



SOLUÇÃO EXATA PARA O PROBLEMA DE ALOCAÇÃO DE ESPECTRO,
REGENERADORES E MODULAÇÃO EM REDES ÓPTICAS ELÁSTICAS

Fabio David

Tese de Doutorado apresentada ao Programa de Pós-graduação em Engenharia de Sistemas e Computação, COPPE, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Doutor em Engenharia de Sistemas e Computação.

Orientador: José Ferreira de Rezende

Rio de Janeiro
Março de 2024

SOLUÇÃO EXATA PARA O PROBLEMA DE ALOCAÇÃO DE ESPECTRO,
REGENERADORES E MODULAÇÃO EM REDES ÓPTICAS ELÁSTICAS

Fabio David

TESE SUBMETIDA AO CORPO DOCENTE DO INSTITUTO ALBERTO
LUIZ COIMBRA DE PÓS-GRADUAÇÃO E PESQUISA DE ENGENHARIA
DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS
REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE DOUTOR
EM CIÊNCIAS EM ENGENHARIA DE SISTEMAS E COMPUTAÇÃO.

Orientador: José Ferreira de Rezende

Aprovada por: Prof. Diego Gimenez Passos
Prof. Ronaldo Moreira Salles
Prof. Luidi Gelabert Simonetti
Prof. Valmir Carneiro Barbosa

RIO DE JANEIRO, RJ – BRASIL
MARÇO DE 2024

David, Fabio

Solução exata para o problema de alocação de espectro, regeneradores e modulação em redes ópticas elásticas/Fabio David. – Rio de Janeiro: UFRJ/COPPE, 2024.

XIV, 66 p.: il.; 29, 7cm.

Orientador: José Ferreira de Rezende

Tese (doutorado) – UFRJ/COPPE/Programa de Engenharia de Sistemas e Computação, 2024.

Referências Bibliográficas: p. 56 – 61.

1. EON:Redes ópticas elásticas. 2. RMSA:Routing, Modulation and Spectrum Allocation. 3. ILP. I. Rezende, José Ferreira de. II. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia de Sistemas e Computação. III. Título.

*Aos meus filhos, Paula e Eduardo,
que, numa grata inversão
acadêmica, se tornaram os meus
exemplos a serem seguidos...*

Agradecimentos

Primeiramente, gostaria de expressar meu sincero agradecimento a todos os professores e funcionários do Programa de Engenharia de Sistemas e Computação (PESC). Em especial, aos meus orientadores Prof. José Ferreira de Rezende e Prof. Valmir Carneiro Barbosa, pelo empenho e incentivo durante todo o desenvolvimento do trabalho. As inúmeras reuniões remotas durante o período da pandemia facilitaram a superação desta difícil fase que todos vivemos. Aprendi muito com ambos, não apenas no aspecto técnico-científico, mas também no comportamento acadêmico.

Gostaria de agradecer aos membros da banca pelas correções e sugestões ao presente trabalho. Todas foram devidamente anotadas e servirão de norte em nossas pesquisas futuras.

Aos amigos que fiz no laboratório LAND e em todo o PESC, de modo geral. Em especial, ao meu amigo e colega Davi Brilhante, que me acompanhou durante todo o percurso do doutorado, trocando ideias sobre os mais diversos assuntos nas longas filas do bandeirão.

Também agradeço ao Núcleo de Computação Eletrônica (NCE) por ter me incentivado e apoiado durante todo o tempo.

Por fim, agradeço muito o apoio da minha família, e em especial minha irmã Silvia, que resolveu praticamente sozinha todos os problemas antes que chegassem a mim.

Não poderia esquecer de mencionar Einstein e Lola, um casal de fiéis *dachshunds* que participaram ativamente de praticamente todas as reuniões remotas com meus orientadores.

Resumo da Tese apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Doutor em Ciências (D.Sc.)

SOLUÇÃO EXATA PARA O PROBLEMA DE ALOCAÇÃO DE ESPECTRO, REGENERADORES E MODULAÇÃO EM REDES ÓPTICAS ELÁSTICAS

Fabio David

Março/2024

Orientador: José Ferreira de Rezende

Programa: Engenharia de Sistemas e Computação

O surgimento de novos serviços na Internet, como computação em nuvem, aplicações em tempo real, redes 5G, IoT, *streaming* de mídia em alta resolução, apenas para citar alguns exemplos, tem impulsionado a busca por maiores taxas de transmissão. Com os constantes avanços tecnológicos, através do desenvolvimento de dispositivos de transmissão e modulações mais eficientes, consegue-se atingir taxas de transmissão cada vez maiores, capazes de percorrer distâncias de centenas ou até milhares de quilômetros. Em redes de comunicação de dados, tirar o máximo proveito dos recursos disponíveis do meio físico é uma preocupação constante. Especificamente, mesmo em se tratando de transmissão de dados sobre fibras ópticas, o espectro óptico deve ser usado de forma criteriosa, apesar de sua enorme capacidade. Na última década, um novo paradigma surgiu denominado redes ópticas elásticas ou simplesmente EON (do inglês *Elastic Optical Networks*), justamente com o objetivo de tornar o uso de fibras ópticas ainda mais eficiente, reduzindo o desperdício característico de tecnologias atualmente empregadas. Um problema essencial a ser tratado em EON ficou conhecido como RMSA (*Routing, Modulation and Spectrum Allocation*). Neste trabalho, apresentamos um modelo ILP denominado RMSA-BP para solução exata do problema RMSA em EON. Dada uma topologia e um conjunto de demandas, é definida a melhor rota, a frequência de espectro e modulação a ser adotada para atender cada demanda. Também é definido o número mínimo de regeneradores necessários. Adicionalmente, e se diferenciando das demais propostas, o RMSA-BP também minimiza a taxa de bloqueio (número de demandas que não podem ser atendidas devido à falta de recursos) em seu objetivo principal. Apesar de toda a complexidade do problema, o modelo RMSA-BP atingiu bons resultados, obtendo a solução exata em tempos satisfatórios.

Abstract of Thesis presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Doctor of Science (D.Sc.)

EXACT SOLUTION TO THE SPECTRUM ALLOCATION, REGENERATORS AND MODULATION PROBLEM IN ELASTIC OPTICAL NETWORKS

Fabio David

March/2024

Advisor: José Ferreira de Rezende

Department: Systems Engineering and Computer Science

The emergence of new services on the Internet, such as cloud computing, real-time applications, 5G networks, IoT, high-resolution media streaming, just to name a few examples, has been driving the ever-growing need for higher transmission rates and bandwidth. The development of more efficient transmission devices and modulations has resulted in higher transmission rates, capable of covering distances of hundreds or even thousands of kilometers. In data communication networks, it is always necessary to make the most of the available resources of the physical medium. Specifically, even when it comes to data transmission over optical fibers, the optical spectrum must be used with care, despite its enormous capacity. In the last decade, a new paradigm has emerged, called Elastic Optical Networks (EON), precisely aiming to make the use of optical fibers even more efficient, by reducing spectrum waste found on currently employed technologies. An essential problem to be addressed in EON is known as RMSA (Routing, Modulation, and Spectrum Allocation). In this work, we present an ILP model called RMSA-BP for the exact solution of the RMSA problem in EON. Given a topology and a set of demands, our model identifies the most efficient route to meet each demand, from its origin to its destination, and specifies which spectrum frequency and modulation should be adopted on each link of the route. It also determines the minimum number and location of specific transmission devices, known as regenerators, to be allocated at some intermediate nodes of each route. Furthermore, unlike other proposals, RMSA-BP minimizes the blocking probability (the number of demands that cannot be met due to lack of resources). Despite the complexity of the problem, the RMSA-BP model achieved the optimal result in satisfactory processing times.

Sumário

Lista de Figuras	x
Lista de Tabelas	xii
Lista de Abreviaturas	xiii
1 Introdução	1
1.1 Objetivos e Contribuições	3
1.2 Estrutura do texto	4
2 Revisão de Conceitos	6
2.1 Conceitos básicos	6
2.1.1 Fibras ópticas e modulações	6
2.1.2 SDM	9
2.1.3 Redes Ópticas Elásticas	10
2.2 Regeneradores	11
2.2.1 FSs, Contiguidade e Continuidade	12
2.2.2 Segmentos	13
2.2.3 Fragmentação	15
2.2.4 Circuitos de proteção	16
2.2.5 Exemplo cenário completo	16
2.3 Estado da arte	18
3 RMSA-BP	21
3.1 Objetivos	21
3.2 Formulação RMSA-BP inicial	22
3.2.1 Primeiros testes e observações	27
3.3 RMSA-BPenum	29
3.3.1 Rotas pré-processadas	29
3.3.2 Alocação do espectro	29
3.3.3 Algoritmos de <i>Backtracking</i>	32
3.3.4 Implementação RMSA-BPenum com <i>backtracking</i>	33

3.3.5	Avaliação RMSA-BPenum	35
3.4	RMSA-BP	36
3.4.1	Pré-processamento + ILP	37
4	Implementação e Resultados	42
4.1	Ambiente de desenvolvimento	42
4.2	Estruturas de dados dos arquivos de entrada	43
4.3	Identificando segmentos válidos	44
4.4	Limitação de regeneradores por demanda	45
4.5	RMSA-BPenum	45
4.5.1	Observações no RMSA-BPenum	46
4.6	Resultados RMSA-BP	48
4.6.1	Limites por instância	48
4.6.2	Implementação da função multiobjetivo	48
4.6.3	Indicator Constraints	49
4.7	Resultados	49
4.8	Saturação dos Enlaces	51
4.9	Ajustando o Gurobi	51
5	Conclusão	53
5.1	Trabalhos futuros	55
	Referências Bibliográficas	56

Lista de Figuras

2.1	Evolução com surgimento de novas tecnologias. Fonte: (AGRAWAL [1], 2016 [1])	7
2.2	Evolução da capacidade de transmissão X distância das diversas gerações de sistemas de transmissão óptica. Fonte: (AGRAWAL [1], 2016 [1])	8
2.3	Evolução das taxas de transmissão. Fonte: (AGRAWAL,2021 [2])	8
2.4	Multiplexação por divisão de tempo (TDM)	9
2.5	Multiplexação por divisão de comprimento de onda (WDM)	9
2.6	Space Division Multiplexing. Fonte: (AGRAWAL, 2016 [1])	9
2.7	Comparação da alocação do espectro entre redes WDM e EON. Fonte (GERSTEL <i>et al.</i> , 2012 [3], editada)	10
2.8	a) Concatenação DWDM (desperdício de banda de guarda). b) Concatenação em EON. Fonte (GERSTEL <i>et al.</i> , 2012 [3], editada)	10
2.9	Regeneradores 1R, 2R e 3R	12
2.10	a) Enlace com todos FSs livres. b) FS ₁ alocado para demanda X. c) FS ₁ alocado para demanda X e FS ₂ a FS ₅ alocado para demanda Y. d) FS ₁ alocado para demanda X, FS ₂ a FS ₅ alocado para demanda Y e FS ₆ a FS ₈ alocado para demanda Z. e) FS ₁ alocado para demanda X e FS ₄ a FS ₇ alocado para demanda Y. f) FS ₁ alocado para demanda X e FS ₅ a FS ₈ alocado para demanda Y.	13
2.11	Caminho A-F. a) sem regeneradores b) dois regeneradores	14
2.12	Cenário exemplo. a) topologia b) conjunto de modulações	16
2.13	Exemplo de alocação de FSs	17
3.1	Exemplo de topologia para identificação de segmentos válidos	24
3.2	Topologias: (a)TopoX (b)NFSnet (c) Cost239 (d) Cost266	28
3.3	Estrutura de uma solução para uma demanda	30
3.4	Conjunto de soluções para uma demanda	31
3.5	Tabela de soluções possíveis das demandas 1 a n	32
3.6	Topologia exemplo pré-processamento	37
3.7	Cenários possíveis e BigM	41

4.1	Relação Segmentos X Regeneradores	45
4.2	Uso dos Enlaces para $ D = 30(A), 60(B), 90(C), 120(D)$	52

Lista de Tabelas

2.1	Exemplo de Modulações e suas características [4]	11
3.1	Segmentos viáveis para modulação com alcance 4000 km	28
4.1	Modulações usadas nos testes [4]	43
4.2	Tempo (s) para geração de P , 30 instâncias	44
4.3	Graus dos nós e tamanho dos enlaces por topologia	44
4.4	Número de soluções nos conjuntos Q_d , para todas as demandas	49
4.5	Tempo de pré-processamento (s), 30 instâncias	50
4.6	Valor médio de bloqueios e taxa de utilização, 30 instâncias	50
4.7	<i>Timeouts</i> e tempos de execução (h:m:s), 30 instâncias	51

Lista de Abreviaturas

BCO	<i>Bee Colony Optimization</i> , p. 20
BP	<i>Blocking Probability</i> , p. 4
CC	Continuidade e Contiguidade, p. 3
CE	Capacidade do Enlace em número de FSs, p. 12
CSV	<i>Comma-separated Values</i> , p. 43
DCKP	<i>Disjunctively Constrained Knapsack Problem</i> , p. 36
DPP	<i>Dedicated Path Protection</i> , p. 16
DWDM	<i>Dense Wavelength-Division Multiplexing</i> , p. 1
EON	<i>Elastic Optical Network</i> , p. 2
FS	<i>Frequency Slot</i> , p. 2
GA	<i>Genetic Algorithm</i> , p. 20
GML	<i>Graph Modeling Language</i> , p. 43
ITU-T	<i>ITU Telecommunication Standardization Sector</i> , p. 10
ITU	<i>International Telecommunication Union</i> , p. 10
MCF	Multi Core Fiber, p. 9
OEO	<i>Optical-Electrical-Optical</i> , p. 11
RLP	<i>Regenerator Location Problem</i> , p. 2
RMLSA	<i>Routing, Modulation Level and Spectrum Allocation</i> , p. 3
RMSA	<i>Routing, Modulation and Spectrum Allocation</i> , p. 3
RMSCA	<i>Routing, Modulation, Spectrum, and Core Allocation</i> , p. 20

RSA	<i>Route and Spectrum Allocation</i> , p. 18
RWA	<i>Routing and Wavelength Assignment</i> , p. 18
SBPP	<i>Shared-Backup Path Protection</i> , p. 16
SDM	<i>Space Division Multiplexing</i> , p. 9
SLA	<i>Service Level Agreement</i> , p. 16
TDM	<i>Time Division Multiplexing</i> , p. 1
WDM	<i>Wavelength Division Multiplexing</i> , p. 7

Capítulo 1

Introdução

O surgimento contínuo e crescente de novos serviços e aplicações *online* na Internet, utilização de servidores na nuvem, transferências de grandes volumes de dados, desenvolvimento de aplicativos de tempo real, *streaming* de vídeos em alta resolução e até a evolução das redes de telefonia móveis, incluindo o 5G e suas futuras versões, vem impulsionando cada vez mais a demanda por maiores velocidades e taxas de transmissão de dados, juntamente com a busca por menor latência e variação de latência possível. Fibras ópticas têm demonstrado serem um meio físico adequado para atender a todos esses requisitos ao longo das últimas décadas. Através da evolução constante das tecnologias de transmissão, tem sido possível atingir as taxas de transferência demandadas, ao mesmo tempo possibilitando a transmissão de dados por distâncias cada vez maiores[5].

A tecnologia de transmissão adotada em fibras ópticas evoluiu de tal forma que tornou possível a transmissão de vários fluxos de dados simultaneamente, através de uma única fibra física. A tecnologia *Dense Wavelength-Division Multiplexing* (DWDM) [6], surgida no final da década de 1990, é hoje uma tecnologia bastante consolidada e ainda amplamente utilizada. Para suportar a transmissão de múltiplos sinais distintos simultaneamente em uma única fibra óptica, a tecnologia DWDM divide o espectro óptico em blocos ou *slots* de tamanho fixo, utilizando diferentes comprimentos de onda. Tais blocos, conhecidos como *lambdas* (λ), são alocados de forma exclusiva para cada fluxo de dados, escolhidos de tal forma que um λ não interfira com os seus vizinhos. Desta forma, torna-se mais eficiente do que tecnologias que utilizam a multiplexação por tempo, como *Time Division Multiplexing* (TDM) [7], pois faz com que cada canal criado tenha sua própria banda dedicada e garantida, permitindo que todos canais sejam utilizados simultaneamente.

É importante observar que a transmissão em fibras ópticas, devido à atenuação do sinal que ocorre quando trafegam por distâncias mais longas, pode requerer a amplificação de sinal sendo transmitido ou até mesmo a sua completa regeneração, realizada por dispositivos conhecidos como regeneradores. Regeneradores, apesar

de melhorarem a qualidade e o alcance do sinal transmitido, adicionam custo e complexidade à operação e manutenção do sistema. Um problema bastante estudado em DWDM é o *Regenerator Location Problem* (RLP)[8] [9] [10], provado ser *NP-hard* em [8]. Seu principal objetivo é escolher a melhor rota, simultaneamente à definição e posicionamento do número mínimo de regeneradores necessários para atender um conjunto de demandas em uma dada topologia de rede DWDM.

No entanto, o DWDM também apresenta deficiências. Ao utilizar blocos de tamanho fixo, duas situações podem ocorrer. No primeiro caso, o tamanho de λ pode ser muito maior do que o necessário para transmitir os dados de uma demanda, acarretando em desperdício de espectro. Já no segundo caso, o oposto pode acontecer, onde o tamanho de λ é insuficiente para comportar os dados a serem transmitidos. Nesta situação, não existe uma forma nativa ao DWDM de se utilizar mais de um λ , cabendo à camada de aplicação resolver isso.

Uma das mais promissoras tecnologias em redes de fibras ópticas que surgiram na última década e que desempenhará um papel importante nos próximos anos são as Redes Ópticas Elásticas, ou *Elastic Optical Network* (EON)[3]. Esta nova geração de redes ópticas tem como objetivo fornecer alta flexibilidade e escalabilidade na alocação de espectro e suportar taxas de bits variáveis para diferentes tipos de tráfego, alocando os recursos disponíveis numa rede óptica de acordo com as necessidades de tráfego de cada conexão.

De forma semelhante ao DWDM, EONs permitem o compartilhamento simultâneo de canais de comunicação em uma única fibra óptica, através da divisão do espectro óptico em blocos de frequência, ou *frequency slots* (FSs). Ainda de maneira similar ao DWDM, EON também pode fazer uso de regeneradores em nós intermediários para permitir alcançar distâncias maiores. No entanto, as semelhanças acabam por aqui. Com mudanças relevantes na sua arquitetura, colocam EON em superioridade em relação ao DWDM. Em EON, cada FS, de forma equivalente ao λ , é um bloco de tamanho fixo, no entanto de tamanho bem menor do que o adotado em DWDM. Isto, por si só, reduz o desperdício citado acima (caso I). Adicionalmente, em EON é possível concatenar FSs contíguos, até atingirem o tamanho necessário para acomodar uma determinada demanda (caso II).

Outra diferença fundamental é a possibilidade de modulações distintas serem utilizadas para cada canal em uma mesma fibra, sendo possível escolher a modulação mais eficiente para atender cada demanda. É importante observar que, quanto maior o alcance de uma modulação, menor é a sua capacidade de bits por símbolo transmitido [1][5]. Chama-se caminho de luz, ou *lightpath*, o canal de comunicação estabelecido em uma rede óptica para a transmissão de dados entre dois nós quaisquer da rede, podendo passar por nós intermediários ou não. Esse canal de comunicação estabelecido é de uso exclusivo, dedicado para atender a uma demanda específica de

transmissão de dados, podendo ter um caráter temporário, ou seja, ser alocado por um período de tempo pré-definido.

Desta forma, dada uma demanda e um caminho entre origem e destino, torna-se possível definir qual é a modulação mais eficiente que suporte o comprimento total do caminho, de modo que não haja degradação do sinal. Uma vez definida a modulação mais adequada, e sabendo previamente a sua capacidade de transmissão de símbolos por FS, pode-se calcular o número mínimo de FSs necessários para atender os requisitos de banda da demanda através deste caminho específico, com quase nenhum desperdício do espectro.

Uma outra característica importante em EON é que seus regeneradores, além de serem capazes de amplificar ou regenerar o sinal, também podem alterar a frequência sendo usada pelo sinal de entrada recebido. E mais, é possível até mesmo mudar a sua modulação, antes de transmitir o sinal de saída regenerado.

Finalmente, o uso de regeneradores em nós intermediários de um caminho introduz um novo conceito em EON, denominado *SEGMENTO*. Um segmento é basicamente formado por um *lightpath* entre dois nós da rede, contendo nenhum regenerador em seus nós intermediários. Ou seja, um caminho completo entre dois nós quaisquer é formado por um ou mais segmentos concatenados. Um caminho contendo R regeneradores em seus nós intermediários, é composto por $R + 1$ segmentos. Dois nós vizinhos formam um segmento simples, composto por um único enlace. Todos os enlaces dentro de um mesmo segmento sendo usado para atender uma determinada demanda, devem utilizar a mesma frequência, modulação e número de FSs. Chamamos esta restrição de *CONTINUIDADE*. Além da restrição de continuidade, sempre que uma demanda necessita utilizar mais de um FS em um mesmo enlace, estes devem ser contíguos, criando mais uma restrição chamada *CONTIGUIDADE*. Atender às restrições de continuidade e contiguidade (CC), dificultam ainda mais encontrar uma solução exata para o problema.

1.1 Objetivos e Contribuições

Neste trabalho, apresentamos uma proposta para solução exata de um dos tópicos mais importantes em EON, um problema essencial conhecido como *Routing, Modulation and Spectrum Allocation* (RMSA). De forma semelhante ao RLP das redes DWDM que é classificado como *NP-hard* [8], o RMSA, muitas vezes referenciado como RMLSA (*Routing, Modulation Level and Spectrum Allocation*), também tem como objetivo escolher o caminho e a localização dos regeneradores para atender cada demanda. Mas, adicionalmente, deve definir a modulação mais eficiente em cada um dos segmentos usados pela demanda. Também deve especificar a alocação do espectro a ser usado em cada enlace por onde os dados da demanda trafegam, garantindo

exclusividade de seu uso, evitando conflitos de alocação entre as demandas, e ainda respeitando as restrições de continuidade e contiguidade.

É importante ressaltar que o surgimento de novas propostas para a solução exata do problema RMSA completo ficou estagnado nos últimos anos, devido à alta complexidade do problema. Neste período, surgiram propostas baseadas em heurísticas, bem como propostas que não atendiam por completo as restrições do RMSA como, por exemplo, as restrições de contiguidade e continuidade [4]. Outros trabalhos foram desenvolvidos se propondo a resolver problemas mais específicos ou cenários com características particulares.

Denominado RMSA-BP, o modelo aqui apresentado busca a solução exata para o problema RMSA completo, ao receber um conjunto de parâmetros:

- Topologia da rede em forma de grafo com a descrição dos nós e as respectivas distâncias de cada enlace;
- Conjunto de modulações disponíveis com a respectiva descrição de capacidade de transmissão por FS e alcance máximo de cada modulação;
- Conjunto de demandas a serem atendidas, contendo nó de origem, nó de destino e largura de banda requerida;

De posse destes parâmetros, o RMSA-BP determina a rota a ser usada por cada demanda, a localização dos regeneradores necessários, os FSs a serem utilizados por cada demanda em cada um dos enlaces de sua rota, respeitando sempre as restrições CC. O “BP” adotado no nome do modelo refere-se a *Blocking Probability*, que é um indicador muito utilizado para projeto, dimensionamento e planejamento de redes de telecomunicações, representando a probabilidade de um serviço ou demanda ser negada ou bloqueada, devido à indisponibilidade de recursos necessários para atendê-la num determinado momento. Desta forma, o RMSA-BP tem como objetivo atender o maior número possível de demandas do conjunto recebido, minimizando BP e, ao mesmo tempo, minimizando o uso de regeneradores e espectro de luz. Diferentemente de outras propostas, que utilizam BP apenas para estimar e comparar a qualidade das suas soluções, o RMSA-BP garante o menor BP possível, sendo uma opção atrativa para provedores de acesso, explorando ao máximo os recursos disponíveis e atendendo ao maior número possível de usuários.

1.2 Estrutura do texto

Os demais capítulos deste trabalho estão organizados conforme descrito a seguir. No Capítulo 2 é apresentada uma revisão dos principais conceitos básicos necessários para o entendimento do problema, bem como o atual estado da arte para sua solução.

No Capítulo 3, introduzimos nossas abordagens iniciais do modelo RMSA-BP, através de duas implementações, bem como as deficiências e problemas identificados. Na sequência, a proposta final do RMSA-BP é apresentada. A avaliação dos resultados obtidos, além de diversas observações, é realizada no Capítulo 4. Por último, no Capítulo 5, apresentamos nossas conclusões e descrevemos algumas propostas para trabalhos futuros.

Capítulo 2

Revisão de Conceitos

Este capítulo destina-se à revisão de conceitos fundamentais para o entendimento dos tópicos desta tese, bem como o estado da arte na solução do problema RMSA e suas variantes.

2.1 Conceitos básicos

A seguir, os principais conceitos de redes ópticas são descritos, bem como alguns de seus componentes e arquiteturas.

2.1.1 Fibras ópticas e modulações

As técnicas de modulação de sinal, os equipamentos e tecnologias envolvidas na transmissão de dados utilizando fibras ópticas, bem como as próprias fibras, evoluíram bastante nas últimas décadas, tornando-se peças fundamentais no aumento de capacidade, confiabilidade e alcance das redes de comunicação ópticas [5]. Podemos ter uma ideia desta evolução pelas Figuras 2.1 e 2.2 [1].

Na Figura 2.1, podemos ver o crescimento do *produto BL* conforme o surgimento de novas tecnologias ao longo dos anos. Produto BL, oriundo de *Bandwidth-Length product* ou *Bit rate-Distance product*, é uma métrica comumente adotada em redes ópticas para avaliar a performance de seus enlaces ponto-a-ponto. A linha tracejada mostra a tendência desta evolução. É interessante observar a mudança na inclinação no final da década de 70, quando fibras ópticas começaram a ser usadas como meio de transmissão em redes de comunicação de dados.

A Figura 2.2 mostra o crescimento constante do produto BL exclusivamente em redes ópticas, levando em consideração as diversas gerações de tecnologias de sistemas de comunicação óptica que surgiram no período de 1975-2000, aproximadamente. Pode-se observar que, a cada nova geração emergente de tecnologia óptica, o produto BL dá um salto inicial, mas tende a se estabilizar, à medida que a tecnologia

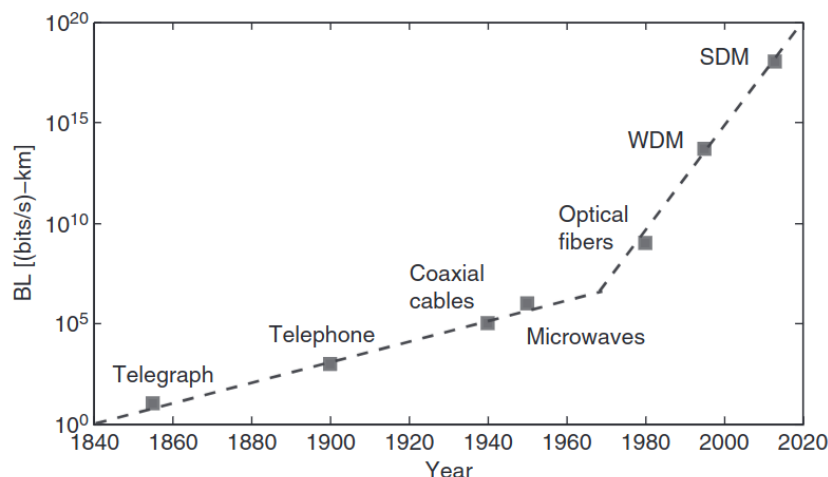


Figura 2.1: Evolução com surgimento de novas tecnologias. Fonte: (AGRAWAL [1], 2016 [1])

vai amadurecendo. Cada nova geração traz alguma alteração fundamental que permite melhorar a capacidade, desempenho e confiabilidade. O gráfico sugere que a capacidade de transmissão por km dobra a cada ano. A Figura 2.3 mostra o crescimento alcançado nas taxas de transmissão.

Muitos dos resultados anunciados hoje em dia, com taxas e distâncias cada vez maiores, batendo recordes como se fosse uma disputa esportiva, são inicialmente obtidos em laboratório, existindo uma certa defasagem até estarem disponíveis comercialmente.

A primeira geração de sistemas de comunicação óptica, surgida no final da década de 70, era extremamente simples. Os sinais do domínio elétrico eram convertidos para o domínio óptico e então enviados por transmissores equipados com lasers, até chegarem a um receptor óptico na outra ponta do enlace ponto-a-ponto, onde era feita a conversão de domínio reversa, através de um foto-receptor [1]. Disponíveis comercialmente somente no início da década de 80, operavam com uma taxa de bits de 45 Mbps e permitiam tráfegar por distâncias de até 10 km, muito superiores ao 1 km de alcance dos cabos coaxiais. O compartilhamento do meio físico era feito através de *Time Division Multiplexing* (TDM), como mostrado na Figura 2.4 [1].

Com o desenvolvimento de novos lasers e ao passar a utilizar frequências com dispersão mínima e, por consequência, menor perda por km, a segunda geração passou a oferecer taxas de 100 Mbps. Com o uso de fibras monomodo ao invés de multimodo, tornou-se possível atingir taxas de até 2 Gbps para distâncias próximas de 50 km. Por sua vez, a terceira geração, através de evolução constante nos tipos de fibras utilizadas, uso de amplificadores, melhoria na sensibilidade dos receptores ópticos, passaram a permitir alcance de até 100 km. A quarta geração se caracteriza principalmente pela introdução do *Wavelength Division Multiplexing* (WDM), onde o

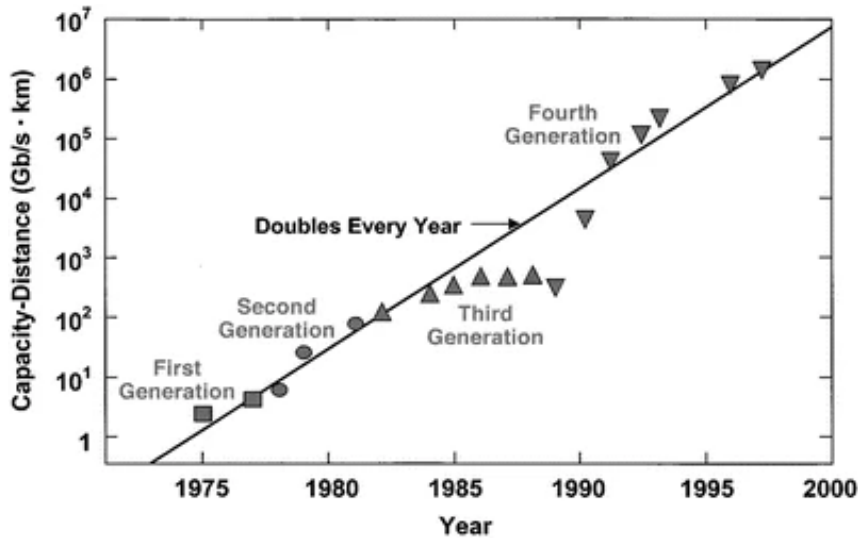


Figura 2.2: Evolução da capacidade de transmissão X distância das diversas gerações de sistemas de transmissão óptica. Fonte: (AGRAWAL [1], 2016 [1])

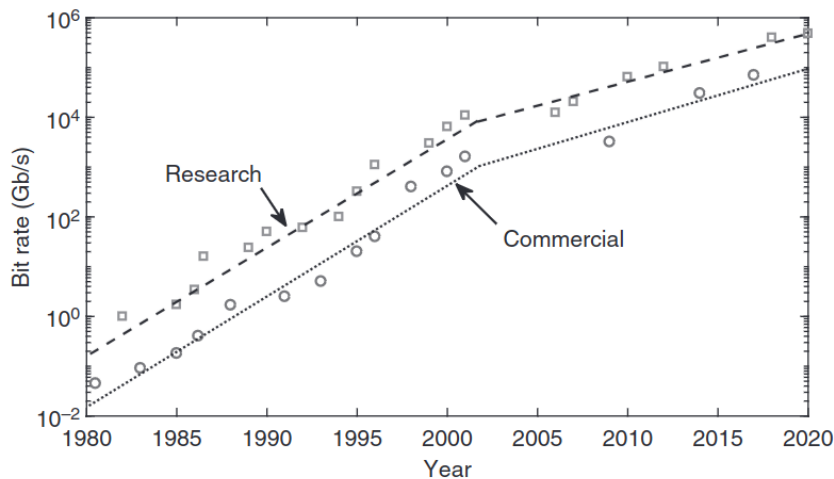


Figura 2.3: Evolução das taxas de transmissão. Fonte: (AGRAWAL,2021 [2])

espectro é dividido em blocos de tamanho fixo, chamados *lambdas* e representados por λ . Desta forma, tornou-se possível a criação de subcanais realmente dedicados, com transmissão simultânea, conforme mostrado na Figura 2.5. O WDM e sua evolução, *Dense Wavelength Division Multiplexing* (DWDM) [6], possibilitaram o aumento da capacidade de transmissão de maneira substancial. Para atingir esse resultado, foram utilizados diversos transmissores configurados para diferentes comprimentos de onda, um para cada canal. WDM e DWDM diferem basicamente no número de blocos e no tamanho da banda de guarda existente entre cada bloco. Bandas de guarda são pequenos blocos de frequência reservados, inseridos entre blocos de dados, para evitar a interferência entre eles.

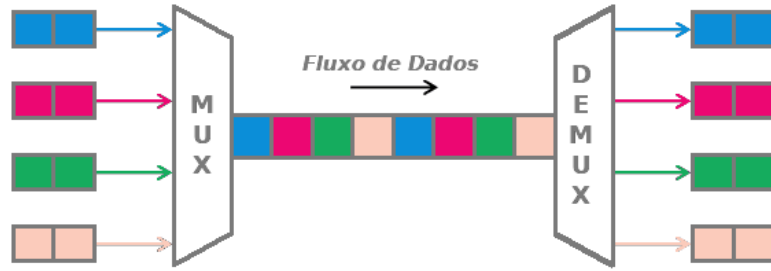


Figura 2.4: Multiplexação por divisão de tempo (TDM)

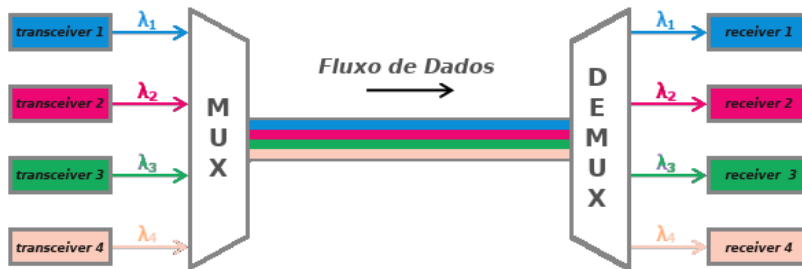


Figura 2.5: Multiplexação por divisão de comprimento de onda (WDM)

2.1.2 SDM

Na constante busca por aumentar a capacidade de transmissão em redes ópticas, o *Space Division Multiplexing* (SDM) tem uma abordagem completamente diferente da multiplexação por divisão de tempo (TDM) ou divisão por comprimento de onda (WDM). A ideia se baseia em utilizar diferentes caminhos dentro de uma mesma fibra. Isso é possibilitado pelo uso de fibras multi-core (MCF) que, como o próprio nome sugere, possuem múltiplos núcleos de transmissão numa única fibra óptica, ou ainda pelo uso de fibras multimodo com modos espaciais separados. O conceito básico é permitir a transmissão simultânea de dados por diferentes caminhos no mesmo espaço físico, aumentando assim a capacidade total de transmissão da rede [1]. A Figura 2.6 mostra o conceito da tecnologia SDM. Na Figura 2.1, SDM aparece no topo. Além disso, é possível utilizar DWDM dentro de um canal SDM, aumentando ainda mais a capacidade e aproveitamento da fibra óptica. Porém, introduz um novo

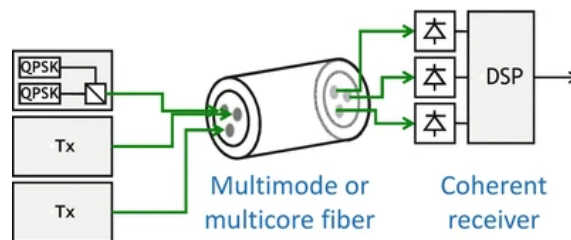


Figura 2.6: Space Division Multiplexing. Fonte: (AGRAWAL, 2016 [1])

problema, que é a interferência entre núcleos, conhecida como *crosstalk*, tornando necessário escolher as frequências cuidadosamente em núcleos mais próximos, visando evitar este efeito.

2.1.3 Redes Ópticas Elásticas

O conceito de *Elastic Optical Network*, ou simplesmente EON, proposto por GERSTEL *et al.* [3], surgiu para corrigir deficiências identificadas e resolver futuras limitações da tecnologia DWDM antecipadamente, como desperdício de espectro óptico e suporte a futuras taxas de transferências [11]. Tradicionalmente, o DWDM divide o espectro em lambdas (λ), utilizando blocos fixos de 50 GHz, seguindo a recomendação do G.694.1 do ITU-T [12], enquanto EON utiliza por padrão blocos ou *frequency slots*(FS) de 12,5 GHz. Como pode ser facilmente observado na Figura 2.7, pelo

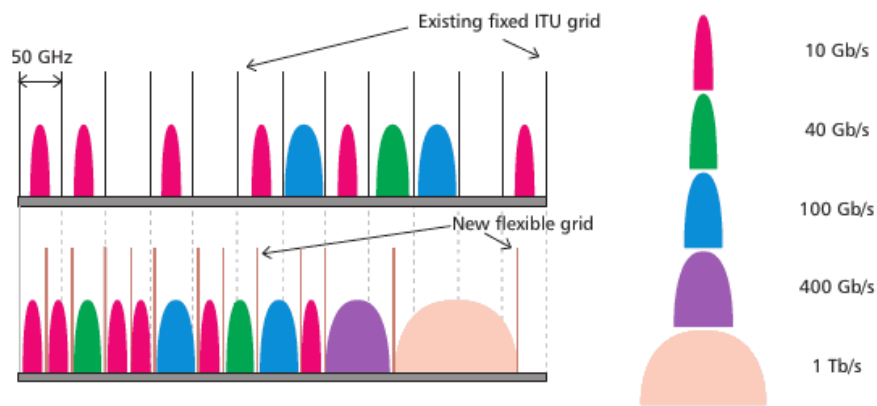


Figura 2.7: Comparação da alocação do espectro entre redes WDM e EON. Fonte (GERSTEL *et al.*, 2012 [3], editada)

fato de EON utilizar FSs menores que o tamanho padrão dos blocos adotado em DWDM, fica evidente o uso mais eficiente do espectro, ao transportar demandas que não requerem a total capacidade de transmissão de um bloco do DWDM. A mesma Figura 2.7 ilustra a capacidade em EON de concatenar FSs de modo a comportar demandas que não “cabem” em um único FS.

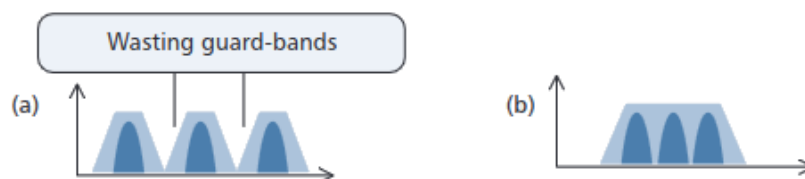


Figura 2.8: a) Concatenação DWDM (desperdício de banda de guarda). b) Concatenação em EON. Fonte (GERSTEL *et al.*, 2012 [3], editada)

A Figura 2.8 mostra como DWDM desperdiça espectro quando utiliza mais de um bloco, ainda mais realçado pela inserção de bandas de guarda.

Tabela 2.1: Exemplo de Modulações e suas características [4]

Modulação	FS Bandwidth (Gbps)	Alcance Máximo (km)
QPSK	25	2 000
8-QAM	37.5	1 000
16-QAM	50	500
32-QAM	62.5	250

Durante a transmissão, modulações com maior eficiência espectral, tendem a ser mais suscetíveis a erros causados pela dispersão fotônica, a medida que as distâncias percorridas pela transmissão aumentam, gerando interferências entre frequências mais próximas [5]. Por este motivo, quanto maior a capacidade de transmissão, menor é o alcance da modulação, conforme mostrado na Tabela 2.1.

2.2 Regeneradores

Regeneradores são dispositivos fundamentais em redes ópticas. Através destes dispositivos, maiores distâncias de transmissão podem ser alcançadas, sinais deteriorados por ruído e/ou atenuação podem ser reconstituídos, garantindo que sua retransmissão seja mais eficaz e confiável.

Regeneradores podem ser classificados de três formas: 1R, 2R e 3R, dependendo do tipo de regeneração de sinal realizada. Muitas vezes, a regeneração do sinal óptico é feita em diferentes domínios físicos, sendo necessária a conversão do domínio óptico para elétrico, processamento do sinal e posterior retorno do domínio elétrico para o domínio óptico novamente. Esta mudança de domínios é conhecida como O-E-O (*Optical-Electrical-Optical*), podendo adicionar atrasos na retransmissão. Regeneradores 1R basicamente amplificam o sinal recebido, atuando somente no domínio óptico. Regeneradores 2R fazem tanto a re-amplificação do sinal como seu condicionamento à forma original, enquanto regeneradores 3R, além da re-amplificação e condicionamento do sinal, atuam na correção e sincronização dos tempos de chegada dos pulsos, garantindo que o sinal regenerado volte ao sincronismo com o sinal original. Como pode ser observado na Figura 2.9, regeneradores do tipo R3 são os mais completos. No entanto, a escolha do tipo de regenerador mais adequado irá depender de cada caso.

Em relação ao uso de regeneradores em redes ópticas, podemos classificar as redes em três categorias, a saber: transparentes, opacas e translúcidas [13]. Em redes ópticas transparentes, nenhum regenerador é utilizado em toda rede, de modo que os sinais ópticos passam através dos diferentes nós sem a necessidade de conversão ou

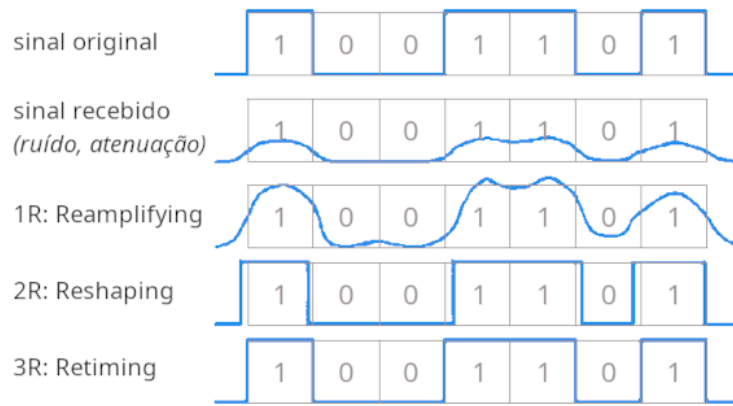


Figura 2.9: Regeneradores 1R, 2R e 3R

processamento. Redes ópticas opacas, ao contrário, possuem regeneração de sinal a cada nó. Por último, em redes ópticas translúcidas, alguns nós podem realizar as funções de comutação e amplificação. Redes ópticas elásticas costumam ser translúcidas. Atualmente, começam a surgir regeneradores 3R atuando somente no domínio óptico, sem a necessidade de O-E-O. No entanto, regeneradores, independentemente do tipo, são empregados somente quando se tornam realmente necessários, devido ao seu elevado custo e necessidade de instalação, configuração e manutenção.

2.2.1 FSs, Contiguidade e Continuidade

O problema de encontrar a rota ótima e a alocação de espectro (RSA) para as demandas em uma matriz de tráfego torna-se mais difícil de resolver do que problemas semelhantes em redes DWDM fixas, principalmente devido à granularidade mais fina do espectro e às restrições de contiguidade e continuidade do espectro.

O espectro óptico de enlace em EON é dividido em FSs. O número total de FSs por enlace óptico irá depender do projeto e infraestrutura da própria rede, além dos equipamentos envolvidos. Com o objetivo de deixar o texto deste trabalho mais claro, identificaremos este número de FSs suportados por enlace como CE (capacidade do enlace) e representado por ξ . Atualmente, o CE comumente adotado é de 160 ou 320 FSs por enlace. Cada FS é identificado pelo seu índice ($FS_1, FS_2, \dots, FS_\xi$). Como visto, FSs podem ser concatenados, contanto que seus índices sejam contíguos.

Com o objetivo de facilitar o entendimento de uso de regeneradores e FSs por enlace, inicialmente analisaremos um enlace com CE de apenas 8 FSs, como mostrado na Figura 2.10. Na Figura 2.10a, temos um enlace com todos os seus FSs livres, ou seja, nenhuma demanda está trafegando por este enlace. Na Figura 2.10b, apenas um FS foi alocado para a demanda X. Em seguida, na Figura 2.10c, uma outra demanda Y alocou 4 FSs neste mesmo enlace. Como o FS_1 já estava alocado para demanda X, Y usou os 4 FSs seguintes. Já na Figura 2.10d, uma terceira demanda

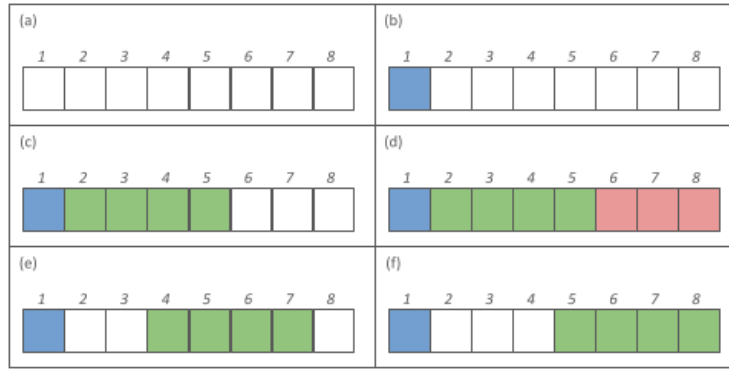


Figura 2.10: a) Enlace com todos FSs livres. b) FS₁ alocado para demanda X. c) FS₁ alocado para demanda X e FS₂ a FS₅ alocado para demanda Y. d) FS₁ alocado para demanda X, FS₂ a FS₅ alocado para demanda Y e FS₆ a FS₈ alocado para demanda Z. e) FS₁ alocado para demanda X e FS₄ a FS₇ alocado para demanda Y. f) FS₁ alocado para demanda X e FS₅ a FS₈ alocado para demanda Y.

Z alocou 3 FSs deste enlace. Neste momento, este enlace se tornou indisponível para outras demandas, por estar saturado, sem FSs livres disponíveis. Como mostrado na Figura 2.10e, caso a demanda Y tivesse alocado os FS₄ até FS₇, não haveria espaço neste enlace para a demanda Z, apesar de existirem 3 FSs livres disponíveis, por não haver contiguidade neste caso. Finalmente, na Figura 2.10f, seria possível a demanda Z utilizar o enlace, utilizando os 3 FSs livres disponíveis.

Quando uma transmissão de dados passa de um enlace para outro através de um nó sem regenerador (ou seja, pertencentes ao mesmo segmento), a mesma quantidade de FSs e seus respectivos índices, além da modulação adotada, têm que ser mantida, dando origem ao conceito de continuidade. Em outras palavras, índices de FS e modulação tem que ser os mesmos em todos os enlaces por onde a demanda trafega, a menos que passem por um nó com regenerador.

Observe que, neste exemplo, o número de FSs alocados para cada demanda é diferente. Uma explicação seria que as demandas têm requisitos de taxa de transmissão distintos, sendo alocado o número mínimo de FSs necessários e suficientes para atender os requisitos de cada demanda. Além disso, as demandas X, Y e Z podem ter origens e destinos completamente diferentes e, por consequência, caminhos distintos, podendo ter enlaces em comum ou não. Portanto, cada demanda pode estar empregando uma modulação diferente.

2.2.2 Segmentos

Definimos como segmento um caminho óptico transparente entre dois nós, ou seja, um *lightpath* entre um nó de origem e um nó de destino, sem regeneradores nos nós intermediários. Por exemplo, para transmitir dados de um nó de origem \mathcal{O} para um nó de destino \mathcal{D} , podemos considerar o caminho como sendo composto por um ou

mais segmentos conectados por intermédio de regeneradores. O primeiro segmento teria sua origem no nó \mathcal{O} e o último segmento do caminho terminaria no nó \mathcal{D} . Podemos observar que o primeiro segmento de um determinado caminho é o único que não possui um regenerador em seu nó de origem. Da mesma forma, o último segmento que compõe o caminho é o único que não possui um regenerador em seu nó de destino. Todos os segmentos intermediários terão regeneradores tanto no nó de origem quanto no nó destino. Um caminho com R regeneradores é formado por $R+1$ segmentos.

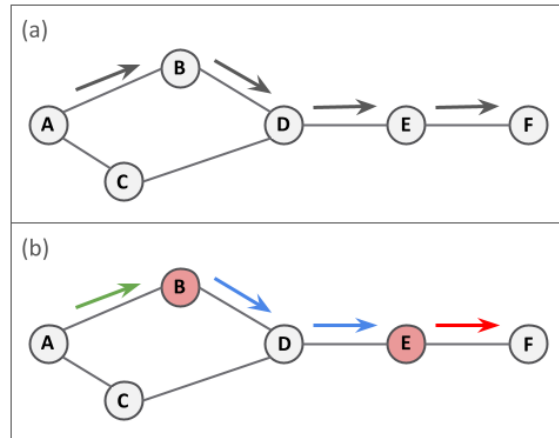


Figura 2.11: Caminho A-F. *a)* sem regeneradores *b)* dois regeneradores

Na Figura 2.11, um mesmo caminho conectando os nós A e F é mostrado em duas situações distintas. Na primeira, representada na Figura 2.11*a*, não é utilizado nenhum regenerador, portanto o caminho A-F é composto por um único segmento, no caso A-B-D-E-F. Na Figura 2.11*b*, pode-se observar que dois regeneradores foram alocados nos nós B e E respectivamente, criando um caminho composto pela concatenação de três segmentos: A-B, B-D-E e E-F. Analisando esta simples topologia, somente dois caminhos são possíveis entre os nós A e F. No entanto, qualquer combinação de posicionamento de regeneradores seria válida, assumindo que nenhum segmento formado fosse superior ao maior alcance das modulações disponíveis. Com isto, um total de 16 possíveis soluções para atender uma única demanda A-F teriam que ser testadas, mesmo numa topologia tão simples. Desta forma, fica claro que cenários com várias demandas e topologias mais complexas, compostas por mais nós e mais enlaces, geram um número muito grande de caminhos válidos. Por consequência, uma quantidade ainda maior de soluções possíveis a serem testadas de modo a não haver conflitos entre elas.

Adição de regeneradores visando economia de espectro

O único momento em que um canal de comunicação pode mudar sua frequência de espectro ou modulação é ao passar por um nó com regenerador. Como um regenerador

determina o final de um segmento, todos os enlaces dentro de um mesmo segmento devem utilizar a mesma frequência e modulação. Ao mudar de modulação, pode-se escolher a modulação mais eficiente para o próximo segmento e, por consequência, possivelmente reduzir o número de FSs necessários.

A introdução de regeneradores em nós intermediários passa a ter uma nova funcionalidade, além da própria regeneração do sinal em si. O uso de regeneradores pode proporcionar redução de uso de FSs. Vamos ilustrar esta situação através do cenário proposto na Figura 2.11. Assumindo que existe uma modulação cujo alcance máximo seja igual ou superior ao tamanho total do segmento A-B-D-E-F e que existem FSs disponíveis, esta seria a melhor solução a ser adotada, por não utilizar nenhum regenerador e, por consequência, reduzir o custo da solução. Por outro lado, supondo que um ou mais enlaces deste segmento estejam próximos da saturação, de modo que não haja mais FSs disponíveis para atender uma nova demanda, seria possível posicionar regeneradores em um ou mais nós intermediários, formando novos segmentos menores. Segmentos menores possivelmente permitirão utilizar modulações de menor alcance, mas de maior eficiência espectral, resultando em necessidade de menos FSs para atender a demanda e, por consequência, liberando espaço nos enlaces utilizados para que mais demandas possam ser atendidas.

2.2.3 Fragmentação

À medida que os FSs são alocados, nem sempre é possível evitar a ocorrência de “buracos” entre os demais FSs alocados para as demandas, resultando no fenômeno de fragmentação do espectro [14]. A fragmentação é causada principalmente por dois fatores:

1. A restrição de continuidade entre enlaces.
2. Conexões temporárias que, ao serem encerradas, liberam os FSs previamente utilizados, criando novos espaços vazios.

Conforme a fragmentação cresce, aumenta o desperdício de espectro, resultando em uma probabilidade cada vez maior de bloqueio.

A desfragmentação pode ser realizada em tempo real, a medida que novas demandas são atendidas, ou *offline*, por intermédio do uso de fibras auxiliares. Em se tratando de provisionamento de recursos para demandas em tempo real, algumas políticas de alocação dinâmicas objetivam minimizar o fenômeno da fragmentação, através de heurísticas variadas na escolha dos índices dos FSs a serem adotados, por exemplo, *First-Fit*, *Largest-First*, *Best-Fit*, *Random-Fit*, *Last-Fit*, *Least-Used*, *Most-Used*, *Exact-Fit* [14]. No caso da desfragmentação ser feita *offline*, todas as demandas já atendidas são avaliadas e remanejadas de forma a evitar fragmentação.

Esta nova disposição é configurada em uma fibra auxiliar, paralela à atualmente em uso. A comutação dos circuitos do caminho anterior para a nova fibra é realizada de forma instantânea, sem causar interrupção nos circuitos, liberando desta forma a fibra anteriormente utilizada.

2.2.4 Circuitos de proteção

As fibras ópticas estão sempre sujeitas a rupturas acidentais, desastres naturais ou até mesmo sabotagem, o que pode resultar na interrupção dos serviços. Para mitigar essas interrupções e, dependendo do SLA (*Service Level Agreement*) contratado na criação do circuito, é comum a implementação de circuitos de proteção. Diversas pesquisas têm sido realizadas na criação de circuitos que funcionam como *backups* para um ou mais circuitos principais.

No contexto das Redes Ópticas Elásticas, existem basicamente duas categorias de circuitos de proteção: circuitos dedicados ou *Dedicated Path Protection* (DPP) e circuitos compartilhados ou *Shared-Backup Path Protection* (SBPP)[15]. Dispositivos ópticos modernos são capazes de detectar a perda de sinal em um circuito e, praticamente de forma instantânea, realizar a comutação para um circuito de proteção. Isso ocorre de tal forma que as camadas superiores do sistema nem mesmo percebem a interrupção.

2.2.5 Exemplo cenário completo

Uma vez que os principais componentes em EON foram apresentados, será mostrado uma sequência completa de alocação de FSs para atender algumas demandas em um cenário bem simples, onde fica visualmente evidente a complexidade da solução do problema RMSA.

No exemplo, usaremos uma rede com topologia em linha, composta por 4 nós, conectados por 3 arestas, com suas respectivas distâncias definidas. Também será utilizado um conjunto fictício de modulações, conforme mostrado na Figura 2.12.

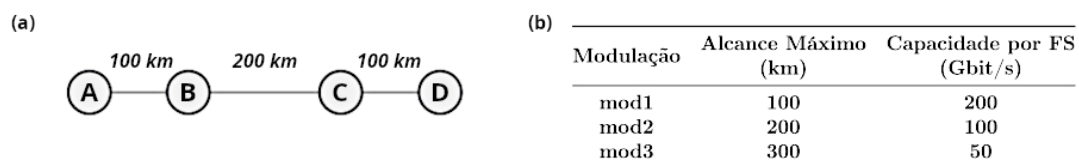


Figura 2.12: Cenário exemplo. a) topologia b) conjunto de modulações

A Figura 2.13 mostra uma possível sequência de alocação de FSs para atender um conjunto de 6 demandas.

Inicialmente, a demanda 1 requisita uma conexão de 300 Gbps, entre os nós A e B, cuja distância é de 100 km e que, portanto, pode ser atendida pela modulação

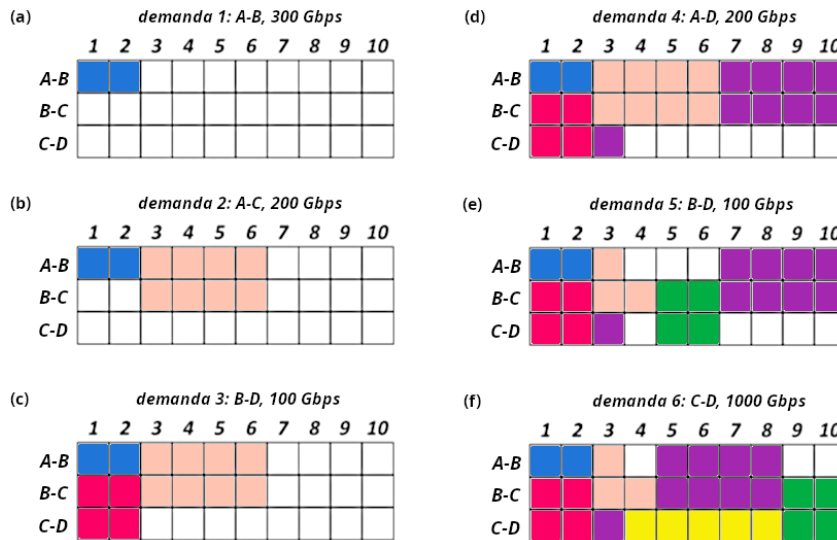


Figura 2.13: Exemplo de alocação de FSs

mod1. Como esta modulação tem capacidade de 200 Gbps por FS, serão necessários 2 FSs no enlace A-B para suportar os 300 Gbps requisitados, conforme pode ser visto na Figura 2.13a.

Em seguida, a demanda 2 requer uma conexão de 200 Gbps entre os nós A-C. A distância entre A e C é de 300 km, que pode ser alcançada sem necessidade do uso de regeneradores pela modulação *mod3*, cuja capacidade é 50 Gbps por FS. Utilizando esta modulação, serão necessários 4 FSs por enlace (Figura 2.13b) para atender esta demanda.

A demanda 3, de B para D com 100 Gbps, também irá utilizar a modulação *mod3*, mas necessitando de apenas 2 FSs por enlace.

A demanda 4, também de 200 Gbps, requer conexão entre A e D, cuja distância é de 400 km, maior do que a modulação de maior alcance. Desta forma, o uso de regenerador se torna obrigatório para atendê-la. Vamos supor que o regenerador seja colocado no nó C, formando dois segmentos: A-B-C e C-D. O segmento A-B-C tem 300 km no total, podendo ser atendido pela modulação *mod3*, usando 4 FSs. Já no segmento C-D, por ter passado por um regenerador, é possível alterar tanto os índices de FSs quando a modulação adotada. Como o segmento C-D tem apenas 100 km, podemos mudar para a modulação *mod1* e, opcionalmente, mudar o índice de FS.

Observe que neste instante, tanto os enlaces A-B quanto B-C estão sem FSs livres disponíveis, completamente saturados. Com isto, a demanda 5 de B para D não poderia ser atendida e, ao ser bloqueada, aumentaria o BP. Uma possível alternativa para liberar FSs nestes enlaces seria alterar a forma que a demanda 2 foi atendida, onde inicialmente foi adotada uma solução sem uso de regeneradores. Com objetivo específico de liberar FSs, podemos adicionar um regenerador em B para atender a

demanda 2, criando os segmentos A-B, de 100 km, e B-C, de 200 km. Usando as modulações *mod1* e *mod2*, respectivamente, seriam liberados 3 FSs no enlace A-B e 2 FSs no enlace B-C, possibilitando a aceitação da demanda 5, conforme observado na Figura 2.13e.

Finalmente, para atender a demanda 6, de 1000 Gbps, seriam necessários 5 FSs no enlace C-D. Embora existam 5 FSs disponíveis, eles não estão contíguos. Alterando os índices dos FS usados pela demanda 4 e pela demanda 5, obtém-se o espaço para atender a demanda 6.

Neste exemplo, foi possível observar a inserção de regeneradores para liberação de FSs, possibilitando o uso de modulações mais eficientes para os segmentos formados. Também foi visto um exemplo de mudança de índices dentro de uma mesma demanda. Adicionalmente, foi observado o problema de fragmentação, que neste caso foi resolvido apenas através de um rearranjo dos índices de FSs já alocados. Em cenários mais complexos, com topologias que possibilitam várias rotas alternativas entre um par de nós quaisquer e uma quantidade maior de demandas a serem atendidas, o número de possibilidades a serem testadas cresce exponencialmente.

2.3 Estado da arte

EON surgiu há pouco mais de uma década, apontado como uma mudança de paradigma fundamental em redes ópticas. Entretanto, ainda não se tornou um padrão de fato. Desafios ainda não completamente resolvidos, como a alocação de espectro eficiente, escolha da modulação mais eficiente levando considerando o comprimento dos enlaces, fragmentação, caminhos de proteção, utilização em conjunto com fibras ópticas *multi-core* e SDM etc., fazem com que EON ainda seja alvo de muitos estudos e pesquisa.

Trabalhos anteriores propuseram soluções para problemas semelhantes em redes WDM, embora bem mais simples, como RLP (*Regenerator Location Problem*) [8] [9] e RWA (*Routing and Wavelength Assignment*) [16] e, posteriormente, RSA (*Route and Spectrum Allocation*) [17][18]. RSA é *NP-hard*, conforme provado em WANG *et al.* 2011 [19] e estendido para RMSA em CHRISTODOULOPOULOS *et al.* 2011 [20] e WANG *et al.* 2012 [21].

Em relação ao problema RMSA, as propostas podem ser classificadas de diferentes formas. Em relação ao conhecimento prévio das demandas a serem atendidas, podemos classificar uma solução para RMSA em *offline* e *online*, às vezes também chamadas de estáticas ou dinâmicas, respectivamente. Soluções para RMSA *offline* tratam, como um todo, um conjunto de demandas previamente conhecido, normalmente com o objetivo de otimizar recursos utilizados. Por sua vez, soluções *online* RMSA tem que decidir em tempo real como atender as demandas uma a uma,

sequencialmente, sem conhecimento prévio das próximas, e tendo que manter intactos os recursos previamente alocados para as demandas anteriores. Nesta categoria, as propostas tentam normalmente reduzir a fragmentação ao definir como cada demanda será atendida. É importante observar que as duas formas de atender RMSA são igualmente relevantes. Soluções *offline* são essenciais para o projeto e dimensionamento da rede, além de poderem ser usadas para auxiliar na desfragmentação de redes em uso. Por melhor que seja a política adotada por uma solução *online*, a fragmentação forçosamente irá ocorrer, em maior ou menor escala, principalmente devido à restrição de contiguidade característica de EON ou circuitos temporários, liberados após um tempo de duração previamente especificado. A partir de algum critério de gatilho baseado em índices de fragmentação ou política de administração/gerenciamento da rede, é possível disparar um processo de desfragmentação. De posse das demandas atendidas até o momento, pode-se aplicar um modelo *offline* para remanejar os recursos da melhor forma possível e então aplicar as novas rotas numa infraestrutura física em paralelo. Após a instalação e configuração dos regeneradores, a comutação dos circuitos ativos para os novos circuitos configurados pode ser realizada de forma transparente, sem interrupção dos serviços.

Por ser considerado *NP-hard*, diferentes heurísticas foram propostas para RMSA. Inclusive, variantes de RMSA também são objeto de pesquisa, adicionando novas características ao problema, como proteção de circuitos, fibras multi-core, *fragment-aware* entre outros.

Ao longo dos anos, alguns levantamentos sobre o estado da tecnologia EON foram surgindo, sendo uns mais genéricos, outros mais específicos referentes a alguma determinada característica. Um dos primeiros estudos sobre o estado da arte em RSA foi CHATTERJEE *et al.* 2015 [17], até hoje bastante citado. SHIRIN ABKENAR e GHAFFARPOUR RAHBAR 2017 [22] apresentam um levantamento de algoritmos bem completo. Mais recentemente, o artigo de RUIZ *et al.* 2021 [23], em sua primeira parte, apresenta um levantamento mais atualizado para RMSA e, na sua segunda parte, propõe um novo algoritmo RMSA, chamado Multi-Path Best-Fit (MP-BF), que utiliza uma estratégia de divisão espectral por múltiplos caminhos, juntamente com política *Best-fit*.

Em se tratando de circuitos de proteção e sobrevivência em redes ópticas elásticas, KLINKOWSKI e WALKOWIAK 2012 [24] propõe uma formulação ILP (*Integer Linear Programming* ou programação linear inteira) e uma heurística para RSA *offline* voltado para DPP (*Data Path Protection*). WALKOWIAK e KLINKOWSKI 2013 [25] propõe também uma formulação ILP e duas novas heurísticas, além de outras heurísticas para RSA, todas voltadas para SBPP (*Shared-Backup Path Protection*). Outra proposta para proteção em EON é feita em MOGHADDAM *et al.* 2018 [26], que além de diferenciar o tipo de tráfego para prover o mecanismo de proteção

mais adequado, possui um esquema de reserva e agendamento. É proposta uma formulação ILP para RMSA para redes transparentes, além de heurísticas para reduzir a complexidade em redes maiores.

Um levantamento completo sobre técnicas para evitar a fragmentação do espectro foi realizado por CHATTERJEE *et al.* 2018 [27], fazendo um estudo comparativo da performance de diversos algoritmos, como, por exemplo, *First-Fit*, *Best-Fit* e *Exact-Fit*.

FUJII *et al.* 2014 [28] aborda o problema de interferência (*crosstalk*) em fibras multi-core (MCF) em EON, propondo um método de alocação *online* que reduz tanto o *crosstalk* quanto a fragmentação. Um levantamento recente sobre uso de SDM em conjunto com EON foi realizado por ÍTALO BRASILEIRO *et al.* 2020 [29], caracterizando um novo problema denominado *Routing, Modulation, Spectrum, and Core Allocation* (RMSCA).

Como esperado, o uso de inteligência artificial (IA) também é fruto de estudos em EON, e um levantamento foi realizado por MATA *et al.* 2018 [30].

MARKOVIĆ 2017 [31] apresenta uma abordagem meta-heurística para RSA *offline*, baseada em *Bee Colony Optimization* (BCO), visando minimizar tanto a utilização do espectro da rede quanto o comprimento médio do caminho. HAI *et al.* 2018 [32] apresenta método baseado em algoritmo genético (GA) para abordar o problema RSA combinado com formulação ILP.

VILLAMAYOR-PAREDES *et al.* 2023 [33] propõe uma solução RMSA, que considera simultaneamente o roteamento e permutação das demandas, através de algoritmo genético (GA), tanto para RMSA *offline* e *online*.

Em relação às soluções exatas para o problema RMSA *offline* em redes ópticas elásticas translúcidas, YILDIZ e KARAŞAN 2017 [4] apresenta uma formulação ILP eficiente, com o emprego de algoritmo *branch-and-price*. Sua formulação é voltada para segmentos, e não arestas, simplificando a escolha da modulação e cálculo dos FSs necessários para atender uma demanda em cada aresta. Faz corretamente a alocação e posicionamento de regeneradores, minimizando seu uso. No entanto, como principal deficiência, não atende às restrições de contiguidade e continuidade.

Nos capítulos seguintes, são mostradas as implementações preliminares realizadas até evoluírem para a formulação final do modelo RMSA-BP e os resultados obtidos.

Capítulo 3

RMSA-BP

Neste capítulo, é mostrado o modelo ILP na sua abordagem inicial, apontando as deficiências encontradas. Em seguida, é apresentada a solução RMSA-BPenum, uma outra forma desenvolvida para resolver o problema RMSA através de um algoritmo de *backtracking*. Mesmo não tendo apresentado escalabilidade para instâncias mais complexas, foram importantes na evolução para chegar ao modelo proposto. Por último, como resultado do aprendizado das implementações anteriores, é apresentada a formulação RMSA-BP, que mostrou ótimos resultados na resolução do problema.

3.1 Objetivos

Como relatado na Seção 2.3, os problemas RSA e RMSA, por serem peças fundamentais a serem resolvidas em EON, foram objeto de muita pesquisa na última década. No entanto, devido à sua característica *NP-hard* [19] [20], parte desses estudos fundamentou-se em heurísticas. Outras propostas que utilizam Programação Linear Inteira (ILP) não garantem soluções exatas. Ao dividir o problema RMSA em duas etapas (primeiro avaliando o roteamento e posicionamento de regeneradores, e depois a alocação de espectro), ou ao reduzir o número de caminhos analisados para atender a uma demanda usando *k-shortest path*, por exemplo, soluções exatas podem ser descartadas.

YILDIZ e KARAŞAN 2017 propõem uma interessante formulação ILP chamada RLP-FON [4], que resolve conjuntamente os problemas de roteamento, nível de modulação e localização de regeneradores. No entanto, apesar de respeitar o número de FSs máximo disponíveis em cada um dos enlaces de fibra óptica da rede (CE), não leva em consideração a alocação do espectro óptico em si, sem a definição dos índices de FSs nem respeitando as restrições de continuidade e contiguidade (CC). O RLP-FON não é, portanto, uma solução para o problema RMSA completo, e sim uma solução para RLP em EON, adicionando escolha de modulação. Todavia, o que torna RLP-FON particularmente interessante é a sua abordagem, orientada

para segmentos e não em arestas individualmente. Desta forma, obtém-se uma simplificação considerável no número de restrições referentes à escolha da melhor modulação a ser adotada em cada segmento e o cálculo do número de FSs utilizados em cada enlace. Em resumo, RLP-FON apresenta três grandes deficiências:

- Por não considerar os índices de FSs, não testa a existência de conflitos entre duas demandas quaisquer, ou seja, aceita soluções onde um mesmo índice possa ser usado por mais de uma demanda. A única garantia que oferece em relação ao uso do espectro é que a soma total de FSs usados por enlace não ultrapasse sua capacidade óptica.
- Da mesma forma, por não reconhecer os índices de FS, não atende as restrições de contiguidade e continuidade (CC).
- Não aceita o bloqueio de demandas, ou seja, ou encontra a solução ótima para todo o conjunto de demandas ou então considera a instância do problema inviável como um todo.

Até onde temos conhecimento, o modelo proposto neste trabalho é a primeira solução exata em EON do problema RMSA completo em redes translúcidas, para processamento *offline* de um conjunto de demandas, minimizando a taxa de bloqueio (BP) e atendendo a todas as restrições de continuidade e contiguidade. Além de definir a localização e alocação de regeneradores pelos diversos nós, de maneira semelhante às soluções propostas para RLP, busca também minimizar o número de regeneradores necessários para atender o conjunto de demandas aceitas. Por último, minimiza o uso do espectro óptico. Importante observar que é necessário um regenerador dedicado para cada fluxo a ser regenerado.

3.2 Formulação RMSA-BP inicial

Inicialmente, desenvolvemos uma formulação semelhante ao modelo apresentado em [4], mas atendendo às restrições de CC. Adicionalmente, introduzimos a funcionalidade de bloqueio de demandas. Nessa abordagem, demandas podem ser bloqueadas (não atendidas), caso não haja recursos disponíveis suficientes na rede. Esta nova funcionalidade, por si só, adiciona grande complexidade na solução do problema, ao ter que decidir quais demandas devem ser atendidas e quais devem ser bloqueadas.

Desta forma, o modelo proposto passa a ter uma função multiobjetivo: primeiramente maximizar o número de demandas atendidas, em seguida minimizar o número de regeneradores e, finalmente, minimizar número total de FSs utilizados por toda a solução.

Representamos a rede e sua topologia por um grafo não direcionado G com conjunto de vértices N e conjunto de arestas E , onde N é o conjunto de nós que compõem a rede G e E representa o seu conjunto de enlaces, sendo que cada enlace $e \in E$ tem seu comprimento especificado em Φ_e . Todos os enlaces da rede representada pelo grafo G possuem a mesma capacidade de uso de espectro, ou seja, capacidade óptica quantificada pelo número total de FSs comportados por cada enlace, que representamos por ξ . No conjunto de modulações M , cada $m \in M$ representa uma modulação, onde Θ_m é capacidade de transmissão por FS da modulação m e Λ_m o seu alcance máximo. Utilizamos P para denotar o conjunto de segmentos válidos em G . Consideramos um segmento válido todo segmento p possível em G , desde que exista uma modulação m com alcance Λ_m maior ou igual ao comprimento total de p . Para cada segmento $p \in P$, s_p representa o nó de origem do segmento p , assim como t_p o seu nó final. O conjunto de demandas a serem atendidas é representado por D , onde cada demanda $d \in D$ é caracterizada pelo nó de origem S_d , nó de destino T_d e a largura de banda requisitada B_d . Sendo G um grafo não direcionado, uma demanda do nó n_a para n_b é equivalente a uma demanda de n_b para n_a . O número de FSs necessários para atender à demanda d no segmento p é denotado por F_p^d , obtido através da divisão de B_d pela capacidade Θ_m por cada FS na modulação $m \in M$ mais eficiente para o segmento p . Assumindo que o conjunto de modulações M é ordenado em ordem crescente pela alcance Λ_m das modulações m que o compõe, entende-se por modulação mais eficiente como sendo a primeira modulação $m \in M$ cujo Λ_m seja igual ou superior ao comprimento total do segmento p . É importante lembrar que é possível que uma demanda d utilize mais de um segmento, com modulações distintas e índices de FS diferentes em cada um destes segmentos. Usamos R_{\max} para denotar o número máximo de regeneradores que podem ser utilizados por uma única demanda. A adoção de R_{\max} é justificada posteriormente. Cabe ressaltar que, apesar das arestas em E serem não direcionadas, cada segmento em P é inerentemente direcionado. Portanto, um enlace pode ser percorrido por um segmento em qualquer uma das duas direções do enlace.

Dado G , o conjunto P de segmentos é gerado considerando cada um dos $|N|$ nós e selecionando todo caminho simples que parte dele cujo comprimento seja menor ou igual ao maior Λ_m disponível em M . Para cada novo caminho selecionado, um novo segmento direcionado partindo do nó inicial é adicionado a P . No final, P conterá necessariamente tanto esse segmento quanto sua versão reversa. Ou seja, cada caminho não direcionado (mais curto ou não) em conformidade com o comprimento máximo permitido, terá sido considerado e dará origem a dois segmentos, um em cada uma das direções do caminho. Um exemplo é fornecido na Figura 3.1.

A topologia representada na Figura 3.1 é composta por 8 nós e 9 enlaces, com seus respectivos comprimentos apresentados em quilômetros. Utilizando, por exemplo,

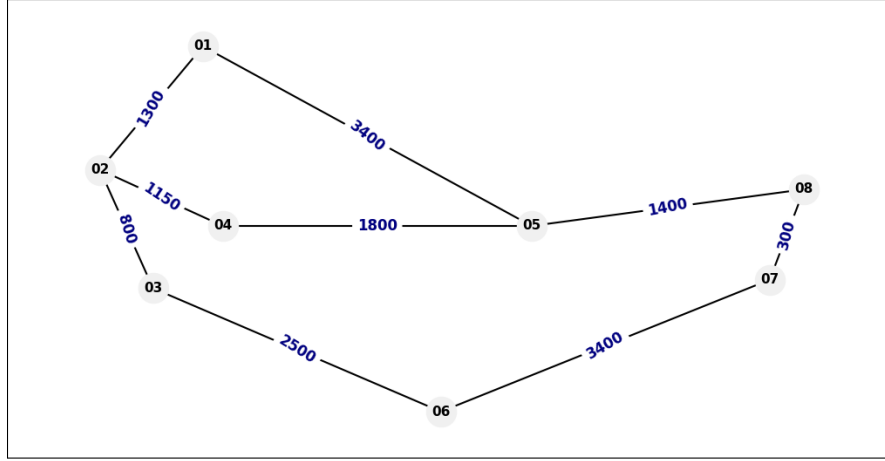


Figura 3.1: Exemplo de topologia para identificação de segmentos válidos

uma modulação m com $\Lambda_m = 4000\text{km}$, dos 186 segmentos encontrados na topologia, apenas 38 seriam viáveis: 1-2, 1-2-3, 1-2-4, 1-5, 2-3, 2-3-6, 2-4, 2-4-5, 3-2-4, 3-2-4-5, 3-6, 4-5, 4-5-8, 4-5-8-7, 5-8, 5-8-7, 6-7, 6-7-8, 7-8 e seus respectivos reversos. O Algoritmo 1 descreve o processo de geração do conjunto P .

Algorithm 1 Geração do conjunto de segmentos P

```

1: Entrada: Conjunto de modulações  $M$ , Grafo  $G$ .
2: Saída: Conjunto de segmentos válidos  $P$  e informação se  $e \in p$ .
3: Alcancemax ← Modulação de maior alcance em  $M$ 
4:  $P \leftarrow \emptyset$  // Inicializa o conjunto de segmentos válidos
5:  $Y \leftarrow \{\}$ . // Inicializa o dicionário  $Y$  para a tupla( $e,p$ )
6: for  $i$  em nós de  $G$  do
7:   for  $j$  em nós de  $G$  do
8:     if  $i \neq j$  then
9:       for caminho em todos os caminhos entre  $i$  e  $j$  do
10:        if comprimento total de caminho  $\leq$  Alcancemax then
11:          Adiciona caminho em  $P$  // Adiciona segmento válido em  $P$ 
12:          for  $e$  em enlaces de  $G$  do
13:             $Y(e,p) \leftarrow e \in p$  // True ou False, se  $e \in p$ 
14:          end for
15:        end if
16:      end for
17:    end if
18:  end for
19: end for
20: Retorna:  $P, Y$ 

```

Para cada $p \in P$ e cada $e \in E$, usamos Y_p^e para indicar se o enlace e faz parte do segmento p . Em caso afirmativo, a direção de travessia de e por p é dada implicitamente pelos nós s_p e t_p . Os Y_p^e s são constantes binárias cujos valores (de 0, 1) são atribuídos juntamente com a determinação do conjunto de segmentos P da rede, conforme detalhado no Algoritmo 1.

Duas variáveis binárias de decisão são utilizadas no modelo. A variável binária a_d , para cada $d \in D$, indica se a demanda d foi bloqueada ou admitida, recebendo os valores 0 e 1, respectivamente. A variável binária x_p^d , para cada $d \in D$ e cada $p \in P$, indica se a demanda d usa o segmento p .

Com o objetivo de simplificar a formulação visualmente, algumas abreviações foram adotadas:

$$R_d = \sum_{p \in P} x_p^d - a_d \quad (D1)$$

$$F_d^e = \sum_{p \in P} F_p^d x_p^d Y_p^e \quad (D2)$$

$$F_d = \sum_{e \in E} F_d^e \quad (D3)$$

Nessas equações, em (D1) definimos R_d como o número de regeneradores usados pela demanda d . Este número pode ser diretamente obtido pelo número de segmentos utilizados pela demanda d , subtraído de 1. Quando a_d igual a 0, significando que a demanda d foi bloqueada, todas as variáveis x_p^d também devem ser iguais a zero. Em (D2), F_d^e representa o número de FSs usados pela demanda d no enlace e . Em (D3), F_d simboliza o total de FSs usados pela demanda d como um todo. Desta forma, nossa formulação, já incluindo BP, mas ainda sem as restrições CC, fica assim:

$$\max \sum_{d \in D} (w_1 a_d - w_2 R_d - w_3 F_d)$$

s.t.

$$x_p^d \leq a_d \quad \forall d \in D, p \in P \quad (C1)$$

$$\sum_{\substack{p \in P \\ s_p = n}} x_p^d - \sum_{\substack{p \in P \\ t_p = n}} x_p^d = \begin{cases} a_d & \text{if } n = S_d \\ -a_d & \text{if } n = T_d \\ 0 & \text{otherwise} \end{cases} \quad \forall d \in D, n \in N \quad (C2)$$

$$R_d \leq R_{\max} \quad \forall d \in D \quad (C3)$$

$$\sum_{d \in D} F_d^e \leq \xi \quad \forall e \in E \quad (C4)$$

$$x_p^d \in \{0, 1\} \quad \forall d \in D, p \in P$$

$$a_d \in \{0, 1\} \quad \forall d \in D$$

Usamos pesos $w_1 \gg w_2 \gg w_3$ na função objetivo apenas para simbolizar que os três objetivos do problema devem ser priorizados na seguinte ordem: 1) Maximizar o número de demandas admitidas; 2) Minimizar o número total de regeneradores usados; 3) Minimizar o número total de FSs usados. A técnica específica que foi

utilizada para impor essa priorização é discutida no Capítulo 4.

Na Restrição (C1) é garantido que, se a demanda d for bloqueada, nenhum segmento $p \in P$ será usado por ela. A Restrição (C2) garante a conservação do fluxo, ou seja, ao atender uma demanda d , o número de fluxos que entram num determinado nó tem que ser igual ao número de fluxos que saem deste mesmo nó, exceto para o nó de origem da demanda (S_d) e seu nó de destino (T_d), onde há apenas um fluxo de saída e um fluxo de entrada, respectivamente. Se cada uma das somatórias que aparecem na definição de (C2) for igual a no máximo 1, então a diferença entre os somatórios estará no intervalo $\{-1, 0, 1\}$. Portanto, as Restrições (C1) e (C2) tratam de como as variáveis de decisão a_d e os x_p^d estão relacionadas. Uma demanda bloqueada d ($a_d = 0$) não utiliza nenhum segmento ($x_p^d = 0$ para todo $p \in P$). Já uma demanda d admitida ($a_d = 1$), por outro lado, deve implicar em um deficit de fluxo unitário positivo em seu nó de origem S_d , um deficit de fluxo unitário negativo em seu nó de destino (T_d) e nenhum deficit de fluxo em qualquer outro nó. Além disso, as Restrições (C3) e (C4) garantem, respectivamente, que nenhuma demanda use mais do que R_{\max} regeneradores e que, consideradas em conjunto, todas demandas usando o enlace e (d tal que $x_p^d Y_p^e = 1$ para algum $p \in P$) não utilizem mais do que a capacidade ξ disponível por enlace.

Para atender aos requisitos de continuidade e contiguidade (CC), é necessário adicionar um novo conjunto de variáveis inteiras, z_d^e , com valores válidos $\{0, 1, \dots, \xi\}$, indicando, se maior que zero, o índice do primeiro FS usado pela demanda d no enlace e , para $d \in D$ e $e \in E$. Além disso, é definida a constante I_p como sendo o primeiro enlace do segmento $p \in P$. Novamente, fazemos uso de atalhos, visando facilitar a visualização e entendimento da formulação.

$$X_d^e = \begin{cases} 1 & \text{if } \sum_{p \in P} x_p^d Y_p^e > 0 \\ 0 & \text{otherwise,} \end{cases}$$

$$O_{d,d'}^e = \begin{cases} 1 & \text{if } z_d^e \leq z_{d'}^e \leq z_d^e + F_d^e - 1 \\ 0 & \text{otherwise.} \end{cases}$$

Nos atalhos criados, X_d^e indica se a demanda d utiliza o enlace e , e $O_{d,d'}^e$ indica se o primeiro FS usado pela demanda d' no enlace e é um dos FSs usados pela demanda d no mesmo enlace e , caracterizando uma superposição de espectro entre duas demandas, fato que não pode ocorrer. Claramente, $X_d^e = 0$ se e somente se $F_d^e = 0$. Os atalhos X_d^e e $O_{d,d'}^e$ podem ser utilizados para impor restrições sobre os valores de z_d^e 's, de modo a garantir o cumprimento dos requisitos de CC. Com isso, podemos adicionar as restrições para CC ao modelo.

$$X_d^e \leq z_d^e \leq X_d^e \xi \quad \forall d \in D, e \in E \quad (\text{C5})$$

$$z_d^e + F_d^e - X_d^e \leq \xi \quad \forall d \in D, e \in E \quad (\text{C6})$$

$$(X_d^e Y_p^e z_d^e - X_d^{I_p} Y_p^{I_p} z_d^{I_p}) x_p^d = 0 \quad \forall d \in D, p \in P, \quad (\text{C7})$$

$$e \in E, e \neq I_p$$

$$\sum_{\substack{d, d' \in D \\ d \neq d'}} O_{d, d'}^e = 0 \quad \forall e \in E \quad (\text{C8})$$

A Restrição (C5) é utilizada para garantir que z_d^e não seja um índice de FS válido ($z_d^e = 0$) quando a demanda d não utiliza o enlace e ($X_d^e = 0$). Se a demanda d utiliza o enlace e ($X_d^e = 1$), a mesma Restrição (C5) garante que seu valor esteja no intervalo $\{1, \dots, \xi\}$. Por sua vez, a Restrição (C6) garante que o valor de z_d^e seja tal que os índices dos demais FSs usados pela demanda d ($F_d^e - 1$) não sejam superiores a ξ . Quando a demanda d utiliza o enlace $e \neq I_p$ no segmento p ($X_d^e Y_p^e = X_d^{I_p} Y_p^{I_p} = 1$), a Restrição (C7) garante a continuidade ao impor $z_d^e = z_d^{I_p}$. Finalmente, a Restrição (C8) garante que não existam conflitos entre duas demandas quaisquer, evitando que um mesmo FS seja alocado para mais de uma demanda. Pela Restrição (C8), não pode haver duas demandas distintas d, d' tais que $z_{d'}^e \in \{z_d^e, \dots, z_d^e + F_d^e - 1\}$ para qualquer enlace e . Vale ressaltar que a Restrição (C8) também serve para garantir a contiguidade nos enlaces. Para provar que também é suficiente, suponha um par de demandas distintas d, d' e um enlace e tal que $F_d^e, F_{d'}^e > 0$ e $z_{d'}^e \in \{z_d^e, \dots, z_d^e + F_d^e - 1\}$ (ou seja, a contiguidade não se mantém para d em e). Então $O_{d, d'}^e = 1$ e a Restrição (C8) é violada para o enlace e . Fica evidente que os F_d^e FS, a partir de z_d^e inclusive, ficam reservados para uso exclusivo da demanda d .

Quando aplicadas, as Restrições (C5)–(C8) englobam a Restrição (C3), que limita o número total de FSs usados em cada enlace e . A única forma da Restrição (C3) ser violada seria no caso de pelo menos um FS ser alocado para mais de uma demanda, o que violaria a Restrição (C8). Logo, a Restrição (C3) torna-se desnecessária quando as restrições de CC são aplicadas ao modelo.

3.2.1 Primeiros testes e observações

Os primeiros experimentos para validação realizados com o modelo utilizaram uma topologia bem simples, criada especificamente para os testes, como mostrado na Figura 3.2a. Posteriormente, também foram utilizadas topologias acadêmicas conhecidas como NFSNET [34], Cost239 [35] e Cost266 [36] para validar o modelo, mostradas nas Figuras 3.2(b), 3.2(c) e 3.2(d), respectivamente.

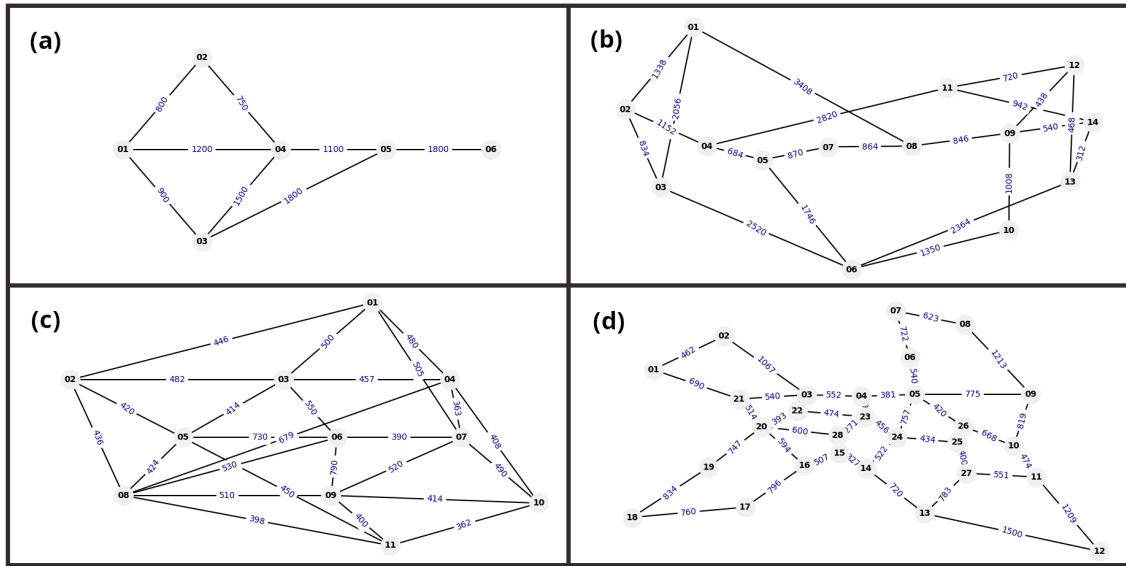


Figura 3.2: Topologias: (a)TopoX (b)NFSnet (c) Cost239 (d) Cost266

Nestes testes, ficou claro que a abordagem por segmentos ao invés de enlaces pode reduzir o problema consideravelmente.

Tabela 3.1: Segmentos viáveis para modulação com alcance 4000 km

Topologia	Nós	Arestas	Segmentos possíveis	Segmentos viáveis
TopoX	6	8	140	80
NFSnet	14	21	14226	300
Cost239	11	25	140040	78248
Cost266	28	41	868914	11960

A Tabela 3.1 mostra as características de cada topologia utilizada nos testes, no que diz respeito ao seu número de nós, enlaces e segmentos. O número de segmentos viáveis formando o conjunto de segmentos P é calculado através do Algoritmo 1, utilizando uma modulação de alcance 4000 km. Obviamente, quanto menor for o maior alcance das modulações Λ_m $m \in M$, menor será o número de segmentos válidos.

Os testes foram divididos em duas etapas: sem considerar as restrições de CC e com as restrições CC. Na fase de testes “sem CC”, todas as instâncias rodaram rapidamente, independente da topologia utilizada e do número de demandas. Como esperado, o número de demandas em D tem influência no tempo de processamento. A medida que mais demandas são atendidas, a solução da instância vai se tornando mais complexa, pois os enlaces vão ficando mais cheios e, ao se aproximarem da saturação, fica cada vez mais difícil a aceitação de novas demandas. Pequenas alterações no modelo não apresentaram melhorias significativas de performance.

A fase de testes “com CC” se mostrou efetiva para topologias simples, como a mostrada na Figura 3.2a. No entanto, utilizando topologias mais complexas, como NFSnet (Figura 3.2b) ou Cost239 (Figura 3.2c), os tempos aumentaram consideravelmente.

3.3 RMSA-BPenum

Após os testes com a formulação apresentada na Seção 3.2 e com o conhecimento adquirido das características do problema, foi feita uma nova abordagem, totalmente distinta da anterior, denominada RMSA-BPenum. RMSA-BPenum utiliza a técnica de *backtracking*, usada para encontrar soluções possíveis para um determinado problema, testando de forma sistemática todas as alternativas previamente enumeradas, selecionando as combinações que aparentam ser promissoras.

3.3.1 Rotas pré-processadas

A primeira avaliação feita foi verificar a viabilidade de se obter, previamente, todas as rotas possíveis entre a origem (S_d) e destino (T_d) para atender cada demanda $d \in D$. Para cada rota, todas as combinações de posicionamento de regeneradores seriam previamente calculadas. Obviamente, isto poderia se tornar inviável, dependendo diretamente do tamanho e da complexidade da topologia, além do número de demandas em D . Uma alternativa seria limitar o número de saltos para atender uma demanda. Em algumas propostas para RMSA, somente *k-shortest paths* são levados em consideração, com objetivo de reduzir o tamanho das variáveis e, por consequência, a complexidade do problema. No entanto, desta forma não seria garantida a solução exata, uma vez que caminhos válidos podem não estar sendo considerados. Por outro lado, limitar o número de regeneradores para atender uma demanda é absolutamente válido em cenários realistas. Como citado na Seção 2.2, regeneradores são dispositivos de custo elevado. Para o provedor de acesso, o uso indiscriminado de regeneradores para atender uma única demanda inviabiliza comercialmente o negócio. Desta forma, incluir o parâmetro R_{\max} para limitar o máximo de regeneradores por demanda, além de manter a otimalidade da solução e mantê-la coerente com um cenário real, viabiliza o pré-cálculo de rotas e posicionamento de regeneradores, como mostrado no Capítulo 4.

3.3.2 Alocação do espectro

Para cada demanda, é possível então identificar um conjunto de soluções, ao avaliar todas as rotas possíveis que podem ser usadas entre S_d e T_d e, dentro de cada rota, os possíveis posicionamentos de regeneradores em nós intermediários. Com estes

dados calculados, podemos automaticamente extrair diversas informações referentes a cada solução encontrada e úteis na solução geral do problema:

- Número de regeneradores usados em cada solução;
- Segmentos formados e seus comprimentos totais;
- Modulação mais eficiente para cada um destes segmentos;
- Número de FSs usados em cada enlace que compõem cada segmento, obtido através da divisão da banda requerida da demanda d pela capacidade de transmissão por FS da modulação adotada no segmento;
- Número total de FSs utilizados por cada solução da demanda d .

De posse destas informações, já é possível resolver o problema RMSA, mas ainda sem atender as restrições CC.

Com o objetivo de implementar as restrições de alocação de espectro, foi desenvolvida uma estrutura de dados, mostrada na Figura 3.3. Esta estrutura é formada, entre outras informações, por um mapa de bits. Cada bit deste mapa representa um dos ξ FSs de cada enlace, sendo atribuído o valor 1 ao bit quando o FS do enlace correspondente está sendo usado na solução.

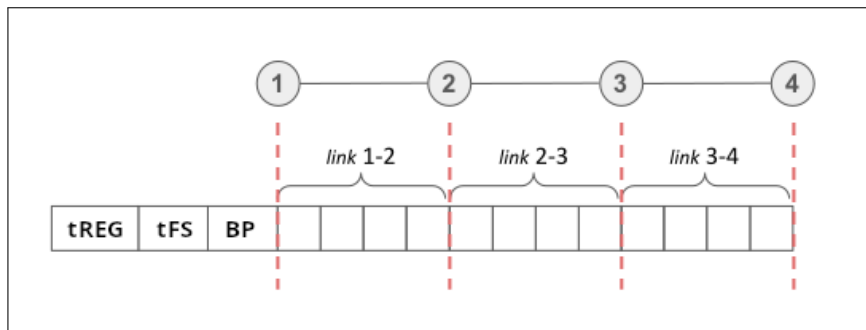


Figura 3.3: Estrutura de uma solução para uma demanda

Para facilitar o entendimento, novamente usaremos como cenário exemplo uma topologia simples em linha, composta por 4 nós e 3 enlaces, cada enlace com $\xi = 4$ FSs. A estrutura mostrada na Figura 3.3 é composta por 4 campos:

1. número de regeneradores usados na solução (tREG)
2. número total de FSs usados na solução (tFS)
3. 1 se demanda bloqueada, 0 se aceita (BP)
4. mapa de bits representando ξ FSs de cada enlace

Portanto, para cada $d \in D$, uma lista contendo todas possíveis soluções \mathcal{S}_d é criada. Adicionalmente, é inserida ao final desta lista uma solução com $BP = 1$ e todos os outros campos valendo 0, inclusive o mapa de bits, representando o caso da demanda não ser admitida. As soluções da lista completa \mathcal{S}_d são ordenadas do menor para o maior custo, calculado em função dos campos BP , $tREG$ e tFS e seus respectivos pesos ($\text{custo} = w_1 BP - w_2 tREG - w_3 tFS$). A lista de soluções para uma demanda d é então representada por $\mathcal{S}_d = \{\mathcal{S}_d^1, \mathcal{S}_d^2, \mathcal{S}_d^3, \dots, \mathcal{S}_d^n\}$, onde \mathcal{S}_d^1 é a solução de menor custo e \mathcal{S}_d^n representa a de maior custo, ou seja, o bloqueio da demanda d . No cenário exemplo, uma lista \mathcal{S}_d criada para uma demanda d , onde $S_d = 1$ e $T_d = 3$, é mostrada na Figura 3.4.

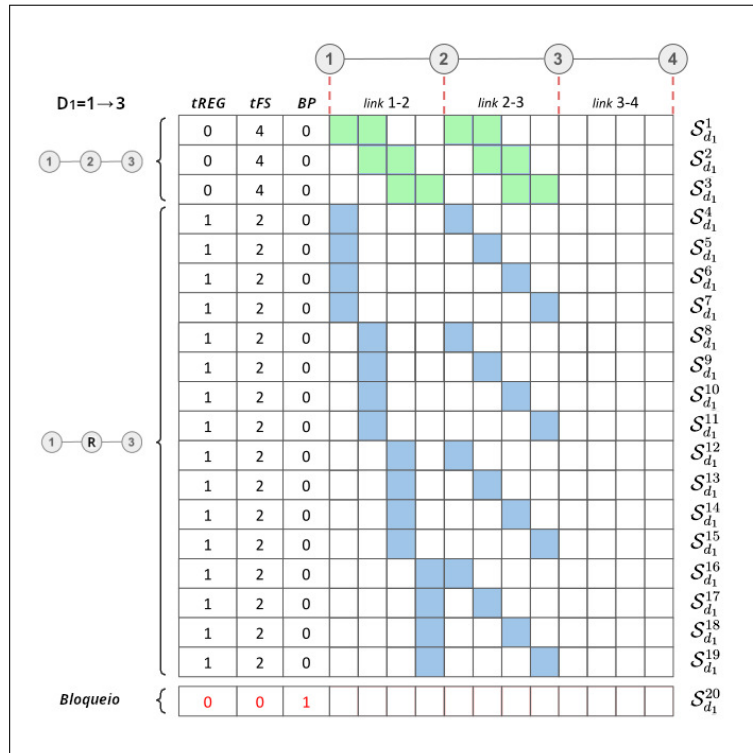


Figura 3.4: Conjunto de soluções para uma demanda

A demanda $d = (1, 3)$ possui uma única rota, passando pelo nó intermediário 2. Neste exemplo, vamos supor que exista uma modulação que comporte o segmento 1–2–3 sem o uso de regeneradores e que necessite de 2 FSs por enlace para atender a demanda d . As três primeiras soluções têm o mesmo custo, apenas diferem nos índices dos FSs alocados. É importante ressaltar que as restrições CC já são implicitamente contempladas durante o pré-processamento. Posicionando um regenerador no nó 2, passamos a ter 2 segmentos (1–2 e 2–3, respectivamente). Ainda a título de ilustração, vamos supor que exista uma modulação que necessite apenas de um único FS em cada um dos segmentos criados. Observe que todas as combinações possíveis de índices de FSs nos dois segmentos são representadas no mapa de bits. Vale ressaltar a inclusão da última solução em \mathcal{S}_d , representando o bloqueio da demanda

d .

O mesmo procedimento é realizado para todas as demandas, gerando uma matriz com todas as soluções, como mostrado na Figura 3.5. Observe que o número de soluções pode variar por demanda.

d_1	$sol_{d_1}^1$	$sol_{d_1}^2$...	$sol_{d_1}^{ \mathcal{S}_1 }$
d_2	$sol_{d_2}^1$	$sol_{d_2}^2$...	$sol_{d_2}^{ \mathcal{S}_2 }$
\vdots				
d_n	$sol_{d_n}^1$	$sol_{d_n}^2$...	$sol_{d_n}^{ \mathcal{S}_n }$

Figura 3.5: Tabela de soluções possíveis das demandas 1 a n

Com essa estrutura montada, resta escolher, para cada demanda d , uma única solução $\sigma_d^x \in \mathcal{S}_d$, onde $x = \{1, \dots, |\mathcal{S}_d|\}$. Esta escolha deve ser feita de tal maneira que minimize a soma dos custos associados a cada solução escolhida para cada demanda.

Outra condição para a escolha das soluções para cada demanda é que não haja conflito entre os FSs escolhidos. Isso pode ser alcançado por meio da operação lógica “E” (AND) entre todos os mapas de bits das soluções selecionadas, par a par. Um resultado igual a 0 indica a ausência de conflitos entre as demandas.

3.3.3 Algoritmos de *Backtracking*

Varrer toda as combinações possíveis de soluções seria inviável em termos computacionais. Portanto, para atingir esse objetivo, foi desenvolvido um algoritmo de *backtracking* para percorrer a árvore de soluções, de forma mais eficiente.

Algoritmos de *backtracking* pertencem à uma classe de algoritmos cujo foco é voltado para explorar eficientemente uma tabela formada por todas as possíveis soluções para um determinado problema [37]. Essa classe de algoritmos somente é aplicável se o problema permitir o conceito de solução candidata parcial; em uma matriz com n linhas, deve ser possível avaliar o custo parcial de k primeiras linhas. Desta forma, se a solução parcial for promissora, ou seja, se há a possibilidade de se encontrar uma solução válida que seja melhor do que a melhor solução completa encontrada até o momento, a busca deve prosseguir. Porém, se a solução parcial nas k primeiras linhas já for pior do que a melhor solução encontrada até o momento considerando as n linhas da tabela, não vale a pena continuar a pesquisa permanecendo nesse ramo. Dessa forma, reduz-se significativamente o número de testes realizados em comparação com algoritmos de enumeração por força bruta, por exemplo.

Portanto, um algoritmo de *backtracking* deve ir atendendo a todas as restrições entre as soluções adotadas a medida que percorre o caminho candidato sendo testado. Mas, além disso, deve ser capaz de retroceder sempre que verifica que as escolhas anteriores irão conduzir a uma solução inválida ou sub-ótima. Dessa maneira, o algoritmo pode explorar eficientemente o espaço de solução, eliminando ramos que não levarão a soluções válidas ou interessantes.

Algoritmos de *backtracking* são muitas vezes implementados empregando o conceito de recursividade. A implementação necessita ser customizada para o problema específico a ser resolvido, envolvendo a definição das restrições e uma forma de calcular o custo parcial de caminhos candidatos sendo analisados.

3.3.4 Implementação RMSA-BPenum com *backtracking*

No nosso caso, a estrutura gerada no pré-processamento e mostrada na Figura 3.3 é perfeitamente adequada para *backtracking*. Os algoritmos 2 e 3 mostram o seu funcionamento.

Algorithm 2 RMSA-BPenum (main)

Monta toda a árvore de soluções e busca melhor solução

```

1: Entrada:
2:   D(demandas), G(topologia), M(modulações),  $\xi$ (FS por link),  $R_{\max}$ 
3: Saída:
4:   Solução exata para o conjunto de demandas  $D$ 
5:
6:    $S \leftarrow \text{ACHASOLUCOES}(D,G,M,\xi,R_{\max})$  // Pré-processamento
7:
8:    $BP, REG, FS \leftarrow \infty$  // Inicializa variáveis da melhor solução
9:    $d \leftarrow 1$  // Demanda de entrada na árvore
10:   $s \leftarrow 0$  // Índice da solução de entrada para demanda  $d$ 
11:
12:  while  $s \neq$  última solução em  $S_d$  do
13:     $s \leftarrow s + 1$ 
14:     $AlocFS \leftarrow 0$  // Mapa de ( $\xi \times$  no. enlaces) bits
15:     $pBP, pREG, pFS \leftarrow 0$  // Inicializa variáveis solução parcial
16:     $\text{RMSA-BPENUM}(d, s, AlocFS, pBP, pREG, pFS)$ 
17:  end while
18:
19:  display BP, REG, FS

```

Algorithm 3 RMSA-BPenum (resolvedor).

Busca melhor solução na árvore recursivamente com *backtracking*

```
1: /* Cálculo do custo em função de BP, REG e FS */
2: function CUSTO(bp, reg, fs)
3:   return ( $w1 \times bp - w2 \times reg - w3 \times fs$ )           // pesos  $w1, w2, w3$ 
4: end function
5:
6: /* Procedure recursiva para varrer toda a árvore */
7: /* Interrompe a descida caso não valha a pena */
8: procedure RMSA-BPENUM(d, s, AlocFS, pBP, pREG, pFS)
9:   if  $AlocFS \wedge MapaBits(s) == 0$  then           // Sem conflitos com  $s$ 
10:    /* Adiciona o custo de  $s$  ao total parcial */
11:     $(tBP, tREG, tFS) \leftarrow (pBP, pREG, pFS) + (BP_s, REG_s, FS_s)$ 
12:
13:    if  $CUSTO(tBP, tREG, tFS) < CUSTO(BP, REG, FS)$  then
14:      if  $d$  é a última demanda then           // NOVA MELHOR SOLUÇÃO
15:         $(BP, REG, FS) \leftarrow (tBP, tREG, tFS)$ 
16:      else
17:        /* Continua a busca */
18:         $AlocFS' \leftarrow AlocFS \vee MapaBits(s)$            // Acrescenta FSs de  $s$ 
19:         $d' \leftarrow d + 1$            //  $d'$  = próxima demanda
20:         $s' \leftarrow 0$            //  $s'$  = primeira solução de  $d'$ 
21:        while  $s' \neq$  última solução em  $S_{d'}$  do
22:           $s' \leftarrow s' + 1$ 
23:          RMSA-BPENUM( $d', s', AlocFS', tBP, tREG, tFS$ )
24:        end while
25:      end if
26:    else
27:      // Custo parcial é igual ou pior do que melhor Custo encontrado
28:      return
29:    end if
30:  else
31:    // Solução  $s$  conflita com soluções anteriores
32:    return
33:  end if
34: end procedure
```

O algoritmo é inicializado atribuindo ∞ ao custo da melhor solução válida até o momento, representado por BP , REG e FS e um mapa de bits representando FS em uso ($AlocFS$) totalmente zerado. A busca pela tabela começa na primeira demanda ($d = 1$), variando a solução s no intervalo $s = \{1, \dots, |S_1|\}$. Para cada uma das soluções de d , avaliam-se as soluções da próxima demanda ($d'=d+1$). Ao avaliar cada solução, verifica-se se há conflito entre o $AlocFS$ e mapa de bits da solução atual em d' . Se existir conflito, volta para d e avança para a próxima solução. Caso não exista conflito, avalia-se o custo parcial até este ponto, obtido pela adição do custo acumulado até a demanda anterior mais o custo da solução em d' sendo avaliada. Se

esse custo parcial for igual ou superior ao melhor custo encontrado até o momento, não vale a pena prosseguir, voltando para a próxima solução de d . Caso o custo parcial seja menor, existem duas possibilidades. No primeiro caso, se d' for a última demanda de D , uma nova melhor solução foi encontrada. Neste caso, atualizamos BP , REG e FS e voltamos para a próxima solução da demanda anterior. Já no segundo caso, ainda existem demandas a serem avaliadas, então atualizamos o estado atual e continuamos a busca na primeira solução de $d'' = d' + 1$. Este processo deve continuar até alcançar a última demanda. Toda vez que chegamos a última solução de uma demanda, prosseguimos na próxima solução da demanda anterior. Este procedimento se repete até ter sido esgotada a avaliação de todas as soluções da primeira demanda.

Observe que o algoritmo descrito é totalmente adequado ao uso de recursividade. Como cada linha da matriz está ordenada de menor custo para o maior, as melhores soluções são encontradas antecipadamente, fazendo com que os cortes sejam cada vez mais eficazes. A inclusão de uma última solução em cada demanda representando o bloqueio da própria, faz que não seja necessário nenhum tratamento diferenciado para o caso de uma demanda ser bloqueada, ou seja, é como se fosse uma solução que nunca gera conflitos com outras demandas, mas com um custo muito elevado.

3.3.5 Avaliação RMSA-BPenum

A implementação de uma solução do problema RMSA sem utilizar nenhum *solver*, apenas por enumeração, foi de grande importância para o desenvolvimento da proposta da formulação RMSA-BP final e já é, por si só, uma contribuição deste trabalho.

Esta implementação serviu para comprovar que o pré-processamento para encontrar os segmentos válidos e os caminhos possíveis entre dois nós de uma topologia é viável. Isto só se tornou possível devido à limitação do número de regeneradores por demanda R_{\max} . Mas mesmo que este processo fosse demasiadamente demorado para se adotar em tempo real, seria possível gerar, em processamento *offline*, um arquivo contendo todas as rotas já pré-calculadas, específico para cada topologia e conjunto de modulações. Dessa maneira, a fase de pré-processamento poderia se limitar unicamente à leitura desse arquivo.

Já o pré-processamento de todas as combinações de alocação de espectro para cada solução se mostrou muito dispendiosa em termos de recursos computacionais, tanto memória quanto processador, chegando a ultrapassar 128 GB de RAM em certos casos. Obviamente, o número de FSs por enlace (ξ) afeta diretamente a complexidade do problema e tempo de processamento.

Concluiu-se que, para $\xi > 40$, o método torna-se muito custoso, tanto no pré-

processamento quanto na pesquisa da solução. O número de demandas claramente também impacta diretamente no tempo de solução.

Por outro lado, a implementação foi extremamente útil em outros aspectos. Primeiramente, serviu para validar os resultados apresentados pela formulação ILP original, através da comparação das soluções obtidas usando ambos os métodos.

Também serviu como mais uma forma de reconhecer que RMSA é *NP-hard*. O problema *Disjunctively Constrained Knapsack Problem* (DCKP) é uma variante do clássico problema da mochila [38][39]. Nesta variante, cada item não apenas possui um valor e um peso, mas também está sujeito a uma restrição adicional na qual determinados itens são incompatíveis entre si, ou seja, itens mutuamente exclusivos, impedindo sua coexistência na mochila simultaneamente. DCKP é classificado como *NP-hard*[38]. Ao examinar a estrutura das soluções de um conjunto de demandas no problema RMSA, podemos conceber cada solução como um possível item a ser colocado na mochila, com seu respectivo custo. A lista de itens incompatíveis com a solução σ_d^x seria composta por todas as outras soluções da própria demanda d , de modo a garantir uma única solução por demanda, e por todas as soluções de demandas diferentes de d cujo mapa de bits entre em conflito com o da solução σ_d^x .

3.4 RMSA-BP

Esta seção mostra a formulação RMSA-BP, na sua versão final. O conhecimento adquirido com as implementações anteriores, tanto utilizando a formulação inicial, quanto a implementação RMSA-BPenum, contribuíram de forma definitiva para se chegar a esta formulação.

A formulação inicial apresentada na Seção 3.2 foi essencial para compreensão completa do problema RMSA, em relação a como cada um dos seus objetivos (minimizar BP, minimizar regeneradores e minimizar utilização do espectro) impactam na sua solução. Além disso, tornou possível entender como a complexidade da topologia, o número de demandas e a saturação dos enlaces influenciam no tempo de processamento da solução.

Nos experimentos realizados, ficou claro que a introdução das Restrições (C5)-(C8), relacionadas a contiguidade e continuidade, exerceu grande influência nos recursos gastos com processador e memória da máquina. Também foi observado que a Restrição (C2), responsável pela conservação de fluxo, poderia ser simplificada e melhorada.

A implementação do algoritmo RMSA-BPenum mostrou a viabilidade de se calcular previamente todos os caminhos, posicionamento e combinações possíveis de regeneradores para atender uma demanda.

Embora a formulação inicial estivesse correta, funcionando perfeitamente em

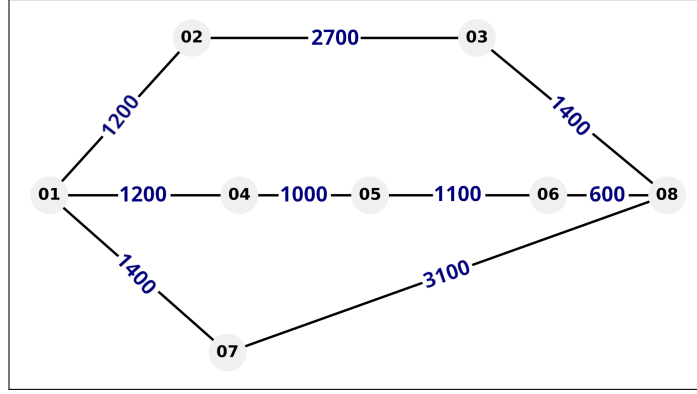


Figura 3.6: Topologia exemplo pré-processamento

topologias mais simples, em certas instâncias rodando com topologias mais complexas, podia demorar excessivamente ou mesmo nem resolver dentro de um período de tempo aceitável, mesmo com um número de demandas reduzido. Diversas pequenas melhorias de implementação ou alterações na formulação não produziram um aumento de performance significativo.

Desta forma, decidiu-se por alterações mais relevantes na formulação, tendo como resultado uma diferença considerável em termos de desempenho, bem como no consumo de recursos computacionais. A seguir, descrevemos estas alterações.

3.4.1 Pré-processamento + ILP

A primeira grande alteração foi a introdução de uma etapa de pré-processamento, anterior à chamada ao resolvidor (*solver*). Um novo conjunto de variáveis $s \in Q_d$ é gerado nesta etapa, para todo $d \in D$, onde s representa uma rota e o posicionamento de regeneradores, se necessários, para atender d . Em outras palavras, de forma semelhante ao RMSA-BPenum, Q_d contém todas as combinações possíveis de regeneradores para cada rota encontrada entre S_d e T_d , respectivamente a origem e destino da demanda d . Cada solução $s \in Q_d$ é o resultado da concatenação de, no máximo, $R_{\max} + 1$ segmentos, onde o nó de origem do primeiro enlace do primeiro segmento é S_d e o destino do último enlace do último segmento é o nó T_d .

Como exemplo do resultado do pré-processamento, será usada a topologia mostrada na Figura 3.6, com $R_{\max} = 2$ e maior alcance de modulação = 4000 km. Considere a demanda $d_1 \in D$, com origem no nó $S_{d_1} = 01$ e destino no nó $T_{d_1} = 08$. Existem três possíveis rotas para atender d_1 : $Rota_1=01-02-03-08$, $Rota_2=01-04-05-06-08$ e $Rota_3=01-07-08$. Os segmentos que podem ser concatenados formando a rota $Rota_1$ são: $s_{d_1}^1=[01-02,02-03-08]$ e $s_{d_1}^2=[01-02,02-03,03-08]$, respeitando R_{\max} e tamanho máximo de segmento equivalente à 4000 km. Na rota $Rota_2$, teríamos as seguintes possibilidades de solução: $s_{d_1}^3=[01-04-05-06-08]$, $s_{d_1}^4=[01-04,04-05-06-08]$, $s_{d_1}^5=[01-04,04-05,05-06-08]$, $s_{d_1}^6=[01-04,04-05-06,06-08]$, $s_{d_1}^7=[01-04-05,05-06-08]$, $s_{d_1}^8=$

[01-04-05,05-06,06-08] e $s_{d_1}^9 = [01-04-05-06,06-08]$. Observe que a solução [01-04,04-05,05-06,06-08] não seria válida neste exemplo, pois ultrapassa R_{\max} regeneradores. Por último, na rota $Rota_3$, teríamos $s_{d_1}^{10} = [01-07,07-08]$. Portanto, neste exemplo, a demanda d_1 teria um total de apenas 10 combinações possíveis utilizando 3 rotas distintas e seria representada por $Q_{d_1} = \{s_{d_1}^1, s_{d_1}^2, \dots, s_{d_1}^{10}\}$.

Com este pré-processamento, ao se adotar uma solução $s^n \in Q_d$ para atender uma demanda d tal que $a_d = 1$, a Restrição (C2) de conservação de fluxo é automaticamente atendida. Adicionalmente, também atende, de forma intrínseca, a Restrição (C3), que limita à R_{\max} o número de regeneradores por demanda. Assim, por intermédio deste pré-processamento, obtém-se $x_p^d(s)$, contendo os valores para as variáveis x_p^d na solução s . De posse de $x_p^d(s)$, podemos extrapolar e obter as seguintes constantes, equivalente ao procedimento realizado em RMSA-BPenum: R_d^s , F_d^s e $X_d^{e,s}$.

O conjunto P_d^s é composto por todos os segmentos p utilizados na solução s da demanda d . A constante R_d^s representa o número de regeneradores usados pela demanda d na solução s e pode ser obtida da seguinte forma:

$$R_d^s = \sum_{p \in P} x_p^d(s) - 1$$

Por sua vez, F_d^s é o número total de FSs usados pela demanda d na solução s .

$$F_d^s = \sum_{e \in E} \sum_{p \in P_d^s} F_p^d Y_p^e$$

Por último, temos $X_d^{e,s}$, indicando se a demanda d utiliza o enlace e na solução s .

$$X_d^{e,s} = \begin{cases} 1 & \text{if } \sum_{s \in Q_d} \sum_{p \in P_d^s} Y_p^e > 0 \\ 0 & \text{otherwise.} \end{cases}$$

De forma equivalente feita anteriormente, iremos utilizar os seguintes atalhos:

$$U_d^e = \sum_{s \in Q_d} \sum_{p \in P_d^s} F_p^d x_p^d Y_p^e > 0$$

$$V_d^e = \begin{cases} 1 & \text{if } \sum_{s \in Q_d} x_d^s X_d^{e,s} > 0 \\ 0 & \text{otherwise.} \end{cases}$$

Na nova formulação, são utilizados três conjuntos de variáveis: x_d^s , z_d^e e $o_{(d,d')}^e$. O conjunto de variáveis binárias x_d^s indica se a demanda d utiliza a solução s , para cada demanda $d \in D$ e $s \in Q_d$. O conjunto de variáveis inteiras z_d^e , já definidas na Seção 3.2, representa o índice de FS usado pela demanda d no enlace e , ou 0 caso o enlace

e não seja utilizado pela demanda d . Finalmente, as variáveis binárias $o_{(d,d')}^e$ e $o_{(d',d)}^e$, para cada par não ordenado (d, d') de demandas distintas $d, d' \in D$ e cada enlace $e \in E$, foram introduzidas para simplificar e garantir a restrição de contiguidade de CC. A variável $o_{(d,d')}^e$, quando assume o valor 1, indica que existe um conflito entre as demandas d e d' em pelo menos um índice de FS no enlace e . Ao assumir o valor 0, indica que o espaço contíguo reservado para atender a demanda d no enlace e não poderá ser usado por nenhuma outra demanda, garantindo sempre a não violação da restrição de contiguidade.

A notação D_u^2 é usada para denotar o conjunto de todos os pares não ordenados (d, d') , onde d e d' pertencem ao conjunto D e são distintos ($d \neq d'$). A formulação também utiliza a notação A_d :

$$A_d = \sum_{s \in Q_d} x_d^s$$

A_d representa o número de soluções em Q_d que a demanda d utiliza. A_d pode valer no máximo 1, pois apenas uma solução pode ser adotada por uma demanda. Quando vale 0, significa que nenhuma solução $s \in Q_d$ foi usada e, portanto, a demanda d foi bloqueada.

A formulação RMSA-BP, incorporando essas alterações na formulação inicial, fica desta forma:

$$\max \sum_{d \in D} \left(w_1 A_d - w_2 \sum_{s \in Q_d} R_d^s x_d^s - w_3 \sum_{s \in Q_d} F_d^s x_d^s \right)$$

s.t.

$$\sum_{s \in Q_d} x_d^s \leq 1 \quad \forall d \in D \quad (\text{R1})$$

$$V_d^e \leq z_d^e \leq V_d^e \xi \quad \forall d \in D, e \in E \quad (\text{R2})$$

$$z_d^e + U_d^e - V_d^e \leq \xi \quad \forall d \in D, e \in E \quad (\text{R3})$$

$$Y_p^e (z_d^e - z_d^{I_p}) x_d^s = 0 \quad \forall d \in D, s \in Q_d, p \in P_d^s, \quad (\text{R4})$$

$$e \in E, e \neq I_p,$$

$$o_{(d,d')}^e + o_{(d',d)}^e \leq 1 \quad \forall (d, d') \in D_u^2, e \in E \quad (\text{R5})$$

$$z_d^e + U_d^e \leq z_{d'}^e + M o_{(d,d')}^e \quad \forall (d, d') \in D_u^2, e \in E \quad (\text{R6})$$

$$z_{d'}^e + U_{d'}^e \leq z_d^e + M o_{(d',d)}^e \quad \forall (d, d') \in D_u^2, e \in E \quad (\text{R7})$$

A Restrição (R1) garante que nenhuma demanda d utilize mais do que uma solução $s \in Q_d$. Obviamente, esta seria uma situação inconsistente, onde uma mesma demanda é atendida mais de uma vez, utilizando soluções s diferentes. As

Restrições (R2) e (R3) são equivalentes as Restrições (C5) e (C6), respectivamente. A Restrição (R2) garante que z_d^e seja igual a 0 caso o enlace e não seja utilizado pela demanda d . Caso contrário, z_d^e deve pertencer ao intervalo $\{1, \dots, \xi\}$. Já a Restrição (R3) garante que nenhum índice de FS ultrapasse o limite da capacidade do enlace (ξ). A Restrição (R4), de forma semelhante a Restrição (C7), implementa a restrição de continuidade, garantindo que todos os enlaces e que compõem um segmento p utilizem o mesmo índice de FS usado no enlace inicial I_p , através da condição $z_d^e = z_d^{I_p}$, desde que d use a solução $s \in Q_d$, o segmento p faça parte desta solução s e e faça parte do segmento p .

As Restrições (R5)-(R7) se referem às restrições de contiguidade e foram redefinidas visando melhoria de performance em relação à Restrição (C8) da formulação inicial, que se mostrou complexa e pouco eficiente na sua implementação. Optou-se por uma maneira mais direta, fazendo uso da técnica chamada *BigM* [40]. Considere a parte M-independente das Restrições (R6) e (R7), ou seja, $z_d^e + F_d^e \leq z_{d'}^e$ e $z_{d'}^e + F_{d'}^e \leq z_d^e$, para cada par não ordenado $(d, d') \in D_u^2$. Claramente, essas condições nunca podem ser simultaneamente satisfeitas, forçando essencialmente uma escolha a priori entre elas. Essa impossibilidade é o que a parte M-dependente das duas restrições ajuda a contornar. Por exemplo, vamos supor que $z_d^e < z_{d'}^e$. De modo a não existirem conflitos entre d e d' , é necessário também que $z_d^e + F_d^e - 1 < z_{d'}^e$. Caso isso seja verdade, a parcela M-dependente da expressão pode ser 0, ou seja, $o_{(d,d')}^e \in \{0, 1\}$, como mostra a Figura 3.7.

O método *BigM* continua a ser amplamente utilizado em pesquisa operacional, sendo essencialmente uma técnica para resolver problemas de programação linear utilizando o algoritmo *Simplex*, mas que contenham restrições do tipo “maior ou igual”, através da introdução de variáveis artificiais para transformar todas as desigualdades no formato exigido pelo *Simplex*. A designação “Big M” está associada a um número de valor grande, representado pela letra M. O valor de M deve ser escolhido cuidadosamente de modo que seja suficientemente grande para que a variável artificial introduzida nunca faça parte de uma solução viável. O método *BigM* pode levar a problemas de estabilidade numérica, especialmente quando os valores de M se tornam muito grandes, podendo causar imprecisões numéricas. A escolha do valor de M adequado é essencial para não impactar na qualidade da solução ótima encontrada.

No entanto, o conceito *BigM* pode ser empregado em outras aplicações, como no caso adotado na nossa solução. Quando aplicado diretamente nas próprias restrições do problema, refere-se a garantir a não violação de uma restrição utilizando *BigM* apenas quando uma variável binária específica associada ao M assumiu um determinado valor.

A Restrição (R3), utilizando as variáveis $o_{(d,d')}^e$ e $o_{(d',d)}^e$, descarta imediatamente

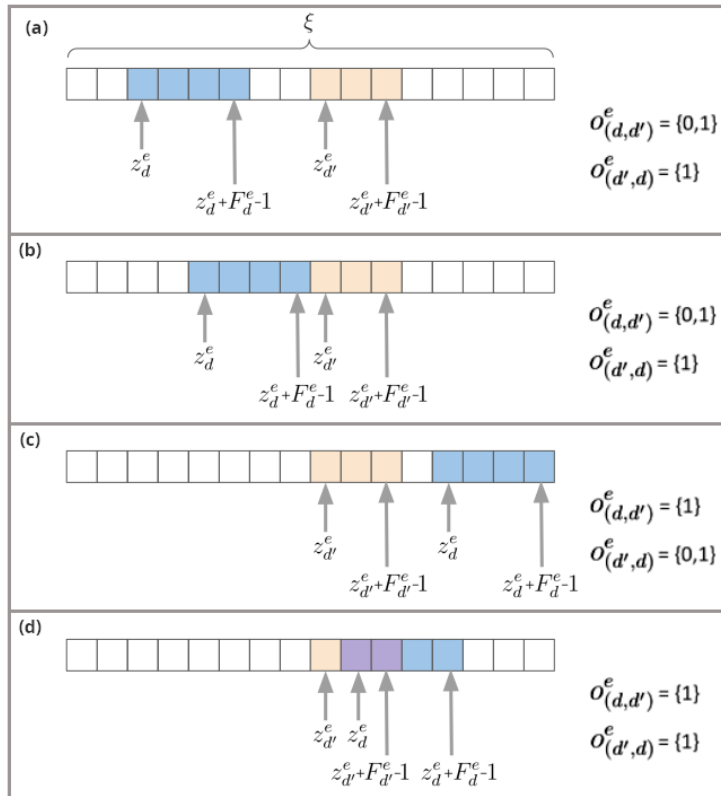


Figura 3.7: Cenários possíveis e BigM

conflitos entre as demandas d e d' . A implementação das Restrições (R4) e (R5) utilizando BigM, em conjunto com a Restrição (R3), se mostrou muito mais simples e eficiente do que a implementação anterior.

No Capítulo 4, são mostrados detalhes da implementação dos métodos descritos, bem como os resultados obtidos. Neste capítulo também é mostrado como o caráter quadrático das Restrições (R2) e (C7) foi contornado.

Capítulo 4

Implementação e Resultados

Neste capítulo são descritos detalhes da implementação dos modelos RMSA-BP e RMSA-BPenum, além de mostrar e analisar os resultados obtidos.

4.1 Ambiente de desenvolvimento

O desenvolvimento e testes das versões iniciais foram realizados em um servidor com sistema operacional Linux, utilizando a distribuição Ubuntu 20.04.6 LTS. O servidor é equipado com um processador Intel(R) Core(TM) i9-10900 CPU @ 2.80GHz, que possui 10 núcleos físicos, e com 128 GB de RAM. A implementação da formulação RMSA-BP foi realizada em sua maior parte em outro servidor, de maior capacidade, dotado de dois processadores AMD EPYC 7763 64-core e 512 GB RAM, usando a distribuição Debian 11.

Em ambos os servidores foram utilizados os seguintes softwares:

1. Gurobi versões 9.1.1 e 9.5.2
2. Python 2.7.18 e posteriormente 3.9.2
3. Biblioteca Networkx 3.1
4. GNU parallel

O software de otimização *Gurobi* [41] foi escolhido para implementar e resolver as formulações de programação linear, não somente por ser reconhecidamente um dos mais rápidos e completos *solvers*, mas principalmente por possuir uma licença acadêmica gratuita e sem restrições. Futuramente, é previsto fazer testes comparativos de performance com outros *solvers*, como *CPLEX* [42], por exemplo.

A linguagem *Python* [43] foi adotada pela facilidade em manipular estruturas e número de bibliotecas disponíveis. A documentação do *Gurobi* atualmente recomenda o seu uso como a melhor opção, apesar de ser reconhecidamente mais lenta do que

códigos compilados em C. A escolha da linguagem *Python* teve pontos positivos e negativos, como veremos adiante.

NetworkX [44] é uma biblioteca *Python* completa para representar e manipular basicamente todos os tipos de grafos. Suas inúmeras funções disponíveis permitem facilmente identificar caminhos mais curtos, determinar todos os caminhos entre dois vértices, etc., e facilitaram bastante a implementação do nosso modelo. Sua integração com a biblioteca *matplotlib* também foi usada para visualizar como as redes se comportavam a medida que iam atendendo as demandas.

O software GNU Parallel [45] é um programa com interface de linha de comando para sistemas Linux, voltado para a execução de comandos em paralelo, como o próprio nome sugere, permitindo tirar o máximo proveito dos núcleos de processador disponíveis de um sistema. Sua adoção permitiu executar diversas instâncias simultaneamente, permitindo explorar ao máximo os recursos disponíveis, sempre com o cuidado de evitar disputa por recursos e trocas de contexto entre os processos.

4.2 Estruturas de dados dos arquivos de entrada

Todos os arquivos contendo os dados para cada instância são passados como parâmetros por linha de comando, como topologia, modulações disponíveis e especificação das demandas.

As topologias foram representadas como grafos no formato GML, que são facilmente lidas pela biblioteca *Networkx* e transformadas em suas estruturas internas para manipulação das mesmas. A estrutura do formato é bem simples, similar ao JSON. Define-se o nome do grafo e, dentro dele os nós (*nodes*) que compõem o grafo, juntamente com seus atributos. Por fim, definem-se os enlaces (*edges*), referenciando quais nós são interligados, além de atributos específicos do próprio enlace, como seu comprimento, por exemplo.

As modulações disponíveis são definidas em um arquivo no formato CSV, contendo o nome da modulação, a capacidade Gbps por FS e o alcance máximo da modulação. A Tabela 4.1 mostra as modulações usadas nos testes.

Tabela 4.1: Modulações usadas nos testes [4]

Modulação	FS Bandwidth (Gbps)	Alcance Máximo (km)
BPSK	12,5	4000
QPSK	25	2000
8-QAM	37,5	1000
16-QAM	50	500
32-QAM	62,5	250
64-QAM	75	125

As demandas para cada instância são especificadas também através de arquivos no formato CSV, contendo o nó de origem, o nó de destino e a largura de banda requerida pela demanda. Esses arquivos foram gerados por um *script* que recebe como parâmetros o número de demandas a serem geradas e a topologia para qual se destinam. As demandas são geradas aleatoriamente, garantindo uma distribuição uniforme dos nós de origem e destino. Além disso, as larguras de banda para cada demanda também são distribuídas de forma homogênea, podendo conter um dos seguintes valores: 10, 40, 100 e 400 Gbps.

4.3 Identificando segmentos válidos

A abordagem voltada a segmentos, ao invés de enlaces, mostrou ser mais simples, pois ao saber o comprimento total de um segmento, imediatamente podemos identificar a modulação mais eficiente que suporte esta distância.

A geração do conjunto P de segmentos válidos foi implementada conforme descrito no Algoritmo 1, utilizando a função `all_simple_paths` da biblioteca *Networkx*. O tempo de processamento é bastante baixo, independente da topologia sendo pesquisada, conforme pode ser visto na Tabela 4.2.

Tabela 4.2: Tempo (s) para geração de P , 30 instâncias

Topologia	Média	Min	Max
TopoX	0,11	0,07	0,22
NFSnet	0,24	0,19	0,34
Cost239	1,04	0,83	1,29
Cost266	18,58	16,63	19,12

Os tempos foram obtidos com 30 amostras de processamento para cada topologia, usando como comprimento máximo 4000 km.

Tabela 4.3: Graus dos nós e tamanho dos enlaces por topologia

Topologia	Nós	Enlaces	Menor grau	Maior grau	Grau médio	Menor enlace	Maior enlace	Enlace médio
TopoX	6	8	1	4	2,67	750	1800	1231,25
NFSnet	14	21	2	4	3,00	312	3408	1299,05
Cost239	11	26	4	6	4,73	362	790	482,62
Cost266	28	41	2	5	2,93	218	1500	622,93

Os tempos para a topologia Cost266 para a geração de P , mesmo pequenos, são bem superiores aos demais. O número de nós e enlaces de menor comprimento possibilitam a formação de muito mais segmentos viáveis, conforme mostrado na Tabela 4.3. No entanto, a topologia que apresenta o maior número de segmentos válidos é a Cost239.

4.4 Limitação de regeneradores por demanda

A adoção de um limite de regeneradores por demanda (R_{\max}) foi importante para reduzir o número de variáveis das instâncias, mas, ao mesmo tempo, manteve um cenário realista. Como dito, devido ao elevado custo de um regenerador, não faz sentido, no sentido comercial, um provedor de acesso utilizar mais de um ou dois regeneradores para atender um única demanda. Além disso, o número de regeneradores aumenta, de forma exponencial, o número de soluções para atender uma demanda, como pode ser visto na Tabela A1. Para cada par ordenado (d, d') onde $d \neq d'$, vemos o número de segmentos válidos para uma modulação de 4000 km, variando $R_{\max} = \{0, \dots, 12, \infty\}$. Observa-se nesta tabela que, a partir de doze regeneradores neste cenário, o número de segmentos válidos atinge o seu limite.

A Figura 4.1 facilita bastante a visualização e compreensão da Tabela A1, onde também é possível verificar que $R_{\max} = \{1, 2\}$ mantém o tamanho do problema administrável na topologia NFSNET.

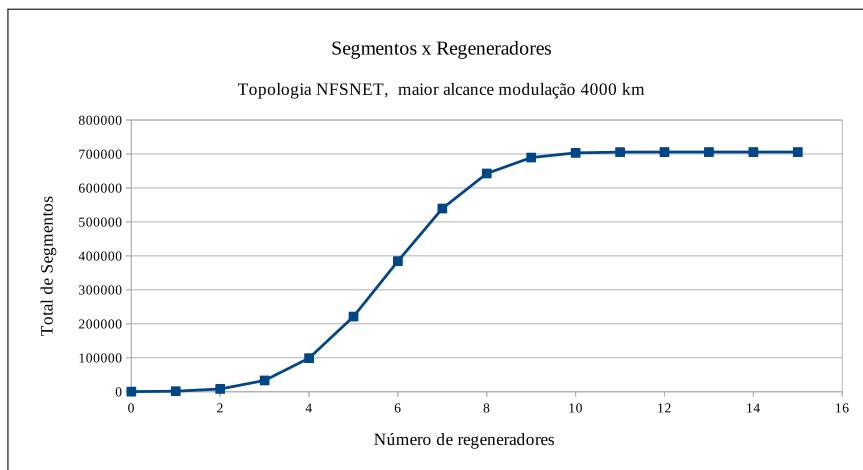


Figura 4.1: Relação Segmentos X Regeneradores

4.5 RMSA-BPenum

Os resultados obtidos no RMSA-BPenum foram fundamentais na validação dos resultados das primeiras versões do RMSA-BP. A implementação aproveitou integralmente toda a parte de inicialização do código Python anterior, basicamente retirando a parte de configuração de variáveis e restrições, bem como as chamadas para o *solver* Gurobi, e introduzindo as rotinas recursivas para a implementação da solução por enumeração e *backtracking*. Todos os parâmetros do modelo são passados pela linha de comando, o que facilitou muito a adição de novos métodos ou formulações conforme mostrado a seguir:

RMSA-BP: rmsaBPv3d.py

```
usage: rmsaBPv3d.py [-h] -f ARQTOPO -d ARQDEMANDAS -m ARQMODULACAO
                  [-a ALPHA] [-c CAP] [-r REGEN]
                  [-ms MAXSEG] [-mr MAXREGPD]
                  [-mh MAXHOP]
                  -fo {rlp-fon,bp,pabp,testes,enum,enumC,newbp}
                  [-mo {minreg,minfs,compact}] [-FM {h,b,d,s}]
                  [-cc] [-s] [-fbw FIXEDBW]
                  [-v] [-nc] [-M] [-w WRT] [-D]
                  [-gp GUROBIPARAM] [-Cp CPARAM] [-is]
```

Uma característica interessante do Python facilitou muito a implementação do código RMSA-BPenum, conforme descrito nos Algoritmos 2 e 3. Na linguagem Python, não existe um limite para o tamanho de um número inteiro, como acontece em praticamente em todas as outras linguagens de programação. Teoricamente, o maior número inteiro que uma variável Python pode armazenar é limitado apenas pela quantidade de memória que o computador possui, o que representa um número inimaginavelmente grande. Isso foi muito útil para as variáveis contendo mapas de bits. Por exemplo, usando a topologia NFSnet (21 enlaces) e $\xi = 160$, teríamos um valor inteiro com $21 * 160 = 3360$ bits! Todas as operações, aritméticas e lógicas, podem utilizar estas estruturas, ou seja, são válidas as instruções “A & B” ou “A | B”, representando “A E B” e “A OU B”, respectivamente. Desta forma, para implementar o código para verificar se existia conflito entre duas demandas, uma única linha de código com um simples “AND” era suficiente.

Entretanto, instâncias com $\xi > 40$ se mostraram extremamente lentas. Com o intuito de tentar acelerar o processo, foram desenvolvidos dois melhoramentos do RMSA-BPenum, ambas utilizando bibliotecas escritas em C, responsáveis por toda a implementação de *backtracking* e recursividade. A primeira biblioteca, embora escrita em C, usava as estruturas e métodos internos do próprio Python. Desta forma, o código em C invocava o método AND do Python passando ponteiros para as estruturas Python que continham os valores dos mapas de bits. Já na segunda biblioteca, estruturas em C nativo e uma implementação do AND e OR para estas estruturas foram desenvolvidas. Ambas as bibliotecas proporcionaram uma melhoria no desempenho, mas não suficientemente satisfatórias.

4.5.1 Observações no RMSA-BPenum

Em muitos casos, a solução exata era atingida rapidamente. No entanto, mesmo fazendo muitos cortes nos ramos antecipadamente por não serem promissores através

de *backtracking*, a busca na árvore prosseguia até o final, de forma a garantir a não existência de uma melhor solução.

Algumas melhorias foram introduzidas para otimizar a pesquisa na árvore. Uma dessas melhorias envolveu a adição de novos campos na estrutura de dados de cada solução, obtidas durante a fase de pré-processamento. Ao invés de se percorrer sequencialmente as soluções de cada demanda, a estrutura passou a incluir ponteiros contendo o índice da próxima solução \mathcal{S}_d , que alterasse o padrão utilizado nos enlaces onde ocorreu algum conflito.

Para ilustrar esse conceito, consideremos o exemplo mostrado na Figura 3.4. Suponhamos que o primeiro FS no *link 1-2* tenha sido alocado pela demanda d_0 . Pela estrutura e algoritmo original, ao se detectar o conflito com a solução $\mathcal{S}_{d_1}^4$, por exemplo, avançaríamos sequencialmente para a próxima solução ($\mathcal{S}_{d_1}^5$) e assim por diante, até atingir a solução $\mathcal{S}_{d_1}^8$. Com a introdução de ponteiros dentro da estrutura de soluções, $\mathcal{S}_{d_1}^4$ apontaria diretamente para $\mathcal{S}_{d_1}^8$, evitando desta forma um grande número de testes. Esta otimização poderia evitar ξ testes, no melhor caso. Embora essa alteração tenha melhorado significativamente o desempenho, a performance ainda não escalava para ξ muito grandes.

O consumo de memória por processo também crescia de forma significativa à medida que ξ aumentava. Ao invés de representar o mapa de bits completo para cada solução, foi criada uma tabela contendo todos os padrões possíveis dentro de um enlace. O mapa de bits de cada solução passou a ser representado pela sequência de índices nesta tabela, um para cada enlace. Essa abordagem proporcionou uma grande economia de memória. Em contrapartida, um pequeno aumento de processador foi observado, gastos na conversão desta sequência de índices em um mapa de bits completo.

Uma observação interessante foi que a ordem das demandas a serem tratadas impactava diretamente no tempo de processamento utilizando RMSA-BPenum. Uma mesma instância, simplesmente reordenando suas demandas segundo algum critério, poderia ter seu tempo de processamento alterado, tanto para melhor quanto para pior. Após essa observação, diversos critérios de ordenação das demandas foram analisados. Alguns desses critérios incluíram ordenar pelo número de soluções para cada demanda (crescente/decrescente), ordenar pelas demandas que mais utilizam os enlaces mais frequentemente utilizados entre todas as soluções, e assim por diante. No entanto, não foi encontrado um critério ideal, que desse bons resultados na maioria das vezes.

4.6 Resultados RMSA-BP

Os experimentos com as implementações anteriores consumiam um tempo considerável de processamento. Algumas instâncias podiam demorar dias até chegar numa solução ou então canceladas, manualmente ou por *timeout*, sem mesmo terem solucionado o problema, embora muitas vezes se encontrassem bem próximas da solução ótima.

4.6.1 Limites por instância

Desta forma, os resultados aqui apresentados dos experimentos com formulação RMSA-BP tiveram como limite 30h de processamento para cada instância executada. Este limite foi implementado através do parâmetro `timeout` do Gnu Parallel, que limita o tempo de execução de cada processo disparado.

O Gurobi, por padrão, inicializa diversas *threads* em paralelo durante algumas etapas do seu processamento, limitados somente ao número de núcleos disponíveis no sistema onde está sendo executado. No entanto, para cada instância, limitamos o Gurobi a paralelizar no máximo 32 *threads*, de modo que fosse possível múltiplas instâncias serem executadas em paralelo. Em conjunto a isso, o número de processos disparados pelo GNU Parallel foi dimensionado de tal forma que nunca ultrapassasse o total de núcleos disponíveis no sistema.

4.6.2 Implementação da função multiobjetivo

Nas funções objetivo das formulações apresentadas, a priorização dos objetivos foi representada pelos pesos w_1, w_2, w_3 . No entanto, a grandeza necessária dos pesos para garantir esta priorização poderia resultar em problemas numéricos durante o processamento da instância, potencialmente reduzindo a precisão da solução encontrada.

Para contornar este problema, foi utilizada uma característica do Gurobi que permite a definição de funções multi-objetivo. Existem diferentes maneiras disponibilizadas pelo *solver* na definição de problemas com múltiplos objetivos. É possível mesclar os diversos objetivos através de pesos ou tratá-los hierarquicamente. A abordagem *blended* otimiza uma combinação ponderada dos objetivos individuais.

Já na abordagem *hierarchical*, é definida uma prioridade para cada objetivo e cada um é tratado individualmente, seguindo a ordem das prioridades definidas. O método *hierarchical* garante que, ao otimizar um objetivo, são consideradas apenas soluções que não degradem os valores dos objetivos de maior prioridade. Foi desta forma que o RMSA-BP foi implementado, através da função `Model.setObjectiveN` no modo hierárquico. O interessante desta abordagem é que torna-se possível acompanhar nos

logs do Gurobi toda a evolução da otimização, para cada objetivo sendo processado.

4.6.3 Indicator Constraints

Devido à natureza intrinsecamente quadrática da formulação da Restrição (R2), bem como da Restrição (C7), seria necessário linearizar estas restrições para garantir que o problema seja de fato caracterizado como ILP. Contudo, ao invés disso, a implementação destas restrições foi realizada utilizando *Indicator Constraints*.

Sabemos que, em problemas de otimização, não existe algo equivalente a um “IF”. No entanto, versões mais recentes do Gurobi introduziram o uso de *Indicator Constraints*, que são um novo tipo de restrição. Permitem ao usuário expressar relações específicas entre variáveis, identificando uma variável binária para controlar se uma determinada restrição linear deve ser ativada ou não. Desta maneira, se comportam de forma similar a um “IF”, deixando a tarefa de conversão/linearização para o *solver* realizar internamente, de forma transparente para o usuário.

De um modo geral, formulações que utilizam *Indicator Constraints* são numericamente mais robustas e precisas do que formulações convencionais [46]. A Restrição (R2) poderia ser escrita como $(x_d^s = 1) \gg ((X_d^{e,s} Y_p^e z_d^e - X_d^{I_p,s} Y_p^{I_p} z_d^{I_p}) = 0)$.

4.7 Resultados

Os dados mostrados a seguir são o resultado da execução de três conjuntos de 30 instâncias cada. O primeiro conjunto utilizou arquivos de demandas contendo 100 demandas ($|D| = 100$). Os outros dois utilizaram arquivos de demandas com $|D| = 110$ e $|D| = 120$, respectivamente. Cada demanda d foi gerada com requisitos de largura de banda $B_d = 100$ Gbps, e os nós de origem S_d e destino T_d foram distribuídos de forma uniformemente aleatória.

Na Tabela 4.4 pode-se observar o tamanho de Q_d , considerando os três conjuntos de instâncias e variando $R_{\max} = \{1, 2\}$.

Tabela 4.4: Número de soluções nos conjuntos Q_d , para todas as demandas

R_{\max}	Mínimo	Máximo	80th Percentile
1	2	44	21
2	14	193	125

O tempo de pré-processamento na topologia NFSNET é bastante baixo, conforme pode ser constatado na Tabela 4.5. Isto é devido ao seu baixo número de segmentos viáveis. Topologias mais complexas, com muitos enlaces curtos e maior número de nós apresentam tempos de pré-processamento muito superiores, pois possuem muito mais segmentos viáveis, aumentando consideravelmente o número de soluções do

conjunto Q_d . Nestes casos, uma alternativa seria gerar arquivos previamente para todo par ordenado (d, d') tal que $d \neq d'$, para todo $d, d' \in D$. Obviamente, isto seria necessário fazer uma única vez e, ao executar o modelo, ler o Q_d respectivo para cada demanda d .

Tabela 4.5: Tempo de pré-processamento (s), 30 instâncias

$ D $	R_{\max}	Mínimo	Maximo	80th Percentile
100	1	0.51123	1.04801	0.60694
110	1	0.53375	1.35035	0.70054
120	1	0.54668	1.17509	0.67801
100	2	0.91118	1.50329	1.12410
110	2	0.93966	2.01165	1.13245
120	2	0.95076	2.47579	1.21260

O tempo de processamento está diretamente relacionado à saturação dos enlaces. Demandas somente são bloqueadas à medida que enlaces vão se aproximando de 100% de utilização. Quando maior for ξ , menor a possibilidade de ocorrer algum bloqueio, sendo necessário cada vez mais demandas para ocupar todo o espectro dos enlaces.

Nos primeiros testes com RMSA-BP utilizando $\xi = 160$ neste cenário, não ocorreu bloqueio em nenhuma das instâncias, tanto para R_{\max} igual a 1 ou 2. Desta forma, para forçar o bloqueio de demandas e analisar o comportamento do modelo diante da saturação de enlaces, foi usado $\xi = 80$ em todos os testes realizados.

Na Tabela 4.6, é mostrado o número médio de demandas bloqueadas ($BD \leq |D|$), a média do número total de regeneradores usados por instância ($TR \leq |D|R_{\max}$) e a média do total de FSs usados por instância ($TF \leq 1680$). O total de FSs disponíveis é calculado através da multiplicação do número de enlaces da topologia e o ξ sendo utilizado. No caso da NFSNET, que possui 21 enlaces, multiplicados pelo valor de ξ , resulta 1680 FSs.

Tabela 4.6: Valor médio de bloqueios e taxa de utilização, 30 instâncias

$ D $	R_{\max}	BD	TR	TFS	R_{\max}	BD	TR	TFS
100	1	1.0	43.4	1 265	2	0.6	42.4	1 263
110	1	4.4	55.1	1 292	2	2.9	59.0	1 283
120	1	7.5	55.7	1 329	2	5.1	64.8	1 319

Na Tabela 4.7, observamos que o número de *timeouts* ($timeouts \leq 30$) foi consistentemente baixo. No entanto, ocorreu uma expressiva variação nos tempos de execução nas instâncias bem-sucedidas. Não obstante, para 24 (80%) das instâncias, esses tempos ficaram abaixo de algumas horas, embora consideravelmente mais para $R_{\max} = 2$ do que para $R_{\max} = 1$.

Complementando as informações nas tabelas, observamos que o número médio de núcleos de CPU utilizados foi entre 20 e 22, durante todo o período de execução. O consumo de memória varia diretamente com o número de demandas, oscilando normalmente entre 4 e 8 GB de RAM. No entanto, em instâncias mais demoradas, algumas instâncias atingiram valores próximos de 50 GB de RAM.

Tabela 4.7: *Timeouts* e tempos de execução (h:m:s), 30 instâncias

$ D $	R_{\max}	<i>Timeouts</i>	Mínimo	Máximo	80th Percentile
100	1	2	00:03:06	15:48:10	00:50:44
110	1	3	00:04:15	14:03:06	01:28:03
120	1	8	00:14:24	14:24:14	01:57:48
100	2	4	00:07:52	29:39:28	04:42:32
110	2	1	00:03:39	12:27:29	03:51:46
120	2	4	00:08:26	14:03:24	08:34:09

4.8 Saturação dos Enlaces

Alguns estudos foram realizados de modo a compreender melhor como a saturação dos enlaces vai se desenvolvendo e como a topologia da rede influencia no uso dos enlaces, à medida que o número de demandas aumenta. Foi utilizado basicamente o mesmo ambiente dos testes, agora com 4 conjuntos de demandas e com $|D| = 30, 60, 90, 120$. Consideramos a utilização de um enlace como sendo a razão entre o número de FSs usados no enlace e ξ . Um mapa de calor com médias ao longo das 30 instâncias de cada conjunto de demandas é mostrado na Figura 4.2. O uso dos enlaces cresce, como esperado, conforme $|D|$ aumenta. Com o mapa foi possível constatar que o enlace 8-9 é sempre um dos mais utilizados em todos os casos, enquanto o enlace 1-2 é um dos menos utilizados. Essas parecem ser propriedades estruturais inerentes à topologia da NSFNET, que, em princípio, talvez não tivessem sido identificadas se alguma heurística tivesse sido utilizada no lugar.

É importante observar que com 120 demandas, a rede está bem próxima da saturação, mas, mesmo assim, boa parte das instâncias com este número de demandas foi resolvida.

4.9 Ajustando o Gurobi

Parte integrante do pacote Gurobi, o GRBTUNE é uma ferramenta para fazer o ajuste fino dos diversos parâmetros de execução que o Gurobi possui. Sua função é adequar a configuração do Gurobi para melhor atender às especificidades de um determinado modelo, e não alterar o modelo propriamente dito.

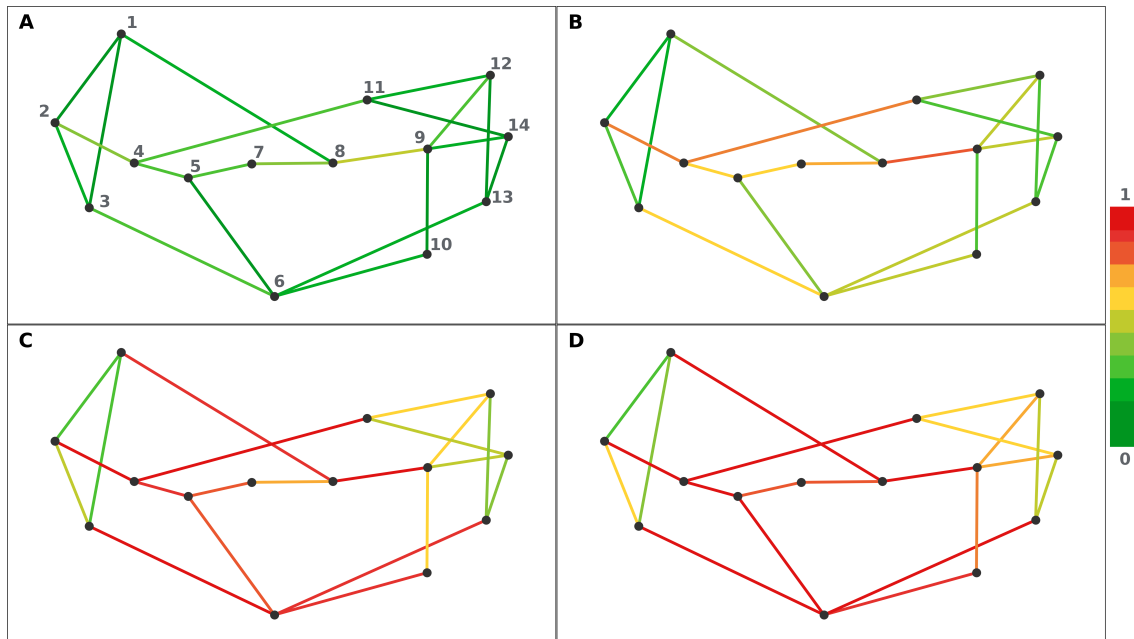


Figura 4.2: Uso dos Enlaces para $|D| = 30(\text{A})$, $60(\text{B})$, $90(\text{C})$, $120(\text{D})$

Com interface por linha de comando, seu uso é muito simples e direto. Basicamente, ele aceita um ou mais modelos no formato MPS, formato muito utilizado para armazenar modelos de programação matemática. Após analisar automaticamente os modelos recebidos, o GRBTUNE tenta resolver repetidamente o modelo, variando parâmetros e ajustando as configurações a cada iteração. A cada iteração, compara os tempos de execução obtidos com a alteração. Se um determinado parâmetro piora o desempenho, o GRBTUNE interrompe e parte para outro parâmetro. Se melhorou, testa combinações do parâmetro atual com outros parâmetros. Ele prossegue neste processo, identificando as combinações de parâmetros que resultaram em um melhor desempenho. Ao terminar, o utilitário exibe as combinações de parâmetros que foram mais eficazes em aprimorar o desempenho na resolução especificamente dos modelos recebidos.

Uma vez que essas configurações são conhecidas, espera-se que as mesmas sejam também eficazes para outras instâncias do mesmo problema. Nos testes realizados, foram observadas reduções significativas no tempo de execução em boa parte dos modelos testados.

Capítulo 5

Conclusão

Neste trabalho foi apresentado um modelo ILP para resolver o problema RMSA (*Routing, Modulation Level and Spectrum Assignment*), parte fundamental em redes ópticas elásticas ou *Elastic Optical Networks* (EON), que constituem um novo paradigma em transmissão de dados utilizando fibras ópticas. O problema RMSA engloba a escolha da rota e da modulação a ser usada entre a origem e o destino de uma conexão de dados, bem como a alocação de espectro óptico. Entende-se como alocação de espectro a definição de qual frequência e o tamanho alocado que cada circuito irá utilizar, mas de modo a não existir conflitos entre eles. Este problema é muito mais complexo quando comparado ao equivalente em redes DWDM (*Dense Wavelength Division Multiplexing*). DWDM divide o espectro em blocos de tamanho fixo, sem possibilidade de concatenação destes blocos de forma transparente, além de ter modulação constante por todo o caminho. Devido às características do EON, com suas regras bem mais flexíveis do que as do DWDM, o problema RMSA torna-se muito mais complexo. Apesar deste grande desafio, e o RMSA ser classificado *NP-hard* na literatura, nosso modelo, denominado RMSA-BP, obteve resultados em tempos satisfatórios.

EON tem sido objeto de estudo durante a última década, e, por consequência, RMSA também. Diferentemente de outras propostas, baseadas em heurísticas ou mesmo formulações ILP, o RMSA-BP tem como objetivo encontrar a solução exata para o problema completo, incluindo restrições de contiguidade e continuidade, minimizando o número de regeneradores necessários e espectro alocado. Adicionalmente, uma outra característica única no RMSA-BP é ter incluído como principal objetivo minimizar o BP (*Blocking Probability*), ou seja, a probabilidade de uma demanda ser bloqueada (não atendida) devido a recursos insuficientes.

O modelo RMSA-BP, composto por múltiplos objetivos hierárquicos, recebe como entrada: a topologia da rede representada por grafo, um conjunto de modulações contendo a eficácia espectral e alcance de cada modulação e um conjunto de demandas, contendo origem, destino e largura de banda requerida. A partir destes dados,

procura atingir o seu principal objetivo, que é atender ao maior número de demandas possível, para, em seguida, minimizar o número de regeneradores. Regeneradores são dispositivos muito caros, e permitem aumentar o alcance da transmissão sendo realizada. Além de regenerar o sinal, em EON é possível mudar a frequência e a própria modulação sendo utilizada. Portanto, o número de combinações possíveis (rotas, localização de regeneradores, escolha de frequência e modulação mais eficiente), sem contar a própria decisão de quais demandas serão atendidas e quais serão bloqueadas, torna a solução exata do problema bastante complexa. A escolha de frequência tem suas próprias restrições: contiguidade dentro do enlace e continuidade de um segmento. Essas restrições adicionam ainda mais dificuldade a um problema já suficientemente complexo.

Apesar de todos estes grandes desafios, o RMSA-BP se propõe a resolver o problema RMSA de forma *offline* para EON em redes translúcidas (redes que podem utilizar regeneradores em alguns de seus nós). Uma decisão que se mostrou importante foi a abordagem voltada a segmentos, ao invés de enlaces, que facilitou a definição da melhor modulação a ser usada, bem como a implementação da restrição de continuidade.

Neste trabalho, foram apresentadas duas implementações antes de se chegar ao modelo RMSA-BP final. A primeira foi uma formulação ILP e a segunda implementação com abordagem totalmente diferente, através de um algoritmo de *backtracking*. Embora com resultados corretos, não apresentaram escalabilidade para cenários e instâncias mais complexas. Mesmo assim, foram fundamentais na compreensão do problema e na identificação dos pontos e das restrições mais difíceis de serem atendidas. Adicionalmente, desempenharam papel importante como forma de validar os resultados obtidos nas soluções encontradas.

A inclusão de um limite de regeneradores (R_{\max}) por demanda foi fundamental para ampliar a escalabilidade, pois possibilitou a redução do conjunto de soluções, mas, ao mesmo tempo, manteve uma abordagem realista, apresentando soluções comercialmente viáveis.

A introdução na função objetivo de minimizar BP mostrou-se muito custosa. Soluções para instâncias sem bloqueios foram obtidas normalmente em tempos de execução inferiores às instâncias com demandas recusadas por falta de recursos disponíveis. Observou-se, como esperado, que o tempo de solução sobe a medida que os enlaces vão se aproximando da saturação, e por consequência, forçando o bloqueio. A decisão de bloquear uma ou outra demanda muda todo o cenário sendo avaliado.

Topologias com muitos enlaces pequenos acabam sendo mais difíceis de resolver do que topologias de dimensões semelhantes, mas com enlaces maiores. Isto se deve ao maior número de segmentos válidos.

O pré-processamento das soluções válidas para cada demanda, que foi empregado

na implementação RMSA-BPenum, foi utilizado na solução final, substituindo restrições associadas à conservação de fluxo. A implementação original de contiguidade e continuidade foi substituída por outra mais simples, usando a abordagem BIG-M, e que se mostrou mais eficiente que a original.

5.1 Trabalhos futuros

Como trabalhos futuros, podemos dar prosseguimento neste estudo de várias maneiras.

Sabendo o valor exato da taxa de bloqueio, número de regeneradores e utilização do espectro para uma determinada instância permitirá avaliar a qualidade de heurísticas existentes ou meta-heurísticas aplicadas para a solução do RMSA. A melhoria destas heurísticas ou mesmo o desenvolvimento de novas propostas, guiadas pela conhecimento prévio da solução exata resultante do RMSA-BP será de grande valia para um importante estudo em EON.

Um outro estudo interessante será, ao invés de minimizar BP, maximizar o lucro, sob o ponto de vista comercial do provedor. Por exemplo, o provedor de acesso pode oferecer preços variáveis em função da distância entre origem e destino ou mesmo largura de banda solicitada. Pode ser mais interessante ao provedor vender um acesso de 100 Gbps do que dez acessos de 10 Gbps. De posse de informações como custo de regeneradores, custo de chassis, capacidade máxima de regeneradores por chassis, custo de instalação, configuração e ajustes dos equipamentos, entre outros, seria possível mudar a função objetivo para maximizar o ganho do provedor, através do cálculo de lucro gerado por cada instância.

A pesquisa de como a ordem das demandas influencia no tempo de solução do problema também será relevante. Entendendo isso, uma ordenação prévia das demandas poderia trazer grande benefício ao processamento.

A adaptação do modelo proposto para utilizar geração de colunas (*Column Generation*) pode ser outra área a ser pesquisada, possivelmente reduzindo o tempo de solução e aumentando ainda mais o número de instâncias e cenários resolvidos.

A comparação da implantação do modelo utilizando outros *solvers*, como CPLEX, pode trazer melhoras no desempenho.

Por último, desenvolver a integração do modelo RMSA-BP ao uso de fibras multi-core (MCF) têm o potencial de expandir significativamente a capacidade de transmissão óptica em redes adotando esta tecnologia.

Referências Bibliográficas

- [1] AGRAWAL, G. P. “Optical Communication: Its History and Recent Progress”. In: *Optics in Our Time*, pp. 177–199, Cham, Springer International Publishing, 2016. ISBN: 978-3-319-31903-2. doi: 10.1007/978-3-319-31903-2_8. Disponível em: <https://doi.org/10.1007/978-3-319-31903-2_8>.
- [2] AGRAWAL, G. *Fiber-Optic Communication Systems*. Wiley, 2021. ISBN: 9781119737384. Disponível em: <<https://books.google.com.br/books?id=IIMxEAAAQBAJ>>.
- [3] GERSTEL, O., JINNO, M., LORD, A., et al. “Elastic optical networking: a new dawn for the optical layer?” *IEEE Communications Magazine*, v. 50, n. 2, pp. s12–s20, 2012. doi: 10.1109/MCOM.2012.6146481.
- [4] YILDIZ, B., KARASAN, O. “Regenerator Location Problem in Flexible Optical Networks”, *Operations Research*, v. 65, pp. 595–620, 2017. doi: 10.1287/opre.2016.1587.
- [5] WINZER, P. J., ESSIAMBRE, R.-J. “2 - Advanced optical modulation formats”. In: Kaminow, I. P., Li, T., Willner, A. E. (Eds.), *Optical Fiber Telecommunications V B (Fifth Edition)*, Optics and Photonics, fifth edition ed., Academic Press, pp. 23–93, Burlington, 2008. doi: <https://doi.org/10.1016/B978-0-12-374172-1.00002-3>. Disponível em: <<https://www.sciencedirect.com/science/article/pii/B9780123741721000023>>.
- [6] KARTALOPOULOS, S. V. *DWDM: Networks, Devices, and Technology*. Wiley-IEEE Press, 2002. ISBN: 9780471269052.
- [7] FOROUZAN, B. A. *Data Communications and Networking 5th Edition*. 5 ed. , McGraw Hill, 2012. ISBN: 9780073376226.
- [8] CHEN, S., LJUBIC, I., RAGHAVAN, S. “The Regenerator Location Problem”, *Networks*, v. 55, pp. 205–220, 05 2010. doi: 10.1002/net.20366.
- [9] CHEN, S., LJUBIĆ, I., RAGHAVAN, S. “The Generalized Regenerator Location Problem”, *INFORMS J. Comput.*, v. 27, pp. 204–220, 2015.

- [10] LUCENA, A., MACULAN, N., SIMONETTI, L. “Reformulations and solution algorithms for the maximum leaf spanning tree problem”, *Computational Management Science*, v. 7, n. 3, pp. 289–311, Jul 2010. ISSN: 1619-6988. doi: 10.1007/s10287-009-0116-5.
- [11] CHATTERJEE, B., OKI, E. *Elastic Optical Networks: Fundamentals, Design, Control and Management*. 1st ed. Boca Raton, CRC Press, aug 2020. ISBN: 9780429465284. doi: <https://doi.org/10.1201/9780429465284>.
- [12] ITU. “ITU Telecommunication Standardization Sector”. <https://www.itu.int/en/ITU-T/Pages/default.aspx>, 2020. Acessado em 6 de janeiro de 2024.
- [13] SHEN, G., TUCKER, R. S. “Translucent optical networks: the way forward [Topics in Optical Communications]”, *IEEE Communications Magazine*, v. 45, n. 2, pp. 48–54, 2007. doi: 10.1109/MCOM.2007.313394.
- [14] CHATTERJEE, B. C., BA, S., OKI, E. “Fragmentation Problems and Management Approaches in Elastic Optical Networks: A Survey”, *IEEE Communications Surveys and Tutorials*, v. 20, n. 1, pp. 183–210, 2018. doi: 10.1109/COMST.2017.2769102.
- [15] WALKOWIAK, K., KLINKOWSKI, M. “Shared backup path protection in elastic optical networks: Modeling and optimization”. In: *2013 9th International Conference on the Design of Reliable Communication Networks (DRCN)*, pp. 187–194, 2013.
- [16] BANERJEE, D., MUKHERJEE, B. “A practical approach for routing and wavelength assignment in large wavelength-routed optical networks”, *IEEE Journal on Selected Areas in Communications*, v. 14, n. 5, pp. 903 – 908, 1996. doi: 10.1109/49.510913.
- [17] CHATTERJEE, B. C., SARMA, N., OKI, E. “Routing and Spectrum Allocation in Elastic Optical Networks: A Tutorial”, *IEEE Communications Surveys and Tutorials*, v. 17, n. 3, pp. 1776–1800, 2015. doi: 10.1109/COMST.2015.2431731.
- [18] SHIRIN ABKENAR, F., GHAFFARPOUR RAHBAR, A. “Study and Analysis of Routing and Spectrum Allocation (RSA) and Routing, Modulation and Spectrum Allocation (RMSA) Algorithms in Elastic Optical Networks (EONs)”, *Optical Switching and Networking*, v. 23, pp. 5–39, 2017. ISSN: 1573-4277. doi: <https://doi.org/10.1016/j.osn.2016.08.003>.

- [19] WANG, Y., CAO, X., PAN, Y. “A study of the routing and spectrum allocation in spectrum-sliced elastic optical path networks”. In: *2011 Proceedings Ieee Infocom*, pp. 1503–1511. IEEE, 2011.
- [20] CHRISTODOULOPOULOS, K., TOMKOS, I., VARVARIGOS, E. A. “Elastic Bandwidth Allocation in Flexible OFDM-Based Optical Networks”, *Journal of Lightwave Technology*, v. 29, n. 9, pp. 1354–1366, 2011. doi: 10.1109/JLT.2011.2125777.
- [21] WANG, Y., CAO, X., HU, Q., et al. “Towards Elastic and Fine-Granular Bandwidth Allocation in Spectrum-Sliced Optical Networks”, *Journal of Optical Communications and Networking*, v. 4, 11 2012. doi: 10.1364/JOCN.4.000906.
- [22] SHIRIN ABKENAR, F., GHAFARPOUR RAHBAR, A. “Study and Analysis of Routing and Spectrum Allocation (RSA) and Routing, Modulation and Spectrum Allocation (RMSA) Algorithms in Elastic Optical Networks (EONs)”, *Optical Switching and Networking*, v. 23, pp. 5–39, 2017. ISSN: 1573-4277. doi: <https://doi.org/10.1016/j.osn.2016.08.003>.
- [23] RUIZ, L., DURÁN BARROSO, R. J., DE MIGUEL, I., et al. “Routing, Modulation and Spectrum Assignment Algorithm Using Multi-Path Routing and Best-Fit”, *IEEE Access*, v. 9, pp. 111633–111650, 2021. doi: 10.1109/ACCESS.2021.3101998.
- [24] KLINKOWSKI, M., WALKOWIAK, K. “Offline RSA algorithms for elastic optical networks with dedicated path protection consideration”. In: *2012 IV International Congress on Ultra Modern Telecommunications and Control Systems*, pp. 670–676, 2012. doi: 10.1109/ICUMT.2012.6459751.
- [25] WALKOWIAK, K., KLINKOWSKI, M. “Shared backup path protection in elastic optical networks: Modeling and optimization”. In: *2013 9th International Conference on the Design of Reliable Communication Networks (DRCN)*, pp. 187–194, 2013.
- [26] MOGHADDAM, E. E., BEYRANVAND, H., SALEHI, J. A. “Routing, Spectrum and Modulation Level Assignment, and Scheduling in Survivable Elastic Optical Networks Supporting Multi-Class Traffic”, *Journal of Lightwave Technology*, v. 36, n. 23, pp. 5451–5461, 2018. doi: 10.1109/JLT.2018.2874820.
- [27] CHATTERJEE, B. C., BA, S., OKI, E. “Fragmentation Problems and Management Approaches in Elastic Optical Networks: A Survey”, *IEEE*

Communications Surveys and Tutorials, v. 20, n. 1, pp. 183–210, 2018. doi: 10.1109/COMST.2017.2769102.

- [28] FUJII, S., HIROTA, Y., TODE, H., et al. “On-demand spectrum and core allocation for reducing crosstalk in multicore fibers in elastic optical networks”, *Journal of Optical Communications and Networking*, v. 6, n. 12, pp. 1059 – 1071, 2014. doi: 10.1109/JOCN.2014.6985898.
- [29] ÍTALO BRASILEIRO, COSTA, L., DRUMMOND, A. “A survey on challenges of Spatial Division Multiplexing enabled elastic optical networks”, *Optical Switching and Networking*, v. 38, pp. 100584, 2020. ISSN: 1573-4277. doi: <https://doi.org/10.1016/j.osn.2020.100584>.
- [30] MATA, J., DE MIGUEL, I., DURÁN, R. J., et al. “Artificial intelligence (AI) methods in optical networks: A comprehensive survey”, *Optical Switching and Networking*, v. 28, pp. 43–57, 2018. ISSN: 1573-4277. doi: <https://doi.org/10.1016/j.osn.2017.12.006>.
- [31] MARKOVIĆ, G. Z. “Routing and spectrum allocation in elastic optical networks using bee colony optimization”, *Photonic Network Communications*, v. 34, n. 3, pp. 356 – 374, 2017. doi: 10.1007/s11107-017-0706-z.
- [32] HAI, D. T., MORVAN, M., GRAVEY, P. “Combining heuristic and exact approaches for solving the routing and spectrum assignment problem”, *IET Optoelectronics*, v. 12, n. 2, pp. 65–72, 2018. doi: <https://doi.org/10.1049/iet-opt.2017.0013>.
- [33] VILLAMAYOR-PAREDES, M. M. R., MAIDANA-BENÍTEZ, L. V., COLBES, J., et al. “Routing, modulation level, and spectrum assignment in elastic optical networks. A route-permutation based genetic algorithms”, *Optical Switching and Networking*, v. 47, pp. 100710, 2023. ISSN: 1573-4277. doi: <https://doi.org/10.1016/j.osn.2022.100710>.
- [34] MILLS, D. L., BRAUN, H. W. “SIGCOMM ’87: Proceedings of the ACM workshop”. In: *Frontiers in computer communications technology*, pp. 191–196, aug 1987. doi: <https://doi.org/10.1145/55482.55502>.
- [35] O’MAHONY, M. “Results from the COST 239 project. Ultra-High Capacity Optical Transmission Networks”. In: *Proceedings of European Conference on Optical Communication*, v. 2, pp. 11–18 vol.2, 1996.
- [36] COLLE, D., VAN BREUSEGEM, E., VERBRUGGE, S., et al. “Chapter 7 Evolution of next-generation optical networks”. In: *Advanced Infrastructure*

for Photonic Networks, *Extended Final Report of COST Action 266*, The Faculty of Electrical Engineering and Computing, University of Zagreb, pp. 205–219, Zagreb, 2003. ISBN: 953-184-064-4.

- [37] CIVICIOGLU, P. “Backtracking Search Optimization Algorithm for numerical optimization problems”, *Applied Mathematics and Computation*, v. 219, n. 15, pp. 8121–8144, 2013. ISSN: 0096-3003. doi: <https://doi.org/10.1016/j.amc.2013.02.017>.
- [38] CACCHIANI, V., IORI, M., LOCATELLI, A., et al. “Knapsack problems — An overview of recent advances. Part II: Multiple, multidimensional, and quadratic knapsack problems”, *Computers and Operations Research*, v. 143, 2022. doi: 10.1016/j.cor.2021.105693. Disponível em: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85125650252&doi=10.1016%2fj.cor.2021.105693&partnerID=40&md5=a1d3a72900592ed74367becc0cc7db6b>.
- [39] HIFI, M., MICHRAFY, M. “Reduction strategies and exact algorithms for the disjunctively constrained knapsack problem”, *Computers & Operations Research*, v. 34, n. 9, pp. 2657–2673, 2007. ISSN: 0305-0548. doi: <https://doi.org/10.1016/j.cor.2005.10.004>.
- [40] GRIVA, I., NASH, S. G., SOFER, A. *Linear and Nonlinear Optimization (2. ed.)*. SIAM, 2008. ISBN: 978-0-89871-661-0.
- [41] GUROBI OPTIMIZATION, LLC. “Gurobi Optimizer Reference Manual”. 2023. Disponível em: <https://www.gurobi.com>.
- [42] CPLEX, I. I. “V12. 1: User’s Manual for CPLEX”, *International Business Machines Corporation*, v. 46, n. 53, pp. 157, 2009.
- [43] VAN ROSSUM, G., DRAKE, F. L. *Python 3 Reference Manual*. Scotts Valley, CA, CreateSpace, 2009. ISBN: 1441412697.
- [44] HAGBERG, A. A., SCHULT, D. A., SWART, P. J. “Exploring Network Structure, Dynamics, and Function using NetworkX”. In: Varoquaux, G., Vaught, T., Millman, J. (Eds.), *Proceedings of the 7th Python in Science Conference*, pp. 11 – 15, Pasadena, CA USA, 2008.
- [45] TANGE, O. “GNU Parallel - The Command-Line Power Tool”, *login: The USENIX Magazine*, v. 36, n. 1, pp. 42–47, Feb 2011. Disponível em: <http://www.gnu.org/s/parallel>.

- [46] BONAMI, P., LODI, A., TRAMONTANI, A., et al. “On mathematical programming with indicator constraints”, *Mathematical Programming*, v. 151, n. 1, pp. 191–223, Jun 2015. ISSN: 1436-4646. doi: 10.1007/s10107-015-0891-4. Disponível em: <<https://doi.org/10.1007/s10107-015-0891-4>>.

Apêndice A

Tabela A1: Segmentos x R_{\max} . NFSNET, 4000 km

orig-dest	rotas	Segmentos válidos variando R_{\max}													
		0	1	2	3	4	5	6	7	8	9	10	11	12	∞
1-2	58	2	4	15	96	460	1453	3098	4857	6115	6714	6894	6924	6926	6926
1-3	58	2	3	14	116	639	1996	4105	6228	7660	8307	8493	8523	8525	8525
1-4	64	1	5	22	173	863	2284	3998	5338	6035	6269	6315	6319	6319	6319
1-5	79	1	6	30	266	1269	3289	5650	7410	8248	8488	8524	8526	8526	8526
1-6	61	0	9	75	563	2120	4756	7531	9471	10382	10660	10710	10714	10714	10714
1-7	97	0	6	40	320	1487	4205	7962	11230	13068	13722	13856	13868	13868	13868
1-8	77	1	5	49	293	1154	3295	6730	10306	12788	13934	14270	14326	14330	14330
1-9	73	0	4	68	319	894	1863	3020	3964	4479	4663	4703	4707	4707	4707
1-10	107	0	6	63	471	2085	5881	11518	17095	20863	22597	23123	23219	23227	23227
1-11	92	0	8	136	875	2843	6083	9659	12410	13912	14495	14651	14677	14679	14679
1-12	101	0	7	123	654	1964	4327	7496	10499	12437	13259	13476	13508	13510	13510
1-13	99	0	6	129	788	2538	5402	8600	11134	12568	13142	13298	13324	13326	13326
1-14	101	0	7	125	665	2002	4416	7639	10673	12620	13443	13660	13692	13694	13694
2-3	58	2	4	19	156	596	1409	2568	3854	4890	5468	5678	5722	5726	5726
2-4	77	1	2	27	254	1007	2511	4539	6442	7692	8257	8423	8451	8453	8453
2-5	79	1	4	58	358	1156	2385	3653	4613	5175	5434	5524	5544	5546	5546
2-6	61	2	11	111	477	1160	2133	3311	4440	5233	5622	5748	5772	5774	5774

Tabela A1: Segmentos x R_{\max} . NFSNET, 4000 km

orig-dest	rotas	Segmentos válidos variando R_{\max}													
		0	1	2	3	4	5	6	7	8	9	10	11	12	∞
2-7	93	1	8	77	390	1263	2830	4776	6502	7568	7998	8098	8108	8108	8108
2-8	67	1	13	84	343	1058	2486	4445	6288	7483	8010	8162	8188	8190	8190
2-9	70	0	15	83	242	577	1141	1869	2553	2990	3166	3206	3210	3210	3210
2-10	104	0	13	98	479	1522	3429	5947	8375	10046	10839	11086	11132	11136	11136
2-11	95	1	19	193	753	1858	3454	5175	6560	7394	7763	7877	7899	7901	7901
2-12	101	0	21	152	516	1276	2604	4425	6275	7617	8285	8499	8538	8541	8541
2-13	99	0	18	174	654	1566	2983	4832	6616	7789	8298	8440	8464	8466	8466
2-14	101	0	21	154	523	1298	2647	4494	6372	7733	8408	8623	8662	8665	8665
3-4	66	1	5	42	229	678	1471	2557	3650	4431	4818	4944	4968	4970	4970
3-5	81	1	7	42	310	1176	2740	4660	6346	7389	7838	7970	7994	7996	7996
3-6	73	1	5	70	550	2028	4665	7806	10430	11982	12621	12795	12823	12825	12825
3-7	96	1	7	52	340	1431	3722	6715	9280	10736	11264	11374	11384	11384	11384
3-8	69	0	10	63	327	1236	3216	5927	8378	9890	10525	10699	10727	10729	10729
3-9	71	0	8	74	284	744	1493	2380	3146	3614	3800	3842	3846	3846	3846
3-10	111	1	6	71	448	1918	5420	10748	16183	19975	21770	22323	22423	22431	22431
3-11	90	0	16	176	870	2555	5182	7933	9889	10826	11116	11168	11172	11172	11172
3-12	102	0	14	145	629	1778	3770	6318	8664	10202	10906	11119	11157	11160	11160
3-13	101	0	11	157	808	2336	4686	7233	9180	10196	10539	10607	10613	10613	10613
3-14	102	0	14	147	641	1810	3846	6453	8849	10414	11126	11340	11378	11381	11381
4-5	63	1	7	72	263	635	1300	2304	3426	4286	4725	4867	4893	4895	4895

Tabela A1: Segmentos x R_{\max} . NFSNET, 4000 km

orig-dest	rotas	Segmentos válidos variando R_{\max}													
		0	1	2	3	4	5	6	7	8	9	10	11	12	∞
4-6	42	1	24	160	466	909	1346	1645	1781	1817	1821	1821	1821	1821	1821
4-7	70	1	12	84	330	795	1467	2335	3192	3774	4024	4084	4090	4090	4090
4-8	52	1	15	79	300	777	1452	2067	2430	2562	2588	2590	2590	2590	2590
4-9	54	2	17	59	165	352	650	1059	1482	1784	1922	1958	1962	1962	1962
4-10	80	1	17	113	460	1194	2277	3533	4719	5596	6074	6252	6292	6296	6296
4-11	91	1	26	154	512	1267	2620	4594	6712	8300	9110	9376	9426	9430	9430
4-12	80	2	27	119	367	888	1774	2935	3992	4619	4849	4896	4900	4900	4900
4-13	78	0	32	163	478	1077	2117	3621	5257	6523	7193	7423	7469	7473	7473
4-14	80	2	28	122	377	908	1804	2971	4029	4656	4886	4933	4937	4937	4937
5-6	60	1	17	114	349	820	1748	3157	4631	5670	6159	6307	6333	6335	6335
5-7	66	1	6	48	191	545	1245	2463	4053	5440	6226	6500	6552	6556	6556
5-8	66	1	9	54	176	491	1291	2580	3829	4579	4849	4901	4905	4905	4905
5-9	64	1	12	50	146	358	731	1251	1756	2070	2188	2212	2214	2214	2214
5-10	98	2	13	84	319	917	2272	4594	7299	9372	10395	10708	10762	10766	10766
5-11	83	2	31	138	425	1077	2208	3556	4608	5136	5299	5327	5329	5329	5329
5-12	92	2	25	102	296	754	1646	2993	4491	5663	6279	6483	6521	6524	6524
5-13	90	2	29	137	379	894	1916	3439	4938	5863	6206	6276	6282	6282	6282
5-14	92	2	26	104	300	771	1689	3072	4607	5803	6427	6632	6670	6673	6673
6-7	66	1	17	97	354	894	1835	3098	4272	5002	5290	5354	5360	5360	5360
6-8	48	2	19	95	353	943	1766	2484	2885	3023	3049	3051	3051	3051	3051

Tabela A1: Segmentos x R_{\max} . NFSNET, 4000 km

orig-dest	rotas	Segmentos válidos variando R_{\max}													
		0	1	2	3	4	5	6	7	8	9	10	11	12	∞
6-9	46	3	18	67	195	412	734	1182	1667	2023	2185	2225	2229	2229	2229
6-10	46	1	16	70	234	567	1091	1851	2782	3623	4141	4343	4387	4391	4391
6-11	52	4	44	166	510	1283	2457	3648	4481	4883	5011	5035	5037	5037	5037
6-12	60	4	30	114	379	979	1893	2839	3495	3783	3853	3860	3860	3860	3860
6-13	67	3	25	123	456	1268	2638	4269	5635	6419	6713	6777	6783	6783	6783
6-14	60	4	30	116	386	995	1922	2884	3550	3840	3910	3917	3917	3917	3917
7-8	66	1	3	41	182	560	1522	3399	5767	7686	8686	9006	9062	9066	9066
7-9	82	1	7	42	157	456	1036	1877	2750	3325	3547	3593	3597	3597	3597
7-10	120	2	9	81	381	1147	2737	5370	8540	11160	12595	13089	13185	13193	13193
7-11	104	4	30	128	439	1225	2863	5229	7524	9017	9660	9836	9864	9866	9866
7-12	114	3	21	91	307	921	2236	4163	6148	7577	8256	8453	8484	8486	8486
7-13	112	2	26	113	362	1064	2589	4888	7217	8771	9441	9621	9649	9651	9651
7-14	114	3	21	92	311	937	2275	4230	6243	7685	8366	8563	8594	8596	8596
8-9	70	1	5	42	219	738	1730	2989	4047	4638	4850	4894	4898	4898	4898
8-10	92	1	9	84	390	1293	3142	5735	8261	9991	10813	11071	11119	11123	11123
8-11	78	4	24	108	430	1457	3514	6214	8628	10122	10754	10928	10956	10958	10958
8-12	86	3	16	78	350	1190	2788	4713	6190	6896	7097	7128	7130	7130	7130
8-13	86	4	21	82	372	1319	3283	5929	8324	9813	10444	10618	10646	10648	10648
8-14	86	3	16	79	356	1212	2841	4799	6290	6998	7199	7230	7232	7232	7232
9-10	46	1	10	52	181	451	876	1517	2346	3133	3635	3835	3879	3883	3883

Tabela A1: Segmentos x R_{\max} . NFSNET, 4000 km

orig-dest	rotas	Segmentos válidos variando R_{\max}													
		0	1	2	3	4	5	6	7	8	9	10	11	12	∞
9-11	48	4	16	52	138	322	677	1150	1546	1745	1799	1805	1805	1805	1805
9-12	61	3	9	32	115	337	789	1527	2448	3257	3717	3873	3901	3903	3903
9-13	54	4	14	31	88	228	463	849	1385	1834	2038	2084	2088	2088	2088
9-14	61	3	9	33	120	349	814	1576	2522	3343	3805	3961	3989	3991	3991
10-11	84	4	30	136	486	1499	3564	6403	9083	10828	11591	11801	11833	11835	11835
10-12	96	3	21	94	366	1151	2741	5049	7500	9385	10395	10746	10816	10822	10822
10-13	96	5	22	97	416	1359	3286	6170	9331	11773	13036	13448	13524	13530	13530
10-14	96	3	21	96	374	1175	2790	5137	7637	9559	10583	10936	11006	11012	11012
11-12	64	3	12	62	229	630	1429	2659	3994	4983	5472	5625	5652	5654	5654
11-13	58	4	18	71	212	526	1111	1925	2711	3214	3414	3458	3462	3462	3462
11-14	64	3	12	63	233	641	1449	2684	4020	5009	5498	5651	5678	5680	5680
12-13	64	3	10	39	148	451	1061	2019	3213	4303	4965	5210	5259	5263	5263
12-14	67	3	10	42	154	460	1080	2006	2974	3616	3864	3914	3918	3918	3918
13-14	64	3	10	39	150	458	1077	2048	3258	4358	5022	5267	5316	5320	5320
Total		150	1332	7962	33012	98671	221125	384448	539396	642568	689478	703270	705616	705790	705790
Média		1.65	14.64	87.50	362.77	1084.30	2429.95	4224.70	5927.43	7061.19	7576.68	7728.24	7754.02	7755.93	7755.93