

UTILIZAÇÃO DE FLUXO MÁXIMO PARA ORDENAÇÃO DE REQUISIÇÃO DO PROBLEMA DE ROTEAMENTO E ALOCAÇÃO DE COMPRIMENTOS DE ONDA

Artur de Alvarenga Silva
Universidade Federal de Ouro Preto
Rua 36, n.115, Loanda, João Monlevade
artur.alvarenga@aluno.ufop.edu.br

Bruna Silva de Moraes
Universidade Federal de Ouro Preto
Rua 36, n.115, Loanda, João Monlevade
bruna.morais1@aluno.ufop.edu.br

Alexandre Xavier Martins
Universidade Federal de Ouro Preto
Rua 36, n.115, Loanda, João Monlevade
xmartins@decea.ufop.br

Thiago Augusto de Oliveira Silva
Universidade Federal de Ouro Preto
Rua 36, n.115, Loanda, João Monlevade
thiago@deenp.ufop.br

RESUMO

A ideia principal da pesquisa é a implementação do algoritmo de fluxo máximo para ordenação das requisições no problema de roteamento e alocação de comprimentos de onda *Routing and Wavelength Assignment* (RWA). Na literatura o problema consiste em atender as demandas definidas na topologia virtual, sendo possível destacar duas abordagens. A primeira variante é o MIN-RWA, no qual o objetivo geral é atender as requisições com o menor número de comprimentos de onda e a outra variação é o MAX-RWA, no qual tem a finalidade de maximizar o número de requisições atendidas com um número fixo de comprimentos de ondas. Neste estudo será considerado o MIN-RWA.

Este estudo aborda uma adaptação do método proposto por [1], esse consiste na ordenação através do tamanho do caminho mínimo de cada requisição. Através da implementação do método de fluxo máximo, pode-se propor dois novos métodos que foram as combinações dos algoritmos de fluxo máximo e o do caminho mínimo, junto a isso foram implementadas três maneiras para definir o método de criação dos grafos. Os resultados obtidos com as variações foram confrontados com o que já está definido na literatura a fim de mensurar os ganhos do presente trabalho.

Palavra-chave: Otimização Combinatória, Telecomunicações, Métodos Heurísticos.

ABSTRACT

The main idea of the research is the implementation of the maximum flow algorithm for ordering requests in the Routing and Wavelength Assignment (RWA) problem. In the literature the problem consists in meeting the demands defined in the virtual topology, being possible to highlight two approaches. The first variant is MIN-RWA, where the overall goal is to fulfill requests with the shortest number of wavelengths and the other variation is MAX-RWA, which aims to maximize the number of requests met with a fixed number of wavelengths. In this study will be considered the MIN-RWA.

This study addresses an adaptation of the method proposed by cite skorin2007routing, which consists of ordering through the minimum path length of each request. By implementing the maximum flow method, one can propose two new methods which were the combinations of the maximum flow algorithms and the minimum path algorithms, along with three ways to define the graph creation method. The results obtained with the variations were compared with what is already defined in the literature in order to measure the gains of the present work.

Keywords: Combinatorial Optimization, Telecommunications, Heuristic Methods.

Como Citar:

SILVA, A.A.; MORAIS, B.S.; MARTINS, A.X.; SILVA, T.A.O. Utilização de Fluxo Máximo para Ordenação de Requisição do Problema de Roteamento e Alocação de Comprimentos de Onda. In: SIMPÓSIO DE PESQUISA OPERACIONAL E LOGÍSTICA DA MARINHA, 19., 2019, Rio de Janeiro, RJ. **Anais** [...]. Rio de Janeiro: Centro de Análises de Sistemas Navais, 2019.

1. INTRODUÇÃO

A demanda para utilização de redes ópticas é advinda do grande crescimento na área de telecomunicações. O diferencial é que a elevada capacidade de transmissão e alto grau de confiabilidade são características fundamentais para atender ao cenário de expansão dos usuários de telecomunicações em geral.

O problema *Routing and Wavelength Assignment* (RWA) na visão de [2] pode ser definido como casos de tráfego dinâmico onde o objetivo é configurar *lightpath* e atribuir comprimentos de onda de uma maneira que minimize a quantidade de caminhos, ou que maximiza o número de conexões que são estabelecidos na rede a qualquer momento.

Segundo Martins [3], para atender cada requisição entre sua origem e destino, inicialmente deve ser determinado um caminho direcionado e posteriormente associar um comprimento de onda a essa requisição. Uma das restrições é que se duas requisições utilizam um mesmo arco (i, j) , elas devem ser alocadas em comprimentos de onda diferentes.

É possível separar o problema em duas linhas de trabalho. O MAX-RWA, que busca maximizar o número de requisições que podem ser alocadas a um determinado número de comprimentos de onda, e o MIN-RWA, que tem foco em minimizar o número de comprimentos de onda para alocar um montante de requisições. Em relação às requisições

também são possíveis duas abordagens do problema, o estático no qual as requisições já estão pré-estabelecidas, e o dinâmico em que o número de comprimentos de onda (λ) é fixo e as requisições chegam ao longo do tempo.

O RWA tem sido objeto de estudo em diversos trabalhos que buscam por meio de heurísticas e de métodos exatos aprimorar os resultados já existentes.

Neste trabalho abordamos o MIN-RWA estático e propomos um aprimoramento no algoritmo BFD proposto por Skorin-Kapov [1]. A divisão do trabalho está organizada como segue. Na seção 1 apresentamos uma breve introdução ao problema, na seção 2 apresentamos uma descrição formal, na seção 3 apresentamos uma revisão de trabalhos mais recentes sobre o RWA, os métodos propostos são apresentados na seção 4, os resultados dos testes computacionais na seção 5 e, por fim, o trabalho é concluído na seção 6.

2. DESCRIÇÃO FORMAL DO PROBLEMA

Seja $G = (V; E)$ onde V representa o conjunto de vértices (comutadores), tal que $v = |V|$, e E o conjunto de arestas (fibras óticas). Nós denotamos por Δ o conjunto de requisições por caminhos óticos onde cada $r \in \Delta$ é definido por um par origem-destino $(s_r; d_r) \in V \times V$. Note que podemos ter duas ou mais diferentes requisições com o mesmo par origem-destino e, neste caso, se elas são roteadas com o mesmo comprimento de onda, então elas devem ser roteadas através de caminhos disjuntos por arcos em G .

Para atender cada requisição entre os pares, origem e destino, necessitamos determinar um caminho direcionado e alocar um comprimento de onda a esta requisição. A restrição que se impõe é que um mesmo comprimento de onda não pode ser utilizado para atender duas ou mais requisições diferentes, se os caminhos dessas requisições utilizam um mesmo arco. Além disso, podemos ter a restrição de continuidade, onde o comprimento de onda alocado a um caminho deve ser utilizado do início ao fim deste caminho, sem conversão.

No MIN-RWA o nosso objetivo é minimizar o número de comprimentos de onda utilizados, ou seja, encontrar a mínima partição de Δ em W subconjuntos tal que as requisições em cada $\Delta_w, w = 1, \dots, W$, podem ser roteadas em caminhos disjuntos por arcos em G .

3. REVISÃO DE LITERATURA

Na literatura são encontradas diversas estratégias para a resolução do MIN-RWA. Algumas dessas estratégias decompõem o problema em duas partes ([4], [5], [6]). Primeiramente é calculada uma rota para cada conexão, em seguida são atribuídos comprimentos de onda para cada uma dessas rotas, de acordo com uma solução de um problema de coloração de grafos ([7]). Outras estratégias tratam dos dois problemas em conjunto ([1] e [8]).

Skorin-Kapov [1] propõe quatro heurísticas construtivas similares baseadas em algoritmos utilizados na resolução do *Bin Packing* ([9]), *First Fit* (FF), *Best Fit* (BF), *First Fit Decreasing* (FFD) e *Best Fit Decreasing* (BFD). As heurísticas FF e BF ordenam aleatoriamente os itens e percorrem o conjunto de *bins* em uma ordem sequencial alocando os itens. O algoritmo FF coloca o item no primeiro *bin* possível enquanto o algoritmo BF

analisa cada *bin* e aloca o item naquele considerado "melhor". No caso do RWA, cada requisição é considerada um item. O conceito de melhor *bin* muda para o problema original e na abordagem adotada para a resolução do RWA. No *Bin Packing* clássico o melhor *bin* é aquele que fica com menos espaço vazio após a alocação de um item, enquanto no RWA o melhor *bin* é aquele que aloca uma requisição utilizando o menor número de arcos [1]. Os algoritmos FFD e BFD ordenam os itens em ordem decrescente de tamanho, de acordo com o tamanho do caminho mínimo da requisição, e depois alocam esses itens da mesma maneira dos algoritmos FF e BF respectivamente.

Algoritmo 1: BFD

Entrada: Δ^O, G
Saída: $S, |w|$

- 1 **início**
- 2 $d \leftarrow \max(\text{diâmetro}(G), \sqrt{|E|});$
- 3 $w \leftarrow$ Cópia de G ;
- 4 $S \leftarrow \{\emptyset\}; \{\text{Conjunto que guarda a solução}\}$
- 5 **para** $i = 1$ até $|\Delta|$ **faça**
- 6 **se** \nexists caminho disjunto $\leq d$ em nenhuma cópia de $G \in w$ **então**
- 7 $w \leftarrow$ Nova cópia de G ;
- 8 **fim**
- 9 Encontre a j -ésima cópia de $G \in w$ em que a i -ésima requisição possui o menor caminho;
- 10 $c \leftarrow$ menor caminho para i no grafo G_j ;
- 11 $S \leftarrow S \cup \{i, c, j\}$;
- 12 Remova os arcos usados em c de G_j ;
- 13 **fim**
- 14 **fim**

Noronha et al. [10] melhoraram o desempenho dos métodos apresentados em Skorin-Kapov [1], trabalhando na estrutura de dados e em estratégias de implementação, tais como, lista de adjacência duplamente encadeada e atualização dinâmica dos caminhos mínimos. Os melhores resultados são observados com uma variante do algoritmo BFD, em que as requisições são ordenadas em ordem decrescente de acordo com o tamanho do menor caminho (tamanho do caminho definido pelo número de arestas) entre a origem e o destino no grafo original. Iterativamente para cada demanda e cada comprimento de onda já atribuído é checado se a requisição pode ser roteada com este mesmo comprimento de onda. O melhor comprimento de onda é definido como aquele com o menor caminho viável entre a origem e o destino. Se nenhum comprimento de onda pode ser atribuído a esta demanda ou se todos os menores caminhos são maiores que um determinado limite, um novo comprimento de onda é gerado e atribuído a esta demanda. A dificuldade do algoritmo BFD [1] é que muitas requisições apresentam o mesmo tamanho. Sendo assim as requisições de mesmo tamanho são ordenadas aleatoriamente.

Para contornar o fato de o BFD ordenar muitas requisições de maneira aleatória, Noronha et al. [10] propuseram um algoritmo genético com chaves randômicas para ordenar as requisições. Os elementos são ordenados segundo seu tamanho e o valor da chave. Os cromossomos são vetores de números reais, denominadas chaves. A cada requisição é associada uma chave de valor entre $[0; 1]$; na população inicial esse valor é gerado aleatoriamente. As chaves são usadas para a decodificação gerando as soluções viáveis. As

requisições são ordenadas de acordo com a soma do tamanho do seu caminho mínimo, entre a origem e o destino, e o valor da chave, depois, então o BFD é aplicado. A resposta do BFD, o número de comprimentos de onda, é usada como *fitness* de cada indivíduo. Experimentos computacionais foram conduzidos sobre as instâncias mais estudadas na literatura bem como as novas instâncias introduzidas em [10].

Martins et al. [11] utilizam a metaheurística ILS combinada com uma busca local baseada no VND. Como solução inicial os autores também utilizam o BFD como proposto na literatura.

O pseudocódigo do BFD é apresentado no Algoritmo 1. O BFD recebe como entrada o grafo G e Δ^O que é uma lista ordenada das requisições em ordem decrescente em relação ao tamanho do caminho mínimo da requisição. Inicialmente, na linha 2, é calculado o parâmetro d que é o valor máximo entre o diâmetro do grafo (maior distância entre qualquer par de vértices) e a raiz quadrada do número de arestas. Na linha 3 é criada a primeira cópia de G , que significa o primeiro comprimento de onda usado. Na quarta linha S é inicializado como um conjunto vazio, ou seja, nenhuma requisição foi alocada a nenhum comprimento de onda. Na linha 5 se inicia a alocação das requisições percorrendo a lista Δ^O . Para cada requisição é verificado há existência ou não em todos os comprimentos de onda já criados de um caminho disjunto que tenha o caminho menor ou igual a d . Caso exista um ou mais caminhos que atendam essa restrição, o menor deles é selecionado e é incluído em S o índice i da requisição, o caminho usado pela requisição e o índice do grafo (comprimento de onda) em que o caminho foi encontrado, linha 11. Na linha 12 os arcos utilizados no grafo são removidos. Como saída são retornados a solução S e o número de comprimentos de onda utilizados.

O objetivo deste trabalho é a adaptação do BFD proposto por [1], com a implementação do algoritmo de fluxo máximo combinado com métodos de criação de grafos, visando melhorar a geração de soluções iniciais para o RWA. O benefício que as soluções iniciais podem agregar, é que obtendo resultados melhores o desempenho das heurísticas de aprimoramento também pode ser melhorado.

4. MÉTODOS PROPOSTOS

Como visto anteriormente, uma das dificuldades do BFD em obter melhores resultados é o fato de muitas requisições terem o mesmo tamanho de caminho mínimo e com isso a ordem de alocação acaba sendo muito influenciada por fatores aleatórios. Com isso, surgiu a ideia de se utilizar a informação do Fluxo Máximo entre os pares origem-destino de cada requisição, partindo do pressuposto de que uma requisição com maior Fluxo Máximo possui um número maior de caminhos disjuntos em arcos, possibilitando maiores opções de escolha. Sendo assim, para cada requisição é calculado o valor do Fluxo Máximo e essa informação é usada para definir a ordem em que cada requisição será alocada.

4.1. Combinação do Fluxo Máximo e Caminho Mínimo

Com os resultados do fluxo máximo e caminho mínimo é possível aplicar dois métodos de ordenação na lista de requisição. No primeiro método a ordenação é primeiramente realizada levando em consideração o valor do fluxo máximo, então requisições com valores menores ficam nas primeiras posições, em caso de empate no valor do Fluxo

Máximo as requisições são priorizadas pelo tamanho do caminho mínimo, sempre respeitando a ordem decrescente. A outra alternativa é considerar o tamanho do caminho mínimo de cada requisição e caso possuam valores iguais será analisado o valor do Fluxo Máximo de cada uma.

4.2. Criação Grafos

Cada grafo representa uma cópia do grafo original e defini um novo comprimento de onda onde as requisições devem ser alocadas. No decorrer da pesquisa além da implementação dos diferentes métodos de ordenação de requisições, foi possível a realização de três formas para que seja definido como será a criação dos grafos.

4.2.1. Não Limitando a Criação de Grafos

A criação de grafos nesta metodologia é de acordo com a demanda das requisições, não tendo nenhuma limitação para a quantidade de grafos que são criados. As requisições são alocadas de acordo com o algoritmo que foi proposto por [1], onde as requisições são atendidas caso o comprimento de onda encontrado seja menor que o diâmetro do Grafo ($d = \text{Diâmetro}(G)$), caso não encontre nenhum comprimento de onda compatível com esta restrição em nenhum dos grafos é criado um novo comprimento de onda, como demonstrado no Algoritmo 1.

4.2.2. Limitando a Criação de Grafos

Nesta variação do BFD a primeira iteração ocorre conforme descrito no Algoritmo 1. Esta primeira iteração serve para fornecer um limite superior para o método. A partir da primeira iteração para cada novo grafo (comprimento de onda) criado é verificado se o limite já foi alcançado, caso a resposta seja verdadeira a iteração é interrompida e passa-se para uma nova iteração, caso contrário a alocação é realizada e continua-se o método. O limite é atualizado sempre que se encontrar uma solução com um número menor de comprimentos de onda.

4.2.3. Criando os Grafos Antes de Atender as Requisições

Nesta variação também utilizamos a primeira iteração do Algoritmo 1 para gerar um limite superior (L). A diferença aqui é que com o limite superior disponível, nas iterações seguintes sempre será gerada o número de grafos igual a $L - 1$ no início da iteração. Com essa variação esperasse que as primeiras requisições alocadas, sejam sempre alocadas utilizando um caminho que seja do tamanho de seu caminho mínimo e com isso se obtenha soluções melhores. Assim como no método anterior o limite é atualizado sempre que se encontrar uma solução com um número menor de comprimentos de onda.

5. TESTES COMPUTACIONAIS

Os resultados foram obtidos a partir de três métodos de criação de grafos: a primeira é não limitando a criação (NLC), a segunda é a limitando a criação (LC) e a terceira é a criação dos grafos antes da alocação das requisições (CGA). Em cada método de geração de grafos foi implementado quatro maneiras se fazer a ordenação da lista de requisições: considerando somente o caminho mínimo (CM) proposto por [1], considerando somente o fluxo máximo (FM), combinação de fluxo máximo primeiro e depois o caminho mínimo (FM-CM) e combinação de caminho mínimo primeiro e depois o fluxo máximo (CM-FM). Os testes computacionais foram realizados em um computador com processador Intel Core i7 com 2.40 GHz com 8 GB de RAM, executando Microsoft Windows 10 e os algoritmos foram programados em C++.

As instâncias utilizadas nos testes foram as mesmas apresentadas em [11] e [10]. Na Tabela 1 são apresentadas a instância, a quantidade de vértices, a quantidade de arestas e de requisições.

Tabela 1: Instâncias Utilizadas

Instância	Vértices	Arestas	Requisições	Instância	Vértices	Arestas	Requisições
att	90	137	359	mesh4x25.20	100	200	1975
att2	71	175	2918	mesh4x25.40	100	200	3894
brasil	27	70	13707	mesh4x25.60	100	200	5967
eon	20	39	373	mesh4x25.80	100	200	7959
fin	31	51	930	mesh4x25.100	100	200	9900
giul	39	86	14732	mesh5x20.20	100	200	1975
nsf3	14	21	285	mesh5x20.40	100	200	3894
nsf12	14	21	551	mesh5x20.60	100	200	5967
nsf212	14	22	551	mesh5x20.80	100	200	7959
sun	27	51	952	mesh5x20.100	100	200	9900
zib54	54	80	12230	mesh6x17.20	100	204	1975
y3-40-5	100	175	3996	mesh6x17.40	100	204	3894
y3-60-5	100	175	5990	mesh6x17.60	100	204	5967
y3-80-1	100	172	7959	mesh6x17.80	100	204	7959
y3-80-5	100	175	7924	mesh6x17.100	100	204	10302
y4-20-4	100	221	1989	mesh8x13.20	100	208	1975
y4-60-5	100	215	5990	mesh8x13.40	100	208	3894
y4-80-1	100	220	7959	mesh8x13.60	100	208	5967
y4-80-5	100	215	7924	mesh8x13.80	100	208	7959
y4-100-1	100	220	9900	mesh8x13.100	100	208	10712
y5-60-1	100	285	5967	mesh10x10.20	100	200	1975
y5-80-1	100	285	7959	mesh10x10.40	100	200	3894
y5-80-2	100	252	7911	mesh10x10.60	100	200	5997
y5-100-1	100	285	9900	mesh10x10.80	100	200	7959
y5-100-2	100	252	9900	mesh10x10.100	100	200	9900

Para a obtenção dos resultados, cada instância foi submetida a todos os métodos de criação de grafos e de ordenação da lista de requisição, sendo assim foram executadas onze vezes em cada variação com uma limitação de cinco minutos. A partir dos resultados foram feitos os cálculos dos *gaps* em percentual utilizando a fórmula $(UB-LB)/LB$, onde LB representa o menor valor encontrado para aquela instância e UB o maior encontrado pelo método.

Com os resultados obtidos é possível realizar dois tipos comparações, a primeira comparação é feita para os métodos de criação de grafos, sendo observado em qual meto-

dologia obteve melhores resultados e qual foi o *gap* dos dados apresentados. A segunda comparação é feita entre as diferentes maneiras de ordenação da lista de requisição realizadas no presente trabalho, confrontando com o método que foi proposto por [1].

Os métodos de criação de grafo possuem grande influência sobre os resultados. Na Tabela 2 são apresentados os resultados para cada instância, considerando a variação dos métodos utilizados para criar grafos em que o método de ordenação utilizado nas requisições foi o critério do caminho mínimo. Na primeira coluna é apresentado o nome da instância, na coluna 2 e 3 são apresentados o *gap* e o número de comprimentos onda encontrados para cada instância para o BFD da literatura (NLC), as colunas seguintes apresentam a mesma informação para as variações LC e CGA. A implementação do método que limita a criação de grafos trouxe ganhos em relação a quantidade de grafos necessário para alocar as requisições e um diferencial quando analisado o *gap* médio das soluções. Além disso o critério que cria os grafos antes da alocação apresentou resultados satisfatórios quanto comparado com o método que não limita a criação, e isso reflete nos *gaps* médios que também foram satisfatórios.

Tabela 2: Métodos de Criação de Grafos (Ordenação CM)

Instância	NLC		LC		CGA		Instância	NLC		LC		CGA	
	<i>gap</i> (%)	<i>w</i>	<i>gap</i> (%)	<i>w</i>	<i>gap</i> (%)	<i>w</i>		<i>gap</i> (%)	<i>w</i>	<i>gap</i> (%)	<i>w</i>	<i>gap</i> (%)	<i>w</i>
att	16,00	27	4,00	25	12,00	27	mesh4x25.20	2,94	69	0,00	68	1,47	69
att2	1,77	114	0,88	113	0,00	113	mesh4x25.40	2,33	129	0,00	129	2,33	129
brasil	0,00	48	0,00	48	0,00	48	mesh4x25.60	2,05	197	1,03	196	0,51	195
eon	0,00	22	0,00	22	0,00	22	mesh4x25.80	1,92	261	0,77	260	0,00	260
fin	4,26	47	0,00	47	2,13	47	mesh4x25.100	1,26	319	0,95	319	0,32	318
giul	2,83	397	2,06	396	2,57	397	mesh5x20.20	3,70	55	1,85	54	1,85	55
nsf3	10,71	29	0,00	28	0,00	28	mesh5x20.40	1,94	103	0,00	103	1,94	103
nsf12	5,00	40	0,00	40	10,00	41	mesh5x20.60	1,27	158	0,64	157	0,00	157
nsf212	12,82	40	0,00	39	10,26	40	mesh5x20.80	1,93	208	0,97	208	0,48	207
sun	10,17	62	3,39	60	8,47	62	mesh5x20.100	1,18	256	0,79	255	0,39	254
zib54	8,20	600	7,13	594	8,20	596	mesh6x17.20	2,17	46	0,00	46	2,17	46
y3-40-5	3,23	63	1,61	63	1,61	62	mesh6x17.40	2,30	88	1,15	87	2,30	88
y3-60-5	3,33	92	2,22	91	1,11	90	mesh6x17.60	2,27	133	0,76	133	0,76	132
y3-80-1	1,57	127	0,79	127	1,57	128	mesh6x17.80	1,14	177	0,57	176	0,00	176
y3-80-5	2,54	120	1,69	120	1,69	118	mesh6x17.100	2,28	222	1,37	222	0,00	219
y4-20-4	9,52	22	4,76	22	4,76	22	mesh8x13.20	2,78	36	0,00	36	0,00	36
y4-60-5	6,78	63	6,78	62	1,69	59	mesh8x13.40	2,99	68	1,49	67	1,49	68
y4-80-1	8,11	79	6,76	79	0,00	74	mesh8x13.60	1,98	102	0,99	102	2,97	102
y4-80-5	6,49	81	5,19	81	1,30	77	mesh8x13.80	1,48	136	0,74	135	1,48	135
y4-100-1	7,69	97	6,59	97	1,10	91	mesh8x13.100	4,71	176	3,53	176	0,59	170
y5-60-1	5,26	40	2,63	39	0,00	38	mesh10x10.20	19,35	36	6,45	33	0,00	31
y5-80-1	6,12	52	6,12	51	4,08	49	mesh10x10.40	23,21	68	8,93	61	1,79	56
y5-80-2	1,61	63	1,61	62	1,61	62	mesh10x10.60	22,62	102	9,52	91	1,19	84
y5-100-1	6,67	63	5,00	63	1,67	60	mesh10x10.80	9,91	121	9,01	121	0,00	111
y5-100-2	1,30	77	0,00	77	1,30	77	mesh10x10.100	10,29	149	9,56	148	0,00	136
<i>Gap</i> médio	5,68		2,77		3,09		<i>Gap</i> médio	5,20		2,44		0,96	

A tabela 3 trazem os resultados obtidos com o método de criação de grafos NLC, em que agora há a variação no método de ordenação das requisições. Relembrando aqui que a estratégia NLC, com a ordenação CM é o BFD tradicional da literatura.

Tabela 3: Não limitando grafos - NLC

Ordenação Instancia	CM		FM		FM-CM		CM-FM		Ordenação Instancia	CM		FM		FM-CM		CM-FM	
	gap(%)	w	gap(%)	w	gap(%)	w	gap(%)	w		gap(%)	w	gap(%)	w	gap(%)	w	gap(%)	w
att	16,00	27	12,00	26	12,00	26	12,00	27	mesh4x25.20	2,94	69	4,41	69	2,94	69	2,94	69
att2	1,77	114	1,77	113	0,88	114	1,77	114	mesh4x25.40	2,33	129	3,88	132	1,55	130	2,33	130
brasil	0,00	48	2,08	48	4,17	48	0,00	48	mesh4x25.60	2,05	197	3,59	200	2,05	196	2,05	197
eon	0,00	22	4,55	22	0,00	22	0,00	22	mesh4x25.80	1,92	261	2,69	266	2,31	265	1,92	261
fin	4,26	47	6,38	48	6,38	48	2,13	47	mesh4x25.100	1,26	319	3,79	327	3,79	327	1,26	319
giul	2,83	397	1,80	392	2,06	392	2,57	397	mesh5x20.20	3,70	55	5,56	56	3,70	55	3,70	55
nsf3	10,71	29	10,71	28	7,14	28	10,71	28	mesh5x20.40	1,94	103	3,88	106	1,94	104	1,94	103
nsf12	5,00	40	12,5	40	10,00	41	10,00	41	mesh5x20.60	1,27	158	3,18	160	1,27	157	1,27	158
nsf212	12,82	40	15,38	40	7,69	39	10,26	40	mesh5x20.80	1,93	208	3,38	212	2,42	210	2,90	210
sun	10,17	62	3,39	59	3,39	59	6,78	62	mesh5x20.100	1,18	256	3,54	261	3,54	261	1,57	256
zib54	8,20	600	1,25	561	2,32	564	8,56	600	mesh6x17.20	2,17	46	4,35	47	4,35	46	2,17	46
y3-40-5	3,23	63	11,29	66	4,84	63	3,23	63	mesh6x17.40	2,30	88	3,45	89	2,30	88	2,30	88
y3-60-5	3,33	92	10,00	96	4,44	92	3,33	92	mesh6x17.60	2,27	133	2,27	134	0,76	132	1,52	133
y3-80-1	1,57	127	7,87	134	7,87	128	1,57	128	mesh6x17.80	1,14	177	2,27	178	0,57	176	1,14	176
y3-80-5	2,54	120	8,47	125	5,93	120	2,54	119	mesh6x17.100	2,28	222	4,57	228	4,57	228	1,83	222
y4-20-4	9,52	22	14,29	23	14,29	22	4,76	22	mesh8x13.20	2,78	36	5,56	37	5,56	36	2,78	36
y4-60-5	6,78	63	13,56	66	10,17	63	6,78	63	mesh8x13.40	2,99	68	5,97	70	2,99	68	2,99	68
y4-80-1	8,11	79	13,51	83	9,46	80	8,11	79	mesh8x13.60	1,98	102	5,94	106	2,97	102	1,98	102
y4-80-5	6,49	81	12,99	85	9,09	82	6,49	81	mesh8x13.80	1,48	136	4,44	139	1,48	135	1,48	135
y4-100-1	7,69	97	13,19	101	12,09	99	7,69	97	mesh8x13.100	4,71	176	9,41	184	8,82	184	4,12	176
y5-60-1	5,26	40	13,16	42	15,79	40	5,26	40	mesh10x10.20	19,35	36	16,13	35	12,90	33	9,68	33
y5-80-1	6,12	52	14,29	54	10,20	52	6,12	51	mesh10x10.40	23,21	68	17,86	65	14,29	62	10,71	61
y5-80-2	1,61	63	9,68	66	4,84	63	1,61	63	mesh10x10.60	22,62	102	17,86	98	11,90	92	10,71	91
y5-100-1	6,67	63	13,33	66	11,67	64	6,67	63	mesh10x10.80	9,91	121	18,02	130	12,61	122	9,91	121
y5-100-2	1,30	77	10,39	81	6,49	79	1,30	77	mesh10x10.100	10,29	149	18,38	160	16,91	155	10,29	149
Gap médio	5,68		9,51		7,33		5,21		Gap médio	5,20		6,98		5,14		3,82	

Analisando os dados que foram obtidos com a metodologia de NLC, em 32% das instâncias disponíveis, houve melhora em relação à ordenação feita exclusivamente pelo caminho mínimo (CM), o *gap* obtido pela combinação de CM-FM foi o menor no grupo 1 de instâncias, mas analisando o *gap* médio apenas entre CM e FM, o da metodologia de Fluxo foi maior tendo como média 9,51% enquanto o CM obteve 5,68% no grupo de instancias que continham uma variação maior de números de vértices, já no grupo de instancias 2 o método de CM-FM obteve menor *gap* entre todas as metodologias. Vale ressaltar que a combinação FM-CM não alcançou os resultados esperados, na maioria dos cenários não conseguiu trazer melhorias nos resultados, além de possuir o segundo maior *Gap* médio no Grupo 1 e o terceiro maior no grupo 2.

A tabela 4 apresentam os resultados que foram obtidos com a implementação do método LC (que limita a criação de grafos) e são apresentados os resultados e o *gap* em percentual para cada metodologia de ordenação da lista de requisição.

Tabela 4: Limitando grafos - LC

Ordenação Instancia	CM		FM		FM-CM		CM-FM		Ordenação Instancia	CM		FM		FM-CM		CM-FM	
	gap(%)	w	gap(%)	w	gap(%)	w	gap(%)	w		gap(%)	w	gap(%)	w	gap(%)	w	gap(%)	w
att	4,00	25	0,00	25	0,00	25	4,00	25	mesh4x25.20	0,00	68	1,47	69	1,47	68	0,00	68
att2	0,88	113	0,00	113	0,00	113	0,88	113	mesh4x25.40	0,00	129	2,33	131	1,55	130	0,00	129
brasil	0,00	48	0,00	48	0,00	48	0,00	48	mesh4x25.60	1,03	196	2,56	200	2,05	197	1,03	196
eon	0,00	22	0,00	22	0,00	22	0,00	22	mesh4x25.80	0,77	260	2,31	265	1,15	262	0,77	261
fin	0,00	47	2,13	48	2,13	47	0,00	47	mesh4x25.100	0,95	319	3,47	327	1,26	319	0,95	319
giul	2,06	396	0,77	389	1,03	390	2,31	397	mesh5x20.20	1,85	54	1,85	55	1,85	54	1,85	54
nsf3	0,00	28	0,00	28	0,00	28	0,00	28	mesh5x20.40	0,00	103	2,91	105	1,94	104	0,00	103
nsf12	0,00	40	0,00	40	0,00	40	0,00	40	mesh5x20.60	0,64	157	1,91	159	1,27	158	0,64	157
nsf212	0,00	39	0,00	39	0,00	39	0,00	39	mesh5x20.80	0,97	208	2,42	211	1,93	209	0,97	208
sun	3,39	60	0,00	59	0,00	59	3,39	60	mesh5x20.100	0,79	255	2,76	261	1,18	256	0,79	255
zib54	7,13	594	0,18	561	0,53	561	7,13	597	mesh6x17.20	0,00	46	2,17	46	0,00	46	0,00	46
y3-40-5	1,61	63	6,45	66	4,84	64	1,61	63	mesh6x17.40	1,15	87	2,30	88	1,15	87	1,15	87
y3-60-5	2,22	91	6,67	96	6,67	95	2,22	91	mesh6x17.60	0,76	133	1,52	134	0,76	133	0,76	133
y3-80-1	0,79	127	5,51	133	4,72	132	0,79	127	mesh6x17.80	0,57	176	1,14	177	0,57	176	0,57	176
y3-80-5	1,69	120	6,78	125	5,93	124	1,69	120	mesh6x17.100	1,37	222	4,11	227	1,83	222	1,37	222
y4-20-4	4,76	22	9,52	22	4,76	22	4,76	22	mesh8x13.20	0,00	36	2,78	37	0,00	36	0,00	36
y4-60-5	6,78	62	11,86	65	10,17	65	6,78	63	mesh8x13.40	1,49	67	4,48	70	2,99	68	1,49	67
y4-80-1	6,76	79	12,16	82	10,81	82	6,76	79	mesh8x13.60	0,99	102	4,95	105	3,96	103	0,99	101
y4-80-5	5,19	81	6,49	82	9,09	83	5,19	81	mesh8x13.80	0,74	135	2,96	139	2,22	137	0,74	135
y4-100-1	6,59	97	10,99	101	10,99	99	6,59	97	mesh8x13.100	3,53	176	8,24	184	7,06	180	3,53	176
y5-60-1	2,63	39	7,89	41	7,89	41	2,63	39	mesh10x10.20	6,45	33	12,90	34	6,45	33	6,45	33
y5-80-1	6,12	51	10,20	54	10,20	53	6,12	51	mesh10x10.40	8,93	61	16,07	65	14,29	62	8,93	61
y5-80-2	1,61	62	6,45	65	4,84	63	1,61	62	mesh10x10.60	9,52	91	16,67	97	14,29	94	9,52	91
y5-100-1	5,00	63	11,67	66	11,67	66	5,00	63	mesh10x10.80	9,01	121	17,12	129	15,32	126	9,01	121
y5-100-2	0,00	77	5,19	81	5,19	78	0,00	77	mesh10x10.100	9,56	148	17,65	159	15,44	155	9,56	148
Gap médio	2,77		4,84		4,46		2,78		Gap médio	2,44		5,56		4,08		2,44	

A estratégia de limitar a criação de grafos já havia se mostrado mais eficiente como verificado pela Tabela 2. Pela Tabela 4 não haver ganho geral para as metodologias implementadas considerando o caso LC.

A Tabelas 5 seguem o mesmo modelo que foi apresentado pelas tabelas 3 e 4, mas os resultados apresentados nas tabelas seguinte é referente a metodologia de CGA, que cria os grafos no início de cada iteração.

Tabela 5: Criando grafos - CGA

Ordenação Instancia	CM		FM		FM-CM		CM-FM		Ordenação Instancia	CM		FM		FM-CM		CM-FM	
	gap(%)	w	gap(%)	w	gap(%)	w	gap(%)	w		gap(%)	w	gap(%)	w	gap(%)	w	gap(%)	w
att	12,00	27	16,00	26	8,00	26	16,00	27	mesh4x25.20	1,47	69	4,41	70	1,47	69	1,47	69
att2	0,00	113	0,88	113	0,88	113	0,00	113	mesh4x25.40	2,33	129	3,88	133	2,33	130	1,55	130
brasil	0,00	48	2,08	48	0,00	48	0,00	48	mesh4x25.60	0,51	195	3,59	201	2,56	197	0,00	195
eon	0,00	22	0,00	22	0,00	22	0,00	22	mesh4x25.80	0,00	260	2,69	266	2,69	264	0,00	260
fin	2,13	47	6,38	49	6,38	49	2,13	47	mesh4x25.100	0,32	318	3,79	327	2,84	320	0,63	317
giul	2,57	397	1,29	391	1,54	391	2,57	397	mesh5x20.20	1,85	55	5,56	56	3,70	55	1,85	55
nsf3	0,00	28	0,00	28	0,00	28	0,00	28	mesh5x20.40	1,94	103	3,88	105	3,88	104	1,94	103
nsf12	10,00	41	7,50	41	7,50	41	7,50	41	mesh5x20.60	0,00	157	2,55	160	1,91	159	0,00	157
nsf212	10,26	40	10,26	40	10,26	39	7,69	39	mesh5x20.80	0,48	207	2,90	212	2,42	211	0,48	207
sun	8,47	62	3,39	59	3,39	59	8,47	62	mesh5x20.100	0,39	254	3,54	262	2,76	259	0,39	254
zib54	8,20	596	1,07	563	6,42	565	8,02	595	mesh6x17.20	2,17	46	4,35	47	2,17	46	2,17	47
y3-40-5	1,61	62	6,45	65	4,84	64	1,61	62	mesh6x17.40	2,30	88	3,45	89	2,30	87	2,30	87
y3-60-5	1,11	90	6,67	93	5,56	94	1,11	90	mesh6x17.60	0,76	132	2,27	134	1,52	133	0,76	132
y3-80-1	1,57	128	6,30	133	7,87	133	1,57	128	mesh6x17.80	0,00	176	1,70	178	1,14	176	0,00	176
y3-80-5	1,69	118	5,93	123	5,93	122	1,69	118	mesh6x17.100	0,00	219	3,65	226	3,20	224	0,00	219
y4-20-4	4,76	22	9,52	22	4,76	21	4,76	21	mesh8x13.20	0,00	36	2,78	36	0,00	36	2,78	36
y4-60-5	1,69	59	5,08	61	5,08	61	1,69	59	mesh8x13.40	1,49	68	4,48	69	2,99	68	2,99	68
y4-80-1	0,00	74	4,05	75	2,70	76	1,35	74	mesh8x13.60	2,97	102	2,97	103	2,97	103	1,98	102
y4-80-5	1,30	77	3,90	79	3,90	79	1,30	77	mesh8x13.80	1,48	135	2,96	138	2,22	136	0,74	136
y4-100-1	1,10	91	3,30	93	3,30	93	1,10	91	mesh8x13.100	0,59	170	4,12	176	4,12	176	0,59	170
y5-60-1	0,00	38	5,26	39	5,26	39	0,00	38	mesh10x10.20	0,00	31	3,23	32	0,00	31	0,00	31
y5-80-1	4,08	49	6,12	51	6,12	50	4,08	49	mesh10x10.40	1,79	56	7,14	59	3,57	57	1,79	56
y5-80-2	1,61	62	8,06	66	9,68	64	1,61	63	mesh10x10.60	1,19	84	5,95	89	4,76	85	1,19	84
y5-100-1	1,67	60	6,67	63	6,67	63	1,67	60	mesh10x10.80	0,00	111	6,31	118	5,41	116	0,00	111
y5-100-2	1,30	77	7,79	81	10,39	78	1,30	77	mesh10x10.100	0,00	136	6,62	144	6,62	144	0,74	136
Gap médio	3,09		5,36		5,25		3,09		Gap médio	0,96		3,95		2,78		1,05	

A metodologia CGA, Tabela 5, quando avaliado o panorama geral apresentou os melhores resultados apesar de não ser o método com os menores gaps. Essa variação ocorre, pois, neste método já existe uma poda por limite. Levando em consideração as

maneiras para a ordenação da lista de requisições, os resultados dos métodos de fluxo máximo e das combinações do caminho mínimo com fluxo máximo, foram utilizadas para comparar com o método de caminho mínimo proposto por [1]. Com a implementação dos métodos foi possível obter melhoras e em algumas instâncias os resultados foram superiores à heurística desenvolvida por [11].

Comparando com os resultados apresentados por [11], utilizando o mesmo método proposto por [1], pode-se perceber que certas instâncias ocorreram resultados piores, Essa piora nos resultados pode ser atribuída ao fator aleatório, sendo que esse pode ser utilizado para justificar as melhoras que algumas instâncias apresentaram. Junto a isso vale ressaltar que há diferença entre os hardwares utilizados nos dois trabalhos, fator que também pode ter influência sobre os resultados que apresentaram melhoras.

Com os resultados apresentados, pode-se perceber nas instâncias que apresentam uma rede mais complexa, possuindo maior número de vértices e arestas, a metodologia de fluxo obteve ganhos mais significativos, já que a ideia inicial para a ordenação por fluxo em ordem crescente é que primeiramente sejam alocadas as requisições que possuem menores opções de escolha de caminhos disjuntos. Essa ordem de alocação possibilita o melhor aproveitamento da rede, uma vez que as requisições que possuem maior fluxo, ou seja opções de caminho, tendem a ser alocadas com maior facilidade com os comprimentos de onde já disponíveis.

6. CONCLUSÃO

Essa pesquisa apresenta o desenvolvimento e implementação de novos métodos de ordenação e combinação de métodos de criação de grafos para o problema RWA, buscando melhora nos resultados do método de geração de solução inicial proposto por [1].

Através das novas metodologias foi possível observar que a maneira que são criados os grafos tem grande influência nos resultados, tendo valores mais constantes com um *gap* pequeno. Já dentro dos métodos para a ordenação da lista de requisição o método de fluxo máximo trouxe ganhos sendo que a combinação das metodologias fluxo máximo e caminho mínimo conseguiram reduzir trazer ganhos consideráveis no valor da função objetivo.

Para que seja estabelecido qual metodologia apresenta melhor desempenho, tanto de criação de grafos quanto de critério para ordenação da lista de requisição, o ideal é que em trabalhos futuros sejam realizados testes maiores, aproximando do comportamento probabilístico da curva normal. Levando em consideração que o algoritmo possui o fator aleatório, aumentando a base de dados seria possível a realização de estudos estatísticos aplicados nos dados, obtendo assim uma base forte para estabelecer qual é o método apresenta melhor desempenho.

7. REFERÊNCIAS BIBLIOGRÁFICAS

- [1] SKORIN-KAPOV, N. Routing and wavelength assignment in optical networks using bin packing based algorithms. *European Journal of Operational Research*, Elsevier, v. 177, n. 2, p. 1167–1179, 2007. 1, 3, 4, 5, 6, 7, 8, 11
- [2] ZANG, H. et al. A review of routing and wavelength assignment approaches for wavelength-routed optical wdm networks. *Optical networks magazine*, v. 1, n. 1, p. 47–60, 2000. 2
- [3] MARTINS, A. X. *Metaheurísticas e Formulações para a resolução do Problema de Roteamento e Alocação de Comprimentos de Onda em Redes Ópticas*. Tese (Doutorado) — Université Blaise Pascal-Clermont-Ferrand II, 2011. 2
- [4] LI, G.; SIMHA, R. The partition coloring problem and its application to wavelength routing and assignment. In: *Proceedings of the First Workshop on Optical Networks*. Dallas, USA: [s.n.], 2000. p. 1–19. 3
- [5] 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, p. 903–908, 1996. 3
- [6] NORONHA, T.; RIBEIRO, C. Routing and wavelength assignment by partition colouring. *European Journal of Operational Research*, v. 171, n. 3, p. 797–810, 2006. 3
- [7] BRELAZ, D. New methods to color the vertices of a graph. *Communications of the ACM*, v. 256, p. 251–256, 1979. 3
- [8] MANOHAR, P.; MANJUNATH, D.; SHEVGAONKAR, R. Routing and wavelength assignment in optical networks from edge disjoint path algorithms. *IEEE Communications Letters*, v. 6, n. 5, p. 211–213, 2002. 3
- [9] GAREY, M.; JOHNSON, D. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. [S.l.]: Freeman, San Francisco, 1979. 3
- [10] NORONHA, T. F.; RESENDE, M. G.; RIBEIRO, C. C. Efficient implementations of heuristics for routing and wavelength assignment. In: SPRINGER. *International Workshop on Experimental and Efficient Algorithms*. [S.l.], 2008. p. 169–180. 4, 5, 7
- [11] MARTINS, A. X. et al. Variable neighborhood descent with iterated local search for routing and wavelength assignment. *Computers & Operations Research*, Elsevier, v. 39, n. 9, p. 2133–2141, 2012. 5, 7, 11