



Heurísticas para o problema do desenho de redes com estrutura de múltiplos anéis e proveitos

Filipa Catarina da Silva Daniel

Mestrado em Estatística e Investigação Operacional

Especialização em Investigação Operacional

Dissertação orientada por:
Prof.^a Doutora Ana Maria Duarte Silva Alves Paias

Em primeiro lugar quero agradecer à professora Ana Paias, que me orientou nesta dissertação com uma enorme entrega e dedicação. Graças ao seu apoio, acompanhamento e disponibilidade permitiram que me sentisse mais confiante, motivada ao longo de todo o trabalho desenvolvido.

Em segundo lugar, quero agradecer aos meus pilares, pais e irmão, que me fizeram ser o que sou hoje e por me apoiarem sempre em todos os desafios. Assim como, à pessoa que acredita em mim, que me ajuda e motiva todos os dias, o meu namorado. Sem nunca esquecer, o enorme e eterno obrigada à minha avó Alice.

Por fim, mas não menos importante, quero agradecer por toda a companhia e amizade do meu amigo Diogo, por todas as horas de estudo e motivação ao longo destes anos.

Resumo

Esta dissertação procura resolver o problema do desenho de redes com estrutura de múltiplos anéis e proveitos, através de métodos heurísticos.

O problema em estudo consiste no desenho de uma estrutura em anel, pelas vantagens que esta apresenta, ao garantir uma maior qualidade de serviço aos clientes que nela se encontram. O presente problema surge no contexto de serviços de telecomunicações, no qual existe uma empresa que pretende servir um conjunto de clientes, que por sua vez é dividido em dois subconjuntos de acordo com a prioridade que lhes é atribuída: alta prioridade e baixa prioridade. Esta característica determina a obrigatoriedade ou não da presença de cada cliente no desenho final da estrutura que se procura.

Neste problema, o desenho da rede considera múltiplos anéis, definidos previamente, unidos por apenas um vértice central, que como referido, dado o contexto do problema é a empresa prestadora do serviço.

O objetivo na resolução deste problema é a maximização do lucro, que é determinado pela diferença entre os custos associados à estrutura e os proveitos que esta pode gerar. Cada ligação possível entre a empresa e os clientes, e entre clientes tem um custo e um proveito associado. Os custos associados ao desenho da estrutura são determinados de acordo com as ligações necessárias e presentes na solução obtida. Os proveitos gerados por essa solução, estão diretamente associados à garantia do serviço oferecida pela estrutura em anel. Assim, para cada par de clientes que partilham o mesmo anel é gerado um proveito ainda que não apresentem uma ligação direta no desenho da solução.

Por se tratar de um problema resultante da combinação de uma variante do problema do desenho de rotas de veículos (*vehicle routing*) e do problema da determinação da clique de peso máximo (*maximum edge-weighted clique*), ambos são considerados NP-difícil [3]. Assim, o problema do desenho de redes com estrutura de múltiplos anéis e proveitos é considerada também, um NP-difícil.

Para a resolução do problema em estudo desenvolveram-se duas heurísticas construtivas, uma de melhoramento e uma meta-heurística, pesquisa local iterada.

Para comparar os métodos utilizados nesta dissertação, utilizaram-se instâncias geradas a partir de instâncias benchmark para o TSP, com todos os parâmetros necessários. Ao fazer o estudo computacional às heurísticas propostas usando essas instâncias, obtiveram-se resultados que demonstram a diferença na complexidade de cada uma, e ainda em relação à dimensão das diferentes instâncias utilizadas. Uma vez mais, a meta-heurística escolhida conseguiu melhores soluções quando comparadas com as heurísticas construtivas e de melhoramento.

Palavras-Chave: Desenho de redes, heurísticas, meta-heurística, pesquisa local iterada.

Abstract

This dissertation seeks to solve the problem of network design with multiple ring structure and revenues, using heuristic methods.

The problem under study is the design of a ring structure, due to the advantages it presents in guaranteeing a higher quality of service to the customers who are in it. This problem arises in a context of telecommunications services in which there is a company that wants to serve a set of customers. These costumers are divided into two subsets according to the priority assigned to them: high priority and low priority. This characteristic determines whether the presence of each client in the final design of the structure is mandatory.

The network design in this problem consists of multiple rings, previously defined, joined by only one central vertex, which, as mentioned, given the context of the problem is the company providing the service.

The objective in solving this problem is to maximize profit, which is determined by the difference between the costs associated with the structure and the revenue it can generate. Each possible link between the company and the customers, and between customers has an associated cost and revenue. The costs associated with the design of the structure are determined according to the necessary connections present in the solution obtained. The revenue generated by this solution is directly associated with the service guarantee offered by the ring structure. Thus, for each pair of clients sharing the same ring a revenue is generated even if they are not connected by a direct link in the solution.

This problem results from the combination of a variant of the vehicle routing problem and the problem of determining the maximum edge-weighted clique, both NP-hard [3]. Thus, the problem of network design with multiple ring and revenues is also considered to be a difficult NP.

To solve the problem under study, two constructive heuristics, an improvement heuristic and a meta-heuristic, iterated local search, were developed.

To compare the methods used in this dissertation, instances generated from benchmark instances for the TSP were used, with all the necessary parameters.

When doing the computational study of the proposed heuristics using these instances, results were obtained that demonstrate the difference in the complexity of each one, and also in relation to the dimension of the different instances used. Once again, the chosen meta-heuristic achieved better solutions when compared to constructive and improvement heuristics.

Keywords: Network design, heuristics, meta-heuristics, iterated local search.

Índice

Resumo.....	iii
Abstract	v
Índice.....	vii
Lista de Figuras	ix
Lista de Tabelas.....	xi
Lista de Algoritmos	xv
Lista de abreviaturas.....	xvii
1. Introdução.....	1
2. Problema do desenho de redes com estrutura de múltiplos anéis e proveitos	5
2.1. Exemplo Ilustrativo	6
2.2. Revisão Bibliográfica	8
3. Heurísticas para o SmRNDP	11
3.1. Heurísticas Construtivas.....	11
3.1.1. Heurística Construtiva de inserção ponderada	11
3.1.2. Heurística Construtiva de inserção aleatória	16
3.2. Heurística de Melhoramento	19
3.2.1. Fase 1 – Intra-anel	19
3.2.2. Fase 2 - Inter-anel.....	21
3.2.3. Fase 3 - Troca ou Adição de facultativos	23
3.3. Meta-heurística.....	29
3.3.1. Pesquisa Local iterada	29
3.3.1.1. Perturbação.....	29
3.3.1.2. Pesquisa Local	30
4. Experiência Computacional	33
4.1. Instâncias de Teste.....	33
4.2. Heurísticas Construtivas.....	38
4.3. Heurística de melhoramento.....	41
4.4. Pesquisa Local iterada.....	45
4.4.1. Teste ao número de Iterações	48
5. Conclusões	57
5.1. Trabalho Futuro	58
6. Referências	59
Anexos.....	61
A. Resultados das cinco <i>runs</i> da Heurística construtiva ponderada.....	61

B.	Resultados das cinco <i>runs</i> da Heurística de melhoramento	63
C.	Resultados das cinco <i>runs</i> da Meta-heurística com 50 iterações.....	66
D.	Resultados das cinco <i>runs</i> da Meta-heurística com 200 iterações.....	69
E.	Resultados das cinco <i>runs</i> da Meta-heurística com 500 iterações.....	72
F.	Resultados das cinco <i>runs</i> da Meta-heurística com 1000 iterações.....	75
G.	Resultados das cinco <i>runs</i> da Meta-heurística com 2000 iterações.....	78
H.	Resultados das cinco <i>runs</i> da Meta-heurística com 3000 iterações.....	80

Lista de Figuras

Figura 1.1 - Exemplo ilustrativo de uma estrutura em anel	1
Figura 2.2.1 - Exemplo ilustrativo do problema SmRNDP.....	7
Figura 3.1.1.1. – Exemplo ilustrativo da estrutura inicial da Heurística Construtiva de Inserção Ponderada.....	12
Figura 3.1.1.2. – Exemplo ilustrativo da junção de anéis da estrutura inicial da Heurística Construtiva de Inserção Ponderada.....	13
Figura 3.1.1.3. – Exemplo ilustrativo na escolha dos clientes facultativos na estrutura da Heurística Construtiva de Inserção Ponderada	14
Figura 3.1.1.4 – Exemplo ilustrativo das possíveis inserções de clientes facultativos na estrutura da Heurística Construtiva de inserção ponderada	14
Figura 3.1.2.1 – Exemplo ilustrativo das escolhas de clientes facultativos na estrutura da Heurística Construtiva de Inserção Aleatória	16
Figura 3.1.2.2. – Exemplo ilustrativo das possíveis inserções de clientes facultativos na estrutura da Heurística Construtiva de Inserção Aleatória.....	17
Figura 3.2.1.1 – Exemplo ilustrativo da solução corrente.....	20
Figura 3.2.1.2 – Exemplo ilustrativo de uma possível solução obtida depois de aplicado algoritmo intra-anel.....	20
Figura 3.2.2.1 – Exemplo ilustrativo da solução corrente a considerar.....	21
Figura 3.2.2.2 – Exemplo ilustrativo de uma possível solução obtida depois de aplicado o inter-anel	22
Figura 3.2.3.1. – Exemplo ilustrativo das possíveis escolhas de clientes facultativos na estrutura da Heurística de melhoramento de troca de facultativos.....	24
Figura 3.2.4.1. – Exemplo ilustrativo das possíveis escolhas de clientes facultativos na estrutura da Heurística de melhoramento de adição de facultativos	26
Figura 4.2.1. – Exemplo ilustrativo de um dos resultados obtidos pela instância de 20 nodos.....	40
Figura 4.3.1. – Gráfico Comparativo de valores médios de gap obtidos na heurística de melhoramento com soluções iniciais das heurísticas construtivas	43
Figura 4.3.2. – Gráfico Comparativo de valores mínimos de gap obtidos na heurística de melhoramento com soluções iniciais das heurísticas construtivas	44
Figura 4.4.1. – Gráfico Comparativo de valores médios de gap obtidos na heurística de melhoramento e na meta-heurística, com referência inicial na construtiva de inserção aleatória.....	47
Figura 4.4.2. – Gráfico com os tempos médios de CPU obtidos na meta-heurística e apresentados na tabela 4.4.2., por cada instância considerada.....	48
Figura 4.4.1.1. - Gráfico Comparativo de valores médios de gap obtidos pela aplicação de diferentes iterações na meta-heurística, com referência inicial na construtiva de inserção aleatória para a instância de 20 nodos.....	52
Figura 4.4.1.2. - Gráfico Comparativo de valores médios de gap obtidos pela aplicação de diferentes iterações na meta-heurística, com referência inicial na construtiva de inserção aleatória para a instância de 30 nodos.....	52
Figura 4.4.1.3. - Gráfico Comparativo de valores médios de gap obtidos pela aplicação de diferentes iterações na meta-heurística, com referência inicial na construtiva de inserção aleatória para a instância de 50 nodos.....	52
Figura 4.4.1.4. - Gráfico Comparativo de valores médios de gap obtidos pela aplicação de diferentes iterações na meta-heurística, com referência inicial na construtiva de inserção aleatória para a instância de 70 nodos.....	53
Figura 4.4.1.5. - Gráfico Comparativo de valores médios de gap obtidos pela aplicação de diferentes iterações na meta-heurística, com referência inicial na construtiva de inserção aleatória para a instância de 100 nodos.....	53

Figura 4.4.1.7. - Gráfico Comparativo de valores médios de CPU obtidos pela aplicação de diferentes iterações na meta-heurística, com referência inicial na construtiva de inserção aleatória para todas as instâncias	55
Figura 4.4.1.6. - Gráfico Comparativo de valores médios de gap obtidos pela aplicação de diferentes iterações na meta-heurística, com referência inicial na construtiva de inserção aleatória para as instâncias de 136, 152, 202 e 164 nodos	55

Lista de Tabelas

Tabela 4.2.1 – Resultados da Heurística construtiva de inserção ponderada	39
Tabela 4.2.2 – Resultados da Heurística construtiva de inserção aleatória	39
Tabela 4.3.1. – Valores de Referência para o cálculo dos gaps	41
Tabela 4.3.2.- Resultados da Heurística de melhoramento com soluções iniciais da heurística construtiva de inserção ponderada	42
Tabela 4.3.3. - Resultados da Heurística de melhoramento com soluções iniciais de Heurística construtiva de inserção aleatória	42
Tabela 4.3.4. - Resultados em média da Heurística de melhoramento com soluções iniciais em cada uma das Heurísticas construtivas, para todas as instâncias, à exceção da instância de 20 nodos	44
Tabela 4.4.1 - Resultados da Meta-heurística, correspondentes à heurística construtiva de inserção ponderada, com 50 iterações	45
Tabela 4.4.2 - Resultados da Meta-heurística, correspondentes à Heurística construtiva de inserção aleatória, com 50 iterações	46
Tabela 4.4.1.1. - Resultados da Meta-heurística, instâncias de 20, 30, 50, 70 e 100 nodos correspondentes à heurística construtiva de inserção ponderada, com 200 iterações	49
Tabela 4.4.1.2 - Resultados da Meta-heurística, instâncias de 20, 30, 50, 70 e 100 nodos correspondentes à heurística construtiva de inserção aleatória, com 200 iterações	49
Tabela 4.4.1.3. - Resultados da Meta-heurística, instâncias de 20, 30, 50, 70 e 100 nodos correspondentes à heurística construtiva de inserção ponderada, com 500 iterações	49
Tabela 4.4.1.4 - Resultados da Meta-heurística, instâncias de 20, 30, 50, 70 e 100 nodos correspondentes à heurística construtiva de inserção aleatória, com 500 iterações	50
Tabela 4.4.1.5. - Resultados da Meta-heurística, instâncias de 20, 30, 50, 70 e 100 nodos correspondentes à heurística construtiva de inserção ponderada, com 1000 iterações	50
Tabela 4.4.1.6 - Resultados da Meta-heurística, instâncias de 20, 30, 50, 70 e 100 nodos à heurística construtiva de inserção aleatória, com 1000 iterações	50
Tabela 4.4.1.7. - Resultados da Meta-heurística, instâncias de 20, 30, 50, 70 e 100 nodos correspondentes à heurística construtiva de inserção ponderada, com 2000 iterações	50
Tabela 4.4.1.8 - Resultados da Meta-heurística, instâncias de 20, 30, 50, 70 e 100 nodos correspondentes à heurística construtiva de inserção aleatória, com 2000 iterações	51
Tabela 4.4.1.9. - Resultados da Meta-heurística, instâncias de 20, 30, 50, 70 e 100 nodos correspondentes à heurística construtiva de inserção ponderada, com 3000 iterações	51
Tabela 4.4.1.10 - Resultados da Meta-heurística, instâncias de 20, 30, 50, 70 e 100 nodos correspondentes à heurística construtiva de inserção aleatória, com 3000 iterações	51
Tabela 4.4.1.11 - Resultados da Meta-heurística, instâncias de 136, 152, 202 e 264 nodos correspondentes à heurística construtiva de inserção aleatória, com 200 iterações	54
Tabela 4.4.1.12 - Resultados da Meta-heurística, instâncias de 136, 152, 202 e 264 nodos correspondentes à heurística construtiva de inserção aleatória, com 500 iterações	54
Tabela 4.4.1.13 - Resultados da Meta-heurística, instâncias de 136, 152, 202 e 264 nodos correspondentes à heurística construtiva de inserção aleatória, com 1000 iterações	54
A.1. - Resultados das Heurísticas Construtivas para a instância de 20 nodos.....	61
A.2. - Resultados das Heurísticas Construtivas para a instância de 30 nodos.....	61

A.3. - Resultados das Heurísticas Construtivas para a instância de 50 nodos.....	61
A.4. - Resultados das Heurísticas Construtivas para a instância de 70 nodos.....	62
A.5. - Resultados das Heurísticas Construtivas para a instância de 100 nodos.....	62
A.6. - Resultados das Heurísticas Construtivas para a instância de 136 nodos.....	62
A.7. - Resultados das Heurísticas Construtivas para a instância de 152 nodos.....	62
A.8. - Resultados das Heurísticas Construtivas para a instância de 202 nodos.....	63
A.9. - Resultados das Heurísticas Construtivas para a instância de 264 nodos.....	63
B.1. - Resultados da Heurística de melhoria, onde as soluções iniciais das heurísticas construtivas para a instância de 20 nodos.....	63
B.2. - Resultados da Heurística de melhoria, onde as soluções iniciais das heurísticas construtivas para a instância de 30 nodos.....	64
B.3. - Resultados da Heurística de melhoria, onde as soluções iniciais das heurísticas construtivas para a instância de 50 nodos.....	64
B.4. Resultados da Heurística de melhoria, onde as soluções iniciais das heurísticas construtivas para a instância de 70 nodos.....	64
B.5. - Resultados da Heurística de melhoria, onde as soluções iniciais das heurísticas construtivas para a instância de 100 nodos.....	65
B.6. - Resultados da Heurística de melhoria, onde as soluções iniciais das heurísticas construtivas para a instância de 136 nodos.....	65
B.7. - Resultados da Heurística de melhoria, onde as soluções iniciais das heurísticas construtivas para a instância de 152 nodos.....	65
B.8. - Resultados da Heurística de melhoria, onde as soluções iniciais das heurísticas construtivas para a instância de 202 nodos.....	66
B.9. Resultados da Heurística de melhoria, onde as soluções iniciais das heurísticas construtivas para a instância de 264 nodos.....	66
C.1. Resultados da Meta-Heurística, onde as soluções iniciais das heurísticas construtivas para a instância de 20 nodos, com 50 iterações.....	66
C.2. Resultados da Meta-Heurística, onde as soluções iniciais das heurísticas construtivas para a instância de 30 nodos, com 50 iterações.....	67
C.3. Resultados da Meta-Heurística, onde as soluções iniciais das heurísticas construtivas para a instância de 50 nodos, com 50 iterações.....	67
C.4. Resultados da Meta-Heurística, onde as soluções iniciais das heurísticas construtivas para a instância de 70 nodos, com 50 iterações.....	67
C.5. Resultados da Meta-Heurística, onde as soluções iniciais das heurísticas construtivas para a instância de 100 nodos, com 50 iterações.....	68
C.6. Resultados da Meta-Heurística, onde as soluções iniciais das heurísticas construtivas para a instância de 136 nodos, com 50 iterações.....	68
C.7. Resultados da Meta-Heurística, onde as soluções iniciais das heurísticas construtivas para a instância de 152 nodos, com 50 iterações.....	68
C.8. Resultados da Meta-Heurística, onde as soluções iniciais das heurísticas construtivas para a instância de 202 nodos, com 50 iterações.....	69

F.3. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 50 nodos, com 1000 iterações	76
F.4. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 70 nodos, com 1000 iterações	76
F.5. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 100 nodos, com 1000 iterações	77
F.6. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 136 nodos, com 1000 iterações	77
F.7. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 152 nodos, com 1000 iterações	77
F.8. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 202 nodos, com 1000 iterações	78
F.9. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 264 nodos, com 1000 iterações	78
G.1. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 20 nodos, com 2000 iterações	78
G.2. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 30 nodos, com 2000 iterações	79
G.3. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 50 nodos, com 2000 iterações	79
G.4. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 70 nodos, com 2000 iterações	79
G.5. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 100 nodos, com 2000 iterações	80
H.1. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 20 nodos, com 3000 iterações	80
H.2. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 30 nodos, com 3000 iterações	80
H.3. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 50 nodos, com 3000 iterações	81
H.4. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 70 nodos, com 3000 iterações	81
H.5. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 100 nodos, com 3000 iterações	81

Lista de Algoritmos

Algoritmo 3.1.1. Pseudo-código Heurística Construtiva de Inserção Ponderada.....	15
Algoritmo 3.1.2. Pseudo-código Heurística Construtiva de Inserção Aleatória.....	18
Algoritmo 3.2. Pseudo-código Heurística de melhoramento.....	27
Algoritmo 3.3. Pseudo-código Meta-Heurística Pesquisa Local Iterada, ILS.....	31

Lista de abreviaturas

SmRNDP - *Steiner multi-ring network design problem with revenues*

SRNDP - *Steiner ring network design problem with revenues*

ILS - *Iterated Local Search*

CmRSP - *Capacitated m-Ring-Star Problem*

VNS - *Variable Neighborhood Search*

MDRSP - *Multi-Depot Ring-Star Problem*

m – número de anéis a considerar na estrutura em rede

b – número máximo de clientes em cada anel

n - número total de clientes

β – proporção de clientes obrigatórios (em relação ao total de clientes, n)

R – conjunto de clientes obrigatórios;

S – conjunto de clientes facultativos;

α – proporção de clientes por anel (limite superior)

c_{ij} – custo associado à ligação entre todos os pares de clientes e entre os clientes e a empresa

p_{ij} – proveitos associados a todos os pares de clientes e entre os clientes e a empresa

1. Introdução

No mundo em que vivemos, cada vez mais tecnológico e com uma exigência de serviço elevada, as empresas de telecomunicações têm a preocupação de procurar soluções e estratégias que garantam a sua qualidade na resposta aos seus serviços.

As redes de telecomunicações, principalmente quando a comunicação é feita por fibra ótica, optam, com frequência, por estruturas em anel.

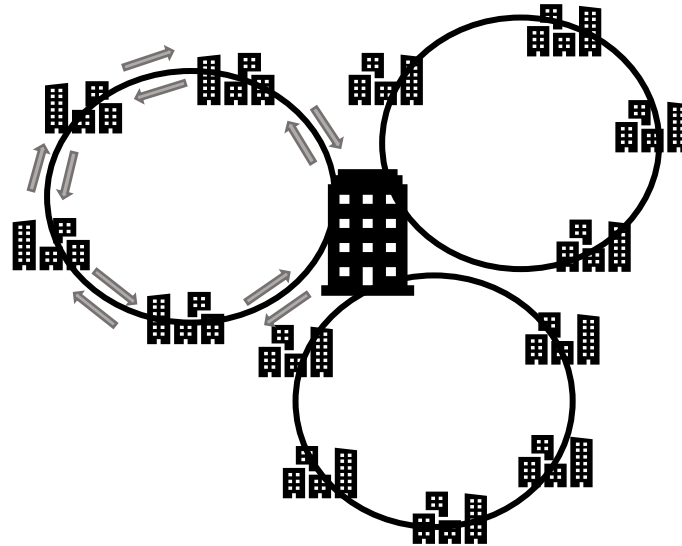


Figura 1.1 - Exemplo ilustrativo de uma estrutura em anel

Estas estruturas representam um círculo de clientes que irão receber/enviar informação para qualquer um dos sentidos desses mesmos círculos (horário e anti-horário), o que permitirá criar duas ligações para cada cliente. A grande vantagem e benefício na escolha destas estruturas passa pela garantia do serviço caso ocorra algum problema numa ligação ou até mesmo num cliente, pois tem sempre uma alternativa que não compromete o serviço.

Para além da garantia do serviço, é ainda necessário corresponder e satisfazer a procura e rapidez que é exigida e esperada pelo cliente. Desta forma, será importante que cada anel tenha um limite de clientes que este pode conter, não só pelo nível de eficácia, mas também pelo número de fibras que cada cabo pode ter.

A construção desta estrutura tem sempre um custo associado, nas ligações necessárias à estrutura que se pretende. Esses custos dependem de diversos fatores, sendo um deles a distância a que os clientes estejam uns dos outros, assim como a distância destes à empresa prestadora. Por isso, é importante desenvolver soluções que permitam diminuir os custos associados às implementações das mesmas.

Além dos custos mencionados, a empresa terá retorno no proveito que cada anel gerar. Isto é, como a comunicação e informação que chega a cada cliente passa também pelos restantes presentes nessa estrutura, os outros clientes que se encontram no mesmo anel, são vistos como garantias de serviço e por isso a empresa gera um proveito por cliente, pelo uso dessas mesmas garantias.

Cada cliente terá ainda uma prioridade associada quanto à necessidade de o incluir na estrutura, assim, os clientes serão divididos em dois grupos de acordo com a sua prioridade. O primeiro grupo será composto pelos clientes que terão de estar obrigatoriamente presentes nos anéis e o segundo grupo será constituído pelos clientes que poderão ser incluídos nos anéis, mas que a sua presença será facultativa.

O desafio ao abordar este problema de desenho de redes com estrutura de múltiplos anéis, passa por encontrar o equilíbrio entre o custo de serviço e o proveito que a empresa prestadora possa alcançar, de forma a respeitar todas as restrições e limitações que estejam associadas. Por esta razão, é importante explorar técnicas, métodos e estratégias na área de Investigação Operacional, que muito tem vindo a contribuir e a evoluir nas mais diversas áreas.

O problema descrito anteriormente foi apresentado na literatura [3], como *Steiner multi-ring network design problem with revenues* (SmRNDP).

Com o objetivo de contextualizar o problema, considere-se o seguinte exemplo: uma empresa de telecomunicações pretende desenhar a sua rede usando uma estrutura de múltiplos anéis para servir um conjunto de clientes. Para o desenho dessa estrutura será necessário ter em conta:

- O único ponto de ligação entre os anéis é a empresa prestadora de serviço, pelo que cada anel terá início e fim na empresa mencionada;
- O número de anéis presentes no desenho da estrutura;
- Os clientes e a sua prioridade: alta prioridade e baixa prioridade, ou seja, clientes obrigatórios e clientes facultativos, respetivamente;
- Os custos associados à construção de ligação entre todos os pares de clientes, assim como entre os clientes e a empresa;
- Os proveitos entre todos os pares de clientes, assim como entre os clientes e a empresa;
- O número máximo de clientes presentes em cada anel, de forma a obter a qualidade de serviço.

Depois de conhecidos os dados e restrições do problema, procede-se ao desenho da rede, isto é, à distribuição dos clientes pelos anéis, de forma a maximizar o valor do lucro total.

O presente problema pode ser visto como uma combinação de uma variante do problema do desenho de rotas de veículos (*vehicle routing*) e do problema da determinação da clique de peso máximo (*maximum edge-weighted clique*). Estes dois problemas são considerados NP-difícil [3], pelo que o problema em estudo SmRNDP é igualmente considerado um problema NP-difícil.

O SmRNDP pode ser formulado através de modelos em programação linear inteira mista e esses modelos podem ser resolvido com o auxílio de um solver comercial. Contudo em [3], devido à dimensão dos problemas resultantes apenas se conseguem resolver para pequenas instâncias. Assim, com o objetivo de resolver instâncias do problema com dimensão média a grande, decidiu-se nesta dissertação explorar um método de resolução heurístico, um método que permite obter soluções admissíveis para o problema.

O objetivo na elaboração de métodos de resolução desta dissertação passa pelo desenvolvimento, exploração e implementação de algumas heurísticas, construtivas e de melhoramento e ainda de uma meta-heurística, *Iterated Local Search* (ILS). As meta-heurísticas têm vindo a provar

que são métodos bastante inteligentes com bons resultados, que surpreendem pela sua aleatoriedade e desconstrução do conceito de soluções consideradas “menos viáveis” numa primeira instância.

Apesar do problema ter sido apresentado numa vertente de prestação de serviços de telecomunicações, esta estrutura ou topologia de rede pode ser encontrada noutros contextos, como a logística, roteamento de veículos, etc.

No próximo capítulo segue-se a apresentação formalizada do problema, diversas variantes e contextos presentes na literatura, assim como os seus métodos de resolução. Para uma melhor visualização do problema, é apresentado um pequeno exemplo ilustrativo, aplicado ao mesmo.

No capítulo 3, são apresentadas as heurísticas utilizadas para a resolução do SmRNDP: duas heurísticas construtivas, uma de melhoramento e ainda uma meta-heurística, pesquisa local iterada [8], conhecida na literatura como *Iterated Local Search* (ILS). Para cada uma delas, será feita uma explicação dos métodos utilizados, acompanhados por alguns exemplos da aplicação dos procedimentos e ainda um pseudo-código.

No capítulo 4, apresenta-se a experiência computacional desenvolvida nesta dissertação, onde se explica a geração das instâncias, as soluções obtidas pelas heurísticas testadas e implementadas em Matlab R2018b e respetivos comentários. Por fim, para a meta-heurística escolhida nesta dissertação, a pesquisa local iterada, ILS, será feito um estudo ao número de iterações a aplicar.

Termina-se a dissertação, com as conclusões retiradas ao desenvolver o presente estudo.

2. Problema do desenho de redes com estrutura de múltiplos anéis e proveitos

O corrente capítulo apresenta o problema do desenho de redes com estrutura de múltiplos anéis e proveitos conhecido na literatura como *Steiner multi-ring network design problem with revenues* e referido daqui em diante por SmRNDP.

Para uma melhor contextualização do trabalho a apresentar nesta dissertação, são referidos estudos feitos a problemas semelhantes, dando a conhecer os diferentes contextos dos mesmos, as suas variantes e alguns métodos de resolução desenvolvidos.

Para terminar este capítulo, é apresentado de um exemplo ilustrativo, de pequena escala para uma melhor visualização do problema a desenvolver.

A evolução da tecnologia na área das telecomunicações levou ao desenvolvimento de problemas de desenho de uma estrutura em rede, que tem vindo a ser estudada e aplicada nas mais diversas áreas, como é possível observar na literatura existente.

Os estudos e aplicações da estrutura em rede têm vindo a ser explorados para a área da tecnologia, como foi referido, mas também para a área de roteamento, com diversas variantes de acordo com o problema que se pretende resolver.

Nesta dissertação, o problema a desenvolver SmRNDP, foi apresentado pela primeira vez em [3], como uma generalização do problema do desenho de uma rede com topologia em anel com vértices Steiner, este último conhecido na literatura como *Steiner ring network design problem with revenues* (SRNDP), apresentado em [6].

Este problema consiste no desenho de uma estrutura em vários anéis que se irão interligar pela empresa (hub). No contexto apresentado, trata-se da distribuição dos clientes pelos anéis que se pretendem construir e que terão como um elo comum, a empresa prestadora de serviços.

Cada cliente, terá uma prioridade atribuída, pelo que se terá clientes obrigatórios (alta prioridade) e clientes facultativos (baixa prioridade). De forma a assegurar a qualidade necessária de serviço, existe um número máximo de clientes em cada anel.

Cada ligação entre dois clientes terá um custo associado, e entre cada par de clientes que partilhe um mesmo anel está associado a um proveito. Quando se inclui um cliente num anel, este irá usufruir da estrutura, ou seja, poderá comunicar com todos os clientes presentes nesse mesmo anel. Assim, a inclusão de um cliente, irá gerar um proveito total, entre o cliente incluído e todos os clientes já presentes no anel onde foi inserido.

Assim, os parâmetros a considerar para o SmRNDP são os seguintes:

- O número total de clientes (n);
- O número de anéis (m);
- O número de clientes obrigatórios (o conjunto destes clientes é representado por R e assume-se que os obrigatórios são os $|R|$ primeiros clientes do conjunto de todos os clientes);
- Capacidade máxima de cada anel (b);

- Matriz de custos C (c_{ij} – custo associado à ligação direta entre o cliente/empresa i e o cliente/empresa j)
- Matriz de proveitos P (p_{ij} – proveito associado a ter o cliente/empresa i no mesmo anel que o cliente/empresa j)

O problema em estudo pode ser representado num grafo não orientado $G=(V,A)$ em que os vértices estão associados aos clientes (vértices de 1 a n) e à empresa (vértice 0). Cada arco está associado à possível ligação entre clientes e entre clientes e a empresa, onde os custos associados às ligações estão presentes numa matriz simétrica de custos, C , e os proveitos gerados em cada anel têm por referência os valores presentes na matriz simétrica de proveitos P , de acordo com os clientes que partilham o mesmo anel e que estão ligados entre si, direta ou indiretamente.

O objetivo do problema será maximizar o lucro gerado pelo desenho da rede ou estrutura, ou seja, maximizar a diferença entre o proveito total e o custo total.

O SmRNDP é um problema NP-difícil, pois pode ser visto como uma combinação de uma variante do problema do desenho de rotas de veículos (*vehicle routing*) e do problema da determinação da clique de peso máximo (*maximum edge-weighted clique*).

2.1. Exemplo Ilustrativo

De seguida apresenta-se uma instância do problema do desenho de redes com estrutura de múltiplos anéis e proveitos com os seguintes parâmetros:

- 2 anéis ($m=2$);

- 6 clientes ($n=6$), dos quais os clientes 1 e 2 são obrigatórios (assinalados na figura seguinte, com um largo tracejado, $R=\{1,2\}$);

- cada anel não poderia ter mais do que 3 clientes ($b=3$);

- matriz dos custos de ligação $C = \begin{bmatrix} 0 & 9 & 2 & 9 & 1 & 10 & 9 \\ 9 & 0 & 10 & 1 & 9 & 10 & 3 \\ 2 & 10 & 0 & 3 & 5 & 2 & 2 \\ 9 & 1 & 3 & 0 & 7 & 6 & 4 \\ 1 & 9 & 5 & 7 & 0 & 8 & 4 \\ 10 & 10 & 2 & 6 & 8 & 0 & 3 \\ 9 & 3 & 2 & 4 & 4 & 3 & 0 \end{bmatrix}$, onde a primeira

linha e coluna corresponde à empresa prestadora, a segunda linha e coluna correspondem ao cliente 1 e assim sucessivamente;

- matriz de proveitos $P = \begin{bmatrix} 0 & 7 & 4 & 10 & 4 & 2 & 4 \\ 7 & 0 & 2 & 6 & 1 & 5 & 2 \\ 4 & 2 & 0 & 1 & 1 & 3 & 9 \\ 10 & 6 & 1 & 0 & 2 & 7 & 5 \\ 4 & 1 & 1 & 2 & 0 & 10 & 6 \\ 2 & 5 & 3 & 7 & 10 & 0 & 8 \\ 4 & 2 & 9 & 5 & 6 & 8 & 0 \end{bmatrix}$, onde a primeira linha e

coluna corresponde à empresa prestadora e as seguintes aos clientes;

Para facilitar a representação da empresa nas figuras apresentadas e nas expressões ao longo desta dissertação, esta é representada pelo vértice 0.

Uma solução admissível do desenho de redes com estrutura de múltiplos anéis e proveitos para esta instância é representada na figura seguinte na qual junto a cada arco se indicam o respetivo custo e proveito:

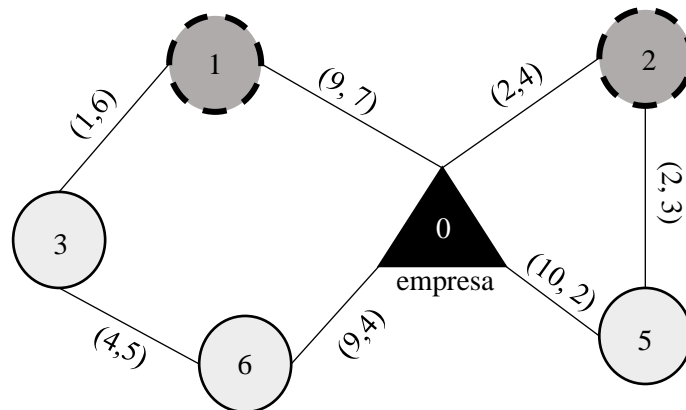


Figura 2.2.1 - Exemplo ilustrativo do problema SmRNDP

O cálculo dos custos é bastante intuitivo, dado que a sua base é a construção das ligações entre os clientes ou entre a empresa e os clientes os proveitos necessitam de uma compreensão e contextualização.

Assim, perante a estrutura apresentada na figura 2.2.1. os custos são calculados da seguinte forma:

$$\text{- Custo total do anel à esquerda} = c(0,1)+c(1,3)+c(3,6)+c(6,0) = 9 + 1 + 4 + 9 = 23$$

$$\text{- Custo total do anel à direita} = c(0, 2) + c(1, 5) + c(5, 0) = 2 + 2 + 10 = 14$$

O cálculo do lucro necessita de uma atenção especial quanto ao cálculo dos proveitos.

Quando algum cliente está presente num anel, este irá usufruir da estrutura onde irá ser inserido, que irá garantir a qualidade do serviço, de acordo com o contexto do problema como referido anteriormente. Assim, quando este se encontra num determinado anel, este está afeto aos restantes clientes nele presentes. Desta forma, os proveitos devem ser calculados dessa forma, sem que haja repetição. Para cada anel, considerem-se i e j distintos, que representam as posições que os nodos ocupam no seu anel (clientes ou empresa), os proveitos são calculados da seguinte forma:

$$\sum_i \sum_j p(i, j), \forall i, j \in \text{anel}, i < j \quad (2.2.1)$$

Considere-se novamente o exemplo apresenta na figura anterior, onde a sequência do anel mais à esquerda é dada por $0 - 1 - 3 - 6 - 0$ ou por $0 - 6 - 3 - 1 - 0$. Assim, de acordo com a equação 2.2.1. os proveitos do anel à esquerda são calculados da seguinte forma:

$$\text{- Proveito total do anel à esquerda} = p(0, 1) + p(0, 3) + p(0, 6) + p(1, 3) + p(1, 6) + p(3, 6) = 7 + 6 + 2 + 10 + 5 + 4 = 34$$

Os proveitos do anel à direita são calculados da seguinte forma:

$$\text{- Proveito total do anel à direita} = p(0, 2) + p(0, 5) + p(2, 5) = 4 + 2 + 3 = 9;$$

Desta forma o lucro será obtido pela diferença entre os proveitos e os custos:

$$\text{Lucro} = \text{proveito total dos anéis} - \text{custo total dos anéis} = (9 + 34) - (14 + 23) = 6$$

2.2. Revisão Bibliográfica

O problema em estudo nesta dissertação trata-se de uma variante do problema de desenho em redes. Na literatura é possível encontrar inúmeras variantes, contextos e métodos de resolução para cada uma delas:

Em [2], os autores (Baldacci et al.), apresentaram o problema *Capacitated m-Ring-Star Problem* (CmRSP), onde se pretende, num contexto de telecomunicações, desenhar uma estrutura em rede que inclui conjuntos de clientes disjuntos, denominados por anéis. Esses anéis serão interligados pela empresa, que será o único ponto em comum entre eles. Para os clientes que inicialmente não estejam incluídos nos anéis, são posteriormente ligados diretamente a um dos clientes presentes num dos anéis, o que permite uma ligação, ainda que indireta, à empresa. A qualidade do serviço é assegurada estabelecendo um número máximo de clientes em cada anel. Os autores formulam o problema em Programação linear inteira mista e resolvem o modelo resultante usando algoritmos de *branch-and-cut*.

Posteriormente em [9], os autores propõem para a resolução do CmRSP o uso de uma heurística construtiva ao aplicar algoritmo de *clustering*, seguido por um algoritmo de melhoria onde é aplicado a troca, remoção de nós Steiner e atribuição de extração, referido na literatura como *Swap, Steiner-Node-Removal and Extraction Assignment*. O algoritmo começa por aplicar as trocas, *Swap*, de forma iterada até que não seja possível mais melhorias. Segue-se a remoção de nós de Steiner, *Steiner-Node-Removal*, e por fim aplica-se a atribuição de extração, *Extraction Assignment*, que consiste na extração de clientes, por uma ordem aleatória e colocá-lo numa posição mais viável, através de iterações.

Em [10], para o problema CmRSP, é utilizada uma meta-heurística para a resolução do mesmo. Esta consiste na escolha de um conjunto de estruturas de vizinhança que variam em tamanho, com o apoio de um algoritmo iterativo com o objetivo de melhorar a solução, até que o critério de paragem seja alcançado.

Mais tarde, em 2009, Baldacci, apresentou uma nova variante do problema, *Multi-Depot Ring-Star Problem* (MDRSP) [1], que se diferencia principalmente pelo número de empresas que a estrutura pode ter e que pode ser vista como uma generalização do problema original CmRSP. Para a sua resolução recorre a técnicas heurísticas ao *utilizar re-optimization procedures, Squeeze and LS-multiple*.

Na literatura, é ainda possível ver variantes do problema que se diferenciam não só pelas suas características, mas também pelo próprio desenho da estrutura, como é o caso dos autores em [7], que em 2017, apresentam um problema, onde a ligação e colocação dos clientes marcam a diferença nesta nova abordagem. Métodos matemáticos exatos foram propostos para a sua resolução, com o apoio de *branch-and-cut algorithms*, anteriormente utilizado para outras variantes do problema. Para além disso, é também possível ver uma abordagem diferente ao problema SmRNDP, onde a inserção de um cliente nos anéis terá um impacto diferente pela forma como os proveitos são considerados e ainda por incluir penalidades quando estes não são escolhidos para um determinado anel. Nesta abordagem, foram exploradas algumas meta-heurísticas, tais como o *simulated annealing* e a *tabu search*.

Apesar do contexto da estrutura abordada ser na sua maioria para a área das telecomunicações, também é possível encontrá-la em problemas de roteamento veículos, quando se pretende desenhar e definir rotas para entregas de mercadoria. Esta vertente do problema é apresentada pelos autores de [4] ou ainda pelos autores de [12].

Todas as experiências, soluções, métodos e conclusões estudados até hoje são interessantes de analisar, para não só perceber o que foi feito até então e explorar métodos desenvolvidos que se possam melhorar, mas também para aprender com as diferentes abordagens às diversas variantes do problema.

O problema em estudo, SmRNDP foi originalmente apresentado Bautzer et al. [3] e surge como uma generalização do problema *Steiner Ring Network Design Problem* (SRNDP), apresentado pelos autores de [6]. Neste trabalho são apresentados modelos matemáticos em PLI mista e são estudados e propostos diferentes cortes que permitam melhorar a qualidade dos limites inferiores fornecidos pela respectiva relaxação linear. Os autores apresentam resultados com um máximo de 41 clientes. Para instâncias de maior dimensão os modelos apresentavam dimensões tais que não foi possível resolvê-los com o solver comercial Cplex.

Com o objetivo de explorar esta variante do problema procurou-se nesta dissertação explorar técnicas heurísticas que permitissem resolver o SmRNDP para qualquer instância, sem limitação na dimensão das mesmas.

3. Heurísticas para o SmRNDP

No corrente capítulo serão apresentadas várias heurísticas para a resolução do SmRNDP.

As heurísticas, são métodos aproximados, que possibilitam obter uma solução admissível para o problema em estudo. Geralmente este tipo de método é aplicado a problemas onde é difícil obter a solução ótima e tem por objetivo obter soluções de boa qualidade num tempo de computação razoável.

Começa-se por apresentar as duas heurísticas construtivas desenvolvidas, seguidas por uma heurística de melhoramento e por fim será apresentada a meta-heurística explorada nesta dissertação.

Apresentam-se também pequenos exemplos ilustrativos para ajudar a explicação e ainda um algoritmo em pseudo-código para cada uma delas.

3.1. Heurísticas Construtivas

Para a resolução deste problema começou-se por desenvolver duas heurísticas construtivas, com o objetivo de obter soluções admissíveis as quais irão servir como ponto de partida para as restantes heurísticas desenvolvidas.

O SmRNDP, consiste no desenho de uma rede em anéis disjuntos que contêm os clientes que serão servidos e que têm a empresa como única ligação entre si. Os clientes dividem-se em dois subgrupos: os clientes obrigatórios (alta prioridade) e os clientes facultativos (baixa prioridade).

De forma a garantir a presença de todos os clientes obrigatórios na solução, decidiu-se nas heurísticas construtivas começar por distribuir e inserir os clientes de alta prioridade nos anéis definidos para a estrutura e só depois, decidir a inclusão dos clientes com baixa prioridade. Ao longo da construção da solução é importante ainda ter em conta o limite máximo de clientes que cada anel poderá ter.

As heurísticas construtivas diferem no modo como inserem os clientes obrigatórios nos anéis: inserção ponderada e inserção aleatória. A inserção dos clientes facultativos em cada uma das heurísticas construtivas são feitas da mesma forma.

3.1.1. Heurística Construtiva de inserção ponderada

A heurística construtiva de inserção ponderada começa por inserir os clientes obrigatórios nos anéis, respeitando a restrição de capacidade de cada anel.

Para a inserção dos clientes obrigatórios desenvolveu-se um algoritmo inspirado na heurística de *Clark and Wright*. Esta heurística há muitos anos que é aplicada em diversos problemas, maioritariamente variantes do TSP, onde o desenho da rota, ou rotas, começa por dividir os clientes em diferentes anéis e vão sendo agrupadas com base na *poupança* que esse agrupamento pode gerar. Esta heurística é bastante versátil, por se conseguir facilmente integrar as restrições que o problema em estudo possa ter.

Começa-se por criar tantos anéis quanto o número de clientes obrigatórios, ou seja, com alta prioridade. Considere-se, por exemplo, uma instância do SmRNDP com 5 clientes obrigatórios. Este algoritmo irá criar 5 anéis interligados pela empresa em questão, com a seguinte estrutura:

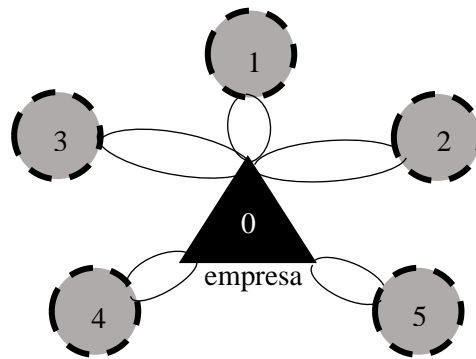


Figura 3.1.1.1. – Exemplo ilustrativo da estrutura inicial da Heurística Construtiva de Inserção Ponderada

Se o número de anéis que se deseja criar, corresponde ao número de clientes obrigatórios, calcula-se o lucro explicado ao detalhe no capítulo anterior e segue-se para a possível inclusão dos clientes facultativos. Caso contrário, o algoritmo irá ser repetido até que o número de anéis na estrutura criada, corresponda ao pretendido na solução. Para tal, calculam-se as conhecidas *poupanças* da heurística *Clark and Wright*, porém no presente problema estas não irão refletir os gastos da estrutura, mas sim, o impacto que a ligação direta de dois clientes de anéis disjuntos terá no lucro final, considerando os respetivos custos e proveitos. É ainda importante ter em atenção o cálculo dos proveitos, explicado e ilustrado no capítulo anterior.

Esta união de anéis só será possível, pela ligação direta de dois clientes que se encontrem em diferentes anéis e ligados ao vértice central representando a empresa. Ou seja, as *poupanças* devem ser calculadas apenas para clientes que se encontrem em anéis disjuntos e ligados ao vértice central.

Depois de analisadas todas as possíveis combinações e as respetivas *poupanças*, escolhe-se aquela que terá o melhor impacto no lucro final da solução, mesmo que este tenha um valor negativo.

De forma a garantir que a solução obtida seja admissível, é necessário ter ainda em conta o número máximo de clientes que cada anel poderá ter. Assim, a união de anéis anteriormente mencionada deve ser feita, apenas se o número de clientes dos dois anéis que se estão a considerar não seja superior ao limite máximo mencionado. Caso o total exceda o limite máximo admitido na solução, deverá escolher-se novamente a melhor *poupança*, de acordo com o objetivo de maximização do valor do lucro, até que a escolha não comprometa a admissibilidade da solução.

Este processo deverá repetir-se até que os anéis presentes na solução correspondam à quantidade que se pretende dos mesmos, ou até que não seja possível juntar mais anéis.

Considere-se o exemplo apresentado na figura 3.1.1.1. onde se pretende que a estrutura final tenha 2 anéis, em que cada um deles possa ter no máximo 3 clientes. O valor do lucro da solução apresentada na figura anterior, deverá ser calculado da seguinte forma:

$$L = \sum_i p(i, 0) - \sum_i c(i, 0) - \sum_i c(0, i)$$

De seguida, começa-se por calcular as *poupanças* da rede apresentada na figura 3.1.1.1. Recorde-se que $R = \{\text{Clientes Obrigatórios}\} = \{1, 2, 3, 4, 5\}$ e $poupança(i, j)$ as *poupanças* para $i, j \in R$, com $i \neq j$.

$$\begin{aligned}
poupan\c{c}a(i, j) = & p(i, j) - c(i, j) + c(i, 0) + c(0, j) + \sum_{k \in \text{anel de } i, k \neq i} p(k, j) + \\
& \sum_{k \in \text{anel de } j, k \neq j} p(k, i), \quad (3.1.1.1) \\
\forall i, j, \in R, i \neq j
\end{aligned}$$

Depois de calculados todos os valores da equa\c{c}o 3.1.1.1., escolhe-se o valor m\c{a}ximo de todos os obtidos (ainda que este seja negativo, deve ser o escolhido). Neste exemplo, considerar-se-\c{a} que o m\c{a}ximo corresponde \c{a} *poupan\c{c}a* (4,5), pelo que a solu\c{c}o corrente \c{e} atualizada como se mostra na figura seguinte e o valor do lucro correspondente na solu\c{c}o corrente \c{e} obtido pela soma da poupan\c{c}a escolhida, neste caso, a *poupan\c{c}a* (4,5), ao valor do lucro anterior.

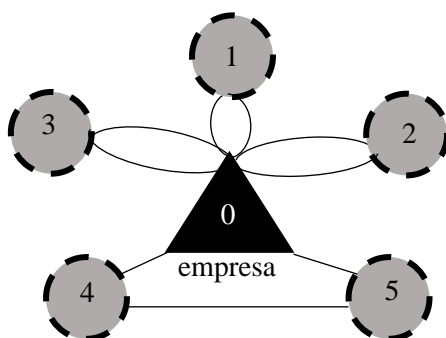


Figura 3.1.1.2. – Exemplo ilustrativo da jun\c{c}o de an\c{e}is da estrutura inicial da Heur\c{i}stica Construtiva de Inser\c{c}o Ponderada

Uma vez que a estrutura apresenta ainda 4 an\c{e}is (>2), continua o processo de jun\c{c}o dos an\c{e}is, com o aux\c{i}lio do c\c{a}lculo das *poupan\c{c}as*, apresentado na equa\c{c}o 3.1.1.1..

Ap\c{os} o c\c{a}lculo de todos as *poupan\c{c}as*, escolhe-se novamente o m\c{a}ximo obtido, que ser\c{a} adicionado ao valor do lucro anterior e atualiza-se a solu\c{c}o de acordo com esta decis\c{a}o.

Este processo deve ser continuado at\c{e} que a solu\c{c}o encontrada corresponda \c{a} estrutura que se pretende, ou seja com o n\c{u}mero de an\c{e}is pretendidos. Na continuidade do processo \c{e} importante ter em aten\c{c}o ao n\c{u}mero m\c{a}ximo de clientes que cada anel pode ter. Por exemplo se a estrutura possuir um anel com o limite m\c{a}ximo conhecido, de 3 clientes, n\c{a}o dever\c{a} ser adicionado mais nenhum, pelo que no c\c{a}lculo das *poupan\c{c}as* n\c{a}o se considera esse anel.

Depois de encontrada uma solu\c{c}o admiss\c{i}vel, onde todas as restri\c{c}oes s\c{a}o respeitadas e todos os clientes obrigat\c{or}ios est\c{a}o presentes segue-se a inclus\c{a}o dos clientes facultativos.

Os clientes facultativos, s\c{a}o os clientes com baixa prioridade, e que podem, ou n\c{a}o, estar presentes na solu\c{c}o, de acordo com o que seja mais vantajoso, tendo em conta o objetivo do problema, a maximiza\c{c}o do lucro.

De seguida descreve-se como \c{e} feita a avalia\c{c}o e poss\c{i}vel inser\c{c}o dos clientes facultativos. Come\c{c}a-se por escolher aleatoriamente um cliente facultativo (que n\c{a}o esteja ainda na solu\c{c}o) e procede-se \c{a} avalia\c{c}o da sua inclus\c{a}o na solu\c{c}o.

Esta avalia\c{c}o \c{e} feita atrav\c{e}s do c\c{a}lculo do impacto no lucro final obtido pela inser\c{c}o deste cliente em todas as posi\c{c}oes poss\c{i}veis nos an\c{e}is. Para al\c{e}m disso \c{e} importante, uma vez mais, ter em conta o n\c{u}mero m\c{a}ximo de clientes por anel, assim, esta avalia\c{c}o apenas ser\c{a} feita para an\c{e}is onde seja poss\c{i}vel aceitar mais clientes, ou seja, onde esse limite ainda n\c{a}o tenha sido atingido.

Cada cliente facultativo em estudo, pode ser inserido na melhor posição possível, desde que o respectivo impacto no lucro seja positivo. Caso não aconteça, segue-se para outro cliente, de prioridade baixa, ainda não avaliado.

Este procedimento deverá ser repetido até que todos os clientes facultativos tenham sido analisados ou até que já não seja possível inserir mais nenhum cliente, por todos os anéis terem atingido a sua capacidade máxima.

Voltando ao exemplo dado anteriormente, considere-se onde a seguinte rede, apresentada à esquerda, com os clientes obrigatórios inseridos na estrutura, e os dois clientes facultativos que poderão ser inseridos na rede, apresentados à direita.

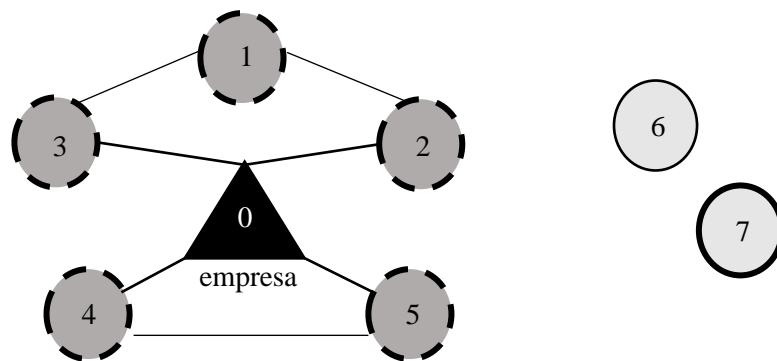


Figura 3.1.1.3. – Exemplo ilustrativo na escolha dos clientes facultativos na estrutura da Heurística Construtiva de Inserção Ponderada

Começa-se por escolher aleatoriamente um dos clientes facultativos, e para esse cliente calcula-se, para cada possibilidade de inserção na rede, o impacto no lucro. Como neste caso, um dos anéis já possui 3 clientes, que corresponde à capacidade máxima por anel definida, só se irá considerar o anel que tem 2 clientes. Considere-se o cliente 7, o primeiro cliente facultativo escolhido para análise. Este poderá tomar as seguintes posições:

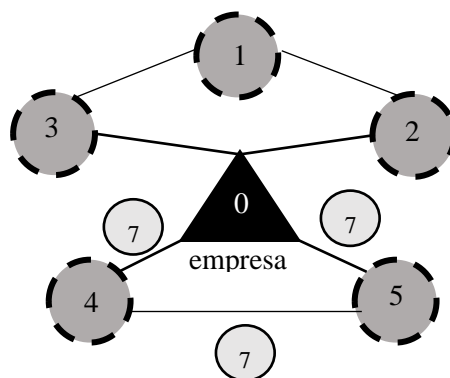


Figura 3.1.1.4 – Exemplo ilustrativo das possíveis inserções de clientes facultativos na estrutura da Heurística Construtiva de inserção ponderada

Para todas as possíveis posições que o cliente 7 pode tomar na estrutura, apresentadas na figura 3.1.1.4., devem ser calculadas as *poupanças*.

Para qualquer cliente a ser inserido na solução, s , tomará o lugar entre dois elementos, que podem ser entre dois clientes ou entre um cliente e a empresa, ou seja, entre o elemento um e o elemento dois, que se designou por e_1 e e_2 respetivamente. Assim, o cálculo deve ser feito da seguinte forma:

$$poupan\c{a}(e_i, e_j) = c(e_i, e_j) - c(e_i, s) - c(e_i, s) + p(s, 0) + \sum_{k \in \text{clientes no anel}} p(k, s) \quad (3.1.1.2)$$

$$\forall e_i, e_j, \in \{\text{Clientes}\} \cup \{0\}, e_i \neq e_j$$

Entre todos os valores obtidos escolhe-se o maior valor. Esse valor ir definir se o cliente facultativo deve ser inserido na respetiva posio, uma vez que se este tomar valores positivos este dever ser includo na estrutura, caso contrrio analisa-se um novo cliente facultativo.

Como referido anteriormente, este processo dever passar por todos os clientes facultativos, at que estes j tenham sido todos analisados ou at que a soluo j no possa aceitar mais nenhum cliente na sua estrutura, pelo que alguns clientes facultativos no cheguem a ser analisados nesta primeira abordagem ao problema.

Algoritmo 3.1.1. Pseudo-cdigo Heurstica Construtiva de Insero Ponderada

$L \leftarrow 0;$

Para cada cliente obrigatrio, $r \in R$

Gerar rota $0 \rightarrow r \rightarrow 0$

$L \leftarrow L + p(r, 0) - c(r, 0) - c(0, r);$

Fim - repetio

Repetir enquanto o nmero de anis na soluo for superior a m

Repetir $\forall i, j, \in R: i \neq j$ e pertencem a anis disjuntos e i, j so extremos do respetivo anel

$$poupan\c{a}(i, j) = p(i, j) - c(i, j) + c(i, 0) + c(0, j) + \sum_{k \in \text{anel de } i, k \neq i} p(k, j) + \sum_{k \in \text{anel de } j, k \neq j} p(k, i), \quad \forall i, j, \in R, i \neq j$$

Fim-repetio

$M_p = \text{Mximo}(poupan\c{a}s)$ e atualizar a soluo fundindo dois anis pela ligao dos clientes (r_1, r_2) para os quais se obteve a poupana M_p

$L \leftarrow L + M_p;$

Fim-repetio

Repetir at que todos os anis tenham atingido a capacidade mxima ou que todos os clientes facultativos j tenham sido analisados

Escolher aleatoriamente um cliente facultativo $s \in S$; Para cada par de elementos (e_i, e_j) , ligados diretamente na soluo corrente, calcular a poupana associada  insero de s entre os clientes (e_i, e_j) :

$$poupan\c{a}(e_i, e_j) = c(e_i, e_j) - c(e_i, s) - c(e_i, s) + p(s, 0) + \sum_{k \in \text{clientes no anel}} p(k, s), \quad \forall e_i, e_j, \in \{\text{Clientes}\} \cup \{0\}, e_i \neq e_j$$

$M_p = \text{Mximo}(poupan\c{a}s)$, se $M_p > 0$ ento atualiza-se a soluo inserindo s na posio correspondente a M_p e atualiza-se o valor da soluo

$L \leftarrow L + M_p; S \leftarrow S \setminus \{s\}$

Fim-repetio

3.1.2. Heurística Construtiva de inserção aleatória

Para além da heurística construtiva anterior, desenvolveu-se uma segunda heurística construtiva, da qual se intitulará de heurística construtiva de inserção aleatória. Esta heurística tem algumas parecenças com a anteriormente explicada, uma vez que os clientes facultativos serão analisados da mesma forma, contudo difere na inserção dos clientes obrigatórios, que neste caso tem uma componente aleatória.

Dado que os clientes obrigatórios têm uma alta prioridade atribuída, são estes os primeiros a ser inseridos na rede. Nesta heurística, a ordem pela qual os clientes serão analisados para serem adicionados na solução deverá ser aleatória. Assim, de forma a garantir a presença de clientes em cada um dos anéis da estrutura, começa-se por atribuir aleatoriamente um cliente em cada um deles.

Considere-se o seguinte exemplo que retrata a heurística construtiva de inserção aleatória até este ponto de situação.

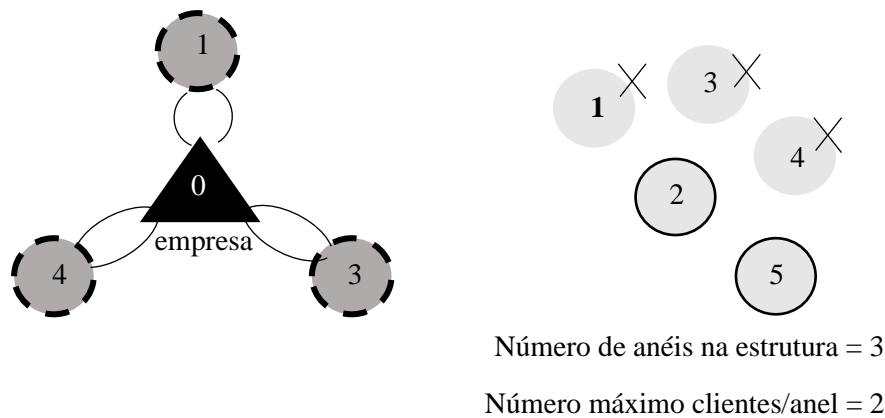


Figura 3.1.2.1 – Exemplo ilustrativo das escolhas de clientes facultativos na estrutura da Heurística Construtiva de Inserção Aleatória

De seguida, escolhe-se aleatoriamente um cliente obrigatório que ainda não esteja presente na estrutura da solução e analisa-se as posições que este pode tomar na estrutura. A inserção do cliente que está a ser analisado nunca pode comprometer a admissibilidade da solução, ou seja, este deve ser adicionado a um dos anéis já presentes na solução e a capacidade dos anéis não poderá ser excedida.

Ao analisar as posições possíveis que o cliente obrigatório escolhido pode tomar na solução, são calculadas as respetivas poupanças, de acordo com o raciocínio do cálculo apresentado na equação 3.1.1.2, porém em vez de se considerar a inserção de um cliente facultativo s , considera-se um cliente obrigatório ainda não presente na solução r . Assim, qualquer cliente obrigatório a ser inserido ainda não presente na solução, tomará o lugar entre dois elementos (que podem ser dois clientes ou um cliente e a empresa), que se designaram por e_1 e e_2 respetivamente e o cliente obrigatório a ser inserido designou-se por r . Assim, o cálculo deve ser feito da seguinte forma:

$$poupança(e_i, e_j) = c(e_i, e_j) - c(e_i, r) - c(e_j, r) + p(r, 0) + \sum_{k \in \text{clientes no anel}} p(k, r) \quad (3.1.2.1)$$

$$\forall e_i, e_j, \in R \cup \{0\}, e_i \neq e_j$$

Considere-se novamente o exemplo anterior, para a análise de inserção de um dos clientes obrigatórios ainda em falta:

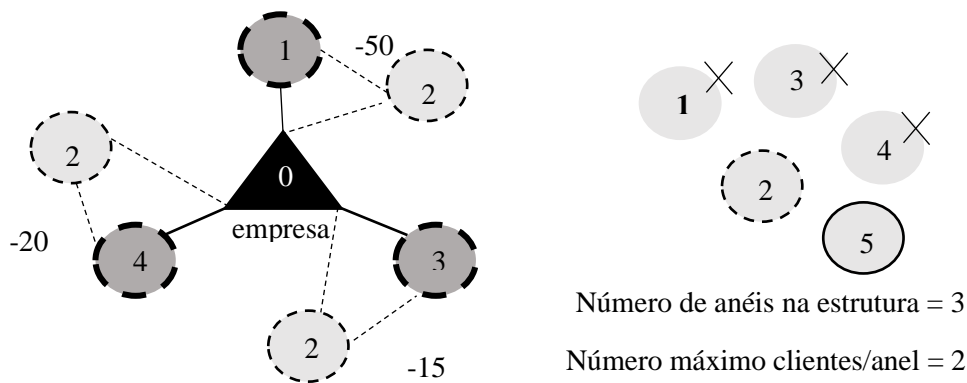


Figura 3.1.2.2. – Exemplo ilustrativo das possíveis inserções de clientes facultativos na estrutura da Heurística Construtiva de Inserção Aleatória

Após o cálculo das *poupanças*, escolhe-se o impacto mais benéfico entre as calculadas para incorporar a solução, mesmo que este tome um valor não positivo. No exemplo representado na figura 3.1.2.2., o cliente obrigatório escolhido foi o cliente apresentado pelo número 2. Este poderá ser inserido na solução nas posições apresentadas nos diferentes anéis, e terá um impacto de -50 se tomar a posição apresentada no anel do cliente 1, o valor -15 no anel do cliente 3 e o valor -20 no anel do cliente 4. Sendo -15 o valor do impacto mais benéfico, ou menos problemático neste caso, quando comparado com os referidos, este deverá ser o escolhido. Se este anel já tivesse atingido o número máximo de clientes que pode conter, escolhe-se o anel do cliente 4, com o valor do impacto de -20 e assim sucessivamente.

Este processo será repetido até que todos os clientes obrigatórios estejam inseridos na solução. De seguida, os clientes facultativos serão analisados como explicado na heurística anterior.

Algoritmo 3.1.2. Pseudo-código Heurística Construtiva de Inserção Aleatória

$L \leftarrow 0$;

Repetir até que cada anel tenha um cliente atribuído

Escolher aleatoriamente um cliente obrigatório e atribuir a um anel

Gerar rota $0 \rightarrow r \rightarrow 0$

$L \leftarrow L + p(r, 0) - c(r, 0) - c(0, r)$;

Fim-repetição

Repetir até que todos os clientes obrigatórios estejam na solução

Escolher aleatoriamente um cliente obrigatório, r

Calcular a poupança associada à inserção de r entre os clientes (e_i, e_j) ligados diretamente na solução corrente

$poupança(e_i, e_j) = c(e_i, e_j) - c(e_i, r) - c(e_i, r) + p(r, 0) + \sum_{k \in \text{clientes no anel}} p(k, r)$

$\forall e_i, e_j, \in R \cup \{0\}, e_i \neq e_j$

$M_p = \text{Máximo}(poupanças)$

Atualização da solução inserindo r na posição associada a M_p

$L \leftarrow L + M_p$

Fim-repetição

Repetir até que todos os anéis tenham atingido a capacidade máxima ou que todos os clientes facultativos já tenham sido analisados

Escolher aleatoriamente um cliente facultativo $s \in S$;

Para cada par de elementos (e_i, e_j) , ligados diretamente na solução corrente, calcular a poupança associada à inserção de s entre os clientes (e_i, e_j) :

$poupança(e_i, e_j) = c(e_i, e_j) - c(e_i, s) - c(e_i, s) + p(s, 0) + \sum_{k \in \text{clientes no anel}} p(k, s)$

$\forall e_i, e_j, \in \{\text{Clientes}\} \cup \{0\}, e_i \neq e_j$

$M_p = \text{Máximo}(poupanças)$, se $M_p > 0$ então atualiza-se a solução inserindo s na posição correspondente a M_p e atualiza-se o valor da solução

$L \leftarrow L + M_p$; $S \leftarrow S \setminus \{s\}$

Fim-repetição

3.2. Heurística de Melhoramento

Após o desenvolvimento das heurísticas construtivas apresentadas anteriormente, segue-se a heurística de melhoramento.

Esta heurística consiste em técnicas de pesquisa local aplicadas à solução inicial, que neste caso poderão ser as soluções obtidas nas heurísticas construtivas. O objetivo será melhorar a solução inicial a considerar, de acordo com o objetivo do problema, mantendo sempre a admissibilidade das soluções. Ou seja, o objetivo será sempre maximizar o valor do lucro total da solução, cumprindo sempre todas as restrições e características do problema já referidos.

Este melhoramento será realizado em várias fases, através da exploração de vizinhanças em cada uma delas. As vizinhanças são definidas da seguinte forma: dado um problema P e um conjunto de soluções admissíveis $F(P)$, uma vizinhança V é uma aplicação $V: F(P) \rightarrow 2^{F(P)}$, que faz corresponder a cada $x \in F(P)$ um subconjunto $V(x) \subseteq F(P)$. Cada fase terá sempre uma vizinhança definida, sendo que esta irá determinar as soluções vizinhas que serão analisadas.

A heurística de melhoramento desenvolvida procura melhorar a solução em várias fases. Começou-se por procurar melhorar e reorganizar cada anel. De seguida analisam-se os anéis e a reorganização que toda a estrutura poderá tomar. Por fim serão analisados todos os clientes facultativos que não pertencem à solução. Este procedimento será repetido sempre sobre a solução atualizada, que será referida daqui em diante, como solução corrente. Desta forma, para a solução corrente, esta será novamente estruturada até que não seja possível ser melhorada em nenhuma das fases referidas.

3.2.1. Fase 1 – Intra-anel

A primeira fase a considerar na heurística de melhoramento é a intra-anel e tem por objetivo explorar as diferentes posições que os clientes pertencentes a cada anel podem tomar, de forma a maximizar o lucro. A fase 1 tem por base a heurística 2-opt, considerada uma heurística de melhoramento que procura soluções que diferem entre si, em apenas duas arestas. A heurística 2-opt só se aplica a matrizes simétricas e dado que o problema consiste na geração de um proveito resultante dos clientes presentes em cada anel, sem direção atribuída, a matriz dos proveitos é simétrica. Para além disso, a matriz dos custos deverá ser também simétrica, caso isso não aconteça a fase 1 da heurística de melhoramento deve ser ignorada passando diretamente para a seguinte, a fase 2.

Esta heurística é bastante usada em inúmeros problemas de determinação de rotas não só pela sua simplicidade, mas pelos bons resultados e eficácia que apresenta na literatura.

Assim, esta é a primeira abordagem melhorativa nesta heurística, que irá permitir explorar as diversas possibilidades de reorganização da estrutura presente na solução corrente a considerar, sem alterar os clientes presentes na sua estrutura. Começa-se por explorar as possíveis melhorias a adotar na solução corrente, ao analisar cada anel da estrutura.

Para o desenvolvimento deste método será necessário ter em conta que as arestas a serem trocadas deverão ser não adjacentes.

Desta forma, a exploração das soluções vizinhas consiste em trocar duas arestas não adjacentes da solução corrente, que por sua vez irá alterar a sequência dos clientes no anel, com o objetivo de verificar se essas as trocas serão benéficas, ou seja, se há um aumento no valor do lucro. Assim, a vizinhança a considerar nesta fase da heurística de melhoramento é dada por: Seja x a solução corrente,

$V_{\text{intra-anel}}(x) = \{x\} \cup \{y: y \text{ é uma solução admissível que se obtém de } x \text{ por troca de duas arestas}\}$

Considere-se o seguinte exemplo, onde a solução inicial a considerar é a seguinte:

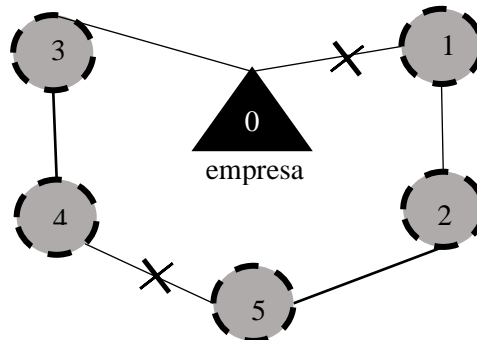


Figura 3.2.1.1 – Exemplo ilustrativo da solução corrente

A partir da ligação entre o vértice central, apresentado como a empresa, e o cliente 1, obtém-se as soluções vizinhas obtidas pela troca de arestas que não lhes sejam adjacentes, ou seja, a ligação entre os clientes 2 e 5, os clientes 5 e 4 e ainda os clientes 3 e 4. Seja a ligação entre os clientes 4 e 5, a aresta escolhida para ser trocada com a aresta que liga a empresa, e o cliente 1.

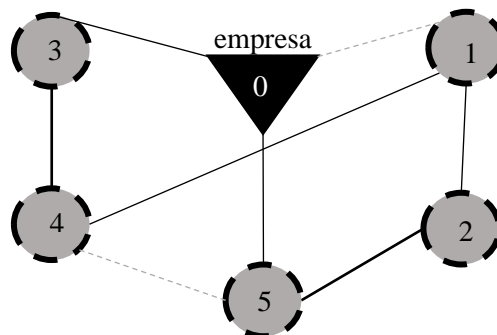


Figura 3.2.1.2 – Exemplo ilustrativo de uma possível solução obtida depois de aplicado algoritmo intra-anel

Nas arestas consideradas para a troca, o resultado obtido seria o apresentado na figura 3.2.1.2. Para cada par de arestas possíveis para a troca calcula-se o impacto no lucro.

Sejam a_k , os elementos do anel que se está a considerar, em que k representa a posição que estes assumem na solução corrente, com (a_i, a_{i+1}) e (a_j, a_{j+1}) os pares de arestas não adjacentes para a troca. Assim, as *poupanças*, ou seja, o cálculo do impacto no lucro, nesta fase da heurística de melhoramento, é dado por:

$$poupança_A((a_i, a_{i+1}), (a_j, a_{j+1})) = c(a_i, a_{i+1}) + c(a_j, a_{j+1}) - c(a_{i+1}, a_{j+1}) - c(a_i, a_j)$$

No exemplo apresentado na figura 3.2.1.2. decidiu-se pela troca das arestas (0,1) e (5,4). Considere-se que a sequência da solução apresentada é dada por 0-1-2-5-4-3-0, então (0,1) corresponde a (a_1, a_2) e (5,4) corresponde a (a_4, a_5) , de acordo com a sequência do anel apresentada, é obtida por:

$$poupan\c{c}a_A((0,1), (5,4)) = c(0,1) + c(5,4) - c(1,4) - c(0,5)$$

Depois de considerado cada par de arestas que pode ser trocado, se o valor do lucro total aumentar, isto é, se a troca das arestas a considerar tiver um valor da *poupança* positivo, a troca deverá ser aceite pelo que esta, passa a ser considerada a solução corrente. Este processo deverá ser repetido até que não seja possível melhorar o valor da solução.

3.2.2. Fase 2 - Inter-anel

Após a exploração das vizinhanças intra-anel, onde as soluções diferem nas posições que os clientes tomam dentro de cada anel, segue-se a exploração de soluções obtidas pelas trocas de clientes entre os anéis da estrutura na solução corrente.

Pretende-se analisar a troca de dois clientes que não partilham o mesmo anel, com o objetivo de aumentar o valor do lucro. Desta forma, a vizinhança a considerar é constituída por soluções admissíveis resultantes da troca entre dois clientes que não pertençam ao mesmo anel. Assim, seja x a solução corrente

$$V_{\text{inter-anel}}(x) = \{x\} \cup \{y: y \text{ é uma solução admissível que se obtém de } x \text{ trocando dois clientes na solução}\}$$

Esta vizinhança, tal como na fase anterior, procura novas soluções, que diferem em dois elementos: na fase 1, difere em duas arestas, enquanto que, nesta fase, difere na posição que dois clientes passam a assumir na estrutura.

Considere-se a seguinte solução corrente, como exemplo:

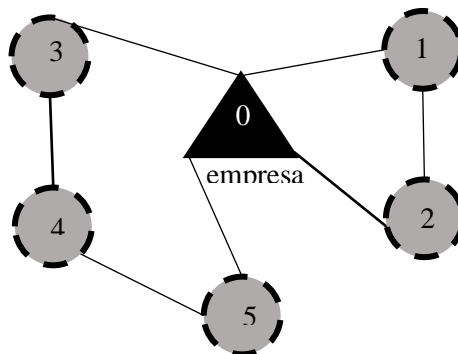


Figura 3.2.2.1 – Exemplo ilustrativo da solução corrente a considerar

As soluções vizinhas da solução corrente, a partir do cliente 1 são todas as que no seu lugar têm um cliente que pertencia a um anel disjunto do seu, e este ocupa o lugar do que o foi substituir na solução. Assim, para este cliente as soluções vizinhanças seriam, a sua troca com os clientes 3, 4 e 5.

Considere-se a troca entre o cliente 1 e 5. A solução obtida seria:

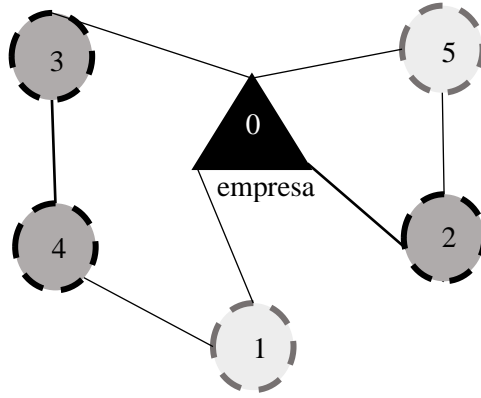


Figura 3.2.2.2 – Exemplo ilustrativo de uma possível solução obtida depois de aplicado o inter-anel

Esta fase tem o objetivo de estudar todas as hipóteses de troca na solução e estas serão aceites e incorporadas na mesma, se a melhoria no valor do lucro for conseguida. Para cada uma das trocas a considerar, é calculado o impacto que a mesma terá na solução, da seguinte forma:

Seja c_i o cliente escolhido do anel i e c_j o cliente escolhido do anel j para trocarem entre si. Esta troca irá alterar não só o custo de cada anel, como também, o proveito gerado em cada um deles, pelo que é necessário considerar $a_{l,k}$, em que k representa a posição que c_l ocupa no anel l . Assim, para cada par de clientes escolhidos para a troca, onde um pertence ao anel l_i e outro ao anel l_j , tem-se:

- **Impacto da troca nos custos**

$$\text{custo}(c_i, c_j) = c(a_{l_i, k-1}, a_{l_j, k}) + c(a_{l_j, k}, a_{l_i, k+1}) - c(a_{l_i, k-1}, a_{l_i, k}) - c(a_{l_i, k}, a_{l_i, k+1}) + c(a_{l_j, k-1}, a_{l_i, k}) + c(a_{l_i, k}, a_{l_j, k+1}) - c(a_{l_j, k-1}, a_{l_j, k}) - c(a_{l_j, k}, a_{l_j, k+1})$$

- **Impacto da troca nos proveitos**

$$\begin{aligned} \text{proveito}(c_i, c_j) = & \sum_{\text{clientes}_j \in \text{elementos do anel } j, \text{ clientes}_j \neq c_j} p(c_i, \text{clientes}_j) - \\ & \sum_{\text{clientes}_i \in \text{elementos do anel } i} p(c_i, \text{clientes}_i) + \\ & \sum_{\text{clientes}_i \in \text{elementos do anel } i, \text{ clientes}_i \neq c_i} p(c_j, \text{clientes}_i) - \\ & \sum_{\text{clientes}_j \in \text{elementos do anel } j} p(c_j, \text{clientes}_j) \end{aligned}$$

Assim, as *poupanças* são dadas por:

$$\text{poupança}_c(c_i, c_j) = p(c_i, c_j) - c(c_i, c_j)$$

Para a solução vizinha apresentada na figura 3.2.2.2., a poupança será dada por:

$$\text{poupança}_c(5, 1) = p(5,2) - p(1,2) + p(1,4) + p(1,3) - p(5,4) + p(5,3) - c(1,0) - c(1,4) + c(0,1) + c(1,2) - c(0,5) - c(5,2) + c(0,5) + c(5,4)$$

Esta troca deve ser aceite se o impacto no lucro da solução corrente for positivo. Sempre que se atualiza a solução o algoritmo continua considerando-a como a solução corrente a partir desse momento. Esta heurística é repetida até que todas as hipóteses tenham sido estudadas e não haja mais melhorias possíveis.

3.2.3. Fase 3 - Troca ou Adição de facultativos

Após a reorganização da solução, é necessário perceber se as escolhas feitas anteriormente, pelas heurísticas construtivas, quanto aos clientes facultativos foram as mais benéficas para a maximização do lucro.

Inicialmente, quando a solução é construída, por qualquer uma das heurísticas construtivas mencionadas, a escolha dos clientes facultativos é feita de forma aleatória e o critério de aceitação dos mesmos na solução é o impacto que a sua inserção tem no lucro, sem que seja feito um estudo geral aos clientes facultativos e a partir daí fazer a escolha de quais é que devem ser incluídos.

Na fase 3, segue-se o segundo objetivo da heurística de melhoramento, a reestruturação da estrutura. A solução corrente quando chega a esta fase da heurística de melhoramento, já foi estudada e alterada quanto à posição que os clientes assumem na estrutura nas duas fases anteriores, intra-anel e inter-anel. Porém os clientes escolhidos nas mesmas mantêm-se na solução.

O objetivo da troca ou inserção de facultativos é perceber se a escolha dos clientes facultativos na solução corrente foi a melhor ou se estes deverão ser trocados por outros, que não estejam incluídos na estrutura. Se a troca analisada não é aceite, segue-se a análise à inserção do cliente facultativo em estudo, para anéis que ainda não tenham atingido a sua capacidade.

Assim, a vizinhança a considerar nesta fase será constituída por todas as soluções que diferem da solução corrente num cliente facultativo, isto é, seja x a solução corrente

$V_{\text{troca de facultativos}}(x) = \{x\} \cup \{y : y \text{ é uma solução admissível que se obtém a partir de } x \text{ trocando um cliente facultativo de um dos anéis por outro cliente facultativo que não se encontra na estrutura}\}$

Começa-se por considerar todos os clientes que não estão presentes na solução inicial, este conjunto é denotado por F . Os clientes presentes neste conjunto serão aqueles que vão ser analisados, ao longo desta fase. Mesmo que algum cliente saia da solução ao ser trocado, em benefício da maximização do lucro, esse mesmo cliente não volta a ser considerado. À medida que se for analisando cada cliente de F , este é eliminado do conjunto.

Depois de reunido o grupo dos clientes facultativos que podem vir a ser incluídos na estrutura, através da troca ou inserção na solução começa-se por escolher um cliente do conjunto F . Para esse cliente são exploradas as soluções vizinhas, em $V_{\text{troca de facultativos}}$, onde as soluções são dadas pela troca de um dos clientes facultativos presentes na solução corrente pelo cliente escolhido de F . De forma a não comprometer a admissibilidade da nova solução, o cliente escolhido da solução corrente para a troca é também um cliente de baixa prioridade.

Na exploração dessas soluções vizinhas, começa-se por calcular as *poupanças*, interpretado como o impacto no lucro da solução em estudo ao efetuar a troca. Assim para cada par de clientes facultativos (s_i, s_j) , onde s_i pertence à solução corrente e s_j pertence a F , grupo de clientes facultativos inicialmente excluídos, considere-se o seguinte exemplo:

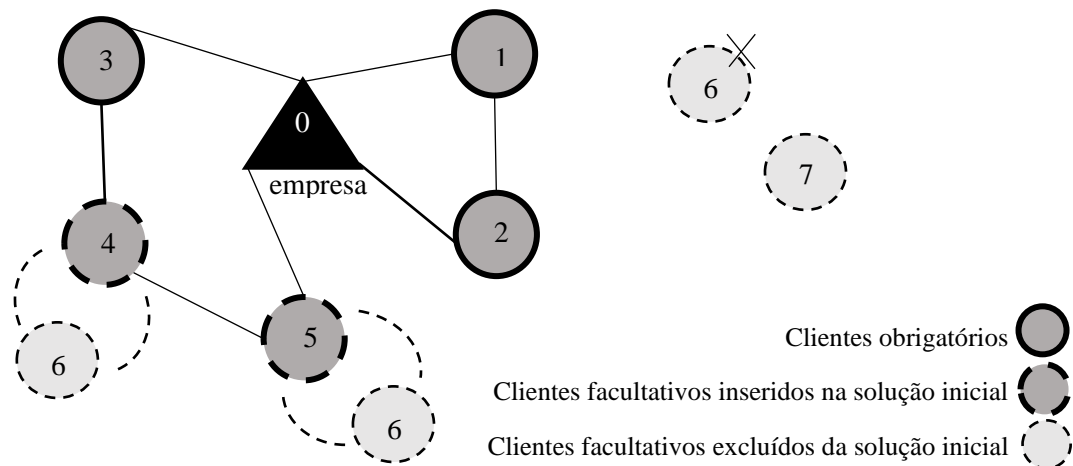


Figura 3.2.3.1. – Exemplo ilustrativo das possíveis escolhas de clientes facultativos na estrutura da Heurística de melhoramento de troca de facultativos

Na figura 3.2.3.1, está representada a estrutura de uma possível solução admissível corrente, onde se pretende a existência de 2 anéis e cada um poderá conter no máximo 3 clientes. Seja $\{1,2,3\}$ o conjunto de clientes obrigatórios neste caso, e os restantes, $\{4,5,6,7\}$ o conjunto dos clientes facultativos. Neste caso tem-se 2 clientes de baixa prioridade incluídos na solução corrente, que são os clientes 4 e 5, e ainda dois clientes que neste exemplo irão fazer parte do conjunto de clientes facultativos inicialmente excluídos, isto é, os clientes 6 e 7.

De acordo com a fase 3 da heurística de melhoramento, para a troca de facultativos, começa-se por analisar a troca entre os pares (4,6), depois (4,7), de seguida (5,6) e por fim (5,7). Para efetuar a análise que se pretende são calculadas *poupanças*, que irão refletir o impacto de cada troca no lucro da solução. Nas *poupanças* a calcular nesta fase, será necessário a sequência dos clientes no anel:

- **Impacto da troca nos custos**

Considere-se o par (s_i, s_j) , em que $s_i \in \{\text{solução corrente}\}$ e $s_j \in \{\text{clientes facultativos inicialmente excluídos}\}$. Para além do par de clientes para a troca, é necessário os vértices adjacentes a s_i , que podem ser dois clientes, ou um cliente e a empresa. Estes vértices são denotados por e , com $e \in \{\text{elementos adjacentes a } s_i \text{ na solução corrente}\}$.

Assim, o cálculo dos custos nas *poupanças* é dado por:

$$\text{custo}(s_i, s_j) = \sum_e c(s_j, e) - \sum_e c(s_i, e) \quad (3.2.3.1)$$

No exemplo apresentado na figura 3.2.3.1., o cálculo dos custos nas *poupanças* para o par (4,5) é dado por:

$$\text{custo}(4, 5) = \sum_{e \in \{3,5\}} c(5, e) - \sum_{e \in \{3,5\}} c(4, e)$$

- **Impacto da troca nos proveitos**

Considere-se o par (s_i, s_j) , em que $s_i \in \{\text{solução corrente}\}$ e $s_j \in \{\text{clientes facultativos inicialmente excluídos}\}$. Seja ainda $a_k \in \{\text{elementos do anel que se está a considerar, sem repetições (a empresa só se considera uma vez)}\}$, assim o cálculo dos proveitos é feito da seguinte forma:

$$\text{proveito}(s_i, s_j) = \sum_{a_k \neq s_j} p(s_j, a_k) - \sum_{a_k \neq s_i} p(s_i, a_k) \quad (3.2.3.2)$$

No exemplo apresentado na figura 3.2.3.1., o cálculo dos custos nas *poupanças* para o par (4,5) é dado por:

$$\text{proveito}(4, 5) = p(4, \text{empresa}) + p(4,3) + p(4,5) - p(6, 0) - p(6,3) - p(6, 5)$$

Desta forma, as *poupanças* são calculadas entre a diferença do proveito (s_i, s_j) e o custo (s_i, s_j), onde $s_i \in \{\text{solução corrente}\}$ e $s_j \in \{\text{clientes facultativos inicialmente excluídos}\}$, ou seja,

$$\text{poupança}_{CTF}(s_i, s_j) = p(s_i, s_j) - c(s_i, s_j) \quad (3.2.3.3)$$

Depois de calculadas as *poupanças*, toma-se a decisão consoante o valor máximo entre todas as obtidas. Se o valor máximo for positivo, significa que o impacto que terá no lucro, vai de encontro com o objetivo do problema, a maximização do lucro, e essa solução passará a ser a solução corrente. Para além disso, atualiza-se o conjunto F, ao retirar o cliente escolhido desse conjunto e o processo repete-se, ou seja, escolhe-se novamente um cliente do conjunto F e analisa-se a possibilidade deste ser trocado e assim sucessivamente:

Caso o valor máximo não seja positivo, para o cliente facultativo escolhido de F e que tem vindo a ser analisado, procede-se à análise de inserção do mesmo na solução, uma vez que a troca não foi possível.

A vizinhança considerada na inserção em estudo é constituída pelas soluções que diferem da solução corrente ao conterem mais um cliente facultativo. Seja x a solução corrente

$$V_{\text{adição de facultativos}}(x) = \{x\} \cup \{y : y \text{ é uma solução admissível que se obtém de } x \text{ adicionando um cliente facultativo a um dos anéis}\}$$

De forma a não perder a admissibilidade esta fase só será iniciada se na solução corrente houver anéis que ainda não tenham atingido a sua capacidade máxima.

Com a vizinhança definida, começa-se por analisar a inserção do cliente facultativo, que não foi possível ser trocado anteriormente na solução corrente, nas posições que este pode tomar. Para isso, calcula-se a *poupança*, ou seja, o impacto que a inserção do cliente facultativo inicialmente excluído tem no lucro.

Após o cálculo da *poupança* decide-se se o cliente facultativo inicialmente excluído deve ser incluído na solução, de acordo com o valor obtido. Se o valor obtido da *poupança* for positivo, a inserção do cliente é aceite e passa a fazer parte da solução corrente.

Considere-se o seguinte exemplo. Para uma rede que deverá ter 2 anéis, em que cada um deles tem uma capacidade máxima de 3 clientes, a solução corrente está representada na figura 3.2.2.1.

Começa-se por analisar as possíveis posições que o cliente facultativo inicialmente excluído pode tomar na estrutura.

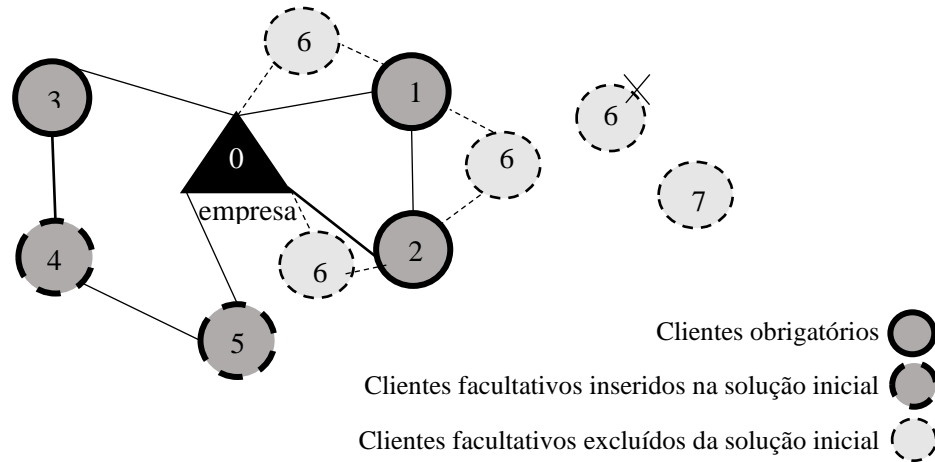


Figura 3.2.4.1. – Exemplo ilustrativo das possíveis escolhas de clientes facultativos na estrutura da Heurística de melhoramento de adição de facultativos

No exemplo representado na figura 3.2.4.1., é possível inserir ainda um cliente *facultativo* apresentado como $s \in \{\text{clientes facultativos inicialmente excluídos}\}$, uma vez que no anel mais à direita ainda não alcançou o número máximo de clientes que este pode conter.

Considere-se o cliente 6, o cliente escolhido dos facultativos inicialmente excluídos escolhido. As poupanças a calcular serão de acordo com as posições que este pode assumir na estrutura, representadas na figura referida.

Sejam e , os elementos do anel que se está a considerar a_k , em que k representa a posição que estes assumem na solução corrente. Neste caso a empresa é representada por a_1 , o cliente 1 por a_2 , o cliente 2 por a_3 e ao voltar à empresa esta deve ser representada por a_4 .

As *poupanças* da inserção entre cada par de elementos adjacentes, (a_k, a_l) , do anel em análise, deve ser calculada da seguinte forma:

$$poupança_{CIF}(a_k, a_l) = \sum_{a \in \text{elementos do anel (sem repetição da empresa)}} p(s, a) - c(a_k, s) - c(a_l, s) + c(a_k, a_l) \quad (3.2.4.1)$$

Considere-se o exemplo anteriormente apresentado. O cálculo da poupança para a inserção do cliente 6, cliente facultativo inicialmente excluído, entre os clientes 1 e 2, é:

$$poupança_{CIF}(1,2) = \sum_{a \in \{\text{empresa}, 1, 2\}} p(6, a) - c(1,6) - c(2,6) + c(1,2)$$

Depois de calculados as *poupanças*, se o valor máximo entre estes for positivo o cliente facultativo inicialmente excluído deve ser inserido na solução, passando esta a solução corrente e atualiza-se o conjunto F , ao retirar o cliente em estudo, desse conjunto. Inicia-se novamente esta fase com os elementos atualizados, para se escolher novamente um cliente desse conjunto e analisar a possibilidade de este ser trocado e caso não seja possível, analisar a possibilidade de este ser inserido.

Esta fase de troca ou adição de facultativos, é repetida até que todos os clientes do conjunto F definido inicialmente tenham sido analisados, ou seja, até que o conjunto esteja vazio.

A sequência das fases intra-anel, inter-anel, troca e adição de facultativos é repetida até que não ocorram alterações à solução-

Algoritmo 3.2. Pseudo-código Heurística de melhoramento

Repetir até que não haja nenhuma alteração da solução corrente em nenhuma das fases; $it=1$;

Fase 1

Repetir até que todas as soluções em $V_{intra-anel}$ sejam consideradas

Para cada par de arestas não adjacentes (a_i, a_{i+1}) e (a_j, a_{j+1}) calcula-se a poupança da troca entre esse mesmo par:

$$poupan\c{c}a_A((a_i, a_{i+1}), (a_j, a_{j+1})) = c(a_i, a_{i+1}) + c(a_j, a_{j+1}) - c(a_{i+1}, a_j) - c(a_i, a_j)$$

Se $poupan\c{c}a_A((a_i, a_{i+1}), (a_j, a_{j+1})) > 0$, atualiza-se a solução com a troca das arestas (a_i, a_{i+1}) e (a_j, a_{j+1})

$$L \leftarrow L + poupan\c{c}a_A((a_i, a_{i+1}), (a_j, a_{j+1}))$$

Fim - se

Fim-repetição

Fase 2

Repetir até que todas as soluções em $V_{inter-anel}$ sejam consideradas

$poupan\c{c}a_C(c_i, c_j) = p(c_i, c_j) - c(c_i, c_j)$, com c_i o cliente escolhido do anel i e c_j o cliente escolhido do anel j

Se $poupan\c{c}a_C(c_i, c_j) > 0$, atualiza-se a solução com a troca que os clientes (c_i, c_j) ocupam na solução;

$$L \leftarrow L + poupan\c{c}a_C(c_i, c_j)$$

Fim - se

Fim - repetição

Algoritmo 3.2. Pseudo-código Heurística de melhoria (continuação)

Fase 3

Seja F o grupo de clientes facultativos inicialmente excluídos e $F'=F$

Repetir até que $F'=\{\}$

Escolher aleatoriamente um cliente de F' , s_j

Calcular $poupança_{CTF}(s_i, s_j) = p(s_i, s_j) - c(s_i, s_j)$, com $s_i \in \{\text{solução corrente}\}$

Se $poupança_{CTF}(s_i, s_j) > 0$, atualiza-se a solução com a troca que do cliente facultativo escolhido s_j , pelo cliente facultativo s_i ,

$$L \leftarrow L + poupança_{CTF}(s_i, s_j)$$

Caso contrário

Se houver pelo menos um anel que não tenha alcançado o número máximo de clientes que pode conter

Calcular o impacto que o cliente facultativo inicialmente excluído escolhido anteriormente na fase 3, tem no lucro

$$poupança_{CIF}(a_k, a_i) = \sum_{a \in \text{elementos do anel (sem repetição da empresa)}} p(s_j, a) - c(a_k, s_j) - c(a_i, s_j) + c(a_k, a_i)$$

Se $poupança_{CIF}(a_k, a_i) > 0$, atualiza-se a solução com a adição do cliente s_j entre os elementos (a_k, a_i) na estrutura

$$L \leftarrow L + poupança_{CIF}(a_k, a_i)$$

Fim - se

Fim- se

Fim - se

$$F'=F'/\{s_j\}$$

Fim - repetição

Fim - repetição

3.3. Meta-heurística

Uma meta-heurística é um método heurístico que procura combinar escolhas aleatórias com o histórico de resultados anteriormente obtidos pelo mesmo, ao explorar as vizinhanças, conceito explicado em 3.2. Esta combinação permite a este método sair de um ótimo local, que acaba por ser um problema nas heurísticas de melhoramento, por exemplo.

As meta-heurística têm vindo a provar, que o seu desempenho é bastante mais sofisticado quando comparado com as heurísticas anteriormente mencionadas. Estas permitem que a solução saia de um ótimo local, ao desconstruir o óbvio na tomada de decisões numa solução. Contam ainda, com uma componente de aleatoriedade que irá enriquecer o estudo das soluções do problema e no resultado obtido.

3.3.1. Pesquisa Local iterada

A meta-heurística escolhida para ser aplicada neste problema foi a pesquisa local iterada, conhecida na literatura como *Iterated Local Search* (ILS).

Esta meta-heurística reúne a aleatoriedade, destruição, reestruturação e pesquisa local, num só algoritmo. Este será sujeito à repetição por iterações que permitem explorar soluções que à primeira vista podem não parecer uma boa escolha na sua construção, por serem exploradas de forma aleatória.

Assim, resulta um algoritmo composto por duas fases, a fase de perturbação e a fase de pesquisa local. O algoritmo da meta-heurística ILS é marcado pela sua elevada eficácia e pela sua simplicidade. Esta meta-heurística procura uma construção modular, para que a sua construção seja mais prática e que seja adaptável para eventuais mudanças.

Começa-se por considerar uma solução inicial que se pretenda melhorar e por isso é sujeita a uma pesquisa local. Este passo é bastante familiar em estudos realizados com heurísticas, onde se constrói uma solução e de seguida é sujeita a melhorias, conhecidas por heurísticas construtivas e heurísticas de melhoramento, já abordadas anteriormente nesta dissertação. Desta forma, a meta-heurística ao combinar todos estes métodos permitem explorar opções que não convirjam num ótimo local, uma desvantagem das heurísticas de melhoramento em muitos exemplos.

Para uma solução inicial que pode ser obtida por heurísticas construtivas, ou ainda pelas melhorias das mesmas, isto é, por heurísticas de melhoramento segue-se a explicação das duas fases do algoritmo.

3.3.1.1. Perturbação

Este processo consiste, tal como o nome indica, em perturbar a solução corrente, com o objetivo inicial de desconstruir a previsibilidade na exploração das soluções. Estas perturbações podem ser feitas e adaptadas a cada problema. Porém, é bastante importante que o processo e a escolha da perturbação a aplicar seja equilibrada, uma vez que se estas forem demasiado pequenas a eficácia da meta-heurística pode ficar comprometida pois poderá convergir para um ótimo local, porém se a perturbação for demasiado grande pode estragar as soluções sem conseguir uma melhoria.

Neste caso, a perturbação a aplicar tem duas fases: a fase da destruição e fase da reposição da admissibilidade.

- **Fase da destruição**

Esta fase tem por objetivo colocar todos os clientes na estrutura da solução, ou seja, colocar os restantes clientes facultativos.

Começa-se por escolher um desses mesmos clientes, de forma aleatória e de seguida, é feita uma análise a todas as posições que este pode tomar na estrutura, e é escolhida a que tiver melhor impacto no valor do lucro.

A análise das posições que cada cliente escolhido pode tomar na solução é feita com base no cálculo de *poupanças* que segue a equação identificada como (3.2.4.1). Esta fase é bastante parecida com a fase 4 da heurística de melhoramento, porém o critério de aceitação neste caso difere. Na corrente fase da meta-heurística, depois de analisadas as *poupanças* para cada cliente, é escolhida aquela que mais beneficia o lucro final, ou aquela que menos prejudica no caso do valor máximo das poupanças tomar um valor negativo. O cliente analisado será inserido na posição correspondente à poupança escolhida.

Com o objetivo de assegurar que todos os anéis têm pelo menos o limite máximo de clientes atingido, de forma a equilibrar a distribuição dos clientes na estrutura, os anéis que não têm a sua capacidade máxima atingida deverão ser os primeiros a incorporar os clientes em análise. Depois de todos os anéis atingirem a sua capacidade máxima, os restantes clientes em falta na solução serão atribuídos a qualquer um dos anéis presentes na estrutura, o que irá destruir a admissibilidade da solução corrente.

Depois da fase de destruição, segue-se para a fase seguinte, a fase da reposição da admissibilidade.

- **Fase da reposição da admissibilidade**

Esta fase consiste, como referido, na reposição da admissibilidade da solução, uma vez que foi destruída anteriormente. O que torna esta solução não admissível quanto ao problema em estudo, é o facto do limite máximo de clientes por anel não ser respeitado. Assim, o objetivo principal é devolver o cumprimento desta restrição.

Começa-se por escolher um dos anéis que excede a sua capacidade máxima. Desse mesmo anel, escolhe-se aleatoriamente um cliente facultativo, para ser retirado da solução. Esta decisão deve ser repetida para todos os anéis que não respeitem o limite máximo, até que a solução volte a recuperar a sua admissibilidade.

A solução resultante das duas fases de perturbação terá na sua estrutura o número de anéis que se pretende e cada um deles terá tantos clientes, quanto o limite máximo restrito no problema.

Depois de perturbar a solução segue-se a pesquisa local à solução obtida.

3.3.1.2. Pesquisa Local

O processo seguinte a aplicar nesta meta-heurística será uma pesquisa local, que pode ser o mesmo método de pesquisa local explicado anteriormente há heurística de melhoramento. Devido a esta estrutura é importante que todos os processos sejam desenvolvidos de forma modular pelas razões mencionadas anteriormente.

Nesta dissertação, como pesquisa local, utilizou-se a heurística de melhoramento anteriormente explicada. No fim do processo de pesquisa local, o resultado do lucro da solução encontrada é comparado com a solução considerada melhor até então, e caso a nova solução seja melhor do que a anterior esta deve ser atualizada. Estes dois processos, perturbação e pesquisa local serão repetidos um número pré-fixado de iterações, dando origem à meta-heurística ILS aplicada e desenvolvida para o problema.

Algoritmo 3.3. Pseudo-código Meta-Heurística Pesquisa Local Iterada, ILS

Dados: uma solução admissível x e número máximo de iterações (Max_it)

Aplicar o algoritmo de Pesquisa Local 3.2 à solução x

Repetir Max_it vezes

// Fase da destruição

Escolher aleatoriamente um cliente facultativo ainda não presente na solução, s

Se houver anéis com a capacidade máxima ainda não atingida

Calcular as *poupanças*, para a inserção de s entre os elementos do anel (a_k, a_l) :

$$poupança(a_k, a_l) = \sum_{a \in \text{elementos do anel (sem repetição da empresa)}} p(s, a) - c(a_k, s) - c(a_l, s) + c(a_k, a_l)$$

Atualiza-se a solução com a adição do cliente s entre os elementos (a_k, a_l) presentes num anel sem a capacidade máxima ainda atingida; $L \leftarrow L + \max(poupança(a_k, a_l))$

Caso contrário

Calcular as *poupanças*, para as posições possíveis de todos os anéis

$$poupança(a_k, a_l) = \sum_{a \in \text{elementos do anel (sem repetição da empresa)}} p(s, a) - c(a_k, s) - c(a_l, s) + c(a_k, a_l)$$

Atualiza-se a solução com a adição do cliente s entre os elementos (a_k, a_l) na estrutura; $L \leftarrow L + \max(poupança(a_k, a_l))$

Fim - se

Fim - repetição

// Fase da reposição da admissibilidade

Repetir até que a capacidade máxima de clientes por anel seja respeitada

Escolher o anel com capacidade máxima excedida e escolher aleatoriamente um cliente facultativo da solução e retirá-lo da solução;

Fim - repetição

Fim - repetição

4. Experiência Computacional

No corrente capítulo apresenta-se a experiência computacional feita nesta dissertação.

As heurísticas propostas no capítulo anterior foram implementadas em linguagem matlab, usando o software Matlab R2018b. A experiência computacional foi realizada, num computador com um processador AMD Ryzen 5 3500U with Radeon Vega Mobile Gfx e com uma memória RAM de 8 GB.

Para testar as heurísticas propostas considerou-se um conjunto de instâncias geradas a partir de instâncias benchmark para o TSP. Na primeira secção do presente capítulo é detalhado o modo como se geraram as instâncias de teste.

De seguida, apresentam-se os resultados para cada uma das heurísticas implementadas, e respetivos comentários.

Por fim, para a meta-heurística escolhida nesta dissertação, a pesquisa local iterada, ILS, será feito um estudo com o objetivo de afinar o número de iterações.

4.1. Instâncias de Teste

As heurísticas implementadas e anteriormente apresentadas foram testadas com instâncias obtidas na biblioteca [11], disponibilizada por Bernardino e Paias, para uma variante do problema do caixeiro viajante. Nesta biblioteca é possível encontrar algumas instâncias com diferentes características, porém para este problema utilizaram-se apenas instâncias simétricas. Para além disso, são ainda gerados elementos de forma aleatória, explicados de seguida.

Para cada instância gerada para testar as heurísticas é crucial a definição de alguns elementos necessários neste problema, o SmRNDP. Esses elementos e as suas denotações, já mencionadas nesta dissertação e que serão importantes de ter conta são:

- O número total de clientes (n);
- O número de anéis (m);
- O número de clientes obrigatórios (o conjunto destes clientes é representado por R e assume-se que os obrigatórios são os $|R|$ primeiros clientes do conjunto de todos os clientes);
- Capacidade máxima de cada anel (b);
- Matriz de custos C , com uma dimensão de $(n+1) \times (n+1)$ (c_{ij} – custo associado à ligação direta entre o cliente/empresa i e o cliente/empresa j);
- Matriz de proveitos P , com uma dimensão de $(n+1) \times (n+1)$ (p_{ij} – proveito associado a ter o cliente/empresa i no mesmo anel que o cliente/empresa j).

Número de clientes

Como referido, foram usadas nove instâncias nesta dissertação, com o objetivo de testar diferentes contextos e tornar os testes mais enriquecedores nas conclusões a retirar, quanto às soluções obtidas a partir de cada uma.

Nas instâncias geradas consideraram-se diferentes dimensões que variam entre os 20 nodos e os 264 nodos presentes na estrutura. O número de nodos $n+1$ é dado pela soma do número de clientes, denotado por n , e o nodo representando a empresa, ou seja, o nodo 0.

Número de anéis

De forma a comparar o comportamento das heurísticas apresentadas no capítulo anterior, decidiu-se considerar 3 anéis para todas as nove instâncias utilizadas, ou seja, $m=3$.

O número de clientes obrigatórios

No conjunto de clientes, assume-se que se encontra ordenado segundo a prioridade atribuída, onde os clientes obrigatórios se encontram em primeiro lugar, seguidos dos clientes facultativos. Assim, de acordo com o número de clientes obrigatórios, presentes em cada instância, o seu conjunto é constituído pelos $|R|$ primeiros clientes do conjunto de todos os clientes e os restantes são o conjunto dos clientes facultativos denotado por S , uma vez que o conjunto dos nodos da estrutura, V , é o resultado entre a união dos conjuntos dos clientes com baixa e alta prioridade e ainda a empresa, ou seja, $V = R \cup S \cup \{0\}$.

Assim, de forma a obter a cardinalidade do conjunto de clientes obrigatórios $|R|$, utilizou-se o método apresentado na literatura pelos autores em [3]. Este método consiste na definição de uma proporção, denotada por β , em relação ao número total de nodos na instância, V .

Desta forma, a proporção que define o número de clientes obrigatórios em cada instância β é de 0.25, pelo que $|R| = \lceil \beta * (n + 1) \rceil$.

Capacidade máxima de clientes em cada anel

Uma das restrições presentes no problema SmRNDP é o número máximo de clientes que cada anel pode conter. Tal como no número de clientes obrigatórios, explicado anteriormente, este será gerado de acordo com o método apresentado em [3], ou seja, pela proporção que cada anel pode conter, tendo em conta que essa quantidade é equilibrada entre eles.

Essa proporção é denotada por α e a capacidade máxima de cada anel b , é dada por $b = \left\lceil \alpha * \left\lceil \frac{n+1}{m} \right\rceil \right\rceil$. Nesta dissertação considerou-se $\alpha=0.5$.

Matriz de custos

No problema em estudo, cada ligação entre dois nodos da estrutura tem um custo associado não negativo. A matriz de custos C é uma matriz quadrada, com dimensão $(n+1) \times (n+1)$. Nesta dissertação considerou-se que a matriz é simétrica, com valores não negativos, $c_{ij} \geq 0$, com $i, j \in V$, e que c_{ii} tomam o valor zero.

Para a geração desta matriz para cada uma das instâncias, utilizou-se a biblioteca [11], disponibilizada por Bernardino e Paias, para uma variante do problema do caixeiro viajante, TSP e que foram aplicadas nesta dissertação ao problema SmRNDP.

De seguida serão apresentadas e referenciadas as matrizes desta biblioteca que foram utilizadas em cada uma das nove instâncias.

Matriz de proveitos

No problema em estudo, cada par de nodos que partilhe um mesmo anel tem associado um proveito, uma vez que quando se inclui um nodo num anel, este pode comunicar com todos os nodos presentes nesse mesmo anel.

Os proveitos referidos, apresentados numa matriz quadrada com dimensão $(n+1) \times (n+1)$, simétrica e com valores não negativos, como na matriz de custo. Porém para esta matriz utilizou-se o software Matlab para a geração aleatória de proveitos, que para qualquer uma das instâncias, os valores que podem tomar estão no intervalo $[0;1000]$.

Para realizar a experiência computacional criaram-se nove cenários para o SmRNDP. Assim, criaram-se nove instâncias, que diferem entre si, nas características acima mencionadas, à exceção do número de anéis que se pretende na estrutura, que serão sempre 3 anéis para cada uma delas.

De seguida, apresentam-se os nove cenários considerados, ou seja as nove instâncias que se encontram num ficheiro *.txt*, e que na sua designação se encontram o número de nodos, $|V|$, número total de clientes, n , número de anéis, m e ainda a capacidade máxima de cada um, b .

'Instancia_20_nodos_n19_m3_b4.txt'

- O número total de clientes é $n = 19$;
- O número de anéis na estrutura a desenhar é $m = 3$;
- O número de clientes obrigatórios no conjunto R , é $|R| = 5$;
- Capacidade máxima de cada anel é $b = 4$ clientes;
- Matriz de custo

O ficheiro recolhido da biblioteca [11], que suportou a geração desta instância é apresentado como "*u159.sftsp_159_17_1.txt*". Esta matriz excede a dimensão que se pretende, 20×20 , pelo que se consideraram apenas as suas primeiras 20 linhas e 20 colunas.

O valor máximo presente nesta matriz é de 3635;

- Matriz de proveito

Esta matriz tem a dimensão 20×20 , e o intervalo de valores que toma $[0;1000]$ os valores dos custos poderão ser maiores do que os proveitos, uma vez que podem tomar no máximo o valor de 3635 enquanto os proveitos tomam no máximo o valor 1000.

'Instancia_30_nodos_n29_m3_b5.txt'

- O número total de clientes é $n = 29$;
- O número de anéis na estrutura a desenhar é $m = 3$;
- O número de clientes obrigatórios no conjunto R , é $|R| = 8$;
- Capacidade máxima de cada anel é $b = 5$ clientes;
- Matriz de custo

O ficheiro recolhido da biblioteca [11], que suportou a geração desta instância é apresentado como "*rat195.sftsp_195_29_1.txt*". Esta matriz excede a dimensão que se pretende, 20×20 , pelo que se consideraram apenas as suas primeiras 30 linhas e 30 colunas.

O valor máximo presente nesta matriz é de 124;

- Matriz de proveito

Esta matriz tem a dimensão 30x30, e o intervalo de valores que toma [0;1000] os valores dos proveitos poderão ser maiores do que os proveitos, uma vez que podem tomar no máximo o valor de 1000 enquanto os custos tomam no máximo o valor 124.

'Instancia_50 nodos_n49_m3_b9.txt'

- O número total de clientes é $n = 49$;
- O número de anéis na estrutura a desenhar é $m = 3$;
- O número de clientes obrigatórios no conjunto R, é $|R| = 13$;
- Capacidade máxima de cada anel é $b = 9$ clientes;
- Matriz de custo

O ficheiro recolhido da biblioteca [11], que suportou a geração desta instância é apresentado como "*pr226.sftsp_226_28_1.txt*". Esta matriz excede a dimensão que se pretende, 20x20, pelo que se consideraram apenas as suas primeiras 50 linhas e 50 colunas.

O valor máximo presente nesta matriz é de 13816;

- Matriz de proveito

Esta matriz tem a dimensão 50x50, e o intervalo de valores que toma [0;1000] os valores dos custos poderão ser maiores do que os proveitos, uma vez que podem tomar no máximo o valor de 13816 enquanto os proveitos tomam no máximo o valor 1000.

'Instancia_70 nodos_n69_m3_b12.txt'

- O número total de clientes é $n = 69$;
- O número de anéis na estrutura a desenhar é $m = 3$;
- O número de clientes obrigatórios no conjunto R, é $|R| = 18$;
- Capacidade máxima de cada anel é $b = 12$ clientes;
- Matriz de custo

O ficheiro recolhido da biblioteca [11], que suportou a geração desta instância é apresentado como "*pr144.sftsp_144_22_1.txt*". Esta matriz excede a dimensão que se pretende, 20x20, pelo que se consideraram apenas as suas primeiras 70 linhas e 70 colunas.

O valor máximo presente nesta matriz é de 8329;

- Matriz de proveito

Esta matriz tem a dimensão 70x70, e o intervalo de valores que toma [0;1000] os valores dos custos poderão ser maiores do que os proveitos, uma vez que podem tomar no máximo o valor de 8329 enquanto os proveitos tomam no máximo o valor 1000.

'Instancia_100 nodos_n99_m3_b17.txt'

- O número total de clientes é $n = 99$;
- O número de anéis na estrutura a desenhar é $m = 3$;
- O número de clientes obrigatórios no conjunto R, é $|R| = 25$;
- Capacidade máxima de cada anel é $b = 17$ clientes;
- Matriz de custo

O ficheiro recolhido da biblioteca [11], que suportou a geração desta instância é apresentado como “*d198.sftsp_198_29_1.txt*”. Esta matriz excede a dimensão que se pretende, 20x20, pelo que se consideraram apenas as suas primeiras 100 linhas e 100 colunas.

O valor máximo presente nesta matriz é de 2232;

- Matriz de proveito

Esta matriz tem a dimensão 100x100, e o intervalo de valores que toma [0;1000] os valores dos custos poderão ser maiores do que os proveitos, uma vez que podem tomar no máximo o valor de 2232 enquanto os proveitos tomam no máximo o valor 1000. Nesta instância os valores máximos que cada uma da matriz pode alcançar são mais equilibrados do que nas instâncias anteriores.

'Instancia_136 nodos_n135_m3_b23.txt'

- O número total de clientes é $n = 135$;
- O número de anéis na estrutura a desenhar é $m = 3$;
- O número de clientes obrigatórios no conjunto R, é $|R| = 34$;
- Capacidade máxima de cada anel é $b = 23$ clientes;
- Matriz de custo

O ficheiro recolhido da biblioteca [11], que suportou a geração desta instância é apresentado como “*pr136.sftsp_136_21_1.txt*”.

O valor máximo presente nesta matriz é de 14076;

- Matriz de proveito

Esta matriz tem a dimensão 136x136, e o intervalo de valores que toma [0;1000] os valores dos custos poderão ser maiores do que os proveitos, uma vez que podem tomar no máximo o valor de 14076 enquanto os proveitos tomam no máximo o valor 1000.

'Instancia_152 nodos_n151_m3_b26.txt'

- O número total de clientes é $n = 151$;
- O número de anéis na estrutura a desenhar é $m = 3$;
- O número de clientes obrigatórios no conjunto R, é $|R| = 38$;
- Capacidade máxima de cada anel é $b = 26$ clientes;
- Matriz de custo

O ficheiro recolhido da biblioteca [11], que suportou a geração desta instância é apresentado como “*pr152.sftsp_152_23_1.txt*”.

O valor máximo presente nesta matriz é de 16429;

- Matriz de proveito

Esta matriz tem a dimensão 152x152, e o intervalo de valores que toma [0;1000] os valores dos custos poderão ser maiores do que os proveitos, uma vez que podem tomar no máximo o valor de 16429 enquanto os proveitos tomam no máximo o valor 1000.

'Instancia_202 nodos_n201_m3_b34.txt'

- O número total de clientes é $n = 201$;
- O número de anéis na estrutura a desenhar é $m = 3$;

- O número de clientes obrigatórios no conjunto R, é $|R| = 51$;
- Capacidade máxima de cada anel é $b = 34$ clientes;
- Matriz de custo

O ficheiro recolhido da biblioteca [11], que suportou a geração desta instância é apresentado como “*gr202.sftsp_202_30_1.txt*”.

O valor máximo presente nesta matriz é de 6535;
- Matriz de proveito

Esta matriz tem a dimensão 202x202, e o intervalo de valores que toma [0;1000] os valores dos custos poderão ser maiores do que os proveitos, uma vez que podem tomar no máximo o valor de 6535 enquanto os proveitos tomam no máximo o valor 1000. Nesta instância os valores máximos que cada uma da matriz pode alcançar são mais equilibrados do que na maioria das instâncias anteriores.

'Instancia_264_nodos_n263_m3_b44.txt'

- O número total de clientes é $n = 263$;
- O número de anéis na estrutura a desenhar é $m = 3$;
- O número de clientes obrigatórios no conjunto R, é $|R| = 66$;
- Capacidade máxima de cada anel é $b = 44$ clientes;
- Matriz de custo

O ficheiro recolhido da biblioteca [11], que suportou a geração desta instância é apresentado como “*pr264.sftsp_264_37_1.txt*”.

O valor máximo presente nesta matriz é de 9016;
- Matriz de proveito

Esta matriz tem a dimensão 264x264, e o intervalo de valores que toma [0;1000] os valores dos custos poderão ser maiores do que os proveitos, uma vez que podem tomar no máximo o valor de 9016 enquanto os proveitos tomam no máximo o valor 1000.

4.2. Heurísticas Construtivas

Apesar das heurísticas construtivas, terem por objetivo nesta dissertação, a inicialização do processo, para a construção de uma solução admissível ao SmRNDP, decidiu-se implementar duas heurísticas construtivas, ligeiramente diferentes, como explicado no capítulo anterior: a heurística construtiva de inserção ponderada e a heurística construtiva de inserção aleatória.

Uma vez que ambas as heurísticas construtivas contam com um fator aleatório, decidiu-se, para cada instância, correr cinco vezes cada uma das heurísticas para melhor analisar as suas soluções e poder tirar conclusões.

De seguida, são apresentados os resultados sintetizados para cada uma das heurísticas construtivas.

Para as heurísticas construtivas, ponderada e aleatória, são apresentados os resultados nas duas tabelas seguintes de acordo com o seguinte formato: na primeira coluna identifica-se a instância, nas três colunas seguintes apresentam-se respectivamente o lucro mínimo, o lucro máximo, o lucro médio e o desvio-padrão do lucro obtidos nas cinco *runs*. Por fim, na última coluna apresenta-se o tempo médio

de processamento em segundos. Os valores de lucro são expressos em unidades monetárias. No anexo A, apresentam-se para cada instância os resultados detalhados de cada *run*.

Tabela 4.2.1 – Resultados da Heurística construtiva de inserção ponderada

Instância	Mínimo Lucro	Máximo Lucro	Média Lucro	Média CPU
20 nodos_n19_m3_b4	-1244	-503	-1003	0
30 nodos_n29_m3_b5	21679	24811	23295	0
50 nodos_n49_m3_b9	23550	27002	26311,6	0
70 nodos_n69_m3_b12	38416	69691	53124,4	0
100 nodos_n99_m3_b17	210099	226553	218295	1
136 nodos_n135_m3_b23	161182	179345	167327,6	1
152 nodos_n151_m3_b26	231987	256169	243907,2	2
202 nodos_n201_m3_b34	559053	575549	569366,6	5
264 nodos_n263_m3_b44	1415302	1446708	1432575,2	22
Média	295558	311703	303689	4

Tabela 4.2.2 – Resultados da Heurística construtiva de inserção aleatória

Instância	Mínimo Lucro	Máximo Lucro	Média Lucro	Média CPU
20 nodos_n19_m3_b4	-2482	-150	-1565,2	0
30 nodos_n29_m3_b5	20963	26323	23705	0
50 nodos_n49_m3_b9	9945	24074	17423,6	0
70 nodos_n69_m3_b12	34330	71231	50646,6	0
100 nodos_n99_m3_b17	134783	145675	141461,2	0
136 nodos_n135_m3_b23	179042	310477	240391,8	1
152 nodos_n151_m3_b26	167393	277271	213963,6	2
202 nodos_n201_m3_b34	561249	585544	574097,8	4
264 nodos_n263_m3_b44	940484	1410905	1037825,2	13
Média	227301	316817	255328	2

Para os resultados apresentados nas tabelas 4.2.1 e 4.2.2 é necessário ter em conta as características que cada uma das instâncias consideradas neste estudo têm, pelo que será preciso ter em conta o intervalo que os valores presentes na matriz de custo podem tomar, assim como o intervalo dos valores na matriz de proveito, de forma a perceber se existe um equilíbrio ou uma discrepância entre eles.

Começando por analisar as duas instâncias de menor dimensão, as instâncias com 20 e 30 nodos, é possível perceber o impacto que os valores dos custos face aos valores dos proveitos têm no valor do lucro da estrutura obtida. Na primeira instância considerada, com 20 nodos, sabe-se que o intervalo dos valores do custo supera bastante o intervalo dos valores dos proveitos, enquanto que na segunda instância, de 30 nodos, acontece o contrário. Devido a esta característica o valor do lucro obtidas nas soluções correspondentes a cada uma das instâncias são bastante dispares.

Contudo, o valor do lucro obtido deve-se também ao facto das heurísticas construtivas, considerarem a inserção à vez, de apenas um cliente.

Considere-se um exemplo de uma das soluções obtidas da instância de 20 nodos, onde a empresa é representada pelo vértice 0, o conjunto dos clientes obrigatórios é $R=\{2,\dots,5\}$ e o conjunto dos clientes facultativos é $S=\{6,\dots,20\}$.

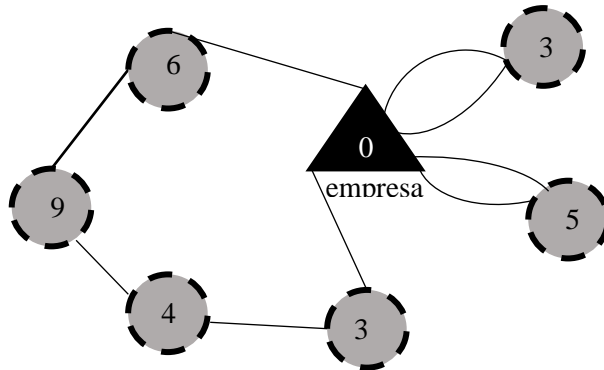


Figura 4.2.1. – Exemplo ilustrativo de um dos resultados obtidos pela instância de 20 nodos

O tipo de solução apresentada na figura 4.2.1., onde existe pelo menos um anel com um único cliente, foram bastante comuns para a maioria das instâncias, para instâncias em que os intervalos dos custos são bastante mais elevados do que para os valores dos proveitos. Uma vez que os clientes são inseridos um a um, apesar de ser espectável para um anel obter um lucro positivo dada a partilha dos clientes presentes gerar proveitos, como explicado em capítulos anteriores. Isto não acontece quando o proveito gerado na inserção de um cliente num determinado anel não compensa face aos custos associados.

Por esta razão acontece em algumas soluções, ter anéis apenas com um cliente, uma vez que ao analisar a inserção de um cliente nesses anéis em instâncias com esta característica não compensa, até que este atinja um certo número de clientes, de forma a que os proveitos gerados entre todos os elementos do mesmo anel seja superior aos custos associados.

Assim, o desenho que se obtém em cada uma das instâncias apresentadas reflete as características dos nove cenários utilizados.

Para além da análise feita individualmente para cada uma das heurísticas construtivas, é importante fazer uma comparação entre estas, para perceber as vantagens e desvantagens de cada uma.

Ao comparar os resultados obtidos quanto à média obtida pelas cinco *runs* de cada uma das instâncias, obtém-se melhores soluções, quanto ao valor médio dos lucros obtidos na heurística construtiva ponderada quando comparada com a heurística construtiva aleatória. Contudo, quando o valor máximo do lucro obtido, os melhores encontram-se na heurística construtiva aleatória. Assim, não é possível retirar uma conclusão, nesta fase da experiência computacional, sobre a melhor heurística construtiva a aplicar.

A nível de rapidez no desempenho, de acordo com os tempos médios de CPU apresentados nas tabelas 4.2.1. e 4.2.2. podemos concluir que a heurística construtiva de inserção ponderada é menos

eficiente do que a heurística construtiva de inserção aleatória. Os valores médios de CPU obtidos nas soluções apresentadas explicam-se pela análise mais cuidada e profunda que é feita para decidir que clientes facultativos a inserir, na heurística construtiva de inserção ponderada do que a heurística construtiva de inserção aleatória, como explicado no capítulo anterior.

4.3. Heurística de melhoramento

Nesta seção analisa-se as soluções da heurística de melhoramento proposta para o conjunto de instâncias geradas.

Nesta dissertação decidiu-se considerar como solução inicial as soluções obtidas com cada uma das heurísticas construtivas.

Os resultados apresentados para esta heurística são calculados e apresentados em *gaps*, com base nos valores dos lucros obtidos na heurística de melhoramento em relação a um valor de referência, com o objetivo de analisar e interpretar melhor a heurística corrente. O cálculo de cada *gap* é feito da seguinte forma:

$$gap (\%) = \frac{ValorReferência - ValorObtido}{|ValorReferência|} * 100 \quad (4.3.1.)$$

Para os valores de referência considerou-se o melhor valor entre as heurísticas construtivas, que se encontram nas tabelas 4.2.1. e 4.2.2.. Assim, para cada uma das nove instâncias consideradas, o valor referência é calculado da seguinte forma:

$$ValorReferência_{Instância} = \text{Máximo}(\text{LucroMáximo}_{\text{HeurísticaConstrutivaInserçãoPonderada}} ; \text{LucroMáximo}_{\text{HeurísticaConstrutivaInserçãoAleatória}}) \quad (4.3.2.)$$

De acordo com a definição do valor de referência para cada instância apresentado na equação anterior, apresentada como 4.3.2., obtiveram-se os seguintes valores:

Tabela 4.3.1. – Valores de Referência para o cálculo dos gaps

Instância	Valor Referência
20 nodos_n19_m3_b4	-150
30 nodos_n29_m3_b5	26323
50 nodos_n49_m3_b9	27002
70 nodos_n69_m3_b12	71231
100 nodos_n99_m3_b17	226553
136 nodos_n135_m3_b23	310477
152 nodos_n151_m3_b26	277271
202 nodos_n201_m3_b34	585544
264 nodos_n263_m3_b44	1446708

De seguida, calculou-se os valores de *gap* definido pela equação 4.3.1, para cada uma das cinco *runs*. Para a heurística de melhoramento testada, considerou-se como solução inicial cada uma das

soluções obtidas nas heurísticas construtivas testadas anteriormente. Desta forma, para cada instância considerada são obtidos cinco resultados como continuação das soluções anteriormente apresentadas.

De seguida, são apresentados nas tabelas 4.3.2. e 4.3.3., os resultados obtidos da heurística de melhoramento, com soluções iniciais da heurística construtiva de inserção ponderada e da heurística construtivas de inserção aleatória, respetivamente.

Em cada uma das seguintes tabelas são apresentados os valores mínimos, máximos e médio de gaps que são obtidos pelas soluções obtidas nos cinco *runs*, em cada uma das instâncias. Para além destes três indicadores, é apresentada a média dos valores gaps obtidos pelas cinco *runs* de cada instância, e ainda a média dos tempos CPU obtidos.

Por fim, encontram-se as médias dos valores de cada uma das colunas presentes nas tabelas: para o gap mínimo, associado ao melhor valor lucro encontrado, pelo que quanto mais negativo for o valor de gap melhor é o valor do lucro obtido, para o gap máximo associado à pior solução encontrada, para o gap médio, para o desvio padrão do gap e ainda para a média CPU.

Os valores dos gaps serão apresentados em percentagem, e os valores de CPU em segundos. Os resultados detalhados, para cada um dos cinco testes por instância, são apresentados no anexo B.

Tabela 4.3.2.- Resultados da Heurística de melhoramento com soluções iniciais da heurística construtiva de inserção ponderada

Instância	Mínimo Gap	Máximo Gap	Média Gap	Média CPU
20 nodos_n19_m3_b4	-510%	-510%	-510%	0
30 nodos_n29_m3_b5	-10%	-6%	-8%	0
50 nodos_n49_m3_b9	-4%	9%	-1%	0
70 nodos_n69_m3_b12	-20%	33%	9%	0
100 nodos_n99_m3_b17	-17%	-13%	-14%	1
136 nodos_n135_m3_b23	25%	28%	27%	3
152 nodos_n151_m3_b26	-15%	-10%	-12%	7
202 nodos_n201_m3_b34	-70%	-3%	-56%	32
264 nodos_n263_m3_b44	-8%	-7%	-8%	46
Média	-70%	-53%	-64%	10

Tabela 4.3.3. - Resultados da Heurística de melhoramento com soluções iniciais de Heurística construtiva de inserção aleatória

Instância	Mínimo Gap	Máximo Gap	Média Gap	Média CPU
20 nodos_n19_m3_b4	-510%	899%	-228%	0
30 nodos_n29_m3_b5	-17%	-9%	-13%	0
50 nodos_n49_m3_b9	-7%	35%	10%	0
70 nodos_n69_m3_b12	-32%	29%	4%	0
100 nodos_n99_m3_b17	-18%	28%	9%	2
136 nodos_n135_m3_b23	-17%	32%	7%	1
152 nodos_n151_m3_b26	-25%	31%	11%	1
202 nodos_n201_m3_b34	-72%	-3%	-20%	18
264 nodos_n263_m3_b44	-11%	28%	12%	37
Média	-79%	119%	-23%	6

Esta análise tem por objetivo observar a qualidade das soluções da heurística de melhoramento e também perceber a influência que a solução inicial pode ter, isto é, qual a melhor heurística construtiva a considerar, pelo que a comparação entre as percentagens obtidas de gaps será direta entre as mesmas instâncias.

Como se pode observar existem muitos valores positivos na coluna dos gaps máximos, devido ao valor anterior do teste onde foi obtido, pois apesar de ter sido melhor quando comparado com o valor obtido na heurística construtiva correspondente, não foi melhor do que o valor considerado para referência. Por exemplo, para a instância de 20 nodos, o valor máximo de gap obtido é de 899%. Este valor foi obtido no teste 4, apresentado nos anexos A e B, onde alcança inicialmente o valor de -2482 pela heurística construtiva de inserção aleatória, e é melhorado pela heurística de melhoramento, onde consegue alcançar o valor de -1498. Apesar disso, quando comparado com o valor de referência, -150, não é suficiente.

De forma a visualizar melhor os resultados, segue-se um gráfico comparativo entre os resultados de valores médios de gaps obtidos:

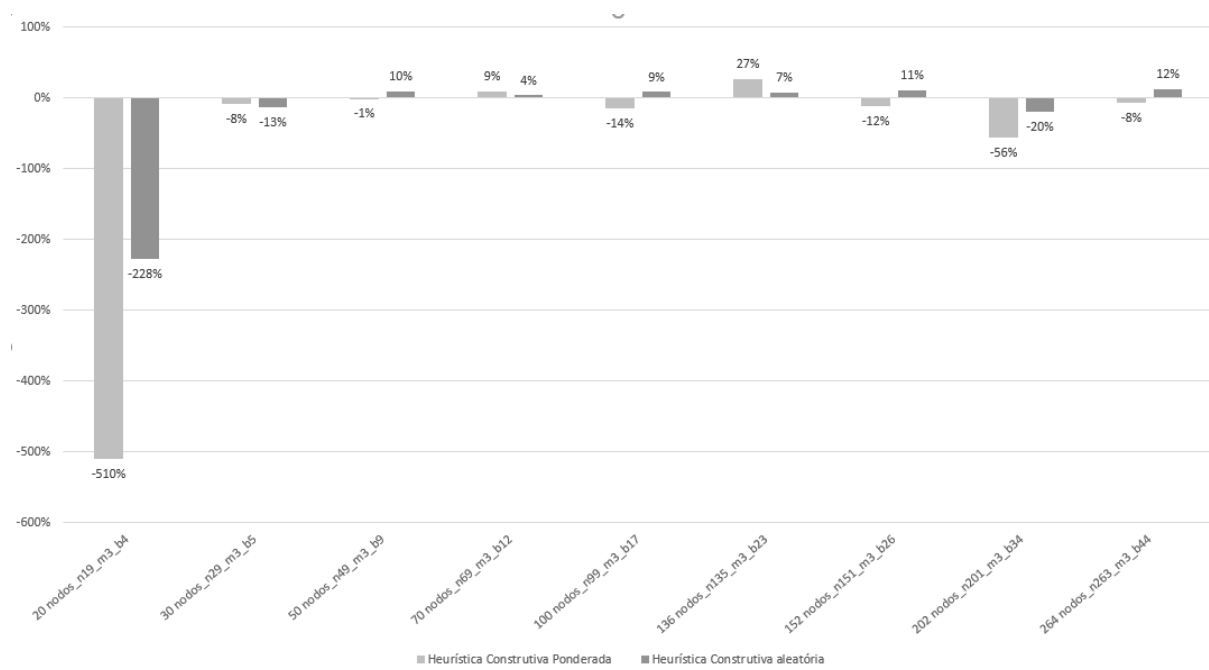


Figura 4.3.1. – Gráfico Comparativo de valores médios de gap obtidos na heurística de melhoramento com soluções iniciais das heurísticas construtivas

Na avaliação dos valores médios é possível perceber que na sua maioria, a heurística construtiva ponderada, apresenta melhores valores para a heurística construtiva aleatória.

Porém, para uma análise mais completa é necessário avaliar os valores mínimos de gaps, isto é, os melhores valores obtidos entre os testes feitos:

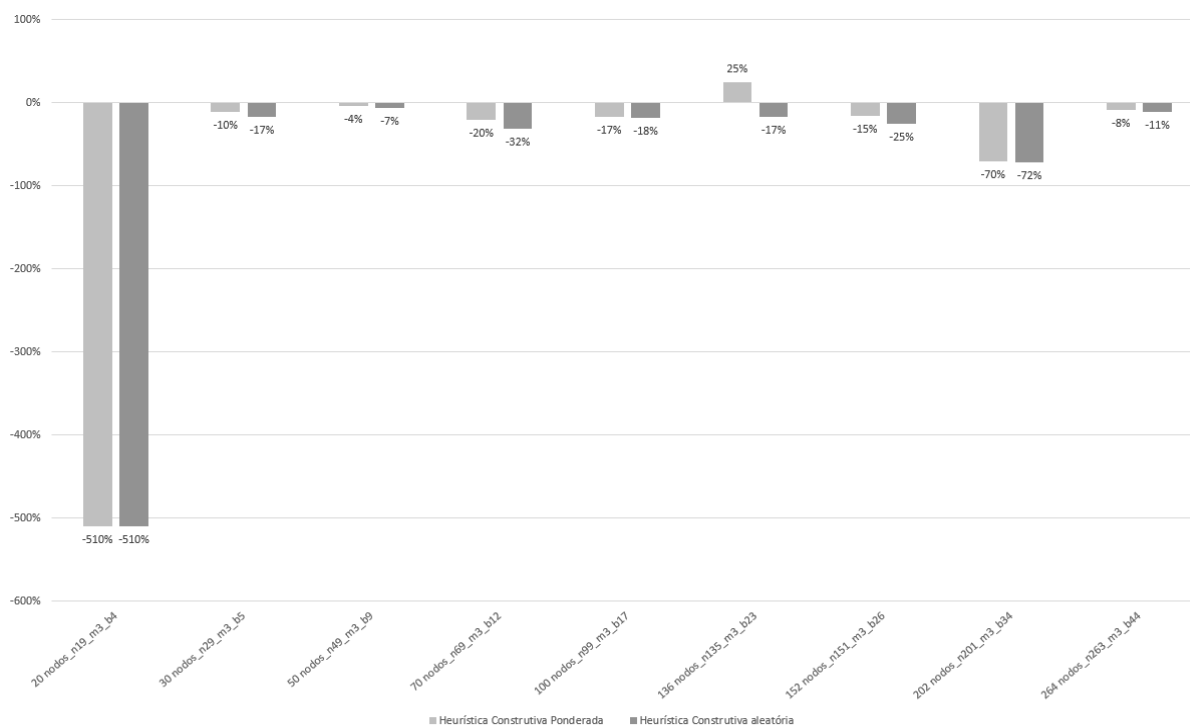


Figura 4.3.2. – Gráfico Comparativo de valores mínimos de gap obtidos na heurística de melhoramento com soluções iniciais das heurísticas construtivas

De acordo com a figura 4.3.2., os melhores resultados obtidos têm como solução inicial as soluções da heurística construtiva aleatória.

Nos gráficos e tabelas anteriormente apresentadas para qualquer valor apresentado para a instância de 20 nodos, gap mínimo, máximo ou médio, quando comparados com os valores das restantes instâncias, são bastante menores. Tal acontece devido ao valor de referência considerado para a instância de 20 nodos ser bastante mais baixa do que os restantes.

Por esta razão, apresentam-se as médias de todas as soluções apresentadas anteriormente, obtida por todas as instâncias dos 30 aos 264 nodos, ficando excluída dos seguintes resultados a instâncias de 20 nodos.

Tabela 4.3.4. - Resultados em média da Heurística de melhoramento com soluções iniciais em cada uma das Heurísticas construtivas, para todas as instâncias, à exceção da instância de 20 nodos

Heurística Inicial	Mínimo Gap	Máximo Gap	Média Gap	Média CPU
Heurística construtiva de inserção ponderada	-36%	-31%	-34%	322
Heurística construtiva de inserção aleatória	-40%	-31%	-36%	1186

De acordo com a tabela 4.3.4, os valores obtidos apesar de não serem tão discrepantes, levam à mesma conclusão retirada anteriormente, onde os melhores resultados são obtidos na heurística construtiva de inserção aleatória, porém em média os resultados obtidos pela heurística construtiva de inserção ponderada apresentarem valores melhores.

Os tempos médios de ambas as heurísticas, são bastante baixos até à instância de 152 nodos incluída. A partir dessa instância, os tempos começam a ser maiores, mas igualmente eficientes face à dimensão que a heurística de melