



**UNIVERSIDADE FEDERAL DA FRONTEIRA SUL
CAMPUS CHAPECÓ
CURSO DE CIÊNCIA DA COMPUTAÇÃO**

YRUÍ GIOVAN NERIS

**PROBLEMA QUADRÁTICO DE ALOCAÇÃO APLICADO AO
MAPEAMENTO DE TOPOLOGIAS DE REDES ÓPTICAS DE
TRANSPORTE DE TELECOMUNICAÇÕES**

**CHAPECÓ
2017**

YRUÍ GIOVAN NERIS

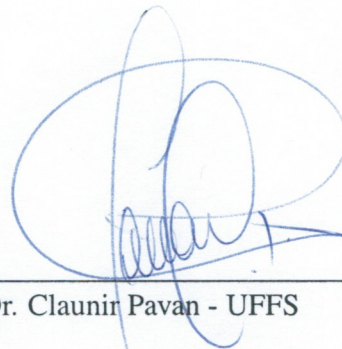
**PROBLEMA QUADRÁTICO DE ALOCAÇÃO APLICADO AO
MAPEAMENTO DE TOPOLOGIAS DE REDES ÓPTICAS DE
TRANSPORTE DE TELECOMUNICAÇÕES**

Trabalho de conclusão de curso de graduação apresentado como requisito para obtenção do grau de Bacharel em Ciência da Computação da Universidade Federal da Fronteira Sul.

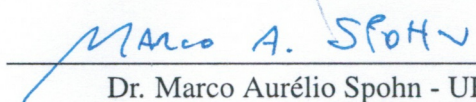
Orientador: Prof. Dr. Claunir Pavan

Este trabalho de conclusão de curso foi defendido e aprovado pela banca em: 14/12/14

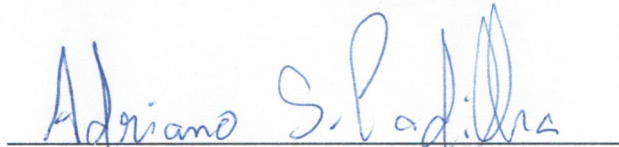
BANCA EXAMINADORA:



Dr. Claunir Pavan - UFFS



Dr. Marco Aurélio Spohn - UFFS



YRUÍ GIOVAN NERIS

**PROBLEMA QUADRÁTICO DE ALOCAÇÃO APLICADO AO
MAPEAMENTO DE TOPOLOGIAS DE REDES ÓPTICAS DE
TRANSPORTE DE TELECOMUNICAÇÕES**

Trabalho de conclusão de curso de graduação
apresentado como requisito para obtenção do
grau de Bacharel em Ciência da Computação da
Universidade Federal da Fronteira Sul.

Orientador: Prof. Dr. Claunir Pavan

CHAPECÓ

2017

YRUI GIOVAN NERIS

**PROBLEMA QUADRÁTICO DE ALOCAÇÃO APLICADO AO
MAPEAMENTO DE TOPOLOGIAS DE REDES ÓPTICAS DE
TRANSPORTE DE TELECOMUNICAÇÕES**

Trabalho de conclusão de curso de graduação apresentado como requisito para obtenção do grau de Bacharel em Ciência da Computação da Universidade Federal da Fronteira Sul.

Orientador: Prof. Dr. Claunir Pavan

Este trabalho de conclusão de curso foi defendido e aprovado pela banca em: ____/____/____

BANCA EXAMINADORA:

Dr. Claunir Pavan - UFFS

Dr. Marco Aurélio Spohn - UFFS

Me. Adriano Sanick Padilha - UFFS

RESUMO

O surgimento de aplicações que utilizam e se beneficiam de altas demandas de tráfego de dados tem causado um crescimento elevado de tráfego nas redes de telecomunicações. As redes ópticas de telecomunicações se demonstram eficientes para atender as demandas existentes, porém a distância é um limitador de alcance por formato de modulação, quanto maior a distância menor será a capacidade de transferência de dados. Neste trabalho foi desenvolvida uma meta-heurística aplicando o problema quadrático de alocação e algoritmo genético, ao mapeamento das redes ópticas de telecomunicações, minimizando a soma do comprimentos de seus enlaces, com a alteração do mapeamento da topologia. Ao todo foram testadas 27 topologias do mundo real, e através da alteração do mapeamento foi possível minimizar o comprimento de 18 redes, observando que as características físicas das rede não possuem influência na possibilidade de minimização, a minimização depende da forma com que as redes estão mapeadas.

Palavras-chave: Redes de telecomunicações; Algoritmo Genético; Problema de alocação quadrática..

ABSTRACT

The emergence of applications that utilize and benefit from high demands on data traffic has caused a high growth of traffic in telecommunications networks. Optical telecommunication networks are efficient to meet existing demands, but distance is a range limitation by modulation format, the longer the distance the smaller the data transfer capacity. In this work, a metaheuristic was developed applying the quadratic assignment problem and genetic algorithm to the mapping of the optical networks of telecommunications, minimizing the sum of the lengths of their links, with the alteration of the mapping of the topology. In total, 27 real-world topologies were tested, and through the mapping change it was possible to minimize the length of 18 networks, noting that physical network characteristics have no influence on the possibility of minimization, minimization depends on the way the networks are mapped .

Keywords: Telecommunications Networks; Genetic Algorithm; Quadratic Assignment Problem..

LISTA DE FIGURAS

Figura 2.1 – Topologia e matriz de fluxo	12
Figura 2.2 – Posição dos nós e matriz de distâncias	13
Figura 2.3 – Mapeamentos	13
Figura 2.4 – Fluxograma de um algoritmo genético.....	15
Figura 3.1 – Indivíduos mapeados para as posições fixas.....	18
Figura 3.2 – Restrição de distância máxima	19
Figura 3.3 – Mapeamento do indivíduo	20
Figura 3.4 – Roleta utilizada no processo de mutação.....	21
Figura 3.5 – Exemplo de cruzamento de ponto único	22
Figura 3.6 – Cruzamento de ponto único aplicado aos indivíduos	23
Figura 3.7 – Ajuste dos indivíduos após o cruzamento	23
Figura 3.8 – Processo de troca utilizado para mutação	24
Figura 3.9 – Processo de inversão utilizado para mutação	24
Figura 3.10 – Processo de inversão inserção para mutação	25
Figura 3.11 – Fluxograma do algoritmo implementado	27
Figura 4.1 – Rede NSFNET original.....	29
Figura 4.2 – Rede NSFNET minimizada	30
Figura 4.3 – Topologia Harary referente a rede NSFNET	30
Figura 4.4 – Mapeamento da Topologia Harary conforme o mapeamento original da rede NSFNET	31
Figura 4.5 – Rede NSFNET - Harary Minimizada	31
Figura 4.6 – Comparação de Tempo de execução entre o Algoritmo Genético e teste de Todas as possibilidades para a Rede NSFNET	32
Figura 4.7 – Ganho em Percentual das Topologias do Mundo Real	38
Figura 4.8 – Ganho em Percentual das Topologias Harary	39
Figura 6.1 – Comparação de quantidade de Bloqueios entre os Mapeamentos da rede CESNET	45
Figura 6.2 – Comparação de quantidade de Bloqueios entre os Mapeamentos da rede ITALY.....	46
Figura 6.3 – Comparação de quantidade de Bloqueios entre os Mapeamentos da rede NSFNET	47
Figura 6.4 – Comparação de quantidade de Bloqueios entre os Mapeamentos da rede RNP	48
Figura 6.5 – Comparação entre as Mutações utilizando a redes Originais CESNET, ITALY, NSFNET e RNP.	50

LISTA DE TABELAS

Tabela 1.1 – Comparação entre formatos de modulação	10
Tabela 4.1 – Redes de Referência.....	33
Tabela 4.2 – Configurações Utilizadas.....	34
Tabela 4.3 – Tabela de Resultados de Topologias com até 14 nós com solução ótima calculada	35
Tabela 4.4 – Tabela de Resultados das topologias com quantidade de nós entre 0 ate 20 sem solução ótima calculada	36
Tabela 4.5 – Tabela de Resultados das topologias com quantidade de nós entre 21 e 40 ...	37
Tabela 4.6 – Tabela de Resultados das topologias com quantidade de nós entre 41 e 60 ...	38
Tabela 4.7 – Tabela de Resultados das topologias com quantidade de nós superior a 60 ...	38
Tabela 6.1 – Tabela de Resultados da quantidade de bloqueios para os Mapeamentos da Rede CESNET	45
Tabela 6.2 – Tabela de Resultados da quantidade de bloqueios para os Mapeamentos da Rede ITALY	46
Tabela 6.3 – Tabela de Resultados da quantidade de bloqueios para os Mapeamentos da Rede NSFNET	47
Tabela 6.4 – Tabela de Resultados da quantidade de bloqueios para os Mapeamentos da Rede RNP	48
Tabela 6.5 – Configurações Utilizadas para o teste de Performance das Mutações.....	49

SUMÁRIO

1 INTRODUÇÃO	9
2 DEFINIÇÃO DO PROBLEMA	11
2.1 Problema Quadrático de Alocação	11
2.2 PQA aplicado a Redes de Telecomunicações	12
2.3 Complexidade	13
2.4 Aplicações	14
2.5 Algoritmo Genético	14
2.5.1 O Problema da Convergência Prematura	16
3 IMPLEMENTAÇÃO	17
3.1 Entrada	17
3.2 População Inicial/Codificação	17
3.2.1 Restrição de distância máxima	18
3.3 Avaliação	20
3.4 Seleção	21
3.5 Cruzamento	22
3.6 Mutação	24
3.7 Renovação da População	25
3.8 Decodificação	26
3.9 Especificação do Algoritmo	26
4 RESULTADOS COMPUTACIONAIS	29
4.1 Análise de Resultados	38
5 CONCLUSÕES	40
REFERÊNCIAS	41
6 APÊNDICE	43
6.1 Simulações Adicionais	43
6.2 Problema do Roteamento e Alocação de Espectro	43
6.2.1 Resultados	44
6.3 Teste de Performance das Mutações	49
6.3.1 Resultados	49

1 INTRODUÇÃO

A importância das redes de telecomunicações e a demanda do tráfego de dados vem crescendo significativamente, impulsionados por aplicações multimídia que são utilizadas diariamente por milhões de pessoas para obter e trocar informação. Esse crescimento por sua vez continua evoluindo, impulsionado por aplicações que se beneficiam e necessitam de uma demanda elevada de transmissão, como as comunicações interativas de vídeo em tempo real [7]. Com esse crescimento surge a necessidade de desenvolver técnicas de aperfeiçoamento das redes, melhorando o desempenho das mesmas e atendendo essa alta demanda requisitada de forma inteligente e eficaz.

A rede óptica tradicional de grade fixa como a DWDM (Multiplexação Densa por Divisão em Comprimento de Onda) que possui os canais com espaçamento de 50 GHz ou 100 GHz - conforme padronizado pela União Internacional de Telecomunicações (ITU) - ao longo do espectro de frequência [2], suporta demandas de até 200 Gb/s, e por possuir a grade fixa desperdiça o espectro, pois somente uma requisição pode ser transmitida por canal. A rede óptica adaptativa com grade flexível denominada, Rede Óptica Elástica (EON) atende demandas de até 1 Tb/s, utilizando o espectro de forma eficiente [6], onde o tamanho do canal varia conforme a requisição, podendo aumentar ou diminuir a quantidade de espectro utilizada.

A redes ópticas elásticas gerenciam o espectro de forma mais eficiente e tem maior capacidade do que as redes tradicionais, mas ambas as tecnologias transmitem a informação através da fibra óptica com utilização da luz, e quanto maior a distância, a intensidade do sinal diminui, requerendo amplificações ou regenerações consoante também ao formato de modulação escolhido.

Na Tabela 1.1 são exibidos os formatos de modulação utilizados em transmissões ópticas, considerando um slot de espectro de 12.5 GHz. É possível identificar que a capacidade interfere no consumo de energia e no alcance total do sinal, quando a capacidade é alta o alcance diminui e o consumo de energia aumenta, quando a capacidade é baixa a distância aumenta e o consumo de energia diminui.

Tabela 1.1: Comparação entre formatos de modulação [5]

Modulação	Capacidade	Consumo de Energia	Alcance
BPSK	12.5 Gb/s	112.374 W	4000 km
QPSK	25 Gb/s	133.416 W	2000 km
8QAM	37.5 Gb/s	154.457 W	1000 km
16QAM	50 Gb/s	175.498 W	500 km
32QAM	62.5 Gb/s	196.539 W	250 km
64QAM	75 Gb/s	217.581 W	125 km

Tendo em vista essas características, percebe-se a importância de dimensionar as topologias de redes minimizando a soma dos comprimentos dos enlaces entre os nós. Nesse trabalho foi aplicada a técnica de Algoritmo Genético com o problema matemático PQA (Problema Quadrático de Alocação), à topologias existentes no mundo real e que já passaram por um tratamento de melhoramento, apresentadas no trabalho How Reliable Are the Real-World OpticalTransport Networks? [16].

2 DEFINIÇÃO DO PROBLEMA

A topologia da rede física pode ser representada por um grafo G com um conjunto $V(G)$ que representa os nós da rede, e um conjunto $E(G)$ que representa os enlaces bidirecionais.

Dois nós u e v são adjacentes quando existe pelo menos um enlace $E(G)$ os interligando. As topologias devem ser 2-aresta-conexo, considerando que redes ópticas de transporte de telecomunicações precisam suportar falhas nos enlaces (todos os pares de vértices estão ligados por, pelo menos, dois caminhos disjuntos em arestas). O grau de um nó v é o número de nós adjacente a v em G .

2.1 Problema Quadrático de Alocação

O problema quadrático de alocação (PQA) foi introduzido por Koopmans e Beckmann em 1957 como um modelo matemático para a localização de atividades econômicas indivisíveis, sendo utilizado frequentemente para descrever um problema de localização [11]. O PQA é definido como a alocação de n instalações móveis em n locais fixos, onde apenas uma instalação pode ser alocada em um local e cada instalação está mutuamente conectada por uma rede já definida [19].

