_1	q_2		
0	0	0	0	0	0	0	0	0	0	0		
1	0	0,5	0	0,5	0	0,5	0	0,5	0	0		
2	0	1	0	1	0	1	0	1	0	0		
3	0	1,5	0	1,5	0	1,5	0	1,5	0	0		
4	0	2	0	2	0	2	0	2	0	0		
5	0	2,5	38	0,1253125	38	0,1253125	38	0,1253125	1	0	0,1253125	
6	0	3	38	0,625375	38	0,625375	38	0,625375	1	0	0,125375	
7	0	3,5	38	1,1254375	38	1,1254375	38	1,1254375	1	0	0,1254375	
8	0	4	38	1,6255	38	1,6255	38	1,6255	1	0	0,1255	
9	0	4,5	38	2,1255625	38	2,1255625	38	2,1255625	1	0	0,1255625	
10	0	5	38	0,3696719	38	0,3696719	38	0,3696719	1	0	0,2443594	
11	38	0,2756875	38	0,8697938	76	0,4457882	38	0,2756875	0	1	-0,3496875	0,2756875
12	38	0,77575	38	1,3699156	76	0,9458271	38	0,77575	0	1	-0,3496875	0,27575
13	38	1,2758125	38	1,8700375	76	1,445866	38	1,2758125	0	1	-0,3496875	0,2758125
14	38	1,775875	38	2,3701594	76	1,9459049	38	1,775875	0	1	-0,3496875	0,275875
15	38	2,2759375	38	0,7271258	76	1,3468897	38	0,7271258	1	0	0,3574539	-1,2728742
16	38	0,5200469	38	0,6629031	76	0,400622	76	0,400622	1	1	0,1249345	0,2753095
17	38	1,0201688	38	1,163025	76	0,9006914	76	0,9006914	1	1	0,1249414	0,2753164
18	38	1,5202906	38	1,6631469	76	1,4007607	76	1,4007607	1	1	0,1249482	0,2753232
19	38	2,0204125	38	2,1632688	76	1,90083	76	1,90083	1	1	0,124955	0,27533
20	38	2,5205344	38	1,1920195	76	1,6463501	38	1,1920195	1	0	**0,4648937**	-0,933543
21	38	0,8775008	38	0,9069034	76	0,6446792	76	0,6446792	1	1	0,2440571	0,2750073
22	38	0,8132781	38	1,4070318	76	0,8553444	38	0,8132781	0	1	-0,0874132	0,5375906
23	38	1,3134	38	1,9071602	76	1,355417	38	1,3134	0	1	-0,0873607	0,53765
24	38	1,8135219	38	2,4072885	76	1,8554897	38	1,8135219	0	1	-0,0873081	0,5377094
25	38	2,3136438	38	1,758981	76	1,7666209	38	1,758981	1	0	**0,5669615**	-0,016894
26	38	1,3423945	38	1,2640702	76	1,0018677	76	1,0018677	1	1	0,3571885	0,2747419
27	38	1,0572784	38	1,4493017	76	0,9377687	76	0,9377687	1	1	0,1244906	0,5371466
28	38	1,5574068	38	1,94948	76	1,4378703	76	1,4378703	1	1	0,1244703	0,5371789
29	38	2,0575352	38	2,4496583	76	1,9379718	76	1,9379718	1	1	0,12445	0,5372112
30	38	2,5576635	38	2,422907	76	2,1678531	76	2,1678531	1	1	0,4088721	0,2670231
31	38	1,909356	38	1,7287118	76	1,466514	76	1,466514	1	1	0,4646463	0,2744946
32	38	1,4144452	38	1,6928803	76	1,1814623	76	1,1814623	1	1	0,2436936	0,5367831
33	38	1,5996767	38	2,1930393	76	1,5200684	76	1,5200684	1	1	0,0821981	0,7067903
34	38	2,099855	38	2,6931983	76	2,0201857	76	2,0201857	1	1	0,0822138	0,7067857
35	38	2,6000333	38	2,936648	76	2,3886572	76	2,3886572	1	1	0,2208041	0,5751353
36	38	2,573282	38	2,2954383	76	2,0332369	76	2,0332369	1	1	**0,5667229**	0,2742559
37	38	1,8790868	38	2,0497017	76	1,5383444	76	1,5383444	1	1	0,3568822	0,5364767
38	38	1,8432553	38	2,39644	76	1,6849076	76	1,6849076	1	1	0,1648392	0,7471389
39	38	2,3434143	38	2,8966139	76	2,1850227	76	2,1850227	1	1	0,164837	0,7471524
40	38	2,8435733	38	3,2717243	76	2,6210026	76	2,6210026	1	1	0,2323454	0,6830307
41	38	3,087023	38	2,9591376	76	2,5726715	76	2,5726715	1	1	**0,5394346**	0,4048184
42	38	2,4458133	38	2,5140522	76	2,002721	76	2,002721	1	1	0,4643765	0,5362069
43	38	2,2000767	38	2,6783497	76	1,9480811	76	1,9480811	1	1	0,2631735	0,7666188
44	38	2,546815	38	3,1785215	76	2,3695178	76	2,3695178	1	1	0,1844951	**0,8494494**
45	38	3,0469889	38	3,617765	76	2,8383941	76	2,8383941	1	1	0,2173915	**0,8182084**
46	38	3,4220993	38	3,5969129	76	3,0071889	76	3,0071889	1	1	0,4345174	0,6185317
47	38	3,1095126	38	3,0805224	76	2,5691962	76	2,5691962	1	1	**0,5664752**	0,5359592
48	38	2,6644272	38	3,0536771	76	2,3143042	76	2,3143042	1	1	0,3662231	**0,7759598**
49	38	2,8287247	38	3,4791044	76	2,603851	76	2,603851	1	1	0,2343332	**0,9189434**
50	38	3,3288965	38	3,9495994	76	3,0887527	76	3,0887527	1	1	0,2503586	**0,90373**

5.4.3 Simulações do Algoritmo OMa

Nesta seção, são apresentados os resultados obtidos com a simulação do segundo algoritmo proposto, denominado OMa (Ociosidade Mínima com alocação adjacente), que também foi desenvolvido em linguagem *Python*. Os parâmetros iniciais configurados na simulação do OMa seguem a mesma configuração listada na seção 5.3.3.

Outra vez, para cada valor de β, a simulação foi realizada para os mesmos quatro pares de $(b_1, b_2) = (5, 11)$, $(5, 14)$, $(7, 17)$ e $(13, 28)$ analisados no caso anterior. A nova simulação também compara a saída de quatro algoritmos em execução, todos submetidos à mesma sequência aleatória de solicitações para conexões de b_1 e b_2 *slots* com o intuito de fornecer uma comparação justa.

Nesta simulação, o algoritmo *exact-first-fit* (linha roxa) é substituído pelo algoritmo OMa (linha laranja), que é um algoritmo não voraz, ou seja, é seletivo de forma inteligente, e a comparação é realizada com outros três algoritmos vorazes: o *first-fit* (linha preta), o NA (linha verde), no qual a seleção do vazio é baseada na sua funcionalidade, e o DF (linha azul), novamente utilizado como caso de referência. Permanece a linha vermelha que indica a reta $S \times L$, com a linha tracejada vermelha sinalizando o nível máximo da ocupação espectral, $S = 320$ *slots*.

Assim como na simulação do NA, os novos resultados para o algoritmo OMa avaliam, em função da carga L, a ocupação espectral média, a vazão média e as probabilidades de bloqueio de banda e das requisições dos tipos b_1 e b_2. Além disso, na simulação do algoritmo OMa, outras três novas métricas foram avaliadas:

a) taxa de redução das perdas por fragmentação em relação às perdas por fragmentação do algoritmo FF;

b) taxa de redução das perdas totais em relação às perdas totais do FF;

c) taxa de recuperação da ocupação proporcionada pela substituição do algoritmo FF pelo OMa.

A primeira métrica é calculada conforme a equação (5.31). A taxa de redução das perdas totais (RPT), que incluem as perdas por insuficiência de recursos e as perdas por fragmentação, em relação às perdas totais proporcionadas pelo algoritmo FF, segue a equação (5.45):

$$RPT_A = \left[\frac{(B_{FF} - B_A)}{(SR_A)}\right] \times 100\% \tag{5.45}$$

em que SR_A é o número de *slots* requisitados na simulação para o algoritmo A, expressando assim a redução (em %) da probabilidade de bloqueio de banda gerada pela substituição do FF pelo algoritmo A. Cabe destacar que, no caso do OMa que é um algoritmo não voraz, B_{OMa} considera o total de *slots* perdidos, ou seja, a soma dos bloqueados, por inviabilidade da alocação, com os rejeitados, mesmo que a alocação imediata seja viável, mas não inteligente em termos da evolução da ociosidade esperada no futuro.

A taxa de recuperação da ocupação (ROP) é definida de acordo com a equação (5.46):

$$ROP_A = \left[\frac{(B_{FF} - B_A)}{(SR_{DF})}\right] \times 100\% \tag{5.46}$$

sendo SR_{DF} é o número de *slots* requisitados na simulação para o algoritmo DF, o que evidencia a redução (em %) da probabilidade de bloqueio de banda em relação ao seu valor máximo atingível com os recursos espectrais disponibilizados, ou seja, a redução das perdas causadas pela fragmentação.

As próximas seções exibem as curvas de cada variável analisada para o caso completo com $(b_1, b_2) = (5, 14)$ e $\beta = 2{,}0$ e as curvas da ocupação espectral média para os demais casos avaliados. O resultado completo de todas as simulações realizadas para o algoritmo OMa também está disponível em https://doi.org/10.25824/redu/HT2MVA.

5.4.3.1 Simulações do OMa com $\boldsymbol{\beta = 0,5}$

As Figuras 5.30 a 5.33 apresentam as curvas da ocupação espectral média em função da carga (L) com $\beta = 0{,}5$, respectivamente, para os pares $(b_1, b_2) = (5, 11), (5, 14), (7, 17)$ e $(13, 28)$.

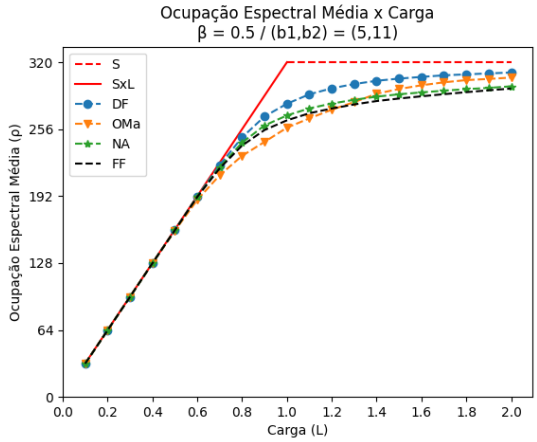


Fig. 5.30 - Ocupação espectral média (ρ) × carga (L) para $\beta = 0{,}5$ e $(b_1, b_2) = (5, 11)$.

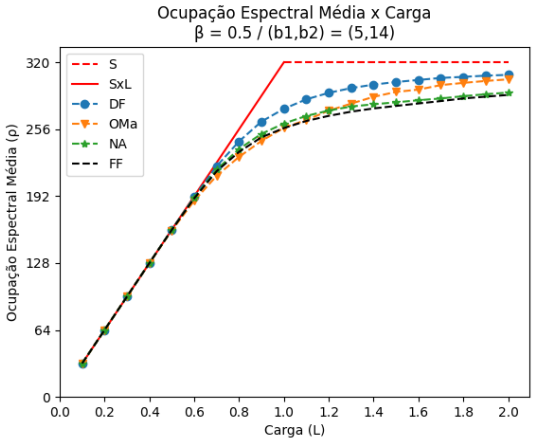


Fig. 5.31 - Ocupação espectral média (ρ) × carga (L) para $\beta = 0{,}5$ e $(b_1, b_2) = (5, 14)$.

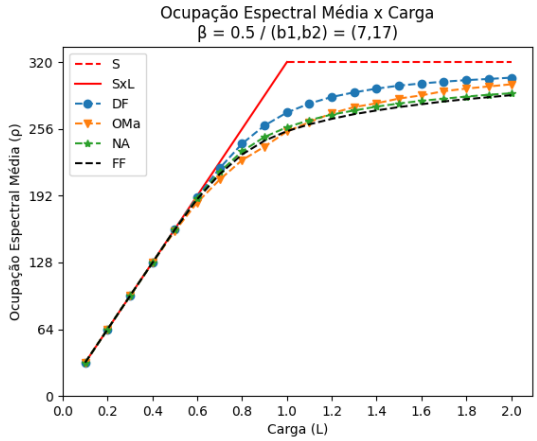


Fig. 5.32 - Ocupação espectral média (ρ) × carga (L) para $\beta = 0{,}5$ e $(b_1, b_2) = (7, 17)$.

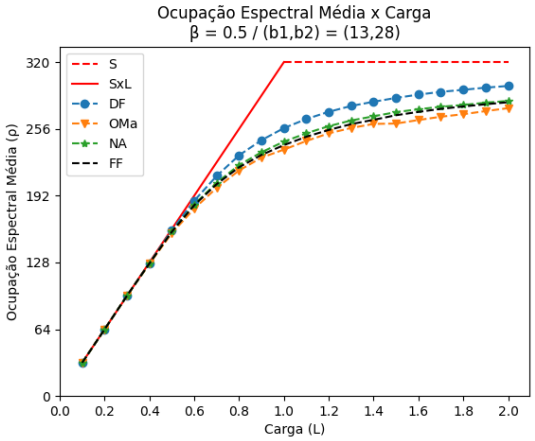


Fig. 5.33 - Ocupação espectral média (ρ) × carga (L) para $\beta = 0{,}5$ e $(b_1, b_2) = (13, 28)$.

A seguir, os Quadros 5.19 a 5.22 apresentam os resultados numéricos dos níveis de ocupação espectral atingidos pelos quatro algoritmos comparados com $\beta = 0{,}5$ para cada ponto de carga (L). Em destaque (amarelo), a faixa de carga na qual o algoritmo OMa supera os algoritmos NA e FF, permanecendo superior a ambos até a carga $L = 2{,}0$. O par $(b_1, b_2) = (13, 28)$ teve um desempenho inferior aos algoritmos NA e FF em toda a faixa de carga avaliada.

Quadro 5.19 - Níveis de ocupação espectral para $\beta = 0{,}5$ e $(b_1, b_2) = (5, 11)$.

(L)	DF	OMa	NA	FF
0,1	31,8574	31,8574	31,8574	31,8574
0,2	63,7162	63,7162	63,7162	63,7162
0,3	95,6795	95,6795	95,6795	95,6795
0,4	127,8158	127,7779	127,8156	127,8155
0,5	159,3934	158,8077	159,3789	159,3652
0,6	191,7861	188,3788	191,4908	191,2685
0,7	221,7037	211,9816	219,8163	218,6074
0,8	248,1344	230,5230	243,1259	240,3132
0,9	268,1357	243,8088	259,5453	255,1942
1,0	280,3090	257,4106	269,1020	264,4418
1,1	289,3192	266,3192	275,9205	271,2357
1,2	295,0655	274,6285	280,4802	275,7338
1,3	299,1466	281,8347	284,0761	279,6288
1,4	302,2572	**289,8797**	287,0848	282,8808
1,5	304,3441	**294,3992**	289,3467	285,3059
1,6	305,9632	**297,9465**	291,1373	287,4102
1,7	307,4928	**300,7831**	292,7843	289,4463
1,8	308,3889	**302,9155**	294,2958	291,3588
1,9	309,3107	**304,0137**	295,4445	293,1716
2,0	310,1475	**305,3194**	296,9613	294,8274

Quadro 5.20 - Níveis de ocupação espectral para $\beta = 0{,}5$ e $(b_1, b_2) = (5, 14)$.

(L)	DF	OMa	NA	FF
0,1	32,0952	32,0952	32,0952	32,0952
0,2	64,0307	64,0307	64,0307	64,0307
0,3	96,0046	96,0044	96,0046	96,0046
0,4	128,0431	127,9991	128,0420	128,0386
0,5	159,9553	159,1865	159,8770	159,7787
0,6	191,2869	187,6448	190,4850	189,9470
0,7	220,6502	211,4567	217,3520	215,8217
0,8	244,3932	229,5510	237,0952	234,3051
0,9	263,3056	244,7764	251,5445	248,2710
1,0	275,4817	257,2887	261,4350	257,0395
1,1	284,5249	264,7559	268,6660	263,8289
1,2	290,7387	273,7001	273,7693	268,7334
1,3	295,2622	**280,4330**	277,6616	272,7378
1,4	298,5995	**286,9174**	279,7755	275,8044
1,5	301,0930	**291,6010**	281,6581	278,2132
1,6	303,0453	**293,8133**	283,6535	280,5597
1,7	304,9798	**298,1162**	285,3438	282,8400
1,8	306,0015	**300,2315**	287,5471	285,3843
1,9	307,1500	**302,1058**	289,3933	287,0598
2,0	307,9228	**303,6106**	290,9712	288,7283

Quadro 5.21 - Níveis de ocupação espectral para $\beta = 0{,}5$ e $(b_1, b_2) = (7, 17)$.

(L)	DF	OMa	NA	FF
0,1	32,0780	32,0780	32,0780	32,0780
0,2	63,9759	63,9759	63,9759	63,9759
0,3	95,9647	95,9629	95,9647	95,9647
0,4	127,6821	127,5597	127,6683	127,6626
0,5	159,8465	158,0852	159,6151	159,5037
0,6	190,7620	185,4878	189,6248	188,7379
0,7	218,3298	207,3545	214,7406	212,6437
0,8	241,8105	226,0640	234,4881	231,4598
0,9	259,1865	238,9592	248,2178	244,6638
1,0	271,6165	253,5334	257,5676	253,7875
1,1	280,2703	262,3938	264,3231	260,3374
1,2	286,5600	**270,9201**	269,7051	265,5900
1,3	290,9759	**276,8561**	273,6098	270,0833
1,4	294,5361	**280,5046**	277,3081	273,6495
1,5	297,4432	**285,0481**	280,3242	276,8864
1,6	299,5192	**288,0932**	282,8597	279,5904
1,7	301,1970	**292,1063**	284,9015	282,2594
1,8	302,6861	**294,5850**	286,9411	284,4288
1,9	303,8167	**296,6353**	288,6786	286,3470
2,0	305,0727	**298,6866**	290,1138	288,2627

Quadro 5.22 - Níveis de ocupação espectral para $\beta = 0{,}5$ e $(b_1, b_2) = (13, 28)$.

(L)	DF	OMa	NA	FF
0,1	31,9259	31,9259	31,9259	31,9259
0,2	63,9139	63,9122	63,9131	63,9131
0,3	96,1855	96,1017	96,1608	96,1600
0,4	127,3855	126,6310	127,0913	127,0983
0,5	158,9879	156,0720	157,5904	157,4043
0,6	187,1487	179,6145	183,5358	182,8779
0,7	210,9019	199,4961	204,3407	203,0527
0,8	230,6825	216,1777	221,3396	219,2590
0,9	245,1680	228,4620	233,6115	231,0407
1,0	256,7329	236,1369	243,7668	240,4702
1,1	265,4417	244,4027	251,7635	248,4640
1,2	272,3568	251,7216	258,6010	254,9550
1,3	277,9541	256,8630	264,0200	260,5535
1,4	281,9918	260,7464	268,0369	264,6793
1,5	285,6605	261,0947	272,0237	269,0781
1,6	288,8275	264,4734	274,7762	272,2448
1,7	291,4486	267,6178	277,1659	275,1638
1,8	293,4275	270,0459	278,9010	277,4436
1,9	295,4990	273,0184	281,1067	279,6160
2,0	297,1076	275,8377	282,7686	281,5024

5.4.3.2 Simulações do OMa com $\boldsymbol{\beta = 1{,}0}$

A seguir, as Figuras 5.34 a 5.37 mostram os níveis da ocupação espectral média versus a carga (L) com $\beta = 1{,}0$, respectivamente, para os pares $(b_1, b_2) = (5, 11), (5, 14), (7, 17)$ e $(13, 28)$.

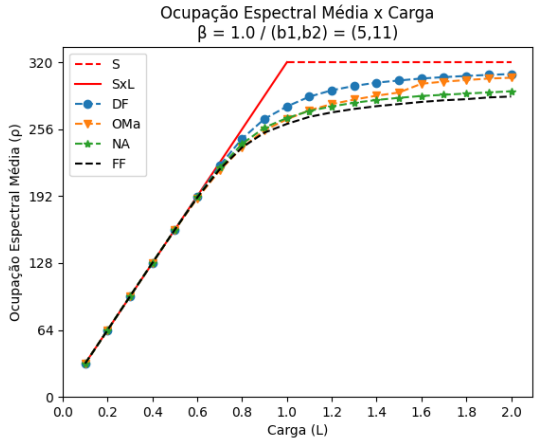


Fig. 5.34 - Ocupação espectral média (ρ) × carga (L) para $\beta = 1{,}0$ e $(b_1, b_2) = (5, 11)$.

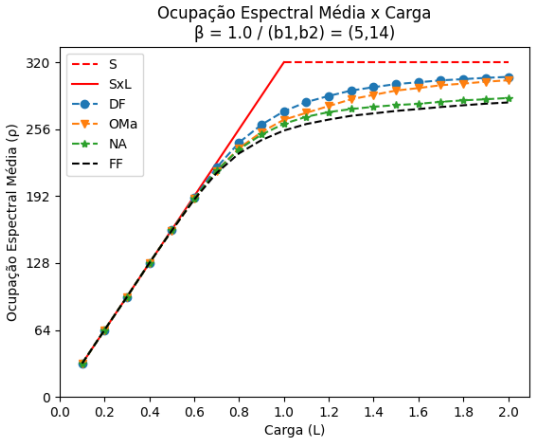


Fig. 5.35 - Ocupação espectral média (ρ) × carga (L) para $\beta = 1{,}0$ e $(b_1, b_2) = (5, 14)$.

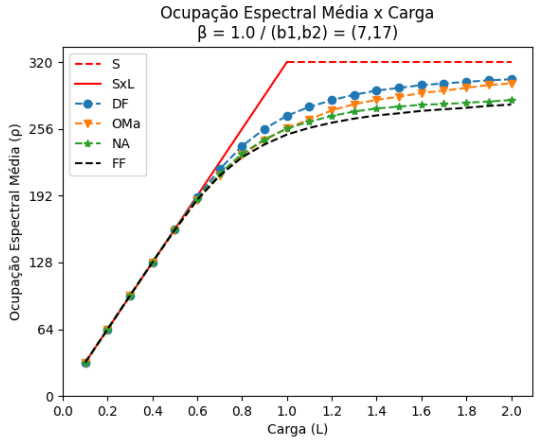


Fig. 5.36 - Ocupação espectral média (ρ) × carga (L) para $\beta = 1{,}0$ e $(b_1, b_2) = (7, 17)$.

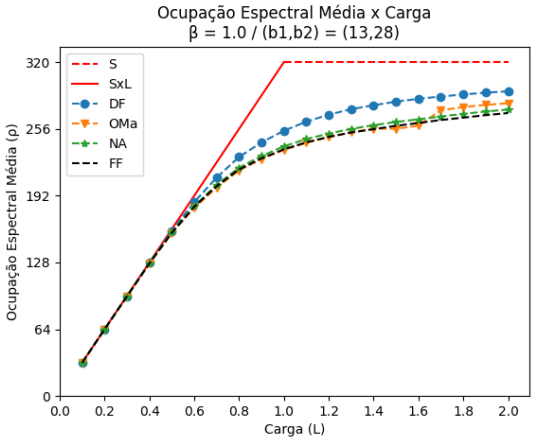


Fig. 5.37 - Ocupação espectral média (ρ) × carga (L) para $\beta = 1{,}0$ e $(b_1, b_2) = (13, 28)$.

Em seguida, os Quadros 5.23 a 5.26 listam os resultados numéricos dos níveis de ocupação espectral produzidos pelos quatro algoritmos comparados com $\beta = 1{,}0$ para cada ponto de carga (L). Novamente, a cor amarela destaca a faixa na qual o algoritmo OMa supera o NA e o FF. Com $\beta = 1{,}0$, a faixa amarela aumenta para todos os pares, inclusive para o par $(b_1, b_2) = (13, 28)$, que supera o NA e o FF para cargas muito elevadas, a partir de $L = 1{,}7$.

Quadro 5.23 - Níveis de ocupação espectral para $\beta = 1{,}0$ e $(b_1, b_2) = (5, 11)$.

(L)	DF	OMa	NA	FF
0,1	31,9452	31,9452	31,9452	31,9452
0,2	63,9136	63,9136	63,9136	63,9136
0,3	96,1811	96,1811	96,1811	96,1811
0,4	128,2196	128,2050	128,2178	128,2179
0,5	159,9611	159,7142	159,9198	159,8964
0,6	191,3882	189,8367	191,0175	190,6723
0,7	221,2852	216,5143	219,1416	217,6588
0,8	246,9201	238,3775	241,9072	238,5579
0,9	265,8390	254,9689	257,5494	252,7089
1,0	277,7050	265,4121	266,6221	261,0342
1,1	287,0526	**274,0680**	273,3850	267,7745
1,2	293,1841	**279,9785**	277,7755	272,0322
1,3	297,3307	**284,3059**	281,2211	275,3433
1,4	300,4832	**287,9945**	283,9027	277,9768
1,5	302,5464	**291,0648**	285,9630	279,9902
1,6	304,4255	**299,4256**	287,7143	282,0301
1,7	305,7649	**301,4444**	288,8468	283,6154
1,8	306,8908	**302,9335**	290,0788	284,6662
1,9	307,9081	**304,3907**	291,0982	286,3473
2,0	308,6941	**305,1477**	292,2253	287,2104

Quadro 5.24 - Níveis de ocupação espectral para $\beta = 1{,}0$ e $(b_1, b_2) = (5, 14)$.

(L)	DF	OMa	NA	FF
0,1	32,0458	32,0458	32,0458	32,0458
0,2	64,0511	64,0511	64,0511	64,0511
0,3	95,7405	95,7397	95,7405	95,7405
0,4	128,1384	128,1229	128,1315	128,1290
0,5	159,3410	159,1228	159,2385	159,1390
0,6	190,4343	189,2272	189,5414	188,7852
0,7	219,4346	216,1547	216,3059	214,1844
0,8	243,2439	**237,8723**	236,8209	232,9368
0,9	260,2682	**253,1729**	250,7600	245,5405
1,0	273,1144	**265,0433**	261,1266	254,7098
1,1	281,8840	**271,6278**	267,8680	260,9132
1,2	287,7741	**278,0961**	272,3208	265,1561
1,3	292,8822	**284,5928**	275,2081	268,7760
1,4	296,1488	**288,7655**	277,3622	271,0671
1,5	299,0872	**292,8952**	279,0201	273,4297
1,6	300,8567	**295,2736**	280,2038	275,2220
1,7	302,6584	**298,0119**	282,1162	277,1744
1,8	303,8447	**299,5275**	283,5708	278,8170
1,9	305,0796	**301,3695**	284,5760	280,3354
2,0	305,9797	**302,7591**	285,8129	281,5003

Quadro 5.25 - Níveis de ocupação espectral para $\beta = 1{,}0$ e $(b_1, b_2) = (7, 17)$.

(L)	DF	OMa	NA	FF
0,1	32,0569	32,0569	32,0569	32,0569
0,2	63,9755	63,9755	63,9755	63,9755
0,3	96,2362	96,2356	96,2362	96,2357
0,4	128,2159	128,1538	128,1924	128,1798
0,5	159,7655	158,8894	159,4413	159,2641
0,6	190,5414	187,7151	188,9409	188,1412
0,7	217,2936	211,4046	213,0711	211,1811
0,8	239,5275	230,9184	232,1755	228,8243
0,9	256,0594	245,1111	245,9081	241,0841
1,0	268,6221	**256,6277**	256,1787	250,5660
1,1	277,0288	**265,0259**	262,9914	256,9730
1,2	283,4937	**273,6858**	268,4100	261,8932
1,3	288,6913	**279,6732**	272,5649	265,8481
1,4	292,6502	**283,7705**	275,5171	268,8741
1,5	295,2772	**286,7938**	277,3243	270,9178
1,6	297,8799	**290,4004**	279,1106	273,2416
1,7	299,5063	**292,6363**	279,7592	274,7573
1,8	301,0216	**295,2656**	281,0107	276,1532
1,9	302,4609	**297,9477**	282,1895	278,0058
2,0	303,5324	**299,5992**	283,5148	279,2664

Quadro 5.26 - Níveis de ocupação espectral para $\beta = 1{,}0$ e $(b_1, b_2) = (13, 28)$.

(L)	DF	OMa	NA	FF
0,1	31,9900	31,9900	31,9900	31,9900
0,2	63,9795	63,9782	63,9787	63,9787
0,3	95,6918	95,6170	95,6568	95,6513
0,4	127,5189	126,8965	127,1894	127,1418
0,5	158,2246	156,0423	156,6803	156,4863
0,6	185,8974	180,1050	182,0696	181,4119
0,7	209,3829	199,8281	202,4065	201,2188
0,8	229,0799	216,1197	218,9002	217,1022
0,9	243,0259	227,0151	229,9175	227,7157
1,0	254,2198	235,8178	239,2938	236,4387
1,1	262,8931	242,6259	246,2125	243,0775
1,2	269,5935	248,0271	251,5044	248,3362
1,3	274,8580	252,8050	255,9807	252,4708
1,4	278,6554	256,1456	259,4929	255,6941
1,5	282,0756	256,0607	262,6159	258,9413
1,6	284,7726	259,0876	265,1420	261,5900
1,7	286,9824	**273,7703**	267,8310	264,5008
1,8	288,9693	**276,8711**	270,3561	266,7282
1,9	290,6446	**278,9720**	272,6282	269,1421
2,0	291,9803	**280,4849**	274,4667	271,1652

5.4.3.3 Simulações do OMa com $\boldsymbol{\beta = 2,0}$

As Figuras 5.38 a 5.41 exibem as curvas da ocupação espectral média em função da carga (L) para os quatros pares (b_1, b_2) analisados com $\beta = 2{,}0$.

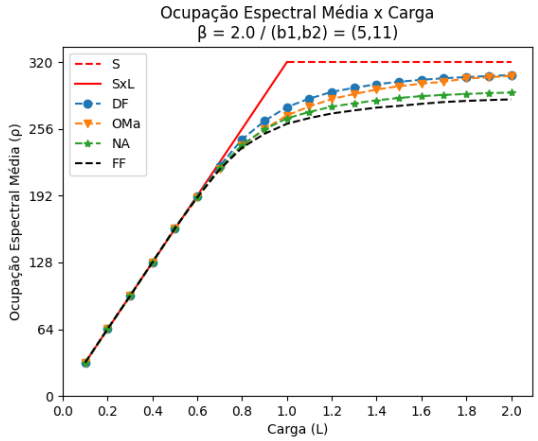


Fig. 5.38 - Ocupação espectral média (ρ) × carga (L) para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 11)$.

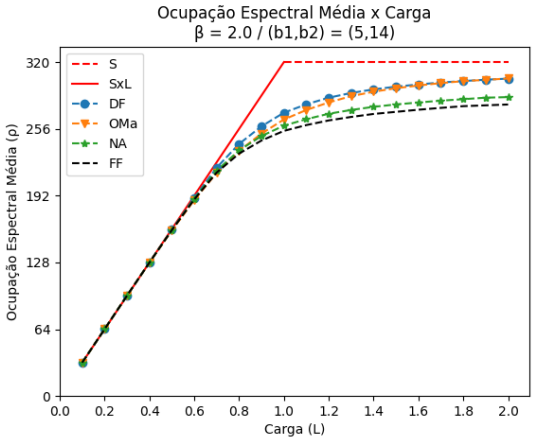


Fig. 5.39 - Ocupação espectral média (ρ) × carga (L) para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$.

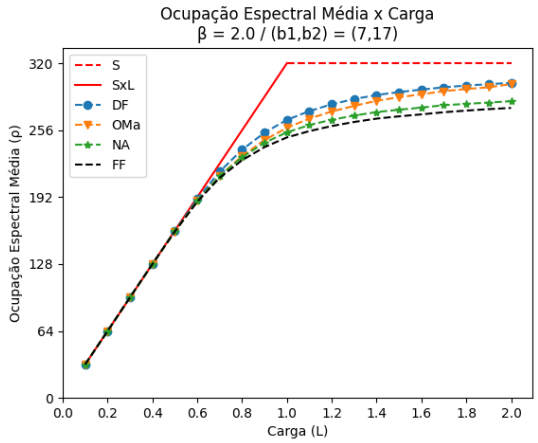


Fig. 5.40 - Ocupação espectral média (ρ) × carga (L) para $\beta = 2{,}0$ e $(b_1, b_2) = (7, 17)$.

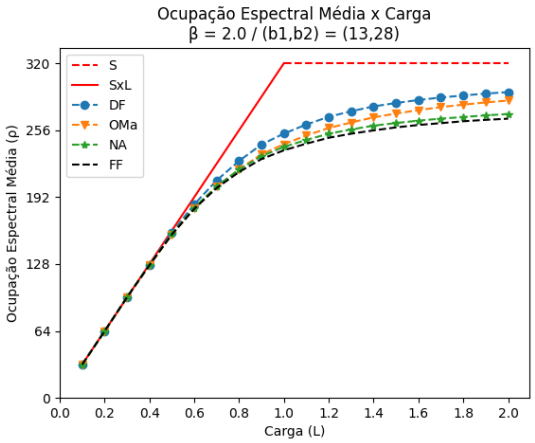


Fig. 5.41 - Ocupação espectral média (ρ) × carga (L) para $\beta = 2{,}0$ e $(b_1, b_2) = (13, 28)$.

Os Quadros 5.27 a 5.30, apresentados a seguir, especificam os valores numéricos dos níveis de ocupação espectral atingidos pelos quatro algoritmos comparados com $\beta = 2{,}0$ para cada ponto de carga (L). Novamente, com o aumento de β, a faixa amarela na qual o OMa supera os outros dois algoritmos também é estendida.

Quadro 5.27 - Níveis de ocupação espectral para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 11)$.

(L)	DF	OMa	NA	FF
0,1	32,0281	32,0281	32,0281	32,0281
0,2	64,2092	64,2092	64,2092	64,2092
0,3	95,8385	95,8381	95,8385	95,8385
0,4	128,1610	128,1483	128,1607	128,1608
0,5	160,6914	160,5117	160,6599	160,6087
0,6	191,2024	190,1550	190,7392	190,4318
0,7	220,5108	217,3529	218,4423	217,1361
0,8	245,9924	239,9700	240,7010	238,0409
0,9	263,5411	**255,7828**	255,4461	251,2644
1,0	276,9313	**268,8012**	266,0897	260,8755
1,1	284,9360	**277,5474**	272,2136	266,3482
1,2	291,4158	**284,5856**	277,2729	270,7348
1,3	295,3464	**289,3603**	280,4874	273,7059
1,4	298,7499	**293,7119**	283,3145	276,3846
1,5	301,1045	**296,8315**	285,5971	278,0229
1,6	303,0264	**299,2718**	287,3505	279,9943
1,7	304,5246	**301,2172**	288,5201	281,6975
1,8	305,5887	**304,2661**	289,4137	282,7776
1,9	306,6182	**305,4190**	290,2996	283,5869
2,0	307,5229	**306,7818**	290,7214	284,2416

Quadro 5.28 - Níveis de ocupação espectral para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$.

(L)	DF	OMa	NA	FF
0,1	32,0503	32,0503	32,0503	32,0503
0,2	64,0903	64,0903	64,0903	64,0903
0,3	96,2029	96,2011	96,2029	96,2029
0,4	127,9203	127,8143	127,9130	127,9013
0,5	159,2512	158,5578	159,1260	159,0114
0,6	189,9024	187,6946	189,0627	188,4301
0,7	218,8711	213,8594	215,9649	214,1689
0,8	241,5635	234,8795	235,3820	232,3100
0,9	258,4640	**251,4700**	249,1286	244,8464
1,0	271,4875	**264,9883**	259,0865	254,0664
1,1	279,5226	**274,0847**	265,2624	259,6220
1,2	285,8911	**281,3264**	270,4838	264,0806
1,3	290,3948	**287,3384**	274,0969	267,4257
1,4	293,8155	**291,5932**	277,0934	270,2822
1,5	296,4514	**294,9651**	279,3929	272,3054
1,6	298,5489	**297,5467**	281,2761	274,3340
1,7	300,4314	**299,8545**	282,9983	276,0969
1,8	301,8052	**301,7213**	284,4254	277,7519
1,9	303,0969	**302,9679**	285,8713	278,5462
2,0	304,1445	**304,2468**	286,4262	279,3472

Quadro 5.29 - Níveis de ocupação espectral para $\beta = 2{,}0$ e $(b_1, b_2) = (7, 17)$.

(L)	DF	OMa	NA	FF
0,1	32,0509	32,0509	32,0509	32,0509
0,2	63,9417	63,9417	63,9417	63,9417
0,3	96,1730	96,1629	96,1719	96,1710
0,4	128,0849	127,9669	128,0437	128,0386
0,5	159,6685	159,1461	159,2664	159,1370
0,6	190,1013	188,2599	188,3344	187,6155
0,7	216,3982	**212,3025**	212,1951	210,5412
0,8	237,3029	**231,2139**	230,1070	227,2797
0,9	253,6268	**246,5668**	244,0087	239,8000
1,0	265,8310	**258,2711**	253,9646	249,0716
1,1	274,2346	**267,0364**	260,8784	255,0899
1,2	280,7579	**273,8900**	265,9692	259,9505
1,3	285,7697	**279,1837**	269,9304	263,8403
1,4	289,6108	**284,0476**	273,2971	267,0394
1,5	292,4144	**287,5263**	275,5150	269,2145
1,6	294,7975	**290,2663**	277,5779	271,1079
1,7	296,9343	**293,2808**	279,8807	273,1328
1,8	298,6852	**295,3141**	281,1121	274,7145
1,9	299,9448	**296,9624**	282,3712	276,1171
2,0	301,2373	**300,0278**	283,6910	277,2508

Quadro 5.30 - Níveis de ocupação espectral para $\beta = 2{,}0$ e $(b_1, b_2) = (13, 28)$.

(L)	DF	OMa	NA	FF
0,1	31,9508	31,9508	31,9508	31,9508
0,2	63,7685	63,7677	63,7681	63,7676
0,3	96,2993	96,2608	96,2621	96,2530
0,4	127,3613	126,7386	127,0196	126,9440
0,5	157,9368	156,0804	156,6144	156,2275
0,6	185,1016	181,2224	181,8306	180,8431
0,7	207,9553	201,8325	202,3952	200,7709
0,8	226,6567	**218,5391**	218,3601	216,0243
0,9	242,2785	**232,9130**	231,1427	228,4085
1,0	252,7339	**242,5674**	239,9994	236,7569
1,1	261,4156	**250,9314**	246,7272	243,1336
1,2	268,7802	**258,0270**	252,6010	248,7286
1,3	274,0775	**263,3239**	256,7206	252,6234
1,4	278,7265	**268,2255**	260,2766	255,7800
1,5	281,9463	**271,8935**	262,8014	258,6519
1,6	284,8152	**275,1535**	265,0921	260,9494
1,7	287,2473	**278,0450**	267,0108	262,6973
1,8	289,1502	**280,3637**	268,5743	264,5053
1,9	290,9426	**282,5064**	270,1208	265,8536
2,0	292,3923	**284,3131**	271,4218	266,9748

5.4.3.4 Caso Completo: Simulação do OMa com $(\boldsymbol{b_1, b_2}) = (\mathbf{5, 14})$ e $\boldsymbol{\beta = 2{,}0}$

O caso completo investigado nesta seção, novamente, apresenta para $(b_1, b_2) = (5, 14)$ e $\beta = 2{,}0$, além da ocupação espectral média, as curvas da vazão média, da probabilidade de bloqueio

de banda e das probabilidades de bloqueio para requisições de b_1 e b_2 *slots*, respectivamente, por meio das Figuras 5.42 a 5.45.

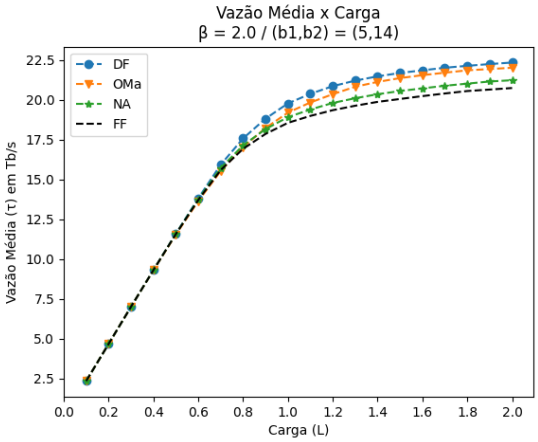


Fig. 5.42 - Vazão média (τ) em Tb/s × carga (L) para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$.

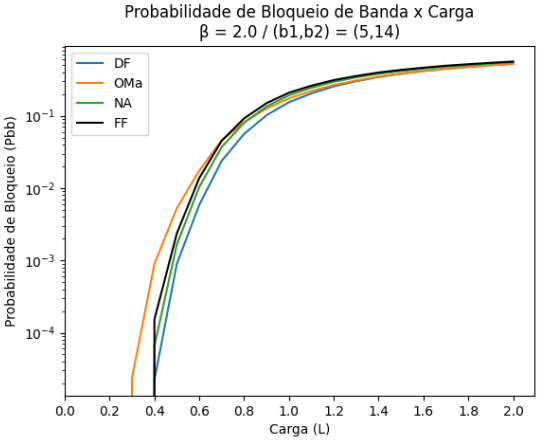


Fig. 5.43 - Probabilidade de bloqueio de banda × carga (L) para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$.

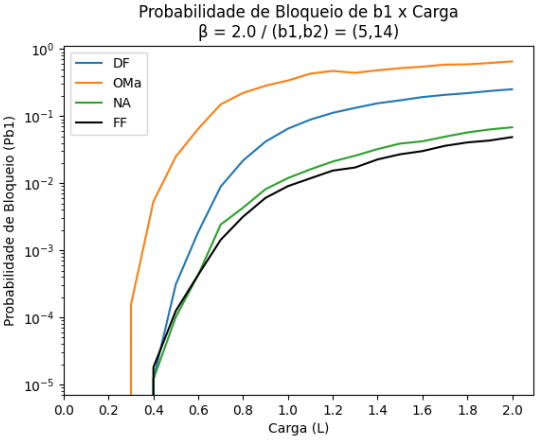


Fig. 5.44 - Probabilidade de bloqueio de b_1 × carga (L) para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$.

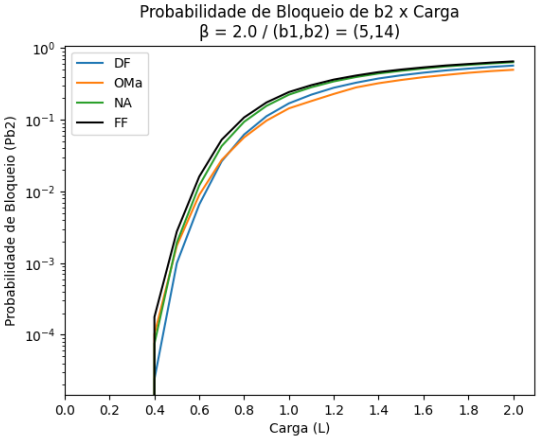


Fig. 5.45 - Probabilidade de bloqueio de b_2 × carga (L) para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$.

O algoritmo OMa apresentou menor probabilidade de bloqueio de banda a partir de $L = 0{,}9$, em comparação aos algoritmos NA e FF na simulação com $\beta = 2{,}0$, e menor probabilidade de bloqueio para requisições de b_2 *slots*, superando inclusive o algoritmo DF a partir de $L = 0{,}8$. Para requisições de b_1 *slots*, o OMa tem a maior probabilidade de bloqueio em comparação aos demais algoritmos, justamente por ser um algoritmo não voraz.

Os Quadros 5.31 e 5.32 apresentam os resultados numéricos alcançados pelos quatro algoritmos comparados, considerando os níveis de ocupação (em %) e a vazão média. Novamente, a faixa de carga (L) na qual o algoritmo OMa supera os algoritmos NA e FF é destacada em amarelo, e assim como acontece na ocupação espectral, o OMa permanece superior a ambos em todos os pontos até a carga $L = 2{,}0$ também para a vazão média.

Quadro 5.31 - Caso OMa: níveis de ocupação em % para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$.

(L)	DF	OMa	NA	FF
0,1	10,02%	10,02%	10,02%	10,02%
0,2	20,03%	20,03%	20,03%	20,03%
0,3	30,06%	30,06%	30,06%	30,06%
0,4	39,98%	39,94%	39,97%	39,97%
0,5	49,77%	49,55%	49,73%	49,69%
0,6	59,34%	58,65%	59,08%	58,88%
0,7	68,40%	66,83%	67,49%	66,93%
0,8	75,49%	73,40%	73,56%	72,60%
0,9	80,77%	**78,58%**	77,85%	76,51%
1,0	84,84%	**82,81%**	80,96%	79,40%
1,1	87,35%	**85,65%**	82,89%	81,13%
1,2	89,34%	**87,91%**	84,53%	82,53%
1,3	90,75%	**89,79%**	85,66%	83,57%
1,4	91,82%	**91,12%**	86,59%	84,46%
1,5	92,64%	**92,18%**	87,31%	85,10%
1,6	93,30%	**92,98%**	87,90%	85,73%
1,7	93,88%	**93,70%**	88,44%	86,28%
1,8	94,31%	**94,29%**	88,88%	86,80%
1,9	94,72%	**94,68%**	89,33%	87,05%
2,0	95,05%	**95,08%**	89,51%	87,30%

Quadro 5.32 - Caso OMa: níveis de vazão média para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$.

(L)	DF	OMa	NA	FF
0,1	2,3307	2,3307	2,3307	2,3307
0,2	4,6605	4,6605	4,6605	4,6605
0,3	6,9966	6,9964	6,9966	6,9966
0,4	9,3037	9,2952	9,3031	9,3023
0,5	11,5831	11,5285	11,5742	11,5660
0,6	13,8129	13,6396	13,7533	13,7081
0,7	15,9225	15,5236	15,7168	15,5889
0,8	17,5790	17,0347	17,1433	16,9242
0,9	18,8198	**18,2298**	18,1652	17,8603
1,0	19,7788	**19,2007**	18,9148	18,5577
1,1	20,3802	**19,8370**	19,3942	18,9932
1,2	20,8629	**20,3574**	19,8074	19,3528
1,3	21,2103	**20,8251**	20,1040	19,6322
1,4	21,4789	**21,1299**	20,3556	19,8744
1,5	21,6920	**21,3695**	20,5558	20,0569
1,6	21,8589	**21,5537**	20,7254	20,2375
1,7	22,0196	**21,7093**	20,8865	20,4030
1,8	22,1405	**21,8576**	21,0207	20,5564
1,9	22,2514	**21,9384**	21,1566	20,6490
2,0	22,3480	**22,0186**	21,2354	20,7462

A seguir, as Figuras 5.46 a 5.48 mostram, respectivamente, as curvas para as taxas de redução das perdas por fragmentação em relação às perdas por fragmentação do algoritmo FF, de redução das perdas totais em relação às perdas totais do FF e de recuperação da ocupação proporcionada pela substituição do algoritmo FF pelo OMa, comparado com o NA.

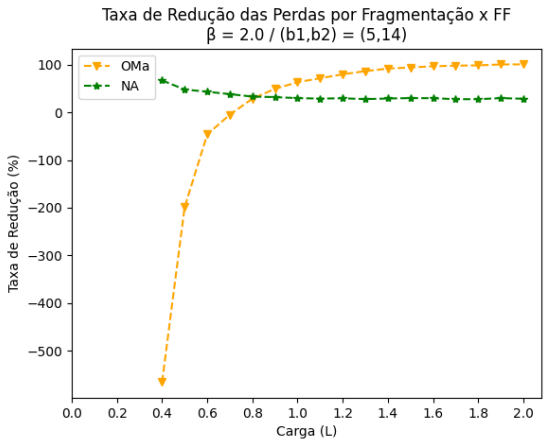


Fig. 5.46 - Caso OMa: redução das perdas por fragmentação × FF.

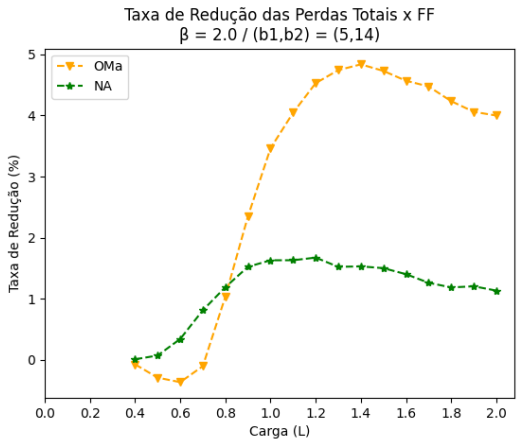


Fig. 5.47 - Caso OMa: redução das perdas totais × FF.

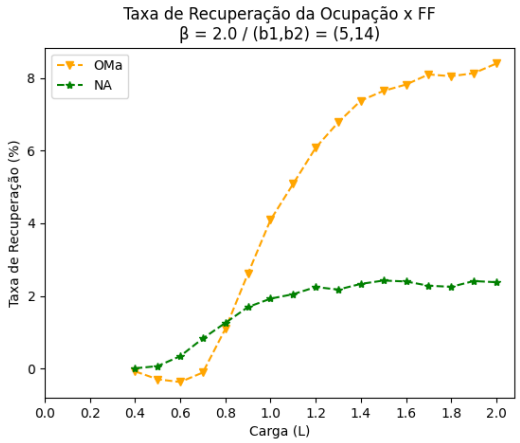


Fig. 5.48 - Caso OMa: recuperação da ocupação × FF.

5.3.3.5 Discussão dos Resultados para a Simulação do OMa

Na redução das perdas por fragmentação em relação às perdas por fragmentação do FF, o algoritmo OMa alcança o desempenho do NA após $L = 0{,}8$ com a taxa de redução em cerca de 30% e, a partir desse ponto, eleva o seu desempenho chegando próximo de 100%.

Comportamento similar é observado nas outras duas métricas analisadas, nas quais o OMa novamente supera o NA após $L = 0{,}8$, aumentando a redução das perdas totais entre 2 a quase 5% na região de cargas elevadas (próximo de $L = 1{,}4$) e, na recuperação da ocupação espectral proporcionada pelo algoritmo em substituição ao FF, o OMa promoveu um incremento que varia, aproximadamente, de 2 a 8,4% na faixa de carga entre 0,8 e 2,0.

Todos os algoritmos simulados, inclusive o Desfragmentador, que não tem perdas por fragmentação, têm perdas causadas apenas pela insuficiência de recursos espectrais. A Figura 5.49 apresenta, para o caso completo avaliado na seção anterior com $(b_1, b_2) = (5, 14)$ e $\beta = 2{,}0$, as curvas da ocupação espectral média para os algoritmos DF e FF acompanhadas da reta dada pela função $S \times L$, que representa a ocupação gerada por um algoritmo hipotético com perda zero que, para isso, precisaria dispor de infinitos *slots*.

Sendo assim, há uma dificuldade de contar diretamente os bloqueios adicionais causados pela fragmentação, uma vez que não é possível identificar esses bloqueios dos causados pela insuficiência de recursos, pois os dois fatores atuam em conjunto, ou seja, a fragmentação atua como um agravamento complexo da insuficiência de recursos.

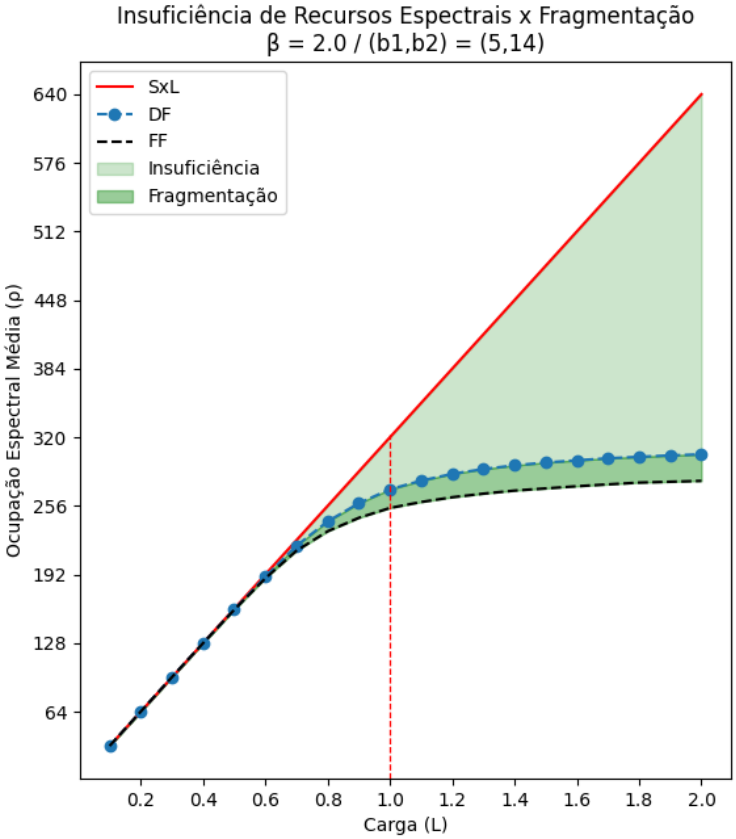


Fig. 5.49 - Insuficiência de recursos espectrais × Fragmentação.

Então, a contribuição da fragmentação para as perdas totais tem que ser obtida por meio da comparação com o algoritmo Desfragmentador, tal qual mostrado na Figura 5.49, em que a zona verde claro indica as perdas por insuficiência de recursos espectrais, e a zona verde escuro sinaliza as perdas por fragmentação adicionadas pelo algoritmo FF em relação ao DF.

De modo geral, o algoritmo OMa tende a superar os algoritmos NA e FF para cargas altas, em torno de $L = 0{,}8$ até $L = 2{,}0$, para os pares $(b_1, b_2) = (5, 11)$, $(5, 14)$ e $(7, 17)$. Isso ocorre por que, justamente para o tráfego com alta intensidade, os vazios são muito pequenos e por isso a alocação adjacente, usada por esta versão do algoritmo OMa, frequentemente é ótima ou até a única possível.

O desempenho do algoritmo OMa com o par $(b_1, b_2) = (13, 28)$, os maiores tamanhos de b_1 e b_2 analisados, teve um resultado aquém comparado com NA e FF em toda a faixa de carga para a simulação com $\beta = 0{,}5$. No caso de $\beta = 1{,}0$, o OMa começa a superar os seus concorrentes a partir de $L = 1{,}7$. Finalmente, com $\beta = 2{,}0$, o OMa ultrapassa o desempenho do NA e FF na faixa de carga a partir de $L = 0{,}8$. Essa performance reforça o bom desempenho do algoritmo OMa em cargas elevadas sob tráfego intenso.

Como $S = 320 = (208 + 112) = (16 \times 13) + (4 \times 28)$, portanto, o melhor casamento de 320 com o perfil do tráfego seria com $\beta = 4/16$ para o par $(b_1, b_2) = (13, 28)$. Nos outros casos, 320 é do sigma 3 para o par $(7, 17)$, do sigma 5 para o par $(5, 14)$ e do sigma 6 para o $(5, 11)$. Assim, os outros casos oferecem algumas (pelo menos três) soluções para o casamento com o perfil do tráfego, enquanto o par $(13, 28)$ só tem uma solução ótima para esse casamento, dada por $\beta = 4/16 = 0{,}25$, que é diferente dos três valores simulados para β.

No caso do par $(5{,}14)$, avaliado no caso completo, tem-se $S = 320 = (64 \times 5) = (50 \times 5) + (5 \times 14) = (36 \times 5) + (10 \times 14) = (22 \times 5) + (15 \times 14) = (8 \times 5) + (20 \times 14)$, de maneira que os valores ótimos de β são zero, $5/50 = 0{,}10$, $10/36 = 0{,}28$, $15/22 = 0{,}68$ e $20/8 = 2{,}5$, ou seja, o estado do sistema nunca fica muito descasado com o perfil do tráfego.

Assim, o algoritmo OMa procura valorizar o casamento dos vazios com o perfil do tráfego, de modo que deve preencher o espectro quando a carga cresce até o infinito desde que S não pertença ao sigma 0, o que não ocorre em nenhum caso em estudo.

Como 320 não é divisível por 7 e nem por 13, espera-se que a ocupação no algoritmo DF se aproxime de $312 = 24 \times 13$ no caso do par $(13, 28)$ ou a $315 = 45 \times 7$ no caso do par $(7, 17)$, ao passo que o algoritmo deve se aproximar de 320 nos outros dois casos quando L tender a infinito. Portanto, é esperado que o algoritmo OMa ultrapasse o DF em algum valor de L e conserve essa vantagem assintoticamente nesses casos. Tal cenário é investigado com mais profundidade no Capítulo 6, mas já é observado para $L = 2{,}0$ no Quadro 5.28, referente à $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$.

6 PERSPECTIVAS FUTURAS

Neste capítulo são tratadas as perspectivas futuras da aplicação de um algoritmo não voraz controlado por aprendizagem de máquina, tal qual a proposta do algoritmo OMa apresentada no Capítulo 5, quando a carga se aproxima da região de sobrecarga ou excede acima de 100%. Essa situação, em que o tráfego bloqueado na primeira fibra pode ser redirecionado por uma segunda fibra, é pouco discutida na literatura, porém se torna relevante com a chegada dos sistemas de multiplexação por divisão espacial (SDM, do inglês *Space Division Multiplexing*).

O crescimento exponencial dos serviços de dados nas últimas décadas levou a comunidade científica de pesquisa em fotônica a explorar uma variedade de novas fibras ópticas e tecnologias relacionadas para substituir as fibras monomodo padrão, que representam a base da maioria dos sistemas ópticos comerciais de alta capacidade. Grande parte desses esforços está voltada para os sistemas SDM, que abrangem uma variedade de tecnologias que dão suporte à transmissão de sinais de dados individuais por vários caminhos espaciais de um canal de fibra óptica comum. A motivação para a adoção de tais sistemas não é apenas multiplicar a capacidade de transporte de informações das fibras ópticas, mas também reduzir o consumo de energia e melhorar a eficiência por meio da integração, do hardware compartilhado e do processamento digital de sinais (DSP, do inglês *Digital Signal Processing*) [64].

Embora a ideia de utilizar a dimensão espacial dos cabos de fibra óptica para multiplexação não seja algo novo, a pesquisa sobre SDM se tornou mais difundida na última década, à medida que os sistemas de fibra monomodo (SMF, do inglês *Single Mode Fiber*) usando multiplexação por divisão de comprimento de onda (WDM, do inglês *Wavelength Division Multiplexing*) com detecção coerente começaram a aproximar-se de seus limites de capacidade teórica. Assim, melhorias nos processos de fabricação das fibras e na tecnologia de processamento de sinais alimentaram um rápido avanço nas pesquisas sobre sistemas SDM, levando à realização de novas fibras e técnicas de acoplamento [64].

Desse modo, nas próximas seções, são avaliados dois casos em que a carga (L) excede acima de 100% a carga total, com L normalizado variando até $4{,}0$ e $10{,}0$. Os casos analisados neste capítulo foram selecionados de acordo com os resultados apresentados no Capítulo 5, sendo: com $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$ aquele que teve o melhor desempenho, e com $\beta = 0{,}5$ e $(b_1, b_2) = (13, 28)$ o de pior desempenho. O objetivo é avaliar a aproximação/ultrapassagem do algoritmo OMa ao DF, o que pode indicar um novo cenário em

sistemas ópticos no qual seja possível se aproximar do esgotamento da primeira fibra, com o tráfego de transbordamento (*overflow*) sendo desviado para uma segunda fibra.

Na Seção 6.5 são discutidas algumas possibilidades de trabalhos futuros para a sequência desta pesquisa. Uma primeira sugestão, propõe estimar a probabilidade de bloqueio de um enlace óptico gerado pelo modelo de tráfego de Erlang quando comparado a um modelo mais realista proposto por Engset, quando ambos os modelos são calibrados para apresentarem a mesma taxa média de chegada de solicitações.

Uma segunda proposta discute uma atualização para o algoritmo OMa: a alocação adjacente, utilizada por essa versão, é ótima para vazios pequenos, mas pode não ser a melhor opção para vazios grandes. A alocação não adjacente em um vazio gera dois novos vazios e, por consequência, várias combinações possíveis para essa alocação, o que resulta em um problema de escalabilidade na análise matemática desse problema.

A terceira ideia levantada trata da possibilidade de implementar um algoritmo híbrido entre o NA e o OMa, reunindo, em uma única versão, as faixas de carga de melhor desempenho de cada algoritmo proposto neste trabalho. Por fim, o capítulo é encerrado com as considerações finais e apresenta a lista dos trabalhos publicados.

6.1 SIMULAÇÕES COM A CARGA ATÉ $L = 4{,}0$

Nesta seção, são apresentados os resultados alcançados com a simulação dos dois algoritmos propostos, NA e OMa, quando a carga varia até $L = 4{,}0$. A seguir, são listados os parâmetros iniciais configurados na simulação:

a) largura de banda $S = 320$ *slots*;
b) duas classes de tráfego caracterizadas por (b_1, b_2);
c) carga L normalizada variando de 0 até $4{,}0$ com incrementos iguais a $0{,}2$;
d) a chegada dos pedidos é poissoniana, sendo as taxas λ_1 e λ_2 expressas em erlangs;
e) fator de perfil do tráfego $\beta = \lambda_2/\lambda_1$, de modo que há:
    a. predominância de requisições do tipo 1, quando $\beta = 0{,}5$;
    b. taxa igualitária de chegada, quando $\beta = 1{,}0$;
    c. predominância de requisições do tipo 2, quando $\beta = 2{,}0$.
f) critério de parada da simulação: 500 mil chegadas.

A simulação foi realizada para dois pares de $(b_1, b_2) = (5, 14)$ e $(13, 28)$, respectivamente com $\beta = 2{,}0$ e $\beta = 0{,}5$, e compara a saída de quatro algoritmos, todos submetidos ao mesmo tráfego de entrada e sob as mesmas condições simuladas: DF (linha azul), FF (linha preta), NA (linha verde) e OMa (linha laranja).

Os resultados da simulação avaliam, em função da carga L, os seguintes parâmetros: ocupação espectral média, vazão média, probabilidades de bloqueio de banda e das requisições dos tipos b_1 e b_2. Além disso, são analisadas as taxas de redução das perdas por fragmentação em relação ao FF, de redução das perdas totais em relação ao FF e de recuperação da ocupação em relação ao FF. O resultado completo das duas simulações realizadas para $L = 4{,}0$, incluindo o código fonte do programa, está disponível em https://doi.org/10.25824/redu/HT2MVA.

6.1.1 Caso $L = 4,0$: simulação do OMa com $\beta = 2,0$ e $(b_1, b_2) = (5, 14)$

As Figuras 6.1 a 6.4 expõem, respectivamente, as curvas da ocupação espectral média em função da carga (L), da probabilidade de bloqueio de banda e das probabilidades de bloqueio para requisições de b_1 e b_2 *slots*, para o par selecionado $(b_1, b_2) = (5, 14)$ analisado com $\beta = 2{,}0$.

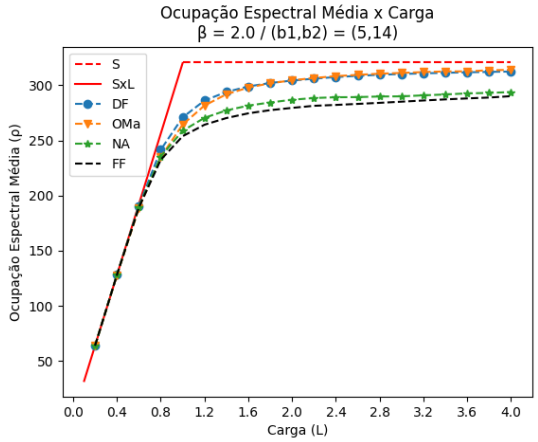


Fig. 6.1 - Caso $L = 4{,}0$: ocupação espectral média (ρ) × carga (L) para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$.

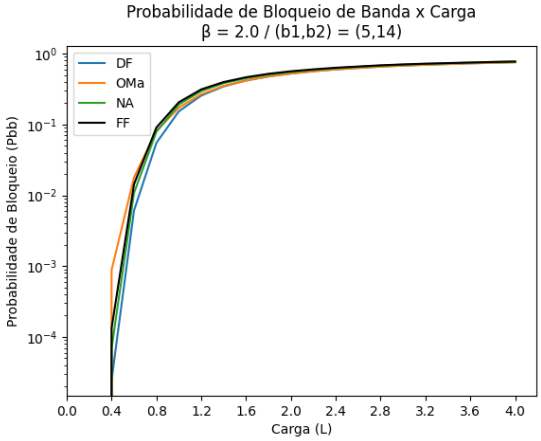


Fig. 6.2 - Caso $L = 4{,}0$: probabilidade de bloqueio de banda × carga (L) para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$.

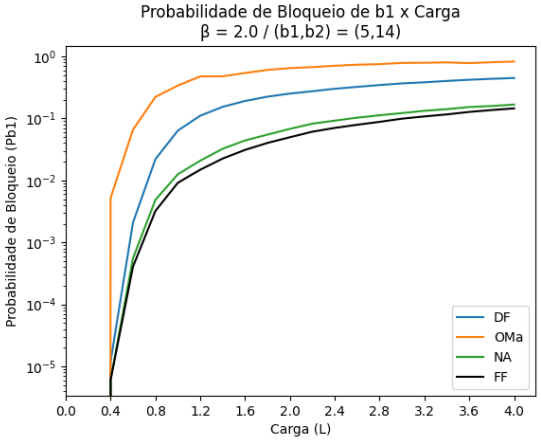


Fig. 6.3 - Caso $L = 4{,}0$: probabilidade de bloqueio de b_1 × carga (L) para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$.

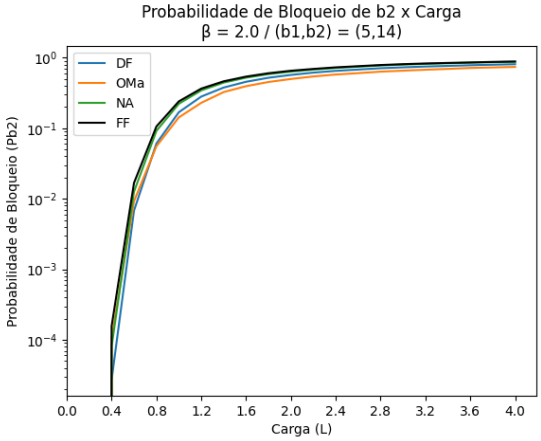


Fig. 6.4 - Caso $L = 4{,}0$: probabilidade de bloqueio de b_2 × carga (L) para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$.

Agora, o algoritmo OMa além de superar os algoritmos NA e FF a partir de $L = 1{,}0$, também apresentou menor probabilidade de bloqueio de banda em relação ao DF a partir de $L = 2{,}0$, o que confirma a expectativa evidenciada na análise do caso completo tratado no Capítulo 5 - Seção 5.4.3.4. Novamente, o algoritmo OMa apresenta menor probabilidade de bloqueio para requisições de b_2 *slots* em comparação ao NA e FF, e supera o algoritmo DF a partir de $L = 0{,}8$. Para requisições de b_1 *slots*, sendo um algoritmo não voraz, o OMa mantém a maior probabilidade de bloqueio em comparação aos demais algoritmos, visto que há predominância na chegada de solicitações por b_2 *slots*.

A seguir, as Figuras 6.5 a 6.8 apresentam, respectivamente, as curvas para a vazão média versus carga (L) e para as taxas de redução das perdas por fragmentação em relação às perdas por fragmentação do FF, de redução das perdas totais em relação às perdas totais do FF e de recuperação da ocupação proporcionada pela substituição do algoritmo FF pelo OMa, na comparação com o NA.

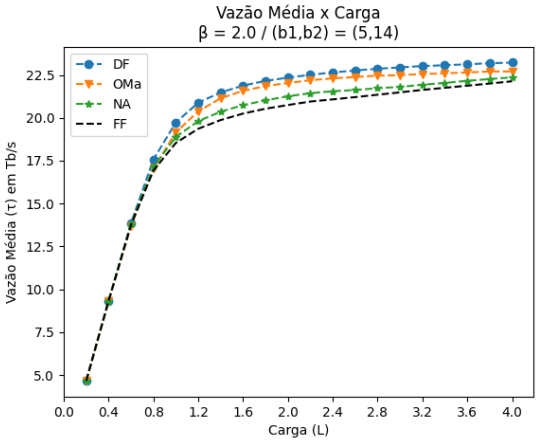


Fig. 6.5 - Caso $L = 4{,}0$: vazão média (τ) em Tb/s × carga (L) para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$.

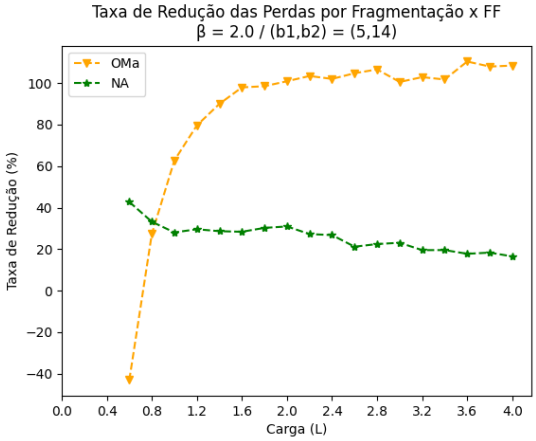


Fig. 6.6 - Caso $L = 4{,}0$: redução das perdas por fragmentação × FF para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$.

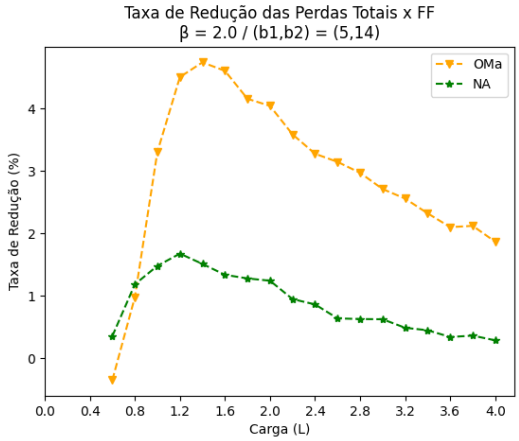


Fig. 6.7 - Caso $L = 4{,}0$: redução das perdas totais × FF para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$.

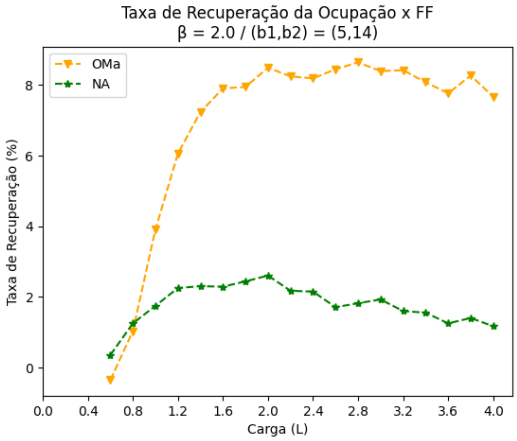


Fig. 6.8 - Caso $L = 4{,}0$: recuperação da ocupação × FF para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$.

Os Quadros 6.1 e 6.2, exibidos a seguir, especificam os valores numéricos dos níveis de ocupação espectral média e vazão média, respectivamente, obtidos pelos quatro algoritmos comparados com $\beta = 2{,}0$ e carga variando até $L = 4{,}0$. Em amarelo, é destacada a faixa de valores de carga na qual o algoritmo OMa supera os seus concorrentes (NA e FF) e, com a fonte em vermelho, a faixa na qual o OMa também supera nos níveis de ocupação o limite teórico (para os algoritmos vorazes) atingido pelo algoritmo DF, o que acontece a partir de $L = 2{,}0$.

Quadro 6.1 - Caso $L = 4{,}0$: níveis de ocupação espectral para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$.

(L)	DF	OMa	NA	FF
0,2	63,9459	63,9459	63,9459	63,9459
0,4	128,1242	128,0107	128,1182	128,1088
0,6	190,6059	188,3228	189,7349	189,0799
0,8	241,4381	234,8258	235,3245	232,3400
1,0	270,7871	**264,3151**	258,6087	253,7632
1,2	286,1047	**281,5613**	270,4333	264,1641
1,4	293,8598	**291,7547**	277,1349	270,2165
1,6	298,8056	**297,7995**	281,4766	274,4854
1,8	301,9872	**301,7281**	284,3269	277,3645
2,0	304,1616	**304,3954**	286,7883	279,4123
2,2	305,8697	**306,5164**	288,3535	281,2989
2,4	307,3291	**308,1780**	289,0461	282,0962
2,6	308,4126	**309,3321**	289,1891	283,0242
2,8	309,4199	**310,4274**	289,7371	283,9794
3,0	310,0800	**311,0768**	289,8667	285,0399
3,2	310,7200	**311,8716**	290,7694	286,1168
3,4	311,1894	**312,4402**	291,5850	287,1176
3,6	311,6417	**312,5151**	292,5531	287,9973
3,8	312,0220	**313,5304**	292,9882	288,7922
4,0	312,3310	**313,8559**	293,7569	290,0302

Quadro 6.2 - Caso $L = 4{,}0$: níveis de vazão média para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$.

(L)	DF	OMa	NA	FF
0,2	4,6508	4,6508	4,6508	4,6508
0,4	9,3168	9,3078	9,3163	9,3157
0,6	13,8626	13,6835	13,8008	13,7540
0,8	17,5707	17,0323	17,1398	16,9272
1,0	19,7312	**19,1548**	18,8825	18,5378
1,2	20,8824	**20,3746**	19,8078	19,3632
1,4	21,4823	**21,1430**	20,3585	19,8698
1,6	21,8788	**21,5747**	20,7389	20,2482
1,8	22,1524	**21,8477**	21,0182	20,5318
2,0	22,3444	**22,0331**	21,2569	20,7460
2,2	22,5102	**22,1923**	21,4351	20,9502
2,4	22,6476	**22,3043**	21,5509	21,0751
2,6	22,7583	**22,3794**	21,6214	21,2072
2,8	22,8634	**22,4605**	21,7276	21,3465
3,0	22,9382	**22,4846**	21,8010	21,4856
3,2	23,0200	**22,5529**	21,9281	21,6286
3,4	23,0644	**22,5947**	22,0330	21,7507
3,6	23,1231	**22,6462**	22,1599	21,8739
3,8	23,1833	**22,7015**	22,2615	21,9983
4,0	23,2246	**22,7025**	22,3609	22,1313

6.1.2 Caso $L = 4,0$: simulação do OMa com $\beta = 0,5$ e $(b_1, b_2) = (13, 28)$

As Figuras 6.9 a 6.12 mostram, respectivamente, as curvas da ocupação espectral média versus carga (L), da probabilidade de bloqueio de banda e das probabilidades de bloqueio para pedidos por b_1 e b_2 *slots*, do par selecionado $(b_1, b_2) = (13, 28)$ analisado com $\beta = 0{,}5$.

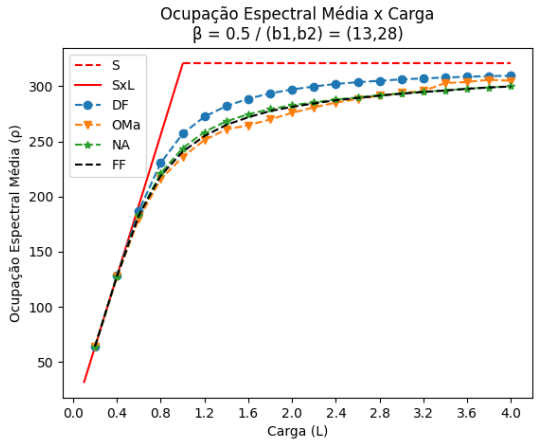


Fig. 6.9 - Caso $L = 4{,}0$: ocupação espectral média (ρ) × carga (L) para $\beta = 0{,}5$ e $(b_1, b_2) = (13, 28)$.

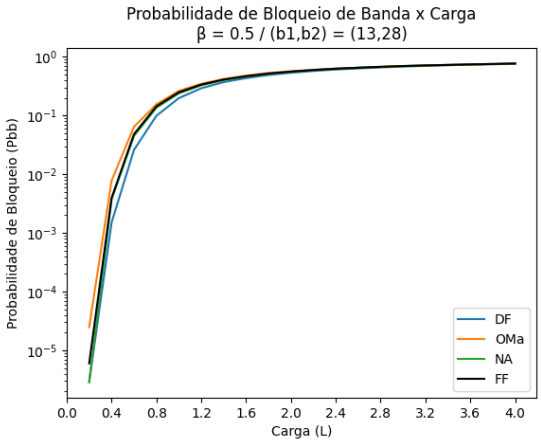


Fig. 6.10 - Caso $L = 4{,}0$: probabilidade de bloqueio de banda × carga (L) para $\beta = 0{,}5$ e $(b_1, b_2) = (13, 28)$.

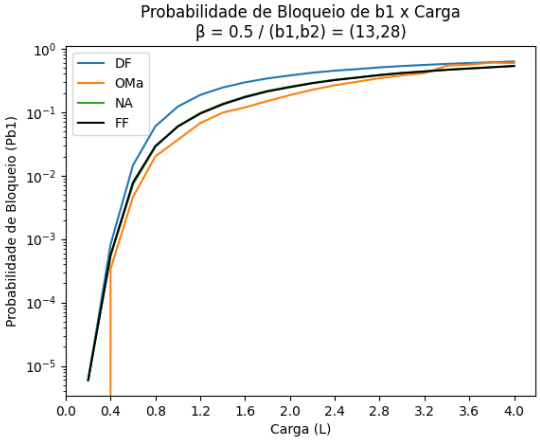


Fig. 6.11 - Caso $L = 4{,}0$: probabilidade de bloqueio de b_1 × carga (L) para $\beta = 0{,}5$ e $(b_1, b_2) = (13, 28)$.

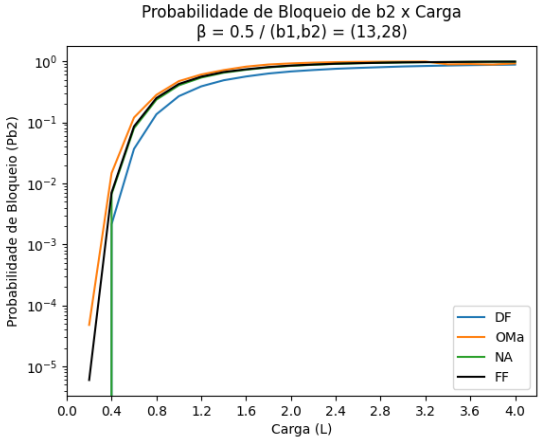


Fig. 6.12 - Caso $L = 4{,}0$: probabilidade de bloqueio de b_2 × carga (L) para $\beta = 0{,}5$ e $(b_1, b_2) = (13, 28)$.

Neste caso, o de pior desempenho entre os casos analisados no Capítulo 05, verifica-se uma tentativa de aproximação do algoritmo OMa em relação ao DF, o que ocorre a partir de $L = 3{,}4$. No bloqueio de banda, o algoritmo OMa fica abaixo do DF e consegue superar NA e FF a partir de $L = 2{,}8$, região na qual apresenta menor probabilidade de bloqueio.

Para requisições de b_2 *slots*, em comparação ao NA e FF, o algoritmo OMa apresenta menor probabilidade de bloqueio a partir de $L = 3{,}4$, o que está em conformidade com a curva da ocupação espectral. Para requisições de b_1 *slots*, o OMa tem a menor probabilidade de bloqueio em comparação aos demais algoritmos até $L = 3{,}4$, quando é superado pelo FF e NA.

A seguir, as Figuras 6.13 a 6.16 exibem, respectivamente, as curvas da vazão média em função da carga (L) e das taxas de redução das perdas por fragmentação em relação às perdas por fragmentação do algoritmo FF, de redução das perdas totais em relação às perdas totais do FF e de recuperação da ocupação proporcionada pela substituição do algoritmo FF pelo OMa, comparado com o NA.

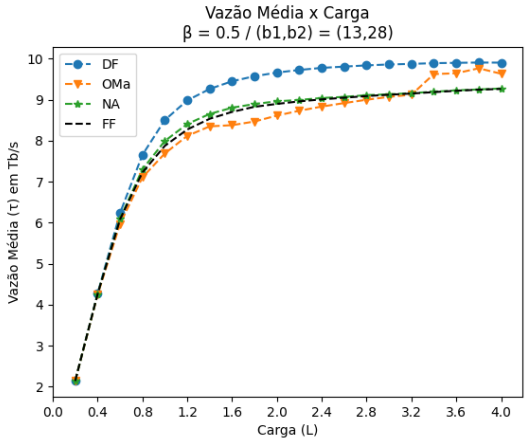


Fig. 6.13 - Caso $L = 4{,}0$: vazão média (τ) em Tb/s × carga (L) para $\beta = 0{,}5$ e $(b_1, b_2) = (13, 28)$.

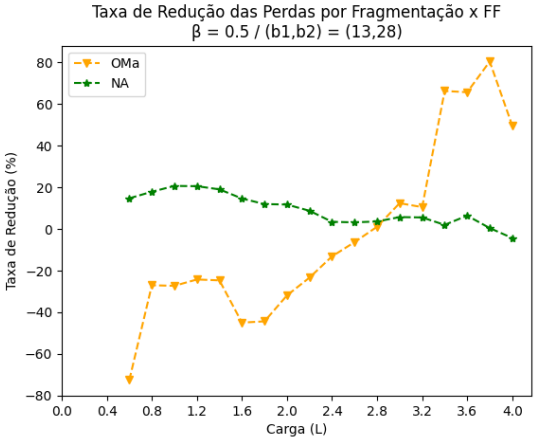


Fig. 6.14 - Caso $L = 4{,}0$: redução das perdas por fragmentação × FF para $\beta = 0{,}5$ e $(b_1, b_2) = (13, 28)$.

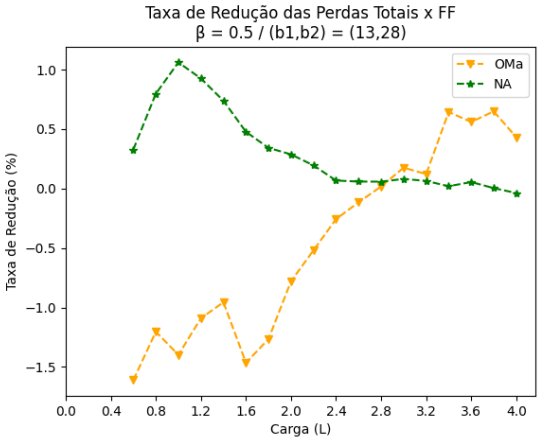


Fig. 6.15 - Caso $L = 4{,}0$: redução das perdas totais × FF para $\beta = 0{,}5$ e $(b_1, b_2) = (13, 28)$.

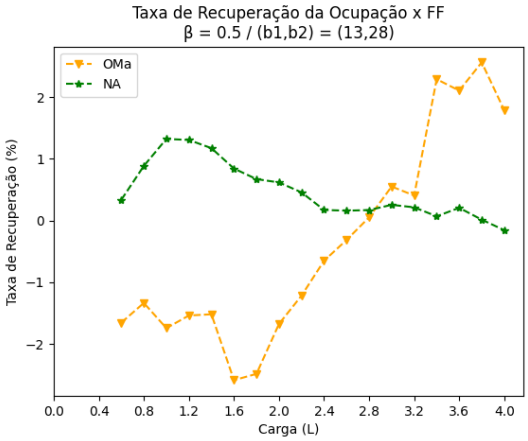


Fig. 6.16 - Caso $L = 4{,}0$: recuperação da ocupação × FF para $\beta = 0{,}5$ e $(b_1, b_2) = (13, 28)$.

Os Quadros 6.3 e 6.4, exibidos a seguir, especificam os valores numéricos dos níveis de ocupação espectral e vazão média, respectivamente, gerados pelos quatro algoritmos comparados com $\beta = 0{,}5$ e carga variando até $L = 4{,}0$. No destaque (em amarelo), a faixa de carga na qual o OMa ultrapassa NA e FF, tanto nos níveis de ocupação quanto da vazão.

Quadro 6.3 - Caso $L = 4{,}0$: níveis de ocupação espectral para $\beta = 0{,}5$ e $(b_1, b_2) = (13, 28)$.

(L)	DF	OMa	NA	FF
0,2	64,1850	64,1836	64,1850	64,1848
0,4	128,1001	127,2953	127,8209	127,7960
0,6	187,1508	179,8191	183,5502	182,9432
0,8	230,4515	215,7869	221,1256	218,9541
1,0	256,8777	235,8696	243,8856	240,5883
1,2	272,3984	251,2213	258,5823	254,8738
1,4	282,0075	260,8119	268,1892	264,9784
1,6	288,6184	264,5739	274,5982	271,8432
1,8	293,4882	269,8230	279,2104	277,4828
2,0	297,0271	275,8807	282,8047	281,2285
2,2	299,7191	280,8014	285,2469	284,3820
2,4	302,0418	285,1133	287,9313	287,2031
2,6	303,5598	288,3788	289,7328	289,3698
2,8	304,9332	291,3223	291,8854	291,3628
3,0	306,1794	**293,8369**	293,2994	293,2123
3,2	307,0249	**295,7002**	294,8606	294,7145
3,4	307,8591	**302,9132**	296,1840	296,0727
3,6	308,5282	**303,8123**	297,7708	297,5734
3,8	309,1104	**305,8003**	298,8925	298,8233
4,0	309,4428	**304,9479**	299,8451	299,6788

Quadro 6.4 - Caso $L = 4{,}0$: níveis de vazão média para $\beta = 0{,}5$ e $(b_1, b_2) = (13, 28)$.

(L)	DF	OMa	NA	FF
0,2	2,1394	2,1394	2,1394	2,1394
0,4	4,2704	4,2415	4,2603	4,2595
0,6	6,2342	5,9678	6,1023	6,0809
0,8	7,6559	7,1082	7,3041	7,2270
1,0	8,5056	7,6904	7,9947	7,8759
1,2	8,9841	8,1183	8,4084	8,2742
1,4	9,2635	8,3511	8,6540	8,5363
1,6	9,4520	8,3803	8,8066	8,7042
1,8	9,5742	8,4691	8,8955	8,8285
2,0	9,6631	8,6144	8,9623	8,8987
2,2	9,7272	8,7326	8,9958	8,9604
2,4	9,7776	8,8339	9,0432	9,0091
2,6	9,8087	8,9191	9,0700	9,0502
2,8	9,8387	9,0004	9,1119	9,0901
3,0	9,8611	9,0706	9,1315	9,1231
3,2	9,8752	9,1226	9,1627	9,1518
3,4	9,8929	**9,6262**	9,1957	9,1864
3,6	9,9009	**9,6465**	9,2292	9,2189
3,8	9,9082	**9,7671**	9,2514	9,2456
4,0	9,9058	**9,6347**	9,2740	9,2642

6.2 SIMULAÇÕES COM A CARGA ATÉ $L = 10{,}0$

A seguir, são apresentados os resultados alcançados com a simulação dos dois algoritmos propostos, NA e OMa, agora a carga variando até $L = 10{,}0$. Os parâmetros iniciais configurados nesta simulação são os mesmos que foram apresentados na Seção 6.1, apenas com a mudança da carga L normalizada, cuja variação vai de 0 até 10,0 com incrementos de 0,5.

Novamente, a simulação foi realizada para $(b_1, b_2) = (5, 14)$ e $(13, 28)$, respectivamente com $\beta = 2{,}0$ e $\beta = 0{,}5$, e compara a saída dos quatro algoritmos, todos submetidos ao mesmo tráfego de entrada e sob as mesmas condições simuladas: DF (linha azul), FF (linha preta), NA (linha verde) e OMa (linha laranja). O resultado completo das duas simulações realizadas para $L = 10{,}0$, incluindo o código fonte do programa, está disponível em https://doi.org/10.25824/redu/HT2MVA.

6.2.1 Caso $L = 10,0$: simulação do OMa com $\beta = 2,0$ e $(b_1, b_2) = (5, 14)$

As Figuras 6.17 a 6.20 ilustram, respectivamente, as curvas da ocupação espectral média em função da carga (L), da probabilidade de bloqueio de banda e das probabilidades de bloqueio para solicitações de b_1 e b_2 *slots*, do par selecionado $(b_1, b_2) = (5, 14)$ analisado com $\beta = 2{,}0$.

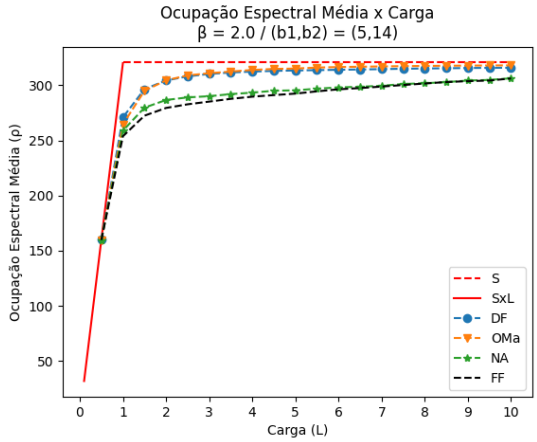


Fig. 6.17 - Caso $L = 10{,}0$: ocupação espectral média (ρ) × carga (L) para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$.

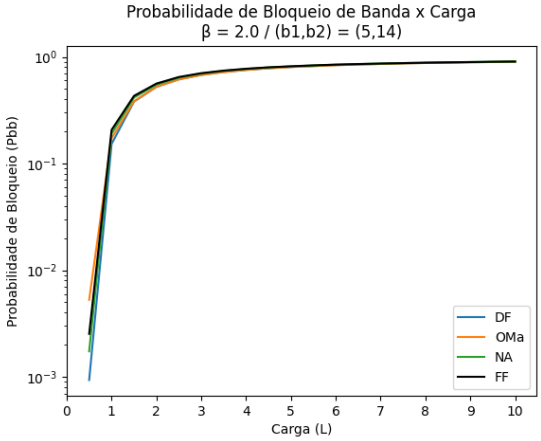


Fig. 6.18 - Caso $L = 10{,}0$: probabilidade de bloqueio de banda × carga (L) para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$.

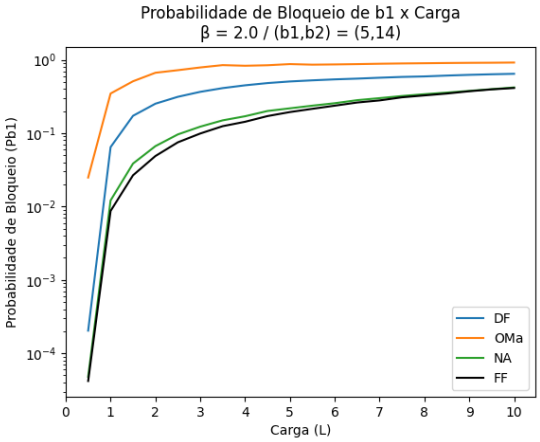


Fig. 6.19 - Caso $L = 10{,}0$: probabilidade de bloqueio de b_1 × carga (L) para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$.

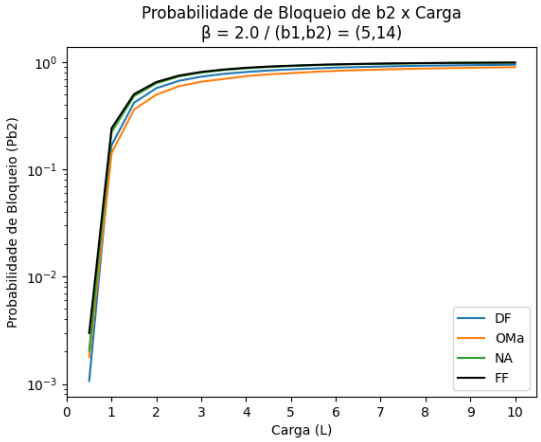


Fig. 6.20 - Caso $L = 10{,}0$: probabilidade de bloqueio de b_2 × carga (L) para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$.

Com a extensão da carga até $L = 10{,}0$, verifica-se que o OMa supera os algoritmos NA e FF a partir de $L = 1{,}0$, como já observado na Seção 6.1.1, e mantém essa vantagem em toda a faixa de carga analisada. Além disso, a partir de $L = 2{,}0$, o OMa também ultrapassa os níveis de ocupação alcançados pelo DF (destaque em vermelho). Mantém-se a menor probabilidade de bloqueio de banda em relação ao DF a partir de $L = 2{,}0$, e menor probabilidade de bloqueio para requisições de b_2 *slots* em comparação ao NA e FF, e superando o algoritmo DF a partir de $L = 1{,}0$. Para a probabilidade de bloqueio de requisições de b_1 *slots*, o comportamento observado é a extensão do que foi verificado na Seção 6.1.1.

Em seguida, as Figuras 6.21 a 6.24 ilustram, respectivamente, as curvas da vazão média versus carga (L) e das taxas de redução das perdas por fragmentação × FF, de redução das perdas totais × FF e de recuperação da ocupação proporcionada pela substituição do algoritmo FF pelo OMa, na comparação com o NA.

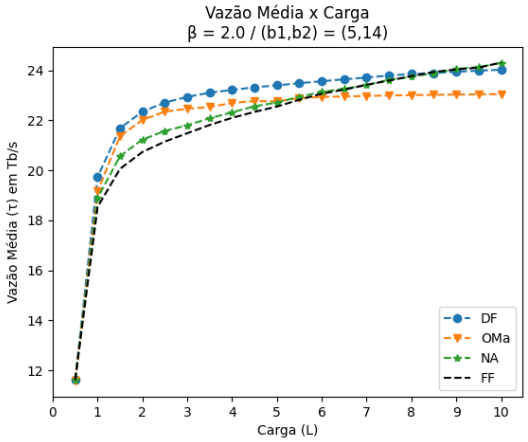


Fig. 6.21 - Caso $L = 10{,}0$: vazão média (τ) em Tb/s × carga (L) para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$.

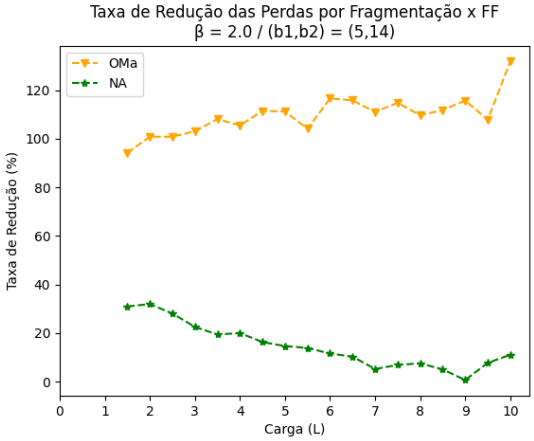


Fig. 6.22 - Caso $L = 10{,}0$: redução das perdas por fragmentação × FF para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$.

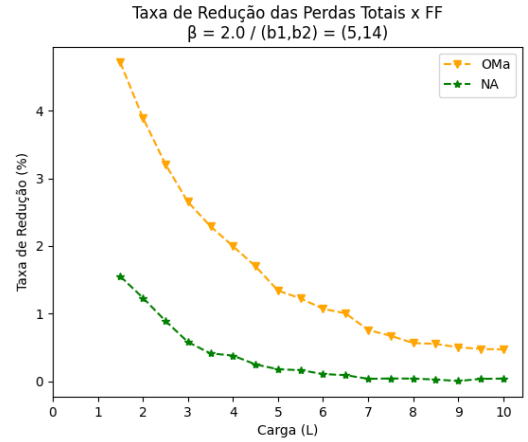


Fig. 6.23 - Caso $L = 10{,}0$: redução das perdas totais × FF para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$.

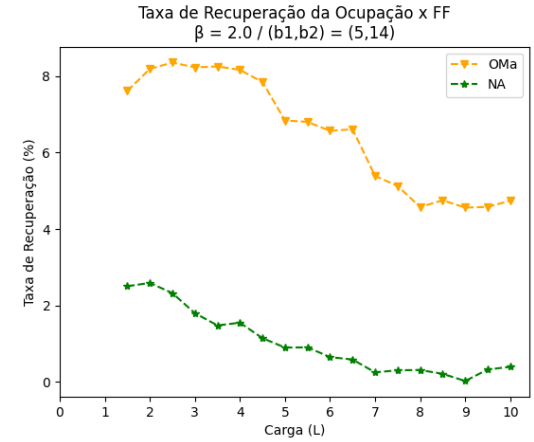


Fig. 6.24 - Caso $L = 10{,}0$: recuperação da ocupação × FF para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$.

Os Quadros 6.5 e 6.6 listam os valores numéricos dos níveis de ocupação espectral média e vazão média, respectivamente, obtidos pelos quatro algoritmos comparados com $\beta = 2{,}0$ e carga variando até $L = 10{,}0$. Em amarelo, novamente, é destacada a faixa de valores de carga na qual o algoritmo OMa ultrapassa os algoritmos NA e FF, e com a fonte destacada em vermelho a faixa na qual o OMa consegue ultrapassar os níveis de ocupação atingidos pelo DF, o que acontece a partir de $L = 2{,}0$.

Quadro 6.5 - Caso $L = 10{,}0$: níveis de ocupação espectral para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$.

(L)	DF	OMa	NA	FF
0,5	159,9177	159,2275	159,7935	159,6683
1,0	270,8480	**264,5859**	258,7970	253,7338
1,5	296,3671	**294,8661**	279,6070	272,3605
2,0	304,2501	**304,3812**	286,5456	279,3523
2,5	307,9730	**308,8582**	288,9201	282,6497
3,0	310,1259	**310,7856**	289,9850	285,1071
3,5	311,4706	**312,3199**	291,8849	287,6071
4,0	312,3034	**313,7960**	293,2266	289,5711
4,5	312,9037	**314,6908**	294,9327	291,0057
5,0	313,3173	**315,0349**	295,3227	292,3459
5,5	313,6087	**315,9276**	296,7918	294,3946
6,0	313,9249	**316,4108**	297,9812	296,2506
6,5	314,2114	**316,6746**	298,5655	297,3796
7,0	314,4968	**316,9557**	299,4187	298,7163
7,5	314,8662	**317,1989**	301,0577	300,1269
8,0	315,0886	**317,4026**	301,7482	301,4922
8,5	315,2878	**317,5829**	302,7617	302,8422
9,0	315,5408	**317,7125**	304,1804	303,6445
9,5	315,6677	**317,8396**	304,7210	304,1954
10	315,7577	**317,9434**	306,2372	306,0967

Quadro 6.6 - Caso $L = 10{,}0$: níveis de vazão média para $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$.

(L)	DF	OMa	NA	FF
0,5	11,6312	11,5768	11,6224	11,6134
1,0	19,7352	**19,1712**	18,8965	18,5361
1,5	21,6841	**21,3665**	20,5700	20,0602
2,0	22,3483	**22,0165**	21,2356	20,7360
2,5	22,7116	**22,3501**	21,5751	21,1512
3,0	22,9387	**22,4634**	21,8047	21,4852
3,5	23,1063	**22,5316**	22,0855	21,8161
4,0	23,2184	**22,6984**	22,3188	22,1019
4,5	23,3211	**22,7722**	22,5640	22,3384
5,0	23,3998	**22,7609**	22,7132	22,5514
5,5	23,4853	22,8899	22,9426	22,8252
6,0	23,5662	22,9370	23,1416	23,0667
6,5	23,6408	22,9563	23,2612	23,2298
7,0	23,7127	22,9770	23,4103	23,4210
7,5	23,7932	22,9948	23,6209	23,5917
8,0	23,8564	23,0097	23,7459	23,7680
8,5	23,8901	23,0230	23,8878	23,9349
9,0	23,9411	23,0324	24,0550	24,0342
9,5	23,9862	23,0416	24,1333	24,1077
10	24,0258	23,0493	24,2923	24,3126

6.2.2 Caso $L = 10,0$: simulação do OMa com $\beta = 0,5$ e $(b_1, b_2) = (13, 28)$

As Figuras 6.25 a 6.28 apresentam, em sequência, as curvas da ocupação espectral média versus carga (L), da probabilidade de bloqueio de banda e das probabilidades de bloqueio para requisições de b_1 e b_2 *slots*, do par selecionado $(b_1, b_2) = (13, 28)$ avaliado com $\beta = 0{,}5$.

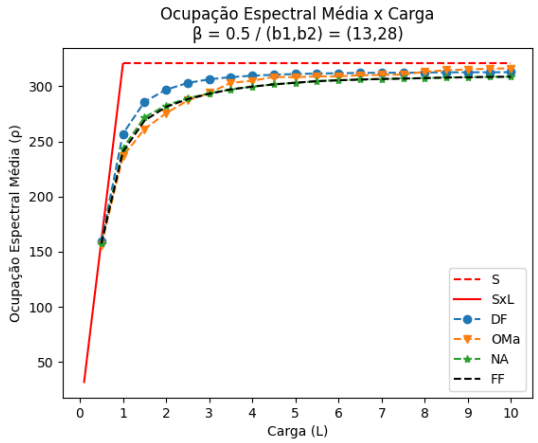


Fig. 6.25 - Caso $L = 10{,}0$: ocupação espectral média (ρ) × carga (L) para $\beta = 0{,}5$ e $(b_1, b_2) = (13, 28)$.

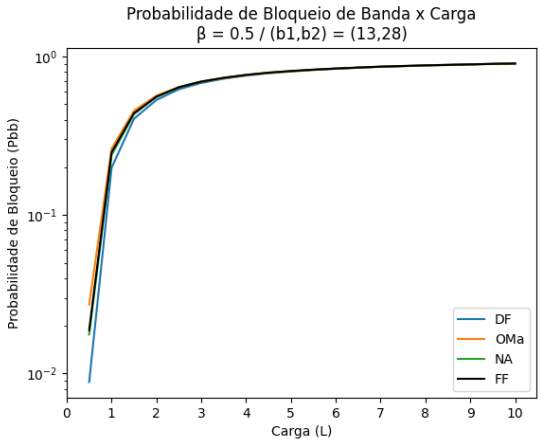


Fig. 6.26 - Caso $L = 10{,}0$: probabilidade de bloqueio de banda × carga (L) para $\beta = 0{,}5$ e $(b_1, b_2) = (13, 28)$.

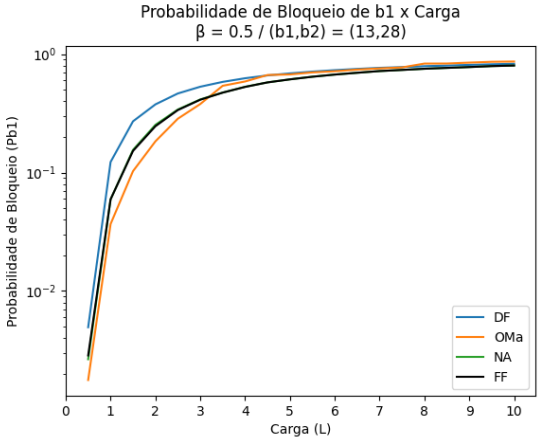


Fig. 6.27 - Caso $L = 10{,}0$: probabilidade de bloqueio de b_1 × carga (L) para $\beta = 0{,}5$ e $(b_1, b_2) = (13, 28)$.

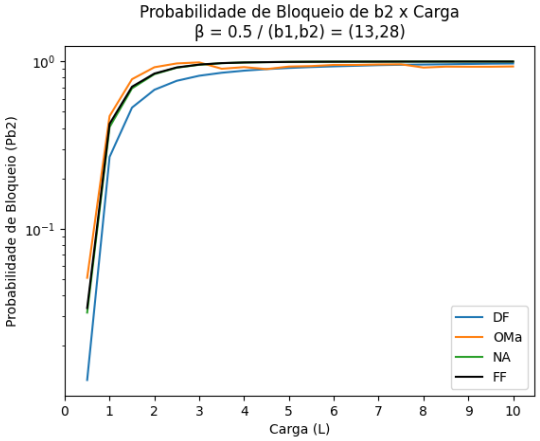


Fig. 6.28 - Caso $L = 10{,}0$: probabilidade de bloqueio de b_2 × carga (L) para $\beta = 0{,}5$ e $(b_1, b_2) = (13, 28)$.

Com a extensão da faixa de carga até 10,0, verifica-se que a tentativa de aproximação iniciada em $L = 3{,}4$, notada na Seção 6.1.2, se consolida, e a ultrapassagem sobre o DF ocorre próximo de $L = 8{,}0$. No caso do bloqueio de banda, o OMa mantém vantagem sobre NA e FF, agora verificada a partir de $L = 3{,}0$, e supera o DF a partir de $L = 8{,}0$.

Para requisições de b_1 *slots*, o OMa mantém a menor probabilidade de bloqueio em comparação aos demais algoritmos até $L = 3{,}4$, quando é superado por FF e NA e, novamente em $L = 7{,}5$, é ultrapassado pelo DF. Para pedidos de b_2 *slots*, o OMa mantém sobre NA e FF a menor probabilidade de bloqueio a partir de $L = 3{,}4$ e, em $L = 7{,}5$, supera também o DF.

As Figuras 6.29 a 6.32 destacam as curvas da vazão média em função da carga (L) e das taxas de redução das perdas por fragmentação × FF, de redução das perdas totais × FF e de recuperação da ocupação proporcionada pela substituição do algoritmo FF pelo OMa, comparado com o NA.

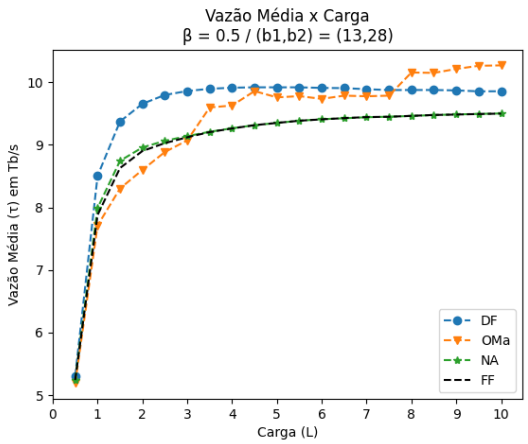


Fig. 6.29 - Caso $L = 10{,}0$: vazão média (τ) em Tb/s × carga (L) para $\beta = 0{,}5$ e $(b_1, b_2) = (13, 28)$.

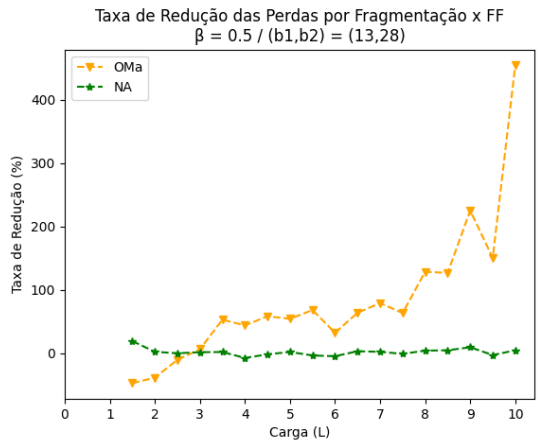


Fig. 6.30 - Caso $L = 10{,}0$: redução das perdas por fragmentação × FF para $\beta = 0{,}5$ e $(b_1, b_2) = (13, 28)$.

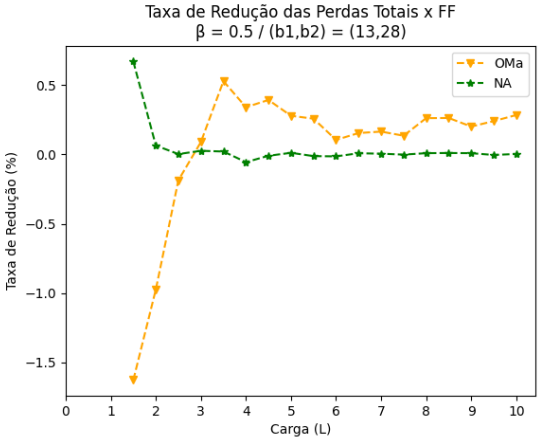


Fig. 6.31 - Caso $L = 10{,}0$: redução das perdas totais × FF para $\beta = 0{,}5$ e $(b_1, b_2) = (13, 28)$.

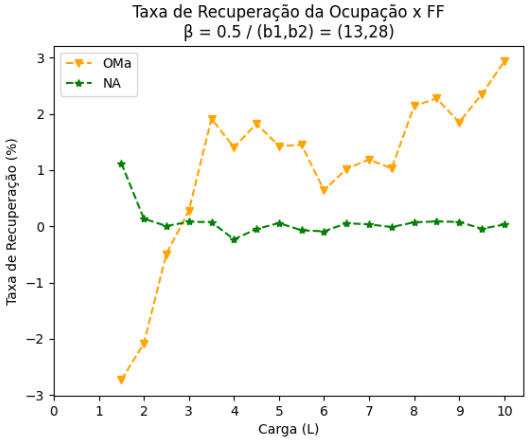


Fig. 6.32 - Caso $L = 10{,}0$: recuperação da ocupação × FF para $\beta = 0{,}5$ e $(b_1, b_2) = (13, 28)$.

Os Quadros 6.7 e 6.8 detalham os valores numéricos dos níveis de ocupação espectral e vazão média, respectivamente, alcançados pelos quatro algoritmos comparados com $\beta = 0{,}5$ e $L = 10{,}0$. No destaque (em amarelo), a faixa de carga na qual o OMa ultrapassa NA e FF, tanto nos níveis da ocupação como da vazão.

Quadro 6.7 - Caso $L = 10{,}0$: níveis de ocupação espectral para $\beta = 0{,}5$ e $(b_1, b_2) = (13, 28)$.

(L)	DF	OMa	NA	FF
0,5	159,0656	156,0603	157,6292	157,4435
1,0	256,8412	236,4653	243,8281	240,7109
1,5	285,7742	261,1298	271,8338	268,7108
2,0	296,8159	275,4634	282,5894	281,2295
2,5	302,8821	286,9978	289,0737	288,2018
3,0	306,1813	**293,8204**	293,4360	293,0629
3,5	308,2083	**303,0119**	297,1111	297,0775
4,0	309,4759	**304,8840**	299,6312	299,5926
4,5	310,3573	**308,3067**	301,6063	301,6018
5,0	311,0034	**308,0867**	303,0280	303,1229
5,5	311,4660	**308,8879**	304,4614	304,5151
6,0	311,7163	**308,8401**	305,3754	305,3146
6,5	311,9983	**309,9110**	305,9497	305,9646
7,0	312,1401	**310,1693**	306,4949	306,5288
7,5	312,1658	**310,5258**	306,8543	306,8351
8,0	312,3396	**313,0963**	307,2554	307,3276
8,5	312,5142	**314,4081**	307,7819	307,7815
9,0	312,5829	**315,2072**	308,1273	308,0721
9,5	312,5807	**315,8695**	308,4143	308,3642
10	312,6915	**316,0720**	308,5861	308,5300

Quadro 6.8 - Caso $L = 10{,}0$: níveis de vazão média para $\beta = 0{,}5$ e $(b_1, b_2) = (13, 28)$.

(L)	DF	OMa	NA	FF
0,5	5,3009	5,1924	5,2487	5,2422
1,0	8,5045	7,7110	7,9912	7,8796
1,5	9,3733	8,3023	8,7446	8,6293
2,0	9,6556	8,5992	8,9585	8,9014
2,5	9,7965	8,8850	9,0641	9,0276
3,0	9,8599	9,0714	9,1389	9,1209
3,5	9,8954	**9,5971**	9,2086	9,2043
4,0	9,9128	**9,6282**	9,2647	9,2613
4,5	9,9192	**9,8586**	9,3139	9,3144
5,0	9,9188	**9,7583**	9,3500	9,3501
5,5	9,9190	**9,7780**	9,3847	9,3850
6,0	9,9095	**9,7348**	9,4094	9,4071
6,5	9,9060	**9,7858**	9,4242	9,4267
7,0	9,8889	**9,7760**	9,4410	9,4433
7,5	9,8754	**9,7860**	9,4495	9,4485
8,0	9,8761	**10,1557**	9,4613	9,4632
8,5	9,8762	**10,1495**	9,4766	9,4769
9,0	9,8698	**10,2126**	9,4853	9,4842
9,5	9,8547	**10,2628**	9,4943	9,4926
10	9,8561	**10,2696**	9,5009	9,4992

Destacado em azul, aparecem os pontos de carga nos quais o NA foi superado pelo FF. Esse comportamento anômalo ocorre justamente quando a ocupação do DF está próxima de 312, que é o maior múltiplo de 13 inferior a 320.

A partir daí, os algoritmos vorazes em geral, inclusive o DF, não conseguem avançar mais em direção ao preenchimento total do espectro, por falta da visão estratégica de futuro proporcionada pela inteligência inerente ao aprendizado de máquina. Observe, por exemplo, que há um recuo do DF entre $L = 9{,}0$ e $L = 9{,}5$.

6.3 DISCUSSÃO DOS RESULTADOS

As simulações realizadas com a extensão de L até $4{,}0$ e $10{,}0$ confirmam a expectativa evidenciada no Capítulo 5, em que o comportamento das curvas da ocupação espectral já indicavam uma tentativa de aproximação por parte do OMa em relação ao DF, especialmente nos casos em que o estado do sistema se aproxima de um "casamento" com o perfil do tráfego. Tal situação é verificada no caso de melhor desempenho, com $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$, no qual a ultrapassagem acontece a partir de $L = 2{,}0$, com a manutenção dessa vantagem até $L = 4{,}0$ e, posteriormente, até $L = 10{,}0$.

Ainda no caso com $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$ verifica-se, em relação ao desempenho do NA, que o OMa foi superior na comparação, apresentando um bom desempenho na redução das perdas por fragmentação, com uma taxa em torno de 100%. Na redução das perdas totais, atingiu um pico de $4{,}7\%$ em $L = 1{,}4$ e manteve-se acima do NA.

Quanto à recuperação da ocupação × FF, apresentou uma variação decrescente entre 8% e 4% com o aumento da carga. Já no caso com $\beta = 0{,}5$ e $(b_1, b_2) = (13, 28)$, o de pior desempenho entre os casos avaliados, foi percebida uma melhora significativa e consistente do OMa, a partir de $L = 3{,}0$ em todas as métricas avaliadas, cujo desempenho supera o NA e, inclusive, supera também o DF a partir de $L = 8{,}0$ nos níveis de ocupação espectral e vazão. Aqui, merece destaque o desempenho da taxa de redução das perdas por fragmentação, na qual o algoritmo OMa ultrapassa a taxa de 100% próximo de $L = 8{,}0$.

É importante ressaltar que o algoritmo OMa foi otimizado para maximizar a ocupação espectral, ou seja, foi configurado como recompensa uma métrica que minimiza a ociosidade do *slot*. Em outra configuração personalizada, por exemplo, seria possível otimizar o algoritmo para maximizar a vazão média com o ajuste por outro tipo de recompensa.

Mesmo com o foco na ocupação espectral, o algoritmo OMa apresentou um bom desempenho na vazão média na comparação realizada. Com $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$, ele foi superior ao NA e FF na faixa de carga entre $1{,}0$ e $5{,}0$ e ficou abaixo dos outros três

algoritmos a partir de $L = 5{,}0$. Já com $\beta = 0{,}5$ e $(b_1, b_2) = (13, 28)$, o algoritmo OMa foi sempre superior ao NA e FF a partir de $L = 3{,}5$ e ultrapassou o DF a partir de $L = 8{,}0$.

A análise da vazão média deve considerar o fato de que a comparação acontece entre três algoritmos vorazes (DF, NA e FF) e um algoritmo não voraz inteligente (OMa), de modo que a configuração do espectro se torna diferente à medida que o espectro se aproxima do seu preenchimento total.

Com um algoritmo voraz, é esperado que com o aumento do preenchimento do espectro e com a redução dos tamanhos dos buracos disponíveis, haja uma preferência "natural" pelos tamanhos menores por parte dos algoritmos vorazes. Assim, com o aumento da carga, o espectro vai sendo tomado por b_1 *slots*, de modo que quando $b_1 = 5$, tem-se uma eficiência espectral para o serviço de 400 Gb/s de $(400/5) = 80$ Gb/s/*slot*.

Com o algoritmo OMa, que é um algoritmo não voraz inteligente, o preenchimento do espectro é diferente em comparação a um algoritmo voraz, de maneira que sob tráfego intenso o algoritmo OMa será não voraz para alguns tamanhos de vazios, quando só existirem vazios que aumentem a ociosidade para uma determinada requisição.

Sendo assim, na configuração do espectro do OMa pode haver uma maior ocupação por requisições de b_2 *slots*, o que sinaliza no caso com $\beta = 2{,}0$ e $(b_1, b_2) = (5, 14)$ uma redução da eficiência espectral, que acontece para o serviço de 1 Tb/s com $b_2 = 14$, ou seja, $(1000/14) \cong 71{,}4$ Gb/s/*slot*. Com o aumento da carga, o surgimento de novos vazios com política ótima não voraz inteligente pode ser responsável pelas descontinuidades observadas na Fig. 6.29, por exemplo.

6.4 CONCLUSÕES

Enlaces ópticos elásticos oferecem um ganho de eficiência espectral no atendimento a demandas heterogêneas de espectro, ao possibilitarem um ajuste mais preciso às necessidades de cada cliente considerando a taxa de bits e a distância entre origem e destino ao longo do caminho óptico.

Por outro lado, eles introduzem um novo fator de ineficiência, representado pelas perdas de fragmentação, que não ocorrem sob o algoritmo *first-fit* de alocação espectral quando há apenas uma única classe de tráfego, caracterizada por um tamanho único de *slots* por conexão.

Para mitigar esse fator de ineficiência, foram estudados os seguintes algoritmos alternativos, visando reduzir as perdas de fragmentação inerentes ao algoritmo *first-fit*, considerado o estado da arte, apresentados em ordem crescente de desempenho em relação ao *first-fit*:

a) um novo algoritmo (NA), igualmente proativo e voraz, que prioriza a alocação em vazios funcionais, dando preferência aos inflexíveis, com o objetivo de preservar os flexíveis para acomodar flutuações de curto prazo no perfil do tráfego. Os vazios disfuncionais, que não podem ser totalmente preenchidos pelos tamanhos habilitados das conexões em trânsito, são utilizados apenas como último recurso;

b) um algoritmo hipotético capaz de desfragmentar (DF) o espectro imediatamente após cada terminação ou ativação de conexão em um enlace avulso, estabelecendo assim um limite teórico de desempenho para todos os algoritmos vorazes; e

c) um algoritmo inspirado no aprendizado de máquina (OMa), projetado para maximizar a recompensa associada à minimização da ociosidade espectral (com alocação adjacente) ao longo do tempo. Diferentemente da abordagem voraz, o algoritmo não prioriza a aceitação de requisições que, ainda que tecnicamente atendíveis, não contribuam significativamente para a minimização da ociosidade esperada futuramente.

Com o estudo realizado, verifica-se que o ganho de desempenho proporcionado por cada algoritmo avaliado está relacionado à quantidade de informações de que ele dispõe sobre o tráfego. No caso do algoritmo *first-fit*, que não dispõe de nenhuma informação sobre o tráfego, observa-se o pior desempenho entre os algoritmos analisados. O algoritmo *exact-fit* incorpora, em relação ao *first-fit*, um conhecimento mais detalhado das características do tráfego, pois busca alocar recursos que correspondam exatamente às necessidades de cada requisição, considerando o estado completo do enlace.

Por sua vez, o algoritmo NA também utiliza uma política voraz, porém com o uso de alguma inteligência adicional. O NA, ao conhecer previamente o tamanho das requisições das classes de tráfego, pode selecionar os vazios de acordo com uma hierarquia de critérios pré-

estabelecidos que priorizam determinados tipos de vazios funcionais em detrimento dos disfuncionais.

O ganho de desempenho apresentado pelo algoritmo NA em relação aos algoritmos *first-fit* e *exact-first-fit* expõe uma limitação associada ao uso de políticas vorazes sobre o tráfego, representada pelo limite teórico, entre os vorazes, imposto pelo algoritmo DF.

Sendo assim, o refinamento do algoritmo proposto passou por ampliar o conhecimento do algoritmo sobre o perfil do tráfego, incluindo a ciência das taxas de chegada λ_1 e λ_2, além da adição de inteligência na seleção do vazio gerada pela maximização de uma recompensa representada pela minimização da ociosidade média do vazio, com o uso de uma política não voraz. A abordagem não voraz, juntamente com a inteligência, proporcionou a aproximação/ultrapassagem do algoritmo OMa em relação ao DF sob tráfego intenso.

Os resultados apresentados confirmam que o algoritmo OMa oferece uma abordagem promissora para mitigar as perdas de fragmentação em enlaces ópticos elásticos, especialmente em cenários de carga alta. O desempenho superior do OMa, especialmente em termos de ocupação espectral e probabilidade de bloqueio de banda, destaca a sua viabilidade para implementação em sistemas ópticos de próxima geração, incluindo aqueles baseados em multiplexação por divisão espacial (SDM).

O algoritmo OMa apresentou desempenho superior ao do algoritmo desfragmentado e, consequentemente, a qualquer outro algoritmo voraz. No entanto, esse desempenho superior foi observado apenas em condições de tráfego de alta intensidade, caracterizadas por sobrecargas próximas ou superiores a 100%. Nessas situações, o algoritmo alcança ocupações próximas à capacidade total do espectro, mas às custas de probabilidades de bloqueio muito elevadas, o que se torna desvantajoso no contexto de uma única fibra. Por outro lado, em sistemas baseados em SDM, esses bloqueios não seriam definitivos, uma vez que as conexões rejeitadas por políticas não vorazes em uma fibra (ou núcleo) poderiam ser redirecionadas para uma segunda fibra, depois para uma terceira, e assim sucessivamente, evitando perdas e viabilizando o preenchimento total do espectro na fibra original.

Este estudo sugere que, quanto maior o conhecimento do algoritmo sobre os parâmetros do tráfego, melhor será seu desempenho na otimização da ocupação do espectro disponível. Para alcançar esse objetivo, o compromisso com a voracidade na alocação de recursos deve ser flexibilizado, permitindo que o desempenho do algoritmo supere o impacto da desfragmentação do espectro, mesmo assumindo que esta ocorra de forma ágil.

6.5 TRABALHOS FUTUROS

Recomenda-se a continuidade desta pesquisa para explorar ainda mais as capacidades dos algoritmos propostos NA e OMa, incluindo a sua integração com outras técnicas avançadas de processamento de sinais, *machine learning* e otimização de tráfego, visando a maximização da eficiência e capacidade dos sistemas de comunicação óptica.

Uma primeira possibilidade considerada é a utilização de um modelo de tráfego mais realista em substituição ao modelo de tráfego de Erlang, que foi empregado neste trabalho para simular as fontes geradoras de tráfego.

Em redes de comutação de circuitos ópticos, as fontes geradoras de tráfego, tais como os transponders em redes de fibra óptica, solicitam recursos espectrais que são os *slots* de frequência. Enquanto uma fonte está ocupada executando uma chamada, ela fica impedida de realizar outra chamada e, portanto, o tráfego gerado por qualquer fonte individual não pode ser estritamente poissoniano.

No entanto, o modelo clássico de tráfego de Erlang é poissoniano e amplamente utilizado em modelos analíticos para caracterizar o desempenho de enlaces ópticos [61]. Essa abordagem só é assintoticamente correta quando o número de fontes de tráfego é levado ao infinito, com cada uma produzindo apenas uma taxa infinitesimal para que as fontes infinitas forneçam o tráfego finito que é oferecido à rede. Em [65], os autores comparam as probabilidades de bloqueio previstas pelos dois modelos em um enlace óptico com uma determinada taxa média de chegadas de solicitações, dado um número finito de fontes idênticas no modelo de Engset.

Como cada fonte pode suportar apenas uma conexão por vez, os nós devem ser equipados com várias fontes para preencher o espectro disponível. O número de fontes de solicitações de *slots* é um dado do sistema no modelo de Engset e é suposto infinito no modelo de Erlang por uma questão de simplicidade [65].

Assim, a taxa de chegada de solicitações de conexões por fonte tem que ser infinitesimal no modelo de Erlang, para que a taxa total observada de chegada de solicitações ao sistema seja finita. Como o número de conexões ativas suportadas pelo espectro é sempre finito e o número de fontes é infinito, a flutuação na taxa de chegada torna-se insignificante no modelo de Erlang, de modo que o tráfego se torna assintoticamente poissoniano, ao contrário do modelo de Engset [65].

Para ser útil, um modelo deve ser simples, então o modelo de Erlang foi, historicamente, uma boa escolha na ausência de outros requisitos, como quando se considerava que as fibras ópticas abrigavam uma largura de banda “infinita”. No entanto, à medida que o espectro óptico se torna um recurso escasso nos dias de hoje, a busca por um modelo alternativo mais realista passou a ser recomendável [65].

Uma segunda vertente para a continuação desta pesquisa é a proposta de uma atualização para o algoritmo OMa, na qual a alocação adjacente a uma extremidade do vazio não seja mais a única utilizada pelo algoritmo. Em vazios pequenos, na maioria das vezes, a alocação adjacente é ótima ou a única possível. Porém, em vazios grandes, a alocação adjacente pode não ser a melhor opção na busca pela minimização da ociosidade de um vazio e, por consequência, a redução das perdas por fragmentação.

Para um vazio grande, a conexão pode ser alocada de forma não adjacente a uma extremidade do vazio, o que vai produzir, não um, mas dois novos vazios, um à esquerda e outro à direita da conexão alocada. Dessa forma, surgem várias possibilidades de alocação de uma conexão, de forma que os tamanhos dos dois novos vazios produzidos podem ser “ajustados” para gerar dois vazios otimizados para reduzir a ociosidade de uma requisição. Porém, a análise matemática desse problema se torna mais complexa quanto maior for o buraco escolhido para a alocação e esse problema merece uma investigação mais detalhada.

Uma terceira possibilidade para a continuidade desta pesquisa é o desenvolvimento de um algoritmo híbrido entre os comportamentos do NA e do OMa. Os resultados das simulações mostraram que, em diferentes faixas de carga, cada algoritmo teve um desempenho superior aos algoritmos vorazes disponíveis na literatura, FF e EF.

Sendo assim, seria possível configurar uma nova versão de algoritmo para selecionar, por ponto de carga, qual forma de alocação seria escolhida, a voraz do NA ou a não voraz inteligente do OMa, de modo a unificar a melhor performance de cada algoritmo.

As redes ópticas elásticas representam uma inovação promissora no campo das telecomunicações, permitindo uma utilização mais eficiente do espectro óptico. As perspectivas futuras de pesquisa nessa área se concentram na economia de recursos espectrais, em que técnicas avançadas de alocação de espectro são desenvolvidas para otimizar a distribuição dos canais de comunicação. Isso inclui o uso de algoritmos inteligentes e de *machine learning* para prever e adaptar a demanda de tráfego em tempo real, permitindo que o espectro seja utilizado de maneira mais eficiente e reduzindo a necessidade de alocação excessiva de largura de banda.

Além da economia de recursos espectrais, a redução das perdas por fragmentação do espectro óptico é uma área crítica de investigação. Pesquisas futuras visam desenvolver

métodos para minimizar essa fragmentação, como técnicas de reconfiguração dinâmica da rede e algoritmos de desfragmentação, além de novos algoritmos inteligentes de alocação espectral, como foi a proposta deste trabalho. Esses avanços têm o potencial de aumentar significativamente a capacidade e a flexibilidade das EONs, permitindo assim que as redes de comunicação sejam mais robustas e capazes de lidar com o crescimento exponencial do tráfego de dados.

REFERÊNCIAS BIBLIOGRÁFICAS

[1] SOUZA, Isaac Diego Tavares de. **Imparcialidade Versus Eficiência na Alocação Espectral em Redes Ópticas Elásticas**. 2017. 102 f. Dissertação (Mestrado) - Curso de Engenharia Elétrica, Faculdade de Engenharia Elétrica e de Computação, Universidade Estadual de Campinas, Campinas, 2017. Disponível em: https://repositorio.unicamp.br/Acervo/Detalhe/989857. Acesso em: 1 jun. 2021.

[2] FERREIRA, Eliezer Emanuel. **Eficiência e Imparcialidade em uma Rede Óptica Elástica Congestionada**. 2018. 93 f. Dissertação (Mestrado) - Curso de Engenharia Elétrica, Faculdade de Engenharia Elétrica e de Computação, Universidade Estadual de Campinas, Campinas, 2018. Disponível em: https://repositorio.unicamp.br/Acervo/Detalhe/1045564. Acesso em: 1 jun. 2021.

[3] LIRA, Clayton José Natal de. **Utilização de Meta-heurística e de Divisão Espectral para Alocação Eficiente de Espectro em Redes Ópticas Elásticas**. 2016. 58 f. Dissertação (Mestrado) - Curso de Engenharia Elétrica, Universidade Federal de Pernambuco, Recife, 2016. Disponível em: https://repositorio.ufpe.br/handle/123456789/28358?locale=pt_BR. Acesso em: 1 jun. 2021.

[4] BORTOLETTO, Rodrigo Campos. **Métodos Analíticos para o Estudo do Desempenho de Redes Ópticas Elásticas**. 2020. 189 f. Tese (Doutorado) - Curso de Engenharia da Informação, Universidade Federal do ABC, Santo André, 2020.

[5] KUMAR, Anurag; MANJUNATH, D.; KURI, Joy. **Communication Networking**: an analytical approach. [S.L.]: Elsevier, 2004. 930 p.

[6] EVERITT, B. S.; SKRONDAL, A. **The Cambridge Dictionary of Statistics**. 4. ed. Cambridge: Cambridge University Press, 2010. 480 p. Disponível em: https://www.stewartschultz.com/statistics/books/Cambridge%20Dictionary%20Statistics%204th.pdf. Acesso em: 12 mar. 2023.

[7] NOGUEIRA, Fernando. **Modelagem e Simulação**: cadeias de Markov. Universidade Federal de Juiz de Fora, Juiz de Fora, 2009. 39 p. Disponível em: http://leg.ufpr.br/lib/exe/fetch.php/projetos:microarray:cadeiamarkov.pdf. Acesso em: 15 mar. 2023.

[8] KELLY, F. P. **Reversibility and Stochastic Networks**. Cambridge: Cambridge University Press, 2011. 238 p.

[9] RUAS, Weldisson Ferreira. **Análise de Desempenho de Redes de Filas com Tráfico Modelado por Distribuições de Cauda Pesada**. 2010. 101 f. Dissertação (Doutorado) - Curso de Engenharia de Telecomunicações, Instituto Nacional de Telecomunicações - INATEL, Santa Rita do Sapucaí, 2010. Disponível em: https://www2.inatel.br/biblioteca/component/docman/doc_download/5070-analise-de-desempenho-de-redes-de-filas-com-trafico-modelado-por-distribuicoes-de-cauda-pesada. Acesso em: 22 maio 2023.

[10] WALDMAN, Helio; BORTOLETTO, Rodrigo Campos; TROYANO, Michel Scassiotte. Sobre-aprovisionamento de capacidade baseado no modelo de tráfego de Erlang. **Anais do XXVII Workshop de Gerência e Operação de Redes e Serviços (WGRS 2022)**, [S.L.], p. 29-42, maio 2022. Sociedade Brasileira de Computação - SBC. http://dx.doi.org/10.5753/wgrs.2022.223448.

[11] ANGUS, Ian. An Introduction to Erlang B and Erlang C. **Telemanagement**. Ajax, p. 6-8. ago. 2001. Disponível em: https://gente.itam.mx/cacosta/modelado2/ErlangBC.pdf. Acesso em: 26 maio 2023.

[12] IVERSEN, Villy Bæk. **Teletraffic engineering and network planning**. Kongens Lyngby: DTU Fotonik, 2015. 398 p.

[13]CHATTERJEE, Bijoy Chand; OKI, Eiji. **Elastic Optical Networks**: fundamentals, design, control, and management. [S.L.]: CRC Press, 2020. 232 p.

[14]TOMKOS, Ioannis *et al*. A tutorial on the flexible optical networking paradigm: state of the art, trends, and research challenges. **Proceedings of the IEEE**, [S.L.], v. 102, n. 9, p. 1317-1337, set. 2014. IEEE. http://dx.doi.org/10.1109/jproc.2014.2324652.

[15]ROSA, Adriana; CAVDAR, Cicek; CARVALHO, Solon; COSTA, Joao; WOSINSKA, Lena. Spectrum allocation policy modeling for elastic optical networks. **High Capacity Optical Networks And Emerging/Enabling Technologies**, Istambul, p. 242-246, dez. 2012. IEEE. http://dx.doi.org/10.1109/honet.2012.6421472.

[16]WRIGHT, Paul; PARKER, Michael C.; LORD, Andrew. Minimum- and Maximum-Entropy Routing and Spectrum Assignment for Flexgrid Elastic Optical Networking [Invited]. **Journal of Optical Communications and Networking**, [S.L.], v. 7, n. 1, p. A66-A72, 15 jan. 2015. Optica Publishing Group. http://dx.doi.org/10.1364/jocn.7.000a66.

[17]CHATTERJEE, Bijoy Chand; OKI, Eiji. Performance evaluation of spectrum allocation policies for elastic optical networks. **2015 17th International Conference on Transparent Optical Networks (ICTON)**, Budapeste, p. 1-4, jul. 2015. IEEE. http://dx.doi.org/10.1109/icton.2015.7193485.

[18]CANDIA, Marcos Paulo Lobo de. **Heurística para alocação de espectro em redes ópticas elásticas baseada em medidas de fragmentação**. 2014. 77 f. Dissertação (Mestrado) - Curso de Computação Aplicada, Instituto Nacional de Pesquisas Espaciais - INPE, São José dos Campos, 2014. Disponível em: http://urlib.net/8JMKD3MGP7W/3FMFJ4S. Acesso em: 3 jun. 2023.

[19]WALDMAN, Helio. The Impending Optical Network Capacity Crunch. **2018 SBFoton International Optics and Photonics Conference (SBFoton IOPC)**, Campinas, p. 1-4, out. 2018. IEEE. http://dx.doi.org/10.1109/sbfoton-iopc.2018.8610949.

[20]RIBEIRO, Sérgio Ricardo Lima. **Roteamento Multicritério em Redes Ópticas Elásticas Dinâmicas**. 2014. 72 f. Dissertação (Mestrado) - Curso de Engenharia Elétrica, Instituto Federal de Educação, Ciência e Tecnologia da Paraíba - IFPB, João Pessoa, 2014. Disponível em: https://repositorio.ifpb.edu.br/jspui/bitstream/177683/241/1/Roteamento_Multicrit%C3%A9rio_em_Redes_%C3%93pticas_El%C3%A1sticas_Din%C3%A2micas.pdf. Acesso em: 5 jun. 2023.

[21]DIJKSTRA, E. W. A note on two problems in connexion with graphs. **Numerische Mathematik**, v. 1, n. 1, p. 269-271, 1959.

[22]BERTHOLD, Joseph; SALEH, Adel A. M.; BLAIR, Loudon; SIMMONS, Jane M. Optical Networking: past, present, and future. **Journal of Lightwave Technology**, [S.L.], v. 26, n. 9, p. 1104-1118, maio 2008. IEEE. http://dx.doi.org/10.1109/jlt.2008.923609.

[23]RUAN, Lu; ZHENG, Yanwei. Dynamic Survivable Multipath Routing and Spectrum Allocation in OFDM-Based Flexible Optical Networks. **Journal of Optical Communications and Networking**, [S.L.], v. 6, n. 1, p. 77-85, jan. 2014. Optica Publishing Group. http://dx.doi.org/10.1364/jocn.6.000077.

[24]WALDMAN, H.; ALMEIDA, R. C.; BORTOLETTO, R. C.; ASSIS, K. D. An analytical defragmentation bound on the performance of the elastic single link with dynamic traffic. **2015 17th International Conference on Transparent Optical Networks (ICTON)**, Budapeste, p. 1-6, jul. 2015. IEEE. http://dx.doi.org/10.1109/icton.2015.7193691.

[25]IZQUIERDO-ZARAGOZA, Jose-Luis; PAVON-MARINO, Pablo; BUENO-DELGADO, Maria-Victoria. Distance-adaptive online RSA algorithms for heterogeneous flex-grid networks. **2014**

International Conference on Optical Network Design and Modeling, Estocolmo, p. 204-209, maio 2014. IEEE.

[26] WANG, Rui; MUKHERJEE, Biswanath. Spectrum management in heterogeneous bandwidth optical networks. **Optical Switching and Networking**, [S.L.], v. 11, p. 83-91, jan. 2014. Elsevier BV. http://dx.doi.org/10.1016/j.osn.2013.09.003.

[27] SARIDIS, George M.; ALEXANDROPOULOS, Dimitris; ZERVAS, Georgios; SIMEONIDOU, Dimitra. Survey and Evaluation of Space Division Multiplexing: from technologies to optical networks. **IEEE Communications Surveys & Tutorials**, [S.L.], v. 17, n. 4, p. 2136-2156, ago. 2015. IEEE. http://dx.doi.org/10.1109/comst.2015.2466458.

[28] STRAUB, Stefan; KIRSTADTER, Andreas; SCHUPKE, Dominic. Multi-period Planning of WDM-Networks: comparison of incremental and eol approaches. **2006 2nd IEEE/IFIP International Conference in Central Asia on Internet**, Tasquente, set. 2006. IEEE. http://dx.doi.org/10.1109/canet.2006.279271.

[29] VELASCO, Luis; MORALES, Fernando; GIFRE, Lluís; CASTRO, Alberto; DIOS, Oscar González de; RUIZ, Marc. On-Demand Incremental Capacity Planning in Optical Transport Networks. **Journal of Optical Communications and Networking**, [S.L.], v. 8, n. 1, p. 11-22, jan. 2016. Optica Publishing Group. http://dx.doi.org/10.1364/jocn.8.000011.

[30] YIN, Yawei; ZHANG, Mingyang; ZHU, Zuqing; YOO, S. J. B. Fragmentation-Aware Routing, Modulation and Spectrum Assignment Algorithms in Elastic Optical Networks. **Optical Fiber Communication Conference/National Fiber Optic Engineers Conference 2013**, Anaheim, v. 5, n. 10, p. A100-A106, mar. 2013. OSA. http://dx.doi.org/10.1364/ofc.2013.ow3a.5.

[31] FIORANI, Matteo *et al*. Optical Spatial Division Multiplexing for Ultra-High-Capacity Modular Data Centers. **Optical Fiber Communication Conference**, Anaheim, p. 1-3, mar. 2016. OSA. http://dx.doi.org/10.1364/ofc.2016.tu2h.2.

[32] BHATIA, Bhupesh; BHATIA, Ashima Bhatnagar; ISHRAWAT, Rashmi. Comparison of Network Topologies for Optical Fiber Communication. **International Journal of Engineering Research & Technology (IJERT)**, [S.L.], v. 1, n. 10, p. 1-6, dez. 2012. Disponível em: https://www.ijert.org/research/comparison-of-network-topologies-for-optical-fiber-communication-IJERTV1IS10026.pdf. Acesso em: 10 jun. 2023.

[33] INTERNATIONAL IT (São Paulo) (org.). **Topologia de Rede**: conheça os principais tipos. 2021. Disponível em: https://www.internationalit.com/post/topologia-de-rede-conhe%C3%A7a-os-principais-tipos. Acesso em: 10 jun. 2023.

[34] WALDMAN, Helio; ALMEIDA JUNIOR, Raul C.; BORTOLETTO, Rodrigo. Imparcialidade sob Congestionamento em Redes Óticas Elásticas. **Anais de XXXVIII Simpósio Brasileiro de Telecomunicações e Processamento de Sinais (SBrT 2020)**, Florianópolis, p. 1-5, nov. 2020. Sociedade Brasileira de Telecomunicações. http://dx.doi.org/10.14209/sbrt.2020.1570658541.

[35] VALUEHOST (São Paulo) (org.). **O que é**: full mesh network. 2024. Disponível em: https://artigos.valuehost.com.br/glossario/o-que-e-full-mesh-network/. Acesso em: 11 jun. 2023.

[36] CHATTERJEE, Bijoy Chand; BA, Seydou; OKI, Eiji. Fragmentation Problems and Management Approaches in Elastic Optical Networks: a survey. **IEEE Communications Surveys & Tutorials**, [S.L.], v. 20, n. 1, p. 183-210, 2018. IEEE. http://dx.doi.org/10.1109/comst.2017.2769102.

[37] ZHU, Zuqing; LU, Wei; ZHANG, Liang; ANSARI, Nirwan. Dynamic Service Provisioning in Elastic Optical Networks with Hybrid Single-/Multi-Path Routing. **Journal of Lightwave**

Technology, [S.L.], v. 31, n. 1, p. 15-22, jan. 2013. IEEE. http://dx.doi.org/10.1109/jlt.2012.2227683.

[38]ZHU, Ruijie *et al*. Dynamic time and spectrum fragmentation-aware service provisioning in elastic optical networks with multi-path routing. **Optical Fiber Technology**, [S.L.], v. 32, p. 13-22, dez. 2016. Elsevier BV. http://dx.doi.org/10.1016/j.yofte.2016.08.009.

[39]DHARMAWEERA, Madushanka Nishan; ZHAO, Juzi; YAN, Li; KARLSSON, Magnus; AGRELL, Erik. Traffic-Grooming- and Multipath-Routing-Enabled Impairment-Aware Elastic Optical Networks. **Journal of Optical Communications and Networking**, [S.L.], v. 8, n. 2, p. 58-70, fev. 2016. Optica Publishing Group. http://dx.doi.org/10.1364/jocn.8.000058.

[40]MOURA, Pedro M.; FONSECA, Nelson L. S. da; SCARAFICCI, Rafael A. Fragmentation aware routing and spectrum assignment algorithm. **2014 IEEE International Conference on Communications (ICC)**, Sydney, p. 1137-1142, jun. 2014. IEEE. http://dx.doi.org/10.1109/icc.2014.6883474.

[41]MOURA, Pedro M.; SCARAFICCI, Rafael A.; FONSECA, Nelson L. S. da. Algorithm for energy efficient routing, modulation and Spectrum Assignment. **2015 IEEE International Conference on Communications (ICC)**, Londres, p. 5961-5966, jun. 2015. IEEE. http://dx.doi.org/10.1109/icc.2015.7249272.

[42]FADINI, Waya; CHATTERJEE, Bijoy Chand; OKI, Eiji. A subcarrier-slot partition scheme with first-last fit spectrum allocation for elastic optical networks. **Computer Networks**, [S.L.], v. 91, p. 700-711, nov. 2015. Elsevier BV. http://dx.doi.org/10.1016/j.comnet.2015.08.048.

[43]FADINI, Waya; OKI, Eiji. A subcarrier-slot partition scheme for wavelength assignment in elastic optical networks. **2014 IEEE 15th International Conference on High Performance Switching and Routing (HPSR)**, Vancouver, p. 7-12, jul. 2014. IEEE. http://dx.doi.org/10.1109/hpsr.2014.6900874.

[44]LAGRANGE, Xavier; JABBARI, Bijan. **Multiaccess, Mobility and Teletraffic for Wireless Communications**. 6. ed. [S.L.]: Springer, 2013. 362 p.

[45]CHATTERJEE, Bijoy Chand; FADINI, Waya; OKI, Eiji. A spectrum allocation scheme based on first–last-exact fit policy for elastic optical networks. **Journal of Network and Computer Applications**, [S.L.], v. 68, p. 164-172, jun. 2016. Elsevier BV. http://dx.doi.org/10.1016/j.jnca.2016.02.020.

[46]WANG, Rui; MUKHERJEE, Biswananth. Spectrum management in heterogeneous bandwidth networks. **2012 IEEE Global Communications Conference (Globecom)**, Anaheim, p. 2907-2911, dez. 2012. IEEE. http://dx.doi.org/10.1109/glocom.2012.6503558.

[47]ZHANG, Mingyang; SHI, Weiran; GONG, Long; LU, Wei; ZHU, Zuqing. Bandwidth defragmentation in dynamic elastic optical networks with minimum traffic disruptions. **2013 IEEE International Conference on Communications (ICC)**, Budapeste, p. 3894-3898, jun. 2013. IEEE. http://dx.doi.org/10.1109/icc.2013.6655165.

[48]TAKAGI, Tatsumi; HASEGAWA, Hiroshi; SATO, Ken-Ichi; SONE, Yoshiaki; HIRANO, Akira; JINNO, Masahiko. Disruption Minimized Spectrum Defragmentation in Elastic Optical Path Networks that Adopt Distance Adaptive Modulation. **37th European Conference and Exposition on Optical Communications**, Genebra, p. 1-3, set. 2011. OSA. http://dx.doi.org/10.1364/ecoc.2011.mo.2.k.3.

[49]SINGH, Sandeep Kumar; JUKAN, Admela. Efficient Spectrum Defragmentation with Holding-Time Awareness in Elastic Optical Networks. **Journal of Optical Communications and**

Networking, [S.L.], v. 9, n. 3, p. B78-B89, 1 mar. 2017. Optica Publishing Group. http://dx.doi.org/10.1364/jocn.9.000b78.

[50] SINGH, Sandeep Kumar; BZIUK, Wolfgang; JUKAN, Admela. Analytical performance modeling of spectrum defragmentation in elastic optical link networks. **Optical Switching and Networking**, [S.L.], v. 24, p. 25-38, abr. 2017. Elsevier BV. http://dx.doi.org/10.1016/j.osn.2016.11.001.

[51] WALDMAN, H.; ALMEIDA, R. C.; BORTOLETTO, R. C.; ASSIS, K. D. R. Deadlock avoidance under incremental traffic in the elastic single link. **2014 16th International Conference on Transparent Optical Networks (ICTON)**, Graz, p. 1-4, jul. 2014. IEEE. http://dx.doi.org/10.1109/icton.2014.6876565.

[52] WALDMAN, H.; ALMEIDA, R. C.; ASSIS, K.D.R.; BORTOLETTO, R. C. Spectrum-sliced elastic optical networking. **2013 15th International Conference on Transparent Optical Networks (ICTON)**, Cartagena, p. 1-4, jun. 2013. IEEE. http://dx.doi.org/10.1109/icton.2013.6602816.

[53] WALDMAN, Helio; ALMEIDA, Raul C.; BORTOLETTO, Rodrigo C.; SOUZA, Vinícius F. de. A number-theoretic framework for the mitigation of fragmentation loss in elastic optical links. **Photonic Network Communications**, [S.L.], v. 43, n. 1, p. 85-99, mar. 2022. Springer Science and Business Media LLC. http://dx.doi.org/10.1007/s11107-022-00962-2.

[54] WALDMAN, Helio; ALMEIDA, Raul C.; BORTOLETTO, Rodrigo C. Spectral Idleness Minimization in the Elastic Single Link under Incremental Traffic. **2019 IEEE Latin-American Conference on Communications (LATINCOM)**, Salvador, p. 1-6, nov. 2019. IEEE. http://dx.doi.org/10.1109/latincom48065.2019.8937937.

[55] SUTTON, Richard S.; BARTO, Andrew G. **Reinforcement Learning**: an introduction. 2. ed. Cambridge: The MIT Press, 2018. 552 p.

[56] WALDMAN, H.; BORTOLETTO, R. C.; SOUZA, V. F. de; ALMEIDA, R. C. A Minimal Idleness Algorithm for Spectrum Assignment on a Single Elastic Link under Dynamic Traffic. **2023 23rd International Conference on Transparent Optical Networks (ICTON)**, Bucareste, p. 1-4, jul. 2023. IEEE. http://dx.doi.org/10.1109/icton59386.2023.10207312.

[57] JONES, Gareth A.; JONES, J. Mary. **Elementary Number Theory**. Berlim: Springer, 1998. 200 p. §1.2 Bezout's Identify, p. 7-11.

[58] WALDMAN, Helio; BORTOLETTO, Rodrigo C.; SOUZA, Vinícius F. de; ALMEIDA, Raul C. A Proactive Algorithm for the Mitigation of Fragmentation Losses in Elastic Links. **2022 IEEE Latin-American Conference on Communications (LATINCOM)**, Rio de Janeiro, p. 1-6, nov. 2022. IEEE. http://dx.doi.org/10.1109/latincom56090.2022.10000530.

[59] BÓRQUEZ-PAREDES, Danilo; BEGHELLI, Alejandra; LEIVA, Ariel; MURRUGARRA, Ruth. Does fragmentation avoidance improve the performance of dynamic spectrum allocation in elastic optical networks? **Photonic Network Communications**, [S.L.], v. 35, n. 3, p. 287-299, nov. 2017. Springer Science and Business Media LLC. http://dx.doi.org/10.1007/s11107-017-0745-5.

[60] SOUZA, André L. N. *et al*. Parameter Selection in Optical Networks with Variable-Code-Rate Superchannels. **Journal of Optical Communications and Networking**, [S.L.], v. 8, n. 7, p. A152-A161, jun. 2016. Optica Publishing Group. http://dx.doi.org/10.1364/jocn.8.00a152.

[61] ZUKERMAN, Moshe. **Introduction to Queueing Theory and Stochastic Teletraffic Models**. EE Department, City University of Hong Kong. Disponível em: http://www.ee.cityu.edu.hk/~zukerman/classnotes.pdf. Acesso em: 08 jun. 2022.

[62] ROSS, Keith W.; TSANG, Danny. The stochastic knapsack problem. **Proceedings of the 27th IEEE Conference on Decision and Control**, Texas, p. 632-633, dez. 1988. IEEE. http://dx.doi.org/10.1109/cdc.1988.194387.

[63] WANG, Rui; MUKHERJEE, Biswanath. Provisioning in Elastic Optical Networks with Non-Disruptive Defragmentation. **Journal of Lightwave Technology**, [S.L.], v. 31, n. 15, p. 2491-2500, ago. 2013. IEEE. http://dx.doi.org/10.1109/jlt.2013.2268535.

[64] PUTTNAM, Benjamin J.; RADEMACHER, Georg; LUÍS, Ruben S. Space-division multiplexing for optical fiber communications. **Optica**, [S.L.], v. 8, n. 9, p. 1186-1203, set. 2021. Optica Publishing Group. http://dx.doi.org/10.1364/optica.427631.

[65] WALDMAN, Helio; BORTOLETTO, Rodrigo C.; SOUZA, Vinícius F. de. A Comparison between the Erlang and Engset Traffic Models in the Provisioning of an Optical Link. **2024 24th International Conference on Transparent Optical Networks (ICTON)**, Bari, p. 1-4, jul. 2024. IEEE. http://dx.doi.org/10.1109/icton62926.2024.10647249.

APÊNDICE A - TRABALHOS PUBLICADOS

WALDMAN, Helio; ALMEIDA, Raul C.; BORTOLETTO, Rodrigo C.; SOUZA, Vinícius F. de. A number-theoretic framework for the mitigation of fragmentation loss in elastic optical links. **Photonic Network Communications**, [S.L.], v. 43, n. 1, p. 85-99, mar. 2022. Springer Science and Business Media LLC. http://dx.doi.org/10.1007/s11107-022-00962-2.

WALDMAN, Helio; BORTOLETTO, Rodrigo C.; SOUZA, Vinícius F. de; ALMEIDA, Raul C. A Proactive Algorithm for the Mitigation of Fragmentation Losses in Elastic Links. **2022 IEEE Latin-American Conference on Communications (LATINCOM)**, Rio de Janeiro, p. 1-6, nov. 2022. IEEE. http://dx.doi.org/10.1109/latincom56090.2022.10000530.

WALDMAN, H.; BORTOLETTO, R. C.; SOUZA, V. F. de; ALMEIDA, R. C. A Minimal Idleness Algorithm for Spectrum Assignment on a Single Elastic Link under Dynamic Traffic. **2023 23rd International Conference on Transparent Optical Networks (ICTON)**, Bucareste, p. 1-4, jul. 2023. IEEE. http://dx.doi.org/10.1109/icton59386.2023.10207312.

WALDMAN, Helio; BORTOLETTO, Rodrigo Campos; SOUZA, Vinícius Ferreira de; ALMEIDA JÚNIOR, Raul C. **Método para Alocação de Espectro Óptico em um Enlace Óptico sob Tráfego Dinâmico e Mídia de Armazenamento Legível por Computador**. BR n. 102022025807-4 A2. Depósito: 16 dez. 2022. Publicação: 25 jun. 2024. Instituto Nacional da Propriedade Industrial (INPI), p. 1-27, 2024.

WALDMAN, Helio; BORTOLETTO, Rodrigo C.; SOUZA, Vinícius F. de. A Comparison between the Erlang and Engset Traffic Models in the Provisioning of an Optical Link. **2024 24th International Conference on Transparent Optical Networks (ICTON)**, Bari, p. 1-4, jul. 2024. IEEE. http://dx.doi.org/10.1109/icton62926.2024.10647249.