Assumindo que o fluxo entre as instalações i e j é $F = [f_{ij}]$ e a distância entre as instalações k e l é $D = [d_{kl}]$, as instalações devem ser mapeadas aos locais de uma forma que minimize o custo total da rede, que é dado pela soma do produto fluxo \times distância. Considerando n o conjunto dos inteiros positivos maiores que 0 $\{1, 2, \dots, n\}$ e S_n o conjunto de permutações de $\{1, 2, \dots, n\}$, o Problema de Alocação Quadrática pode ser definido como:

$$\min \sum_{i=1}^n \sum_{j=1}^n f_{ij} d_{\pi(i)\pi(j)}$$

para todas as permutações $\pi \in S_n$. O objetivo com a matriz de fluxo F e a matriz de distância D é encontrar a permutação $\pi_0 \in S_n$, que minimiza a soma sobre todos i, j . A notação $d_{\pi(i)\pi(j)}$ refere-se a permutar linhas e colunas da matriz D por permutações $\in \pi$. Ou seja $D^\pi = [d_{ij}^\pi] = d_{\pi(i)\pi(j)}$ para $1 \leq i, j \leq n$ [15].

2.2 PQA aplicado a Redes de Telecomunicações

Utilizando a representação de grafos para representar a topologia, tomamos como exemplo para este trabalho uma topologia que possui 5 nós em diferentes locais. E cada nó possui as suas conexões já definidas, na figura 2.1 a) é apresentada uma representação ilustrativa da topologia. Utilizando como exemplo o nó 0, observamos que ele tem como nós adjacentes os nós 1, 2 e 4. Os enlaces que conectam os nós possuem o valor 1, representando que existe conexão entre os nós, e todos os enlaces têm o mesmo peso/prioridade na topologia. Na figura 2.1 b) a matriz F é a matriz de adjacências, onde a posição $F[i][j] = 1$ representa que os nós i e j estão diretamente conectados por um enlace.

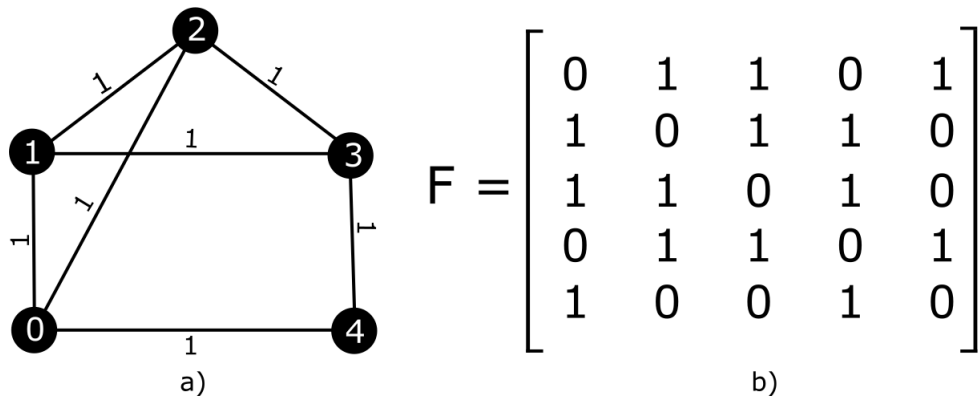


Figura 2.1: Topologia da Rede.

Na figura 2.2 a) é apresentada a posição fixa dos nós sobre um plano e sua distância para todos os outros nós. Considerando estas posições fixas, a topologia apresentada na figura 2.1 deve ser mapeada. Note que o número de nós é o mesmo.

Os números presentes nos enlaces da figura 2.2 a) representam a distância entre os nós. A figura 2.2 b) apresenta a matriz de distância entre todos os nós, em que a posição $D[k][l]$ representa a distância do nó k para o nó l .

Com a topologia a ser mapeada e a posição fixa dos nós definida é possível proceder ao mapeamento. Na figura 2.3 a) está apresentado o mapeamento ideal considerando a minimização da soma dos comprimentos dos enlaces da rede, a partir soma do produto fluxo \times distância temos um comprimento total de 42 quilômetros. A figura 2.3 b) representa outra solução para o mapeamento, porém ela não é a melhor solução, quando trocamos a posição do enlace entre os nós 0 e 2 para a posição 0 e 3, a soma dos comprimentos dos enlaces aumenta para 50 quilômetros.

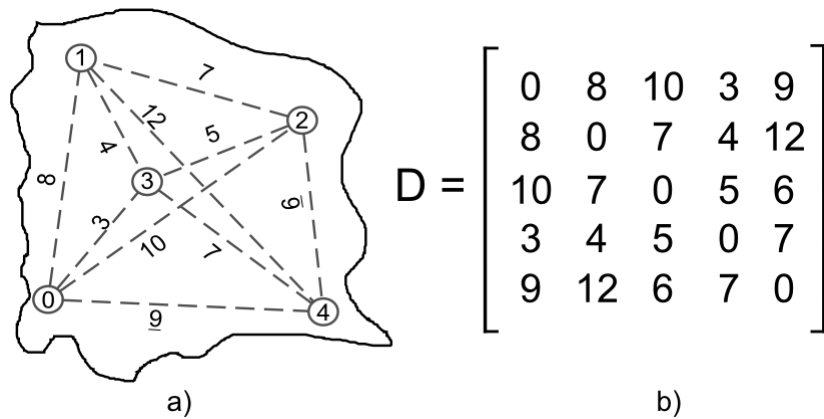


Figura 2.2: a) Posição fixa dos nós onde a Topologia pode ser Mapeada. b) Matriz de distâncias.

A figura 2.3 c) não é uma possível solução para o caso apresentado, pois o mapeamento não respeita a restrição de que as conexões entre os nós não podem ser alteradas, se observarmos a 2.1 a) notamos que o nó 0 se conecta com os nós 1,2 e 4, e no mapeamento c) o nó está se conectando com os nós 1,3 e 4, o mesmo acontece para o nó 2, onde antes ele se conectava com os nós 0,1 e 3, agora está se conectando com os nós 1,3 e 4, isso é devido a troca dos nós 2 e 3 de posição sem manter as suas conexões originais, não sendo possível calcular a distância da rede, pois não existe as conexões $0 \leftrightarrow 3$ e $2 \leftrightarrow 4$ na definição da topologia a ser mapeada.

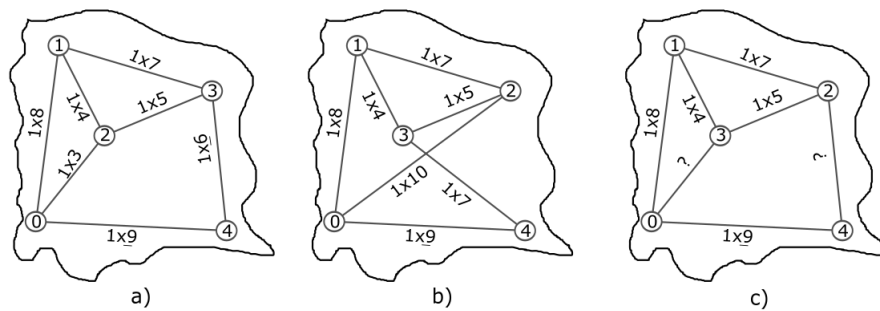


Figura 2.3: Mapeamentos da Topologia.

2.3 Complexidade

A obtenção da solução ótima para o PQA pode ser conseguida pela comparação dos custos de todas as soluções possíveis. Contudo, entre os problemas de otimização combinatória, o PQA está entre os mais difíceis. Em geral, para problemas onde $n > 30$ não é possível chegar na solução ótima em um tempo computacional razoável, pois para toda quantidade n de instalações, temos $n!$ possibilidades [15]. PQA é um problema da classe NP-Hard, ao menos que $P = NP$ [12].

Considerando a rede exemplo, que possui 5 nós, são possíveis 120 mapeamentos distintos.

2.4 Aplicações

Muitos problemas da classe NP-Hard de otimização combinatória e aplicações em tempo real podem ser modelados com PQA, tais como o caixeiro viajante (*traveling salesman problem*), problema do empacotamento (*bin packing problem*), problema de clique máxima (*max clique problem*), layout de memória, processo de agendamento e particionamento de gráfico.

Devido a sua complexidade o PQA foi escolhido como aplicativo de teste para o projeto GRIBB (great international branch-and-bound search), que busca estabelecer uma biblioteca de software para resolver uma grande variedade de problemas de busca paralela, pelo uso de inúmeros computadores em todo o mundo acessados pela Internet [12].

Além de problemas de localização, o PQA aparece em aplicativos de fabricação de computadores, agendamentos, comunicações de processo e balanceamentos de turbinas. Na ergonomia o PQA mostrou que pode ser aplicado ao design do teclado da máquina de escrever, organizando as teclas de um modo que minimize o tempo de digitação de um texto [11].

2.5 Algoritmo Genético

Devido a complexidade do problema não são esperados resultados ótimos em tempo computacional aceitável, com o objetivo de aproximação de um valor ótimo, foi implementado neste trabalho uma meta heurística que utiliza algoritmo genético com características evolutivas para a obtenção de resultados.

Algoritmos genéticos com características evolutivas são técnicas de procura, tendo como objetivo encontrar uma solução aceitável, se baseando na evolução natural das espécies de C. Darwin, e foram inicialmente propostos por Frazee [14]. Atualmente são utilizados em inúmeros problemas de otimização.

Estes algoritmos possuem a característica de manter os indivíduos mais aptos. Como na natureza, os indivíduos mais aptos tem maior probabilidade de sobreviver e se acasalar, gerando novos indivíduos que possuem grande probabilidade de serem igualmente ou mais aptos que os pais, já que herdaram os bons genes dos pais [14].

A ideia de evolução das espécies aplicadas a algoritmo genético tem como primeiro passo a construção de possíveis soluções, e após combinam essas soluções entre elas obtendo uma nova geração de possíveis soluções. Cada solução é considerada um indivíduo na população, e a sua aptidão é dada através do valor correspondente de uma função objetivo. Um indivíduo será mais apto que outro se o valor da função objetivo estiver mais próximo do ótimo.

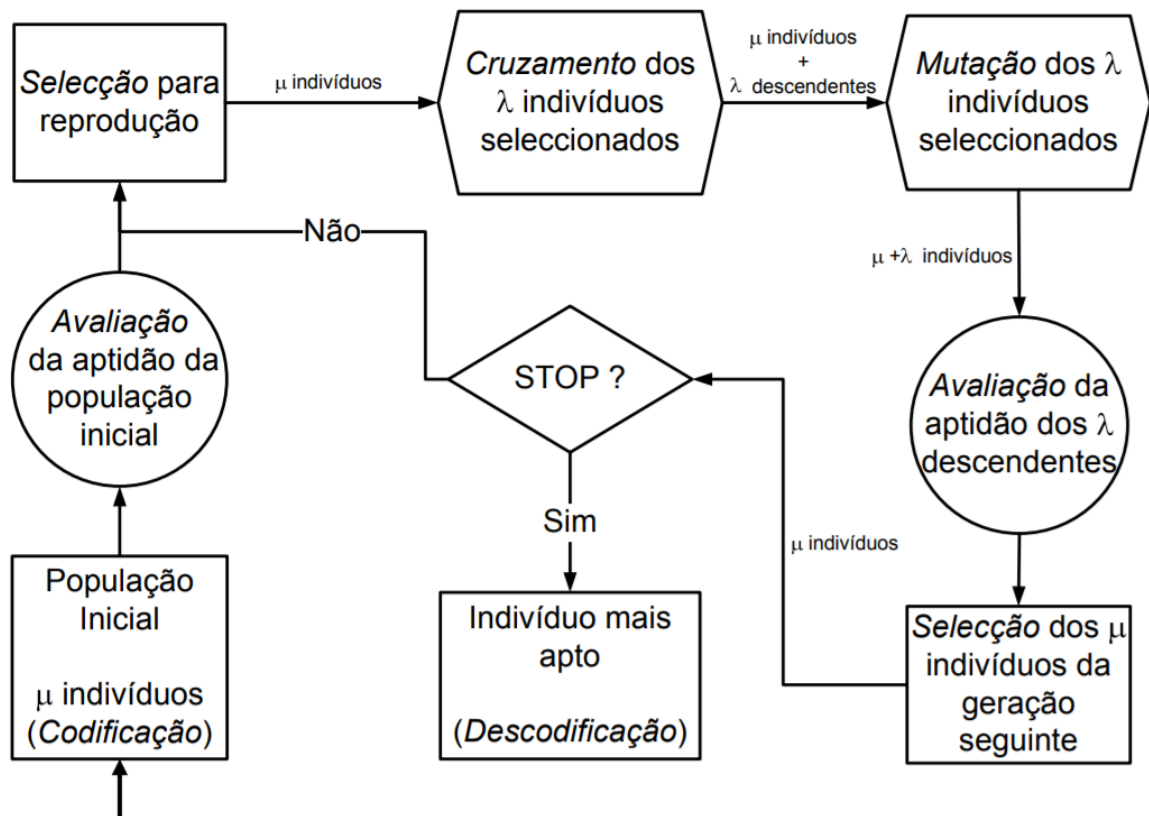


Figura 2.4: Fluxograma de um algoritmo genético [14].

Um algoritmo genético possui as seguintes fases:

- **Codificação:** É primeira etapa do algoritmo, onde são geradas algumas das possíveis soluções, que são também conhecidas como os indivíduos da população inicial.
- **Avaliação:** Após a geração dos indivíduos, eles são avaliados individualmente, ou seja, é calculado o valor da função objetivo. Essa avaliação determina a aptidão dos indivíduos e é utilizada para selecionar os indivíduos vão se reproduzir.
- **Seleção:** Com a aptidão de cada indivíduo reconhecida, é necessário selecionar quais serão os progenitores da próxima geração. Os mais aptos possuem maior probabilidade de cruzamento, porém todos os indivíduos podem ser os progenitores.

- **Cruzamento:** Após a seleção dos indivíduos, eles serão combinados entre si e irão gerar novos indivíduos, que tendem a ser melhores que os indivíduos da população anterior, pois os indivíduos com melhor aptidão tem maior chance de serem selecionados para serem os progenitores.
- **Mutação:** Após o cruzamento alguns indivíduos são selecionados para passarem por um processo de mutação, que altera parte do seu código genético.
- **Decodificação:** Após todos os passos anteriores uma nova população é gerada, se o critério de parada for atendido o algoritmo para, e o indivíduo mais apto será a solução, caso contrário o algoritmo repete o processo até que o critério de parada seja atendido.

Na figura 2.4 é apresentado um fluxograma que contém as diferentes fases do algoritmo genético e a sua sequência.

2.5.1 O Problema da Convergência Prematura

Os algoritmos genéticos padecem do problema de convergência prematura [8], sendo forçados muitas vezes, a forçar a diversificação, violando as características biológicas. O algoritmo tem uma convergência para uma região sub-ótima, e os genes de alguns indivíduos da população rapidamente conseguem dominar toda a população, restringindo a convergência para um local otimizado. Ao entrar em um local ótimo os operadores genéticos não conseguem reproduzir descendentes melhores que os pais, reduzindo a capacidade de busca dos algoritmos. Para contornar esses problemas existem vários métodos e entre eles estão os métodos de seleção restrita, aplicação dinâmica de mutação, restrições para o cruzamento e mutação, e entre outros.

3 IMPLEMENTAÇÃO

Nas próximas seções será descrita a meta heurística implementada neste trabalho. Para exemplificação será utilizada a mesma topologia e posição dos nós presentes na seção 2.2.

A implementação utiliza as matrizes de adjacências F e distâncias D para a obtenção dos resultados, recebe como entrada as ligações bidirecionais dos nós que representa a topologia a ser mapeada, e os valores de latitude e longitude das posições fixas.

3.1 Entrada

A entrada para montar a matriz de distâncias D é dada a partir de um arquivo com o formato "índice do Nó, Latitude, Longitude".

A distância entre os nós é calculada através da fórmula de Haversine [10]:

$$2 \cdot R_e \cdot \arcsin \left(\sqrt{\sin^2\left(\frac{\phi_1 - \phi_2}{2}\right) + \cos(\phi_1) \cdot \cos(\phi_2) \cdot \sin^2\left(\frac{\lambda_1 - \lambda_2}{2}\right)} \right)$$

onde R_e representa o raio da terra, nesse trabalho R_e corresponde a 6373 km, e os termos ϕ e λ são os valores de latitude e longitude respectivamente. A matriz D possuirá todas as distâncias entre os nós, onde $D[i][j]$ é igual a distância entre os nós de posição fixa i e j , e i e j são os identificadores dos nós.

A entrada para a construção da matriz de adjacências F possui a forma "índice do Nó Origem, índice do Nó Destino", significando que existe um enlace entre os nós indicados.

A matriz F representará a topologia a ser mapeada, em que $F[i][j]$ possuirá os valores 1 ou 0, sendo que 1 representa que existe enlace e o 0 representa que não existe enlace entre os nós i e j .

3.2 População Inicial/Codificação

O primeiro passo do algoritmo é a construção da população inicial. Todos os indivíduos são gerados de forma aleatória, constituindo a população P , cada indivíduo da população é representado por um vetor L com o tamanho igual a quantidade de nós (vértices) ($|V(G)|$) a serem mapeados. O valor, t , contido em $L[i]$ representa o nó na posição fixa, sobre o qual o nó com número correspondente ao índice de $L[i]$ será mapeado.

Respeitando a restrição que a relação entre nó e posição fixa é de 1 para 1, não é possível

que o valor t seja mapeado nas posições $L[i]$ e $L[i']$ em um mesmo indivíduo, ou seja, o valor t não pode estar mapeado em mais de uma posição no indivíduo $P[j]$.

Abaixo temos 3 diferentes indivíduos representados em vetores, observe que o índice (número superior) representa o nó a ser mapeado e o valor representa a posição fixa. Tomando como exemplo o indivíduo c e seguindo a ordem de mapeamento {nó \rightarrow posição fixa}, temos o seguinte mapeamento, $\{0 \rightarrow 4'\} \{1 \rightarrow 0'\} \{2 \rightarrow 3'\} \{3 \rightarrow 2'\} \{4 \rightarrow 1'\}$.

Nó:	0	1	2	3	4	0	1	2	3	4	0	1	2	3	4
Posição Fixa:	0'	1'	2'	3'	4'	4'	3'	2'	1'	0'	4'	0'	2'	3'	1'
	(a) Indivíduo					(b) Indivíduo					(c) Indivíduo				

A figura 3.1 representa o mapeamento dos nós para as posições fixas, dos indivíduos a , b e c , onde o valor dentro do círculo representa o nó que foi mapeado, e o valor fora do nó (n') representa a posição fixa. Observe que os enlaces não são exibidos, a forma com que eles são aplicados será demonstrado na sessão 3.3.

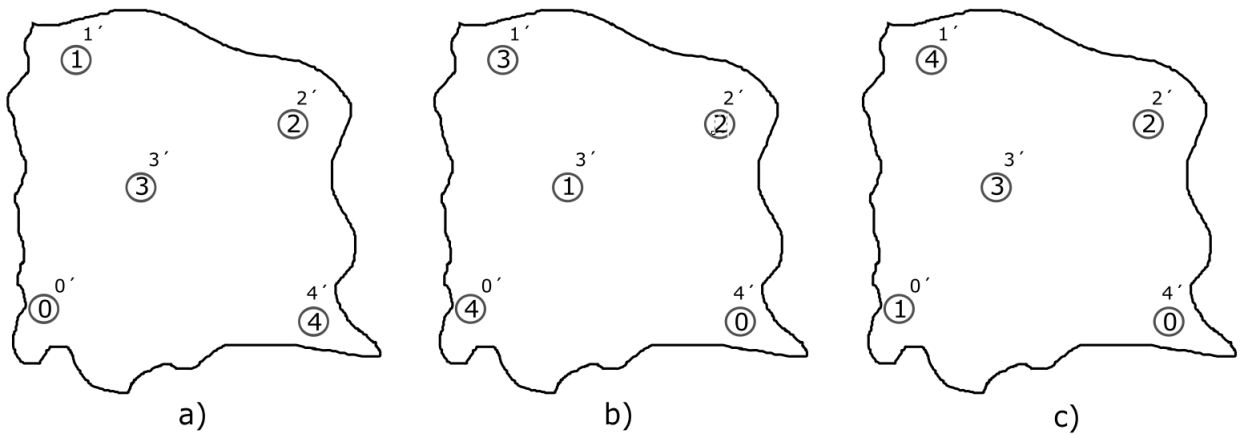


Figura 3.1: Mapeamentos dos Indivíduos.

3.2.1 Restrição de distância máxima

Com o objetivo de diminuir o espaço de busca, foi implementada uma restrição de distância máxima entre dois nós diretamente conectados por um enlace ao gerar a população inicial. Caso o número de enlaces que tenham o comprimento maior que a distância máxima é maior que o permitido, o indivíduo é rejeitado e outro é codificado em seu lugar. O número de enlaces permitidos para cada indivíduo pode variar, nos testes aplicados para indivíduos até 30 enlaces, o número de enlaces superiores a distância máxima foi de 0, ou seja nenhum enlace pode conter a distância maior que a permitida, e para calcular a distância máxima permitida

são utilizadas as matrizes D e F . Primeiramente é verificado o enlace que possui o maior grau utilizando a matriz

$$F = \begin{bmatrix} 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \end{bmatrix}$$

somando as linhas da matriz e identificando qual é a linha que tem o maior somatório, ou seja o maior número de nós adjacentes. Nesse caso possuímos as linhas 0, 1, 2 e 3 iguais a 3, identificando que o vértice com maior grau é igual a 3.

Após a identificação do maior grau da topologia a ser mapeada, e verificado na matriz

$$D = \begin{bmatrix} 0 & 8 & 10 & 3 & 9 \\ 8 & 0 & 7 & 4 & 12 \\ 10 & 7 & 0 & 5 & 6 \\ 3 & 4 & 5 & 0 & 7 \\ 9 & 12 & 6 & 7 & 0 \end{bmatrix}$$

qual é a 3ª (3 é o número de maior grau identificado no processo anterior) menor distância de cada nó de posição fixa, a utilização do 3º nó mais próximo é devido ao maior grau de todos os nós, permitindo que todos os nós possam ser mapeados para qualquer posição fixa. Utilizando as linhas da matriz D , a 3ª menor distância de cada linha, desconsiderando a diagonal principal, são, linha 0 = 9, linha 1 = 8, linha 2 = 7, linha 3 = 5 e a linha 4 = 9. Após a definição do conjunto das maiores 3ªs menores distâncias {9, 8, 7, 5, 9}, identificamos qual é o maior valor do conjunto, no nosso caso é o valor 9, definindo que a distância máxima permitida para o comprimento dos enlaces será 9.

Para o nosso exemplo nenhum enlace poderá conter o comprimento maior que 9. A figura 3.2 representa os enlaces que não são permitidos em um indivíduo do exemplo. Indivíduos que forem gerados com estes enlaces serão rejeitados.

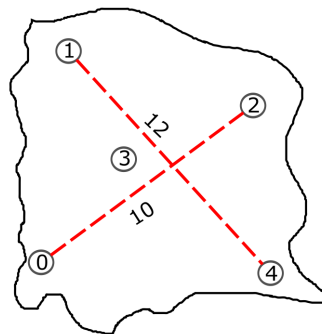


Figura 3.2: Enlaces não permitidos.

3.3 Avaliação

Na avaliação é feito o somatório dos comprimentos dos enlaces da topologia, para efetuar o cálculo são utilizadas as matrizes de adjacências F e de distâncias D .

Vamos utilizar como exemplo o indivíduo c e as matrizes D e F para calcular o a soma

$$\begin{array}{rccccc} \text{Nó:} & 0 & 1 & 2 & 3 & 4 \\ \text{Posição Fixa:} & 4' & 0' & 2' & 3' & 1' \end{array}$$

dos comprimentos dos enlaces do mapeamento.

$$F = \begin{bmatrix} 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \end{bmatrix} \quad D = \begin{bmatrix} 0 & 8 & 10 & 3 & 9 \\ 8 & 0 & 7 & 4 & 12 \\ 10 & 7 & 0 & 5 & 6 \\ 3 & 4 & 5 & 0 & 7 \\ 9 & 12 & 6 & 7 & 0 \end{bmatrix}$$

Primeiramente é verificado se os nós 0 e 1 (índices) possuem ligação, se $F[0][1] = 1$ adicionamos ao custo do indivíduo a distância entre os nós de posição fixa $4'$ e $0'$ (valores do vetor que representa o indivíduo). Esse processo é repetido para todos os pares de nós da matriz superior de F . Os vértices que são vizinhos no nosso exemplo são $F[0][1]$, $F[0][2]$, $F[0][4]$, $F[1][2]$, $F[1][3]$, $F[2][3]$ e $F[3][4]$, e as distâncias a serem somadas entre os nós de posição física, conforme os nós vizinhos são respectivamente $D[4'][0'] = 9$, $D[4'][2'] = 6$, $D[4'][1'] = 12$, $D[0'][2'] = 10$, $D[0'][3'] = 3$, $D[2'][3'] = 5$ e $D[3'][1'] = 4$, gerando o mapeamento da figura 3.3 com um custo total de 49 quilômetros.

Este custo é dado pela expressão $\sum_{i=1}^n \sum_{j=1}^n f_{ij} d_{\pi(i)\pi(j)}$.

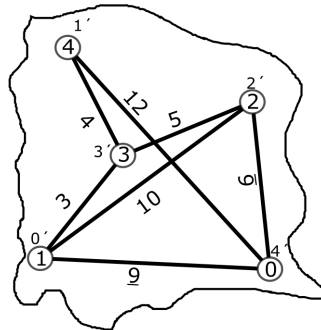


Figura 3.3: Mapeamento do Indivíduo C.

Esse é cálculo do custo do indivíduo, mas observe que ele não respeita a restrição de distância máxima apresentado na seção 3.2.1 ele utiliza dois enlaces com distância maior que

a máxima permitida ($D[0'][2'] = 10$ e $D[4'][1'] = 12$), logo o indivíduo c não é aceito pelo algoritmo genético se gerado na população inicial, mas ele é um possível mapeamento.

3.4 Seleção

O método de seleção utilizado neste trabalho foi o método de Roleta, os pais são selecionados conforme a sua aptidão [13], quanto melhor for o indivíduo melhor será a sua chance de seleção. Após a definição da população, cada indivíduo recebe um valor conforme a sua aptidão, onde os mais aptos recebem um valor maior do que os menos aptos.

Para exemplificação vamos utilizar um roleta que possua 10 bilhetes numerados de 1 a 10, esses bilhetes são divididos entre 3 indivíduos 1, 2 e 3. O indivíduo 1 é o mais apto de todos, após o indivíduo 2 e por fim o menos apto é o indivíduo 3. Para o nosso caso, dizer que um indivíduo é melhor que o outro é provado através da comparação do comprimento total da soma dos enlaces de cada indivíduo, quanto menor o comprimento mais apto será o indivíduo. O indivíduo 1 recebe 5 bilhetes {1,2,3,4,5}, o indivíduo 2 recebe 3 bilhetes {6,7,8} e o indivíduo 3 recebe 2 bilhetes {9,10}, conforme a figura 3.4, a forma de distribuição é feita ordenadamente pela aptidão dos indivíduos, onde os mais aptos recebem os primeiros bilhetes.

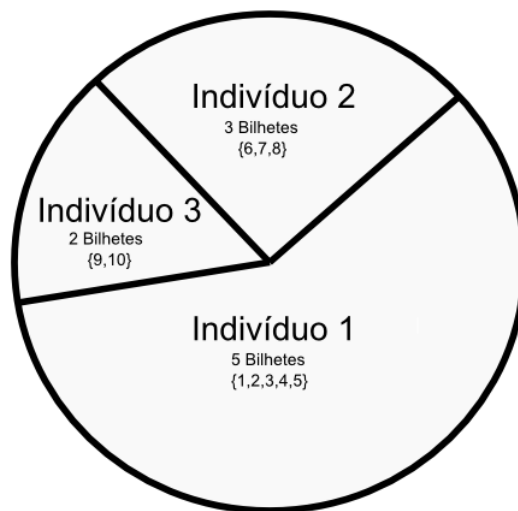


Figura 3.4: Roleta que contém os bilhetes dos Indivíduos 1, 2 e 3.

Com a distribuição dos bilhetes feita o próximo passo é sortear os indivíduos para fazer o cruzamento entre eles, o programa sorteia 1 número aleatório entre 1 e 10 (Número do primeiro e do último bilhete), se o número sorteado for ≥ 1 e ≤ 5 o indivíduo selecionado será o 1, se o número for ≥ 6 e ≤ 8 o indivíduo selecionado será o 2 e se o número for ≥ 9 e ≤ 10 o indivíduo selecionado será o 3. O mesmo processo é repetido duas vezes, para selecionar 2

indivíduos, não existindo controle para que não seja possível selecionar duas vezes o mesmo indivíduo, ou seja, o mesmo indivíduo pode ser selecionado duas vezes para o cruzamento, pois o número sorteado é feito de forma aleatória.

Para este trabalho os valores dos bilhetes variam de 0 a 1, onde os primeiros bilhetes (iniciando em 0) são para os melhores indivíduos da população. Nesse processo é possível dar maior chances para os indivíduos mais aptos ou permitir maior exploração entre os indivíduos, aumentando o número de bilhetes concedidos para os menos aptos [1], porém os mais aptos sempre irão receber mais bilhetes.

3.5 Cruzamento

Como vimos na seção anterior o cruzamento acontece entre dois indivíduos, esses dois indivíduos serão considerados "pais", e os novos indivíduos que surgiram do cruzamento serão os filhos, ou seja um novo par de indivíduos [3]. A quantidade de cruzamentos que acontece em cada ciclo do algoritmo genético é parametrizado e o método de cruzamento aplicado neste trabalho foi o cruzamento de ponto único [3].

Primeiramente o programa sorteia um número aleatório entre 1 e a quantidade de vértices menos 2, obrigando a sempre existir um cruzamento, pois se selecionarmos alguma das extremidades do indivíduo o cruzamento não acontecerá.

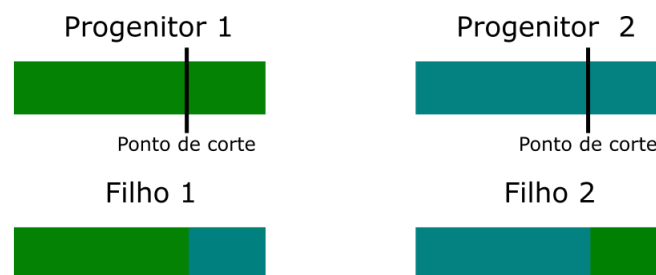


Figura 3.5: Cruzamento de Ponto Único.

O cruzamento de ponto único funciona da seguinte forma, o valor do ponto de corte define onde os indivíduos serão divididos, e cada indivíduo será repartido em 2 partes, a parte anterior ao ponto de corte e a parte posterior ao ponto de corte, gerando 4 partes entre os dois indivíduos. Essas quatro partes geram 2 novos filhos, onde os filhos conterão a primeira parte de um indivíduo e a segunda parte de outro indivíduo, gerando um cruzamento conforme o representado na figura 3.5 onde o Filho 1 é constituído da primeira parte do Progenitor 1 e a segunda parte do Progenitor 2, e o Filho 2 é constituído da primeira parte do Progenitor 2 e a

segunda parte do Progenitor 1.

Para representar o nosso caso vamos utilizar como exemplo os seguintes indivíduos:

Nó:	0	1	2	3	4	0	1	2	3	4
Posição Fixa:	4'	3'	2'	1'	0'	4'	0'	3'	2'	1'

e ponto de corte será igual a 2, gerando os filhos representados na figura 3.6.

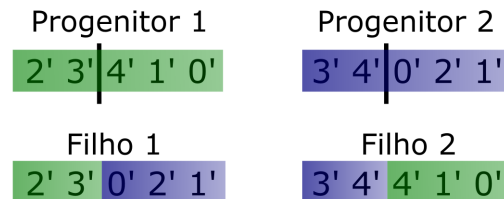


Figura 3.6: Cruzamento de Ponto Único dos Indivíduos.

Conforme descrito na sessão 3.2 em um único indivíduo um valor não pode aparecer mais de uma vez, e em alguns casos o cruzamento acaba gerando indivíduos que não são possíveis mapeamentos, pois a relação entre nó e posição fixa é de 1 para 1, e no caso presente na figura 3.6 podemos observar que o *Filho1* e o *Filho2* não são possíveis mapeamentos, ambos possuem valor repetido, no *Filho1* o valor 2 se repete e no *Filho2* o valor 4 se repete. Nesses casos os indivíduos passam por um tratamento que os ajusta, para se tornarem indivíduos válidos.

O primeiro passo para transformar os novos indivíduos em possíveis soluções, é identificar os nós que se repetem em cada indivíduo, onde os nós que se repetem em um filho são os não que não aparecem no outro filho. Após os filhos são percorridos do início até o fim, e quando chegamos a um nó que é reconhecido como repetido, adicionamos o nó que se repete no outro filho no seu lugar, sempre de forma ordenada, ou seja, o nó que se repete é substituído pelo primeiro menor nó que se repete no outro filho, e quando chegamos no último elemento do indivíduo fazendo essa verificação, o indivíduo se torna válido e uma possível solução para o nosso caso. No fim do processo os Filhos 1 e 2 presentes na figura 3.6 se tornam os indivíduos 1 e 2 respectivamente, presentes na figura 3.7.



Figura 3.7: Indivíduos ajustados após o cruzamento.

3.6 Mutação

Neste trabalho foram utilizadas três formas de mutações, a mutação de troca, inserção e inversão. Para todas as mutações são sorteados 2 valores aleatórios entre 0 e a quantidade de vértices -1, e a seleção de 1 indivíduo aleatório da população. A quantidade de indivíduos que sofrem mutação é parametrizada e acontece a cada ciclo do algoritmo, onde os indivíduos selecionados para a mutação são os que iniciam na população no ciclo, os novos indivíduos que surgem do cruzamento só poderão passar pelo processo de mutação no próximo ciclo, se eles sobreviverem.

Para exemplificação vamos utilizar o seguinte indivíduo $\{4'0'3'2'1'\}$, e os valores aleatórios de 1 e 4. Tendo a definição dos parâmetros utilizados pelos processos de mutação, as mutações funcionam da seguinte forma:

- **Troca:** A troca é um processo simples que irá trocar o valor da posição 1 e da posição 4 entre si, ou seja, os valores $0'$ e $1'$ vão trocar de posição, gerando o indivíduo $\{4'1'3'2'0'\}$ conforme ilustrado na figura 3.8.

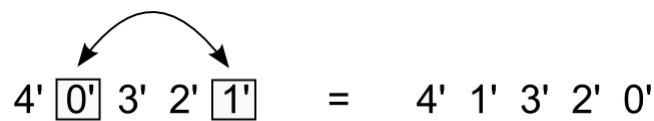


Figura 3.8: Processo de Troca (Mutação).

- **Inversão:** A mutação de inversão irá inverter todos os valores entre as posições 1 e 4 inclusive, ou seja, os valores $0'$, $3'$, $2'$ e $1'$ terão suas posições alteradas, gerando indivíduo $\{4'1'2'3'0'\}$ conforme a figura 3.9

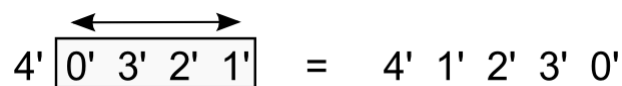


Figura 3.9: Processo de Inversão (Mutação).

- **Inserção:** A mutação de inserção é a mais complexa das 3 mutações, ela seleciona o indivíduo da posição 1 e coloca na posição 4, mas ao invés de trocar um valor com o outro, ele desloca os valores que estão entre as posições 1 e 4 inclusive, ele retira o valor da posição 1 desloca os valores das posições 2,3 e 4 uma casa para a esquerda e insere o valor da posição na posição 4, onde os valores $0'$, $3'$, $2'$ e $1'$ terão suas posições alteradas, o processo efetuado esta ilustrado na figura 3.10.

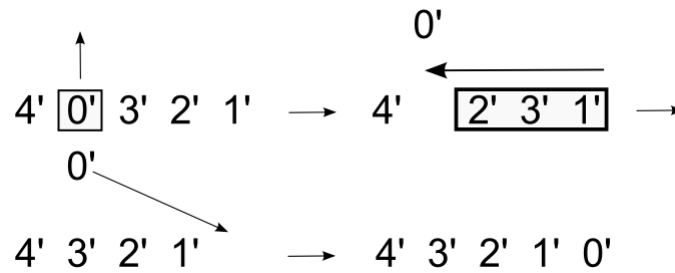


Figura 3.10: Processo de Inserção (Muta  o).

O processo sempre aloca o valor do primeiro n  mero aleat  rio sorteado no lugar do segundo, onde para este caso o primeiro valor (1)    menor que o segundo valor (4), quando o primeiro valor sorteado    maior que o segundo o processo    o mesmo, a   nica altera  o    que ao inv  s dos n  meros entre os valores deslocarem para a esquerda, eles ir  o deslocar para a direita.

3.7 Renova  o da Popula  o

Para este trabalho foi adicionado um novo processo no ciclo do algoritmo, o processo de Renova  o da Popula  o, ele acontece a cada certo n  mero de ciclos que pode ser configurado, no processo a popula  o    alterada, onde 10% dos melhores indiv  duos da popula  o s  o mantidos, e os outros 90% da popula  o s  o gerados novamente com o processo de inicializa  o da popula  o descrito na se  o 3.2. Para a cria  o do novo processo, foram efetuados testes e identificado que a renova  o causa mais impacto do que sempre manter a mesma popula  o dos melhores indiv  duos. Como a inicializa  o da popula  o cont  m restri  o de dist  ncia m  xima a chance de gerar indiv  duos com bons genes    melhor do que gerar indiv  duos de forma aleat  ria, e os objetivos desse novo processo    utilizar essa chance de bons genes, para cruzar com os melhores indiv  duos da popula  o e melhorar o custos dos mesmos, gerar indiv  duos com caracter  sticas diferentes, pois depois de uma quantidade de ciclos a popula  o pode acabar ficando parecida, e evitar que o algoritmo tenha uma converg  ncia precoce para um m  nimo local, pois m  todos que utilizam a aptid  o do indiv  duo nem sempre garantem uma boa converg  ncia do algoritmo [13].

No terceiro processo de renova  o o indiv  duo que corresponde ao mapeamento existente na topologia    adicionado na popula  o e possui o seguinte formato:

N��:	0	1	2	...	n
Posi��o Fixa:	0'	1'	2'	...	n'

O indivíduo que corresponde ao mapeamento da topologia original não é inserido no início, devido a ter grandes possibilidades de ser uma solução ótima e a competição com indivíduos gerados de forma aleatória iria aumentar muito a possibilidade do mapeamento ser escolhido. Ela é inserida somente no terceiro processo de renovação, para que os indivíduos gerados na população inicial possam melhorar suas aptidões, e no momento que ela entrar na população, o outros indivíduos serão tão bons quanto ela, podendo a topologia original contribuir para melhorar a aptidão da população, ou a população melhorar a aptidão do indivíduo da correspondente ao mapeamento da topologia real.

3.8 Descodificação

O processo de finalização e obtenção do resultado é feito através da condição de iterações sem melhoria, ou seja, o algoritmo considera um ciclo todo o processo de Seleção, Cruzamento, Mutação e Seleção da nova população, a cada ciclo a população é ordenada pelo valor de aptidão do indivíduo, onde na posição 0 da população estará sempre o melhor indivíduo, o processo de parada identifica a quantidade de ciclos que o valor da posição 0 não melhorou, ao atingir o valor configurado de iterações sem alteração o processo para e identifica que a melhor solução que o algoritmo encontrou para o caso é o indivíduo da posição 0.

3.9 Especificação do Algoritmo

Nesta seção será descrito o algoritmo implementado com a utilização dos processos descritos anteriormente. Os parâmetros utilizados são:

- **Tamanho da População(TP):** Parâmetro que define qual será o tamanho da população.
- **Iterações sem alteração(ISA):** Esse parâmetro é utilizado para definir o momento de parada do algoritmo, onde ele funciona como um contador que é incrementado em 1 a cada ciclo, para cada vez que o indivíduo da posição 0 não melhorar, no momento que esse indivíduo melhora a sua aptidão o contador é zerado, ou seja, o programa irá finalizar quando chegar ao número de iterações sem melhora, de uma forma continua.
- **Iterações para Renovação(IPR):** Esse parâmetro define o número de ciclos que devem acontecer até a renovação da população, onde o contador é zerado a cada renovação, podendo renovar inúmeras vezes a cada execução.

- **Pressão de seleção(PS):** Esse parâmetro é utilizado para a escolha do indivíduo na população para o cruzamento, onde ele pode dar mais ênfase aos melhores indivíduos, ou aumentar os nível de exploração dando maiores oportunidades para a escolha de indivíduos menos aptos da população [1]. Quanto menor o valor desse parâmetro maior será a chance dos indivíduos mais aptos serem selecionados.
- **Porcentagem de Cruzamento(PC):** Define a quantidade de pares de indivíduos que irão ser cruzados, a porcentagem é calculada sobre o tamanho da população.
- **Porcentagem de Mutação(PM):** Define a quantidade de indivíduos que sofrerão mutação, a porcentagem é calculada sobre o tamanho da população.
- **Porcentagem superior a Distância Máxima($PSDM$):** Parâmetro que define quantos por cento dos enlaces do indivíduo podem ser maiores que a distância máxima calculada, a porcentagem é calculada sobre o total de enlaces da topologia a ser mapeada

O fluxograma do algoritmo implementado pode ser visualizado na Figura 3.11.

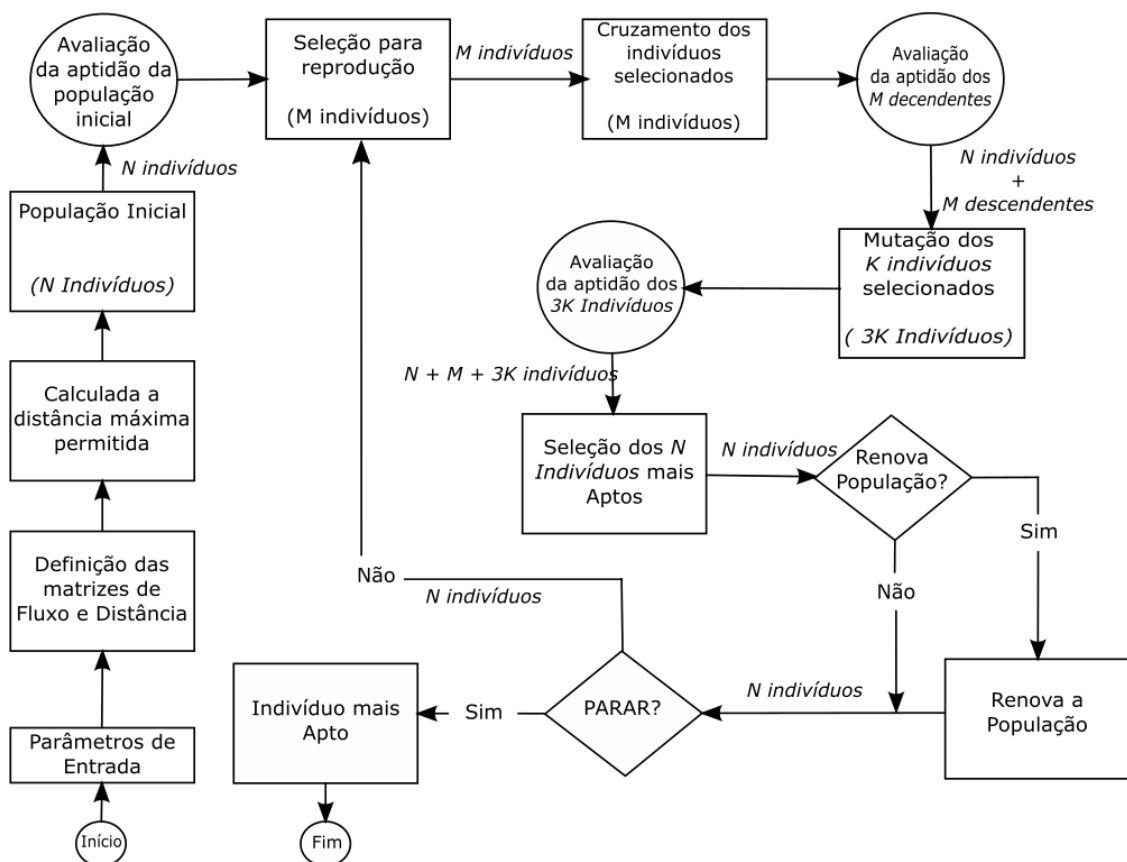


Figura 3.11: Fluxograma do algoritmo implementado.

O primeiro passo do algoritmo é definir as matrizes D e F conforme a entrada, como é descrito na seção 3.1, com a entrada definida é calculada a distância máxima permitida, ver seção 3.2.1. Após são gerados N indivíduos, definidos pelo parâmetro **Tamanho da População**, como é descrito na seção 3.2. A aptidão de cada indivíduo é calculada conforme a seção 3.3. Seguidamente são selecionados os pares de indivíduos com o método de roleta, descrito na seção 3.3. Esses pares indivíduos serão cruzados gerando M novos indivíduos com a utilização do método de cruzamento de ponto único, ver seção 3.5, onde a quantidade de pares selecionados é definida no parâmetro **Porcentagem de Cruzamento**, e para cada par de indivíduos são gerados 2 indivíduos. Na próxima etapa são selecionados K indivíduos onde cada um sofre mutações de 3 formas diferentes, seção 3.6, gerando 3 novos indivíduos, a quantidade K de indivíduos que sofrerão mutação é definida através parâmetro **Porcentagem de Mutação**. Após os novos indivíduos são avaliados e a população tem um total de $N + M + 3K$ indivíduos.

Dos $N + M + 3K$ indivíduos serão selecionados os N melhores indivíduos, se a restrição de Renovação for atendida, ver seção 3.7, 10% dos indivíduos de N serão mantidos e os outros 90% restante serão gerados com o processo de inicialização descrito na seção 3.2. Esse processo se repete até que o critério de parada seja atendido, ver seção 3.8. Quando o processo de parada é atingido o indivíduo mais apto é decodificado e correspondente a melhor solução encontrada.

4 RESULTADOS COMPUTACIONAIS

Neste capítulo serão apresentados os resultados computacionais obtidos através da meta-heurística implementada, aplicando o método matemático PQA e algoritmo genético ao mapeamento de redes ópticas de telecomunicações. Ao todo foram utilizadas 27 topologias existentes para a aplicação dos testes. Todas as topologias selecionadas também passaram por um tratamento de melhoria, apresentadas no trabalho [16], que explora a subclasse das topologias Harary, maximizando a confiabilidade das redes.

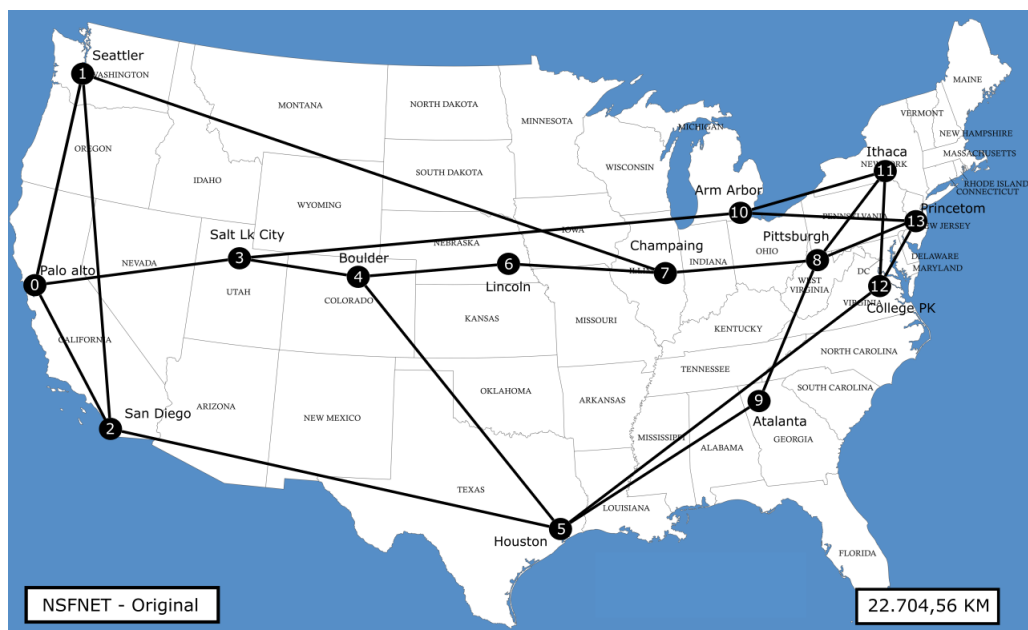


Figura 4.1: Mapeamento Original da rede NSFNET

Como o problema do PQA é um problema da classe NP-Hard, conforme o aumento do número de vértices da topologia, se torna impossível obter resultados exatos em tempo útil. Para redes de até 14 nós foram calculadas todas as possibilidades para obtenção do melhor resultado, para as demais redes os resultados foram obtidos através do algoritmo genético implementado.

Para exemplo vamos utilizar a topologias NSFNET que possui 14 nós e 21 enlaces apresentada na figura 4.1, possuindo o total da soma do comprimento dos enlaces de 22.704,56 quilômetros.

Após aplicar a topologia NFSNET - Original como entrada no algoritmo implementado, foi possível reduzir o comprimento da soma dos enlaces de 22.704,56 para 21.707,94 quilômetros, minimizando 996,62 quilômetros alterando a posição do mapeamento dos nós. A topologia NSFNET - Minimizada é apresentada na figura 4.2.

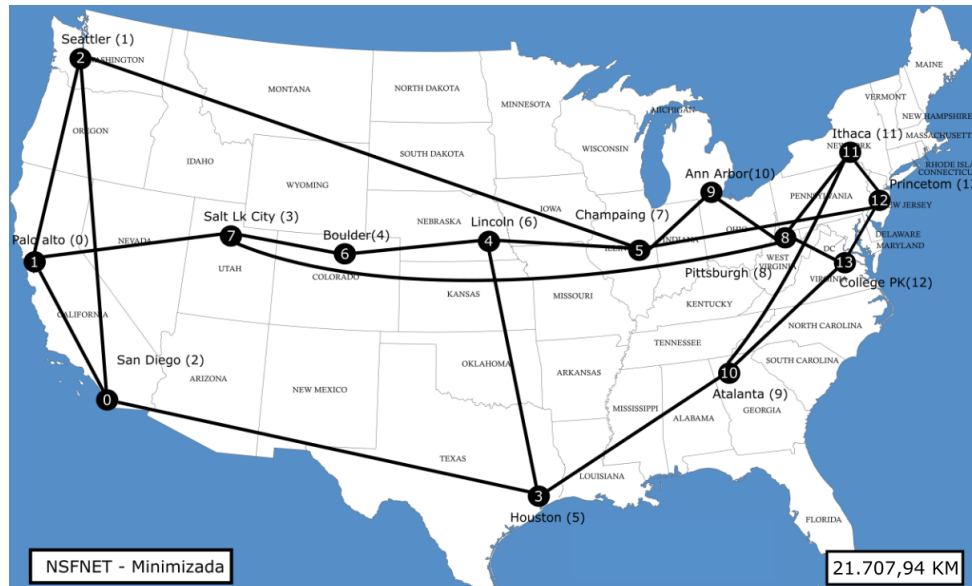


Figura 4.2: Mapeamento Minimizado da rede NSFNET

Na figura 4.3 é apresentada a topologia Harary a ser mapeada sobre os nós de posição fixa da rede NSFNET.

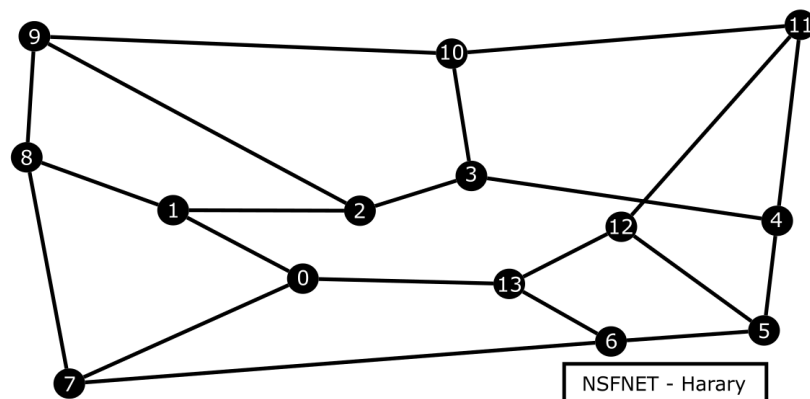


Figura 4.3: Topologia NSFNET - Harary

As topologias Harary não possuem um mapeamento original já definido, só existe a informação da conectividade entre os nós. O mapeamento original da topologia NSFNET - Harary segue o mesmo mapeamento da NSFNET Original, onde o nó é mapeado na posição fixa que corresponde ao seu identificador, gerando o mapeamento presente na figura 4.4

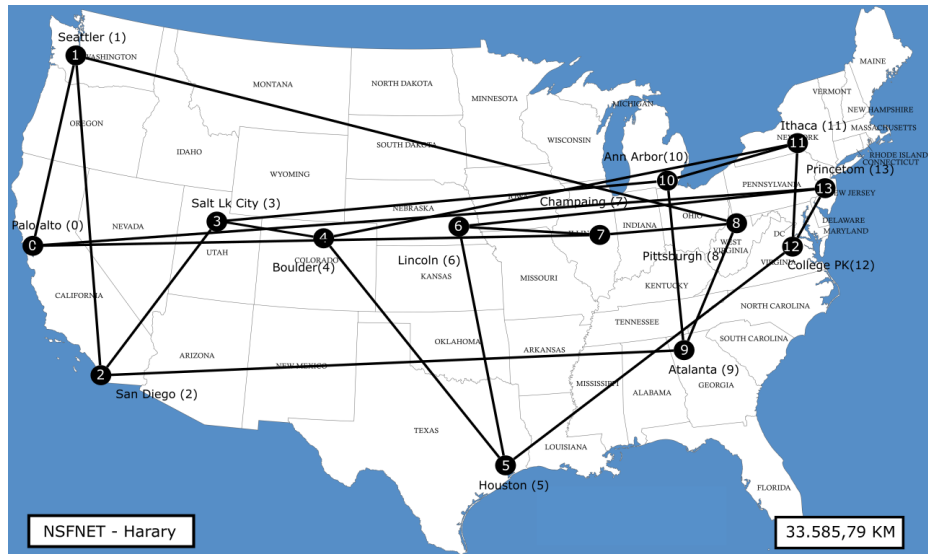


Figura 4.4: Mapeamento NFSNET - Harary conforme a Topologia Original

Após aplicar a topologia Harary da figura 4.3 como entrada no algoritmo implementado, temos como resultado o mapeamento da figura 4.5.

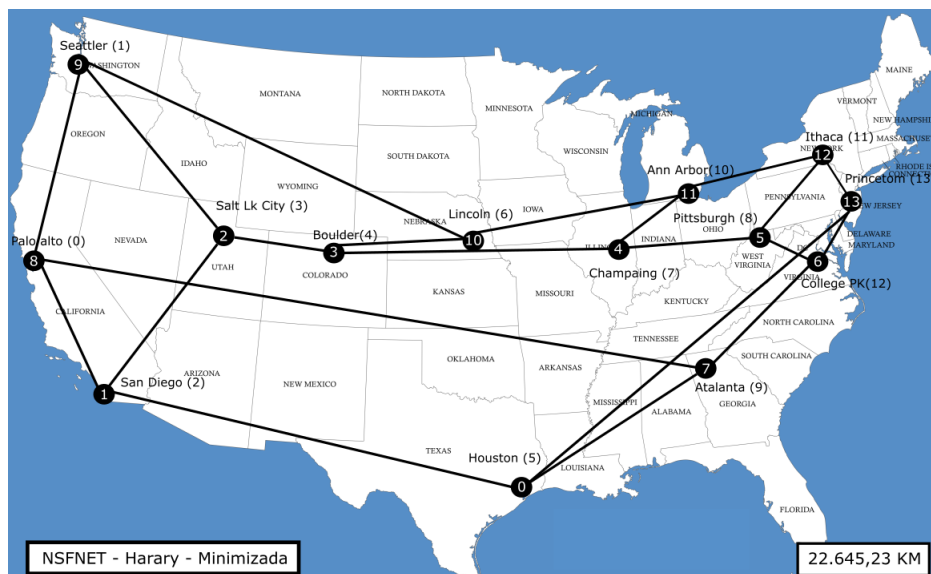


Figura 4.5: Mapeamento Minimizado da rede NSFNET - Harary

Os mapeamentos apresentados para a rede NSFNET, são as soluções ótimas para o caso. A topologia original foi reduzida em 996,62 quilômetros, e o melhor mapeamento para a rede NSFNET - Harary reduziu um total de 59,33 quilômetros referente a topologia original. Na figura 4.6 é apresentada a diferença de tempo de execução entre o algoritmo genético implementado e o teste de todas as possibilidades, levando em consideração que o método retorna o melhor caso encontrado no término da execução, o algoritmo genético inicia em um indivíduo aleatório e termina a sua execução aos 120 segundos pois a condição de parada foi atendida,

o algoritmo de toda as possibilidades inicia no indivíduo que corresponde ao indivíduo original e só traz o resultado ótimo após testar todas as possibilidades, e para chegar ao fim da sua execução leva 22 horas.

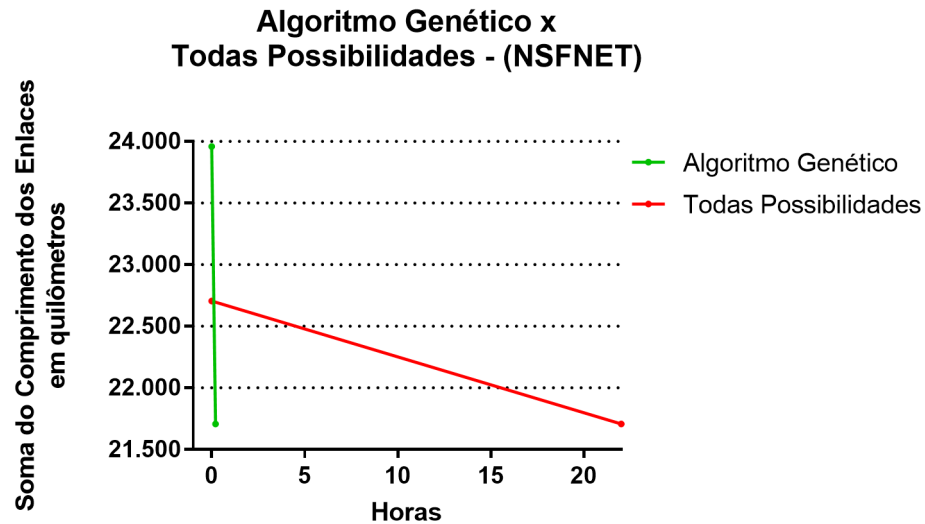


Figura 4.6: Comparação entre os Métodos para a rede NSFNET

Na tabela 4.1 são apresentadas todas as topologias utilizadas, a quantidade de vértices e a quantidade de arestas de cada rede. Observando que para todas as topologia também existe uma topologia da subclasse das topologias Harary.

Tabela 4.1: Redes de Referência.

Redes de Telecomunicações		
Rede	Quantidade de Nós	Quantidade de Enlaces
ARNES	17	20
ARPANET	20	32
AUSTRIA	15	22
CANARIE	19	26
CESNET	12	19
COX	24	40
EON	19	37
GEANT2	32	52
GERMANY	17	26
INTERNET2	56	61
ITALY	14	29
LAMBDARAILUSA	19	23
MEMOREX	19	24
METRONA	33	41
MZIMA	15	19
NEWNET	26	31
NSFNET	14	21
OMNICOM	38	54
PIONIER	21	25
PORTUGAL	26	36
RENATER	27	35
RNP	10	12
SANET	25	28
SPAIN	17	28
USA	100	171
vBNS	12	17
VIA	9	12

Para a obtenção dos melhores resultados, a configuração dos parâmetros altera conforme a quantidade de vértices da topologia. Foram criadas 4 parametrizações, apresentadas na tabela 4.2, a Configuração 1 foi utilizada para topologias com até 20 nós, a Configuração 2 foi utilizada para topologias entre 20 e 30 nós, a Configuração 3 foi utilizada para topologias com a quantidade entre 30 e 60 nós e a Configuração 4 foi utilizada para topologias com quantidade superior a 60 nós. Para a definição dos parâmetros foram efetuados vários testes com alguns tipos de topologias, identificado qual era a iteração média que a população se estabilizava e não melhorava mais.

Os resultados são apresentados em 5 tabelas, uma para cada configuração e uma para as topologias com os resultados ótimos calculados, cada tabela possui até 4 tipos de mapeamentos

Tabela 4.2: Configurações Utilizadas.

Configuração 1		Configuração 2		Configuração 3	
Parâmetro	Valor	Parâmetro	Valor	Parâmetro	Valor
<i>TP</i>	80	<i>TP</i>	90	<i>TP</i>	90
<i>ISA</i>	2.000	<i>ISA</i>	5.000	<i>ISA</i>	10.000
<i>IPR</i>	500	<i>IPR</i>	1.000	<i>IPR</i>	2.000
<i>PS</i>	2	<i>PS</i>	2	<i>PS</i>	3
<i>PC</i>	70%	<i>PC</i>	70%	<i>PC</i>	80%
<i>PM</i>	70%	<i>PM</i>	70%	<i>PM</i>	80%
<i>PSDM</i>	0	<i>PSDM</i>	40%	<i>PSDM</i>	40%

Configuração 4	
Parâmetro	Valor
<i>TP</i>	100
<i>ISA</i>	20.000
<i>IPR</i>	4.000
<i>PS</i>	4
<i>PC</i>	90%
<i>PM</i>	90%
<i>PSDM</i>	55%

por rede, a topologia *Original* do mundo real, a topologia original *Minimizada*, a topologia da classe *Harary* e para topologias de até 14 nós existe o mapeamento *Otimo*. O mapeamento *Otimo* foi calculado através do teste de todas as possibilidades de forma contínua e individual. A meta-heurística implementada foi executada para todas as topologias, e seus resultados foram comparados com os resultados ótimos obtidos, quando existir, e a variação da soma dos comprimentos dos enlaces que os mapeamentos tiveram em relação ao mapeamento da topologia do mundo real da qual surgiram.

As tabelas de resultados possui as seguintes colunas:

- **Topologia:** Representa qual é a Rede do Mundo Real.
- **Mapeamento:** Os tipos de mapeamentos. Observando que o Mapeamento Harary se trata da topologia original que passou por um tratamento.
- **Distância total (KM):** Melhor resultado obtido.
- **Ganho:** Porcentagem de ganho referente ao Mapeamento Original.
- **Tempo de execução(Segundos):** Tempo em segundos para a obtenção do resultado.

Tabela 4.3: Resultados - Configuração 1 e com soluções ótimas (0 - 20 nós).

Resultados Ótimos e Obtidos com a Meta-Heurística				
Topologia	Mapeamento	Distância total (KM)	Ganho	Tempo(Segundos)
CESNET	Original	1.841,95	-	-
	Melhorado(Ótimo)	1.821,27	1,22%	513
	Melhorado	1.821,27	1,22%	100
	Harary(Ótimo)	2.035,75	-10,52%	427
	Harary	2.035,75	-10,52%	94
ITALY	Original	8.393,24	-	-
	Melhorado(Ótimo)	8.231,47	1,92%	77.987
	Melhorado	8.231,47	1,92%	102
	Harary(Ótimo)	8.607,63	-2,55%	76.265
	Harary	8.607,63	-2,55%	100
NSFNET	Original	22.704,56	-	-
	Melhorado(Ótimo)	21.707,94	4,38%	79.222
	Melhorado	21.707,94	4,38%	130
	Harary(Ótimo)	22.645,23	0,26%	80.294
	Harary	22.645,23	0,26%	136
RNP	Original	8970.07	-	-
	Melhorado(Ótimo)	7.951,93	11,35%	0
	Melhorado	7.951,93	11,35%	98
	Harary(Ótimo)	8.353,26	6,87%	0
	Harary	8.353,26	6,87%	97
vBNS	Original	16.259,95	-	-
	Melhorado(Ótimo)	15.858,09	2,47%	530
	Melhorado	15.858,09	2,47%	110
	Harary(Ótimo)	21.282,99	-30,89%	482
	Harary	21.282,99	-30,89%	137
VIA	Original	6708.61	-	-
	Melhorado(Ótimo)	6.708,61	0%	2
	Melhorado	6.708,61	0%	101
	Harary(Ótimo)	7.751,04	-15,53%	1
	Harary	7.751,04	-15,53%	95

Tabela 4.4: Resultados - Configuração 1 (15 - 20 nós).

Resultados Obtidos com a Meta-Heurística				
Topologia	Mapeamento	Distância total (KM)	Ganho	Tempo(Segundos)
ARNES	Original	730,10	-	-
	Melhorado	702,37	3,79%	120
	Harary	775,47	-6,21%	128
ARPANET	Original	26.626,87	-	-
	Melhorado	25.311,31	4,94 %	115
	Harary	26.284,10	1,28%	165
AUSTRIA	Original	2669,80	-	-
	Melhorado	2.145,16	19,65%	120
	Harary	2.178,58	18,39%	119
CANARIE	Original	17.423,60	-	-
	Melhorado	15,822,84	9,18%	113
	Harary	20.009,43	-14,84%	131
EON	Original	23.729,10	-	-
	Melhorado	23.729,10	0%	269
	Harary	27.645,81	-16.50%	171
GERMANY	Original	3.680,29	-	-
	Melhorado	3.512,41	4,56%	110
	Harary	3.891,54	-5,74%	155
LAMBDA RAIL	Original	15.900,32	-	-
	Melhorado	15.786,09	0,17%	174
	Harary	17.669,93	-11,12%	193
MEMOREX	Original	3.459,70	-	-
	Melhorado	3.459,70	0%	240
	Harary	3.861,32	-11,60%	253
MZIMA	Original	16.146,70	-	-
	Melhorado	16.146,70	0%	142
	Harary	19.278,03	-19,39%	162
SPAIN	Original	8.884,38	-	-
	Melhorado	8.080,05	9,05%	127
	Harary	8.688,85	2,20%	164

Tabela 4.5: Resultados - Configuração 2 (21 - 40 nós).

Resultados Obtidos com a Meta-Heurística				
Topologia	Mapeamento	Distância total (KM)	Ganho	Tempo(Segundos)
COX	Original	26.942,62	-	-
	Melhorado	25.295,12	6,11%	1.024
	Harary	26.831,81	0,41%	1.596
GEANT	Original	33.523,96	-	-
	Melhorado	28.781,61	14,14%	721
	Harary	32.182,90	4,81%	131
METRONA	Original	2.982,31	-	-
	Melhorado	2.947,83	1,15%	1.717
	Harary	2.987,54	-0,17%	2.894
NEUNET	Original	16.297,15	-	-
	Melhorado	16.297,15	0%	1.339
	Harary	18.789,68	-15,29%	1.382
OMNICOM	Original	16214,28	-	-
	Melhorado	16.053,28	0,99%	982
	Harary	18.073,05	-11,46%	771
PIONIER	Original	3.031,99	-	-
	Melhorado	3.031,99	0%	1.250
	Harary	3.095,22	-2,08%	1.208
PORTUGAL	Original	6.473,75	-	-
	Melhorado	6.458,04	0,24%	1.635
	Harary	5.942,74	8,20%	1.633
RENATER	Original	5.413,62	-	-
	Melhorado	5.413,62	0%	1.461
	Harary	5.443,42	-0,73%	609
SANET	Original	1.002,67	-	-
	Melhorado	1.002,67	0%	673
	Harary	1.031,49	-2,87%	422

Tabela 4.6: Resultados - Configuração 3 (41 - 60 nós).

Resultados Obtidos com a Meta-Heurística				
Topologia	Mapeamento	Distância total (KM)	Ganho	Tempo(Segundos)
INTERNET2	Original	18197.09	-	-
	Melhorado	18.197,09	0%	3.755
	Harary	19.902,65	-9,37%	3.274

Tabela 4.7: Resultados - Configuração 4 (Maiores que 60 nós).

Resultados Obtidos com a Meta-Heurística				
Topologia	Mapeamento	Distância total (KM)	Ganho	Tempo(Segundos)
USA	Original	47.398,56	-	-
	Melhorado	46.667,13	1,54%	11.700
	Harary	49.470,82	-4,37%	29.700

4.1 Análise de Resultados

A partir dos testes realizados foi possível verificar que 18 das 27 topologias testadas, tiveram a soma do comprimento dos enlaces minimizado com a alteração do mapeamento dos nós da rede. Na figura 4.7 é possível verificar as topologias que tiveram um percentual de ganho referente a topologia original, as topologias que não foram minimizadas são consideradas o mapeamento que minimiza ao máximo a soma do comprimento dos enlaces.

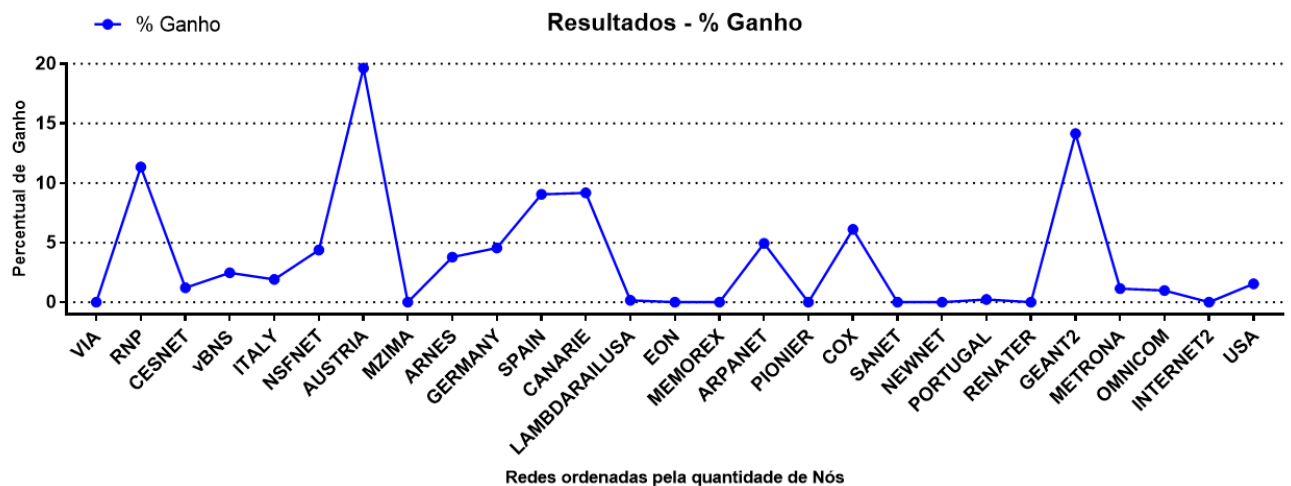


Figura 4.7: Percentual de Ganho das Topologias do Mundo Real

As topologias Harary maximizam a confiabilidade da rede [16], mas os testes provaram que 19 das 27 redes não tiveram ganho em relação a distância da topologia original, a variação pode ser visualizada na figura 4.8, onde é possível ver se a Topologia Harary teve um percentual

de ganho ou perda referente a topologia original. As topologias Harary ao comparadas com as topologias originais das quais surgiram, possuem o mesmo número de nós e de enlaces, porém a distribuição das conexões é diferente. As 19 redes que não tiveram o seu comprimento minimizado além da topologia original, possuíram um custo mais elevado, custo esse que pode se tornar viável ao maximizar confiabilidade da rede.

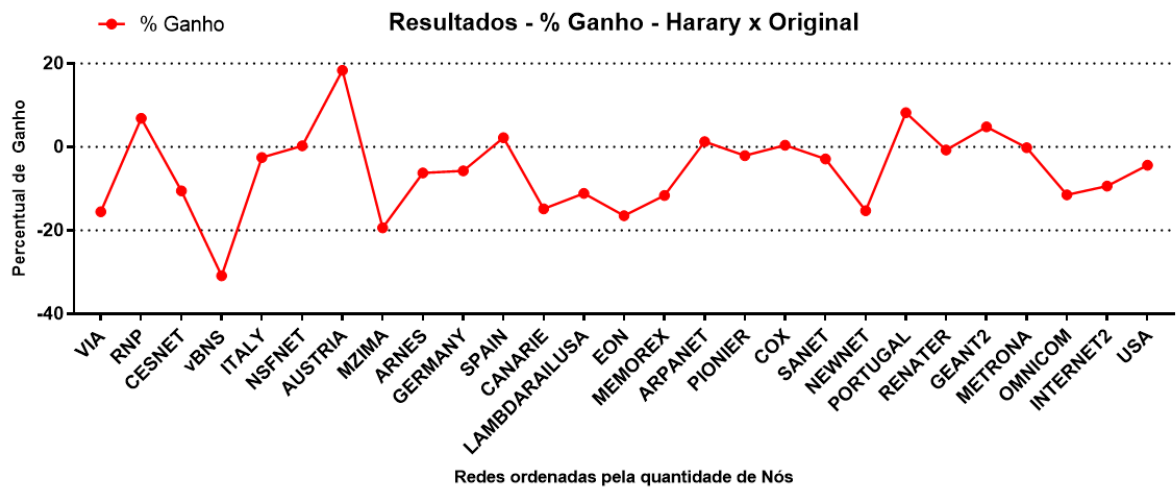


Figura 4.8: Percentual de Ganho das Topologias do Mundo Real

Não existe uma relação com as características da rede e o total que ela pode ser minimizada, a minimização das topologias do mundo real depende exclusivamente da forma com que elas estão mapeadas, e quanto melhores estiverem mapeadas menor será a sua otimização.

5 CONCLUSÕES

Neste trabalho foi estudado o Problema de Alocação Quadrático aplicado ao mapeamento de topologias de redes ópticas. A minimização da soma do comprimento dos enlaces, minimiza o custo da rede e potencializa o sinal transmitido pelo espectro. O PAQ é um problema NP-Difícil, e por isso foi utilizado a técnica algoritmo genético com características evolutivas.

Para aumentar a eficiência do algoritmo genético, foram implementadas restrições que se aplicam a geração da população inicial, com o objetivo de diminuir o espaço de busca. Para evitar a convergência precoce para um mínimo local, foram utilizados 3 tipos de mutações e um processo de renovação da população.

As topologias utilizadas para as simulações são topologias já existentes no mundo real, e foi provado que é possível minimizar a soma do comprimento dos enlaces da rede com a alteração de seu mapeamento, aplicando o PQA com algoritmo genético. Devido a complexidade do problema não foi possível encontrar a solução ótima para todos os casos, mas para os casos onde foi possível calcular a solução ótima, o algoritmo implementado se demonstrou eficaz e chegou aos resultados ótimos em um tempo computacionalmente aceitável.

A maioria das topologias originais testadas, tiveram o seu comprimento minimizado, as topologias que não tiveram seu comprimento minimizado tendem a serem soluções ótimas. As topologias da subclasse Harary não possuem um mapeamento original definido, mas os resultados obtidos se aproximaram da distância da topologia original, onde alguns mapeamentos tiveram um comprimento minimizado ao comparar a topologia original, mas a maioria das topologias tiveram um comprimento superior a topologia original.

Para trabalhos futuros propomos o estudos de novas formas de geração da população inicial, a utilização de outros métodos de cruzamento e mutação. A implementação de outras meta-heurísticas para a comparação de resultados. Criar métodos que sejam capazes de escolher a melhor configuração de parametrização conforme os indivíduos. Aplicar o PAQ ao mapeamento das topologias ópticas usando como valor de aptidão não somente a soma do comprimento dos enlaces, mas também utilizar a taxa de bloqueio e de fragmentação para definir os melhores mapeamentos.

REFERÊNCIAS

- [1] T. Back. Selective pressure in evolutionary algorithms: a characterization of selection mechanisms sign in or purchase. *Proceedings of the First IEEE Conference on Evolutionary Computation. IEEE World Congress on Computational Intelligence*, 1994.
- [2] B. C. Chatterjee, S. Nityananda, and E. Oki. Routing and spectrum allocation in elastic-optical networks: A tutorial. *IEE COMMUNICATION SURVEYS TUTORIALS*, 2015.
- [3] J. Dréo, A. Pétrowski, P. Siarry, and A. Taillard. Metaheuristics for hard optimization. *Springer*, 2006.
- [4] N. Fernandez, J. R. Duran, and I. de Miguel. Designing virtual topologies in exible optical networks. *International Conference On Transparent Optical Networks*, 2014.
- [5] N. Fernández, J. R. Durán, I. Miguel, N. Merayo, J. C. Aguado, P. Fernández, R. M. Lorenzo, and E. J. Abril. Designing virtual topologies in flexible optical networks. *Universidad de Valladolid, Paseo de Belén 15, 47011 Valladolid, Spain*, 2014.
- [6] O. Gerstel, M. Jinno, A. Lord, and S. J. B. Yoo. Elastic optical networking: A new dawn for the optical layer? *Communications Magazine, IEEE*, 2012.
- [7] O. Gerstel, M. Jinno, A. Lord, and S. J. B. Yoo. Spectrum-efficient and scalable elastic optical path network: Architecture, benefits, and enabling technologies. *Communications Magazine, IEEE*, 2012.
- [8] F. Herrera and M. Lozano. Gradual distributed real-coded genetic algorithms.
- [9] A. Horota, G. Figueiredo, and N. da Fonseca. Routing and spectrum assignment algorithm with fragmentation minimization in elastic optical networks. *2014 Brazilian Symposium on Computer Networks and Distributed Systems*, 2014.
- [10] D. Ives, P. Bayvel, and S. J. da Savory. Adapting transmitter power and modulation format to improve optical network performance utilizing the gaussian noise model of nonlinear impairments. *JOURNAL OF LIGHTWAVE TECHNOLOGY, VOL. 32*, 2014.

- [11] P. Ji, Y. Wu, and H. Liu. A solution method for the quadratic assignment problem (qap). *The Sixth International Symposium on Operations Research and Its Applications (ISORA'06)*, 2006.
- [12] E. M. Loiola, N. M. M. de Abreu, P. O. Boaventura-Netto, P. Hahn, and T. Querido. A survey for the quadratic assignment problem. *European Journal of Operational Research* 176 (2007) 657–690, 2007.
- [13] B. L. Miller and D. E. Goldberg. Genetic algorithms, selection schemes, and the varying effects of noise. *Evolutionary Computation*, 1995.
- [14] R. M. D. Morais. Desenho topológico de redes Ópticas. *Universidade de Aveiro Departamento de Matemática*, 2008.
- [15] A. N. H. Zaied and L. A. E. Shawky. A survey of the quadratic assignment problem. *International Journal of Computer Applications* (0975 –8887), 2014.
- [16] C. Pavan, L. S. de Lima, M. H. M. Paiva, and M. E. V. Segatto. How reliable are the real-world optical transport networks? *JOURNAL OF APPLIED MATHEMATICS AND DECISION SCIENCES*, 6(3), 143–153, 2002.
- [17] M. Shadab, Piruthiviraj.P, and S. da Dr. Preeta. Simulation and analysis of blocking probability using rwa algorithm for alloptical networks. *International Journal of Innovative Research in Computer and Communication Engineering*, 2014.
- [18] J. W. Suurballe and R. E. Tarjan. A quick method for finding shortest pairs of disjoint paths. *Networks, [S.l.]*, v.14, n.2, p.325–336, 1984.
- [19] K. Urahama and H. Nishiyuki. Neural algorithms for placement problems. *Proceedings of 1993 International Joint Conference on Neural Networks*, 1993.

6 APÊNDICE

6.1 Simulações Adicionais

Para fins de informação foram efetuadas simulações adicionais sobre os mapeamentos obtidos através do algoritmo genético, para testar cada mapeamento foi aplicado o Problema de Roteamento e Alocação de Espectro, comparando os resultados de Taxa de Fragmentação, Taxa de Bloqueio e Nó Central, para os mapeamentos originais, otimizados e Harary, verificando a eficiência da rede após a sua minimização, e se a alteração do mapeamento tem algum impacto sobre a eficiência da rede. Para testar a performance dos 3 tipos de mutações apresentados na sessão 3.6, eles foram testados individualmente no algoritmo implementado. Foram utilizadas as topologias Originais das redes *CESNET*, *ITALY*, *NSFNET*, *RNP*, que possuem o resultado da solução ótima.

6.2 Problema do Roteamento e Alocação de Espectro

Para verificar a eficiência das topologias geradas será aplicado o Problema do Roteamento e Alocação de Espectro, que a partir de uma topologia e um conjunto de requisições variadas, deve selecionar uma rota para cada requisição e alocar um número de slots para cada requisição, onde as informações devem estar distribuídas de uma forma contínua e contígua no espectro ao longo do caminho [2].

Para os testes efetuados, as requisições demandam 3 tipos de tráfego 30, 75 e 150 Gb/s, que correspondem a 1,2 e 4 slots de frequência utilizando o formato de modulação 8-QAM [4]. A rede possui os enlaces divididos em 320 slots.

O algoritmo irá alocar até 320 (quantidade de slots por enlace) requisições simultâneas na rede, no momento que a rede recebe as 320 requisições permitidas, o algoritmo retira 64 (20%) das requisições, e após realoca as 64 requisições, repetindo esses passos de retirada e inclusão de requisições, até atingir o total de requisições.

Para a alocação da requisição ao longo do espectro, são necessários dois caminhos com espectro suficiente para a demanda requisitada, o primeiro caminho será o caminho principal e o segundo será um caminho de backup, ambos os caminhos devem ser disjuntos de arestas, ou seja, o caminho principal e o caminho secundário não podem compartilhar o mesmo enlace, todos os enlaces de um caminho devem ser diferentes do outro caminho, para atender essa

exigência foi utilizada a técnica de Suurballe [18], utilizando como nós de origem e destino os nós de posição fixa.

Como medida dos resultados, serão consideradas a eficiência espectral medindo a grau de fragmentação do espectro expressada pela função $Fragmentacao = 1 - \frac{MaiorBlocoLivre}{TotaldeBlocosLivres}$ [9] e a taxa de bloqueio que pode ser expressada pela função $TaxaDeBloqueio = \frac{NumerodeRequisiesbloqueadas}{NumeroTotaldeRequisies}$ [17]. A eficiência espectral será medida pelo grau de fragmentação da rede, que após a inclusão e remoção de um conjunto de requisições na rede é verificada a incapacidade de uso de espectro livre, devido a pequenos blocos descontínuos, que não possuem a capacidade suficiente de espectro para atender a futuras requisições. A taxa de bloqueio será calculada através da divisão do número de requisições bloqueadas pelo número total de requisições.

6.2.1 Resultados

Os resultados são demonstrados em forma de gráfico e tabela, o gráfico exibe o número de requisições e o total de bloqueios, e na tabela são exibidos os valores da taxa de bloqueio e fragmentação utilizando as fórmulas apresentadas na seção 6.2, e o nó central da rede, que é escolhido através da quantidade de requisições que passaram pelo nó, onde o nó que tiver mais requisições passando sobre ele será o nó central. Para cada rede foi utilizada a mesma sequência e quantidade de requisições de entrada e saída, comparando os resultados entre os mapeamentos Harary, Melhorado e Original.

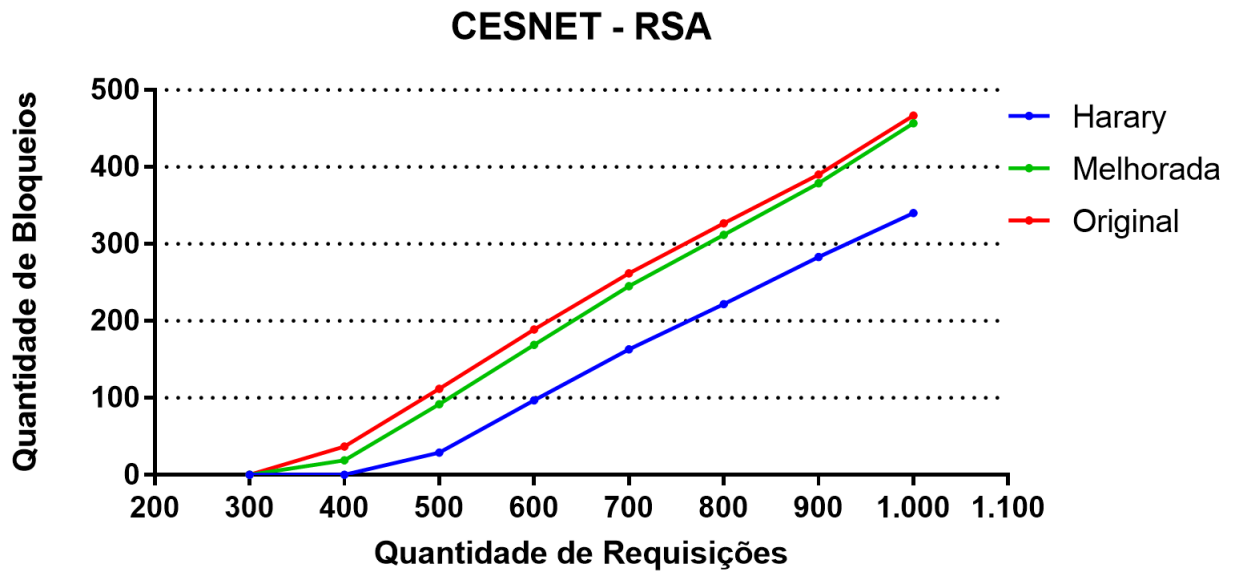


Figura 6.1: Quantidade de bloqueios para os Mapeamentos da rede CESNET

Tabela 6.1: Resultados - Problema de RSA.

CESNET				
Mapeamento	N de Requisições	Taxa de Bloqueio	Fragmentação	Nó Central
Original	1.000	0,34	0,91	Hradec
Melhorado	1.000	0,45	0,93	Praga
Harary	1.000	0,46	0,93	Praga

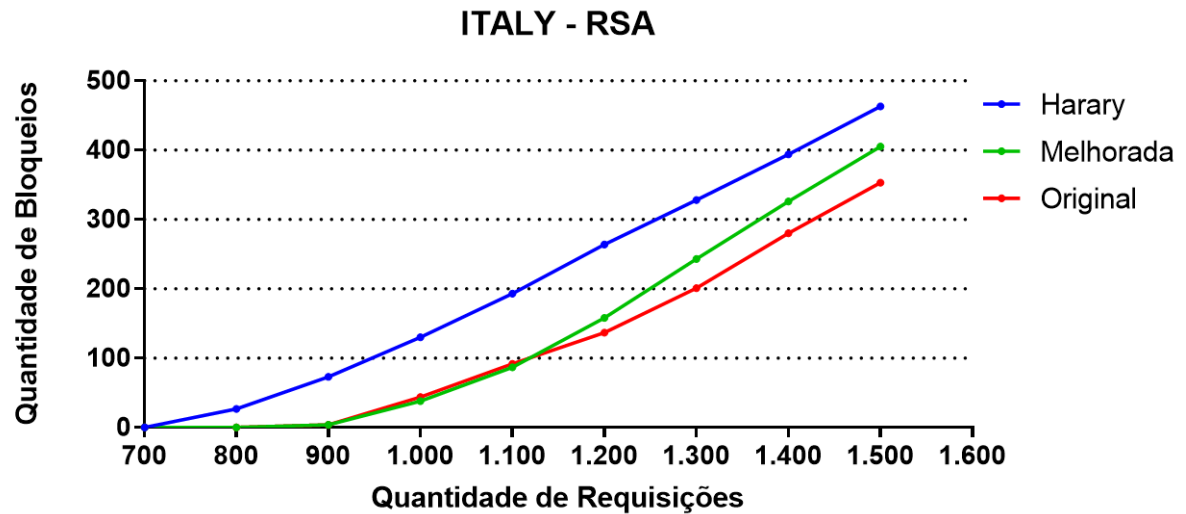


Figura 6.2: Quantidade de bloqueios para os Mapeamentos da rede ITALY

Tabela 6.2: Resultados - Problema de RSA.

ITALY				
Mapeamento	N de Requisições	Taxa de Bloqueio	Fragmentação	Nó Central
Original	1.500	0,30	0,96	Florence
Melhorado	1.500	0,27	0,94	Naples
Harary	1.500	0,23	0,89	Naples

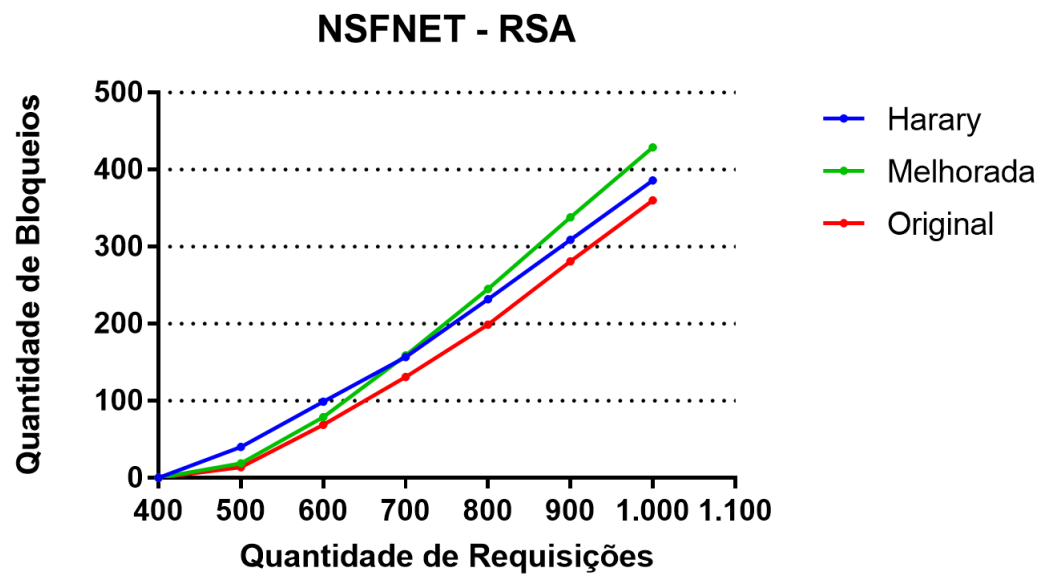


Figura 6.3: Quantidade de bloqueios para os Mapeamentos da rede NSFNET

Tabela 6.3: Resultados - Problema de RSA.

NSFNET				
Mapeamento	N de Requisições	Taxa de Bloqueio	Fragmentação	Nó Central
Original	1.000	0,38	0,95	Boulder
Melhorado	1.000	0,42	0,90	Pittsburgh
Harary	1.000	0,36	0,92	Pittsburgh

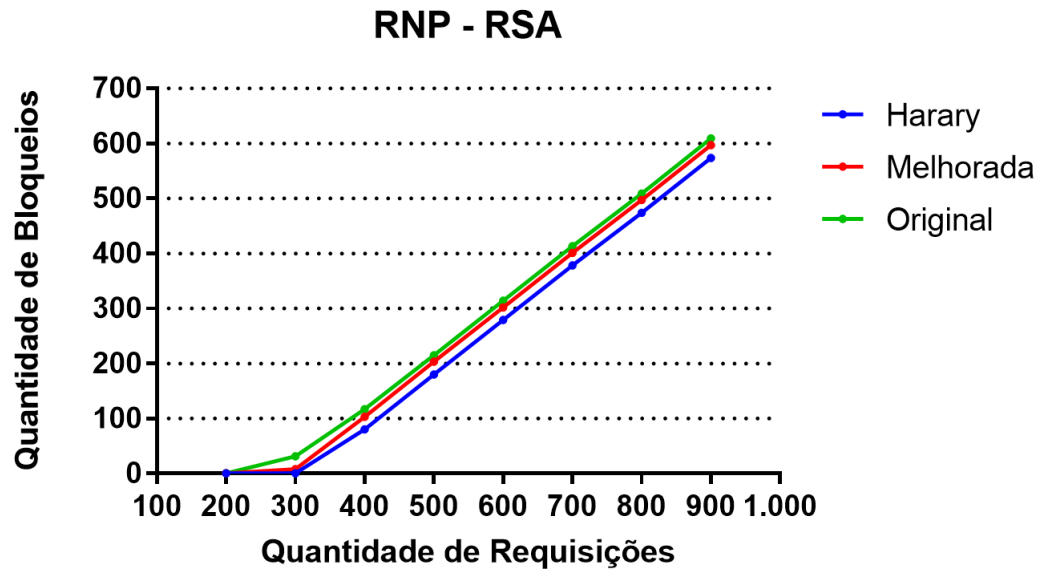


Figura 6.4: Quantidade de bloqueios para os Mapeamentos da rede RNP

Tabela 6.4: Resultados - Problema de RSA.

RNP				
Mapeamento	N de Requisições	Taxa de Bloqueio	Fragmentação	Nó Central
Original	900	0,63	0,78	Rio De Janeiro
Melhorado	900	0,66	0,95	Rio De Janeiro
Harary	900	0,67	0,93	Rio De Janeiro

Após os testes é possível identificar que nem sempre ao minimizar uma rede vai aumentar a sua eficiência, podemos observar que nas redes *CESNET* e *RNP* o mapeamento minimizado da Topologia Harary teve um melhor desempenho ao comparar a taxa de bloqueio e fragmentação, nas redes *NSFNET* e *ITALY* os mapeamentos originais das topologias demonstraram mais eficiência para os testes aplicados.

Com os resultados obtidos é possível afirmar que conforme o mapeamento da rede, a eficiência espectral e o desempenho da rede também alteram.

6.3 Teste de Performance das Mutações

O algoritmo original utiliza as 3 mutações para chegar ao resultado, nesse teste o algoritmo irá usar as mutações individualmente, verificando o impacto causado. Ao todo foram executadas 10 vezes para cada rede e para cada tipo de mutação para 100 ciclos, o resultado final é dado pela média dos resultados. Para o teste de performance das Mutações o processo de Renovação da População não foi utilizado, os indivíduos são todos gerados de forma aleatória e a população não se renova.

6.3.1 Resultados

A configuração utilizada para os testes é apresentada na tabela 6.5 e os resultados foram acompanhados em intervalos de 10 ciclos. Os resultados obtidos para cada mutação conforme a rede pode ser observado na figura 6.5.

Tabela 6.5: Configurações Utilizadas.

Configuração	
Parâmetro	Valor
<i>TP</i>	80
<i>PS</i>	2
<i>PC</i>	70%
<i>PM</i>	70%
<i>PSDM</i>	0

Os testes de Performance das mutações mostraram que a mutação de troca possui mais assertividade que as demais mutações testadas, onde das 40 execuções realizadas ela chegou a 34 vezes ao resultado perfeito, sendo 9 vezes para o teste com a rede CESNET, 9 vezes com a rede ITALY, 6 vezes com a rede NSFNET e 10 vezes com a rede RNP. A segunda melhor mutação identificada para o nosso caso foi a mutação de inversão, chegando ao resultado perfeito 6 vezes para o teste com a rede CESNET, 8 vezes com a rede ITALY, 6 vezes com a rede NSFNET e 10 vezes com a rede RNP. A mutação de inserção foi a rede que teve o pior desempenho onde chegou ao resultado perfeito 6 vezes para o teste com a rede CESNET, 5 vezes com a rede ITALY, 2 vezes com a rede NSFNET e 9 vezes com a rede RNP.

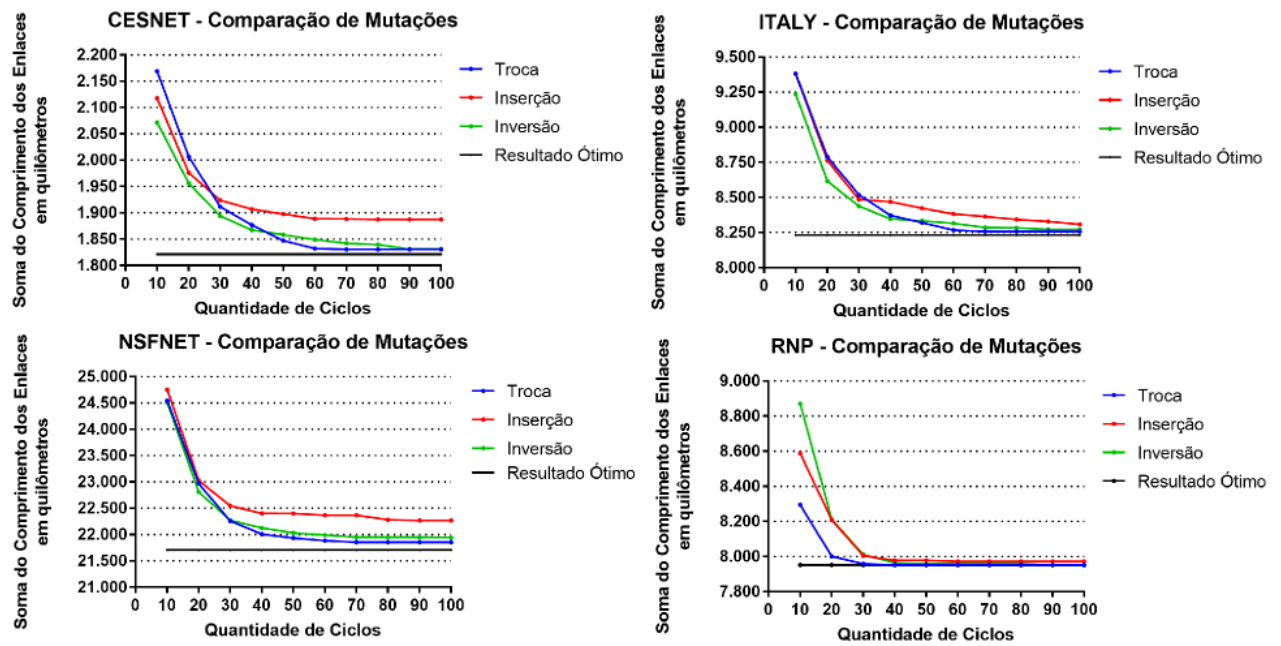


Figura 6.5: Resultado das mutações aplicadas as rede Originais CESNET, ITALY, NSFNET e RNP

A partir dos resultados podemos afirmar que a mutação de Troca teve 85% de exatidão, a mutação de Inversão teve 75% de exatidão e a mutação de Inserção teve 55% de exatidão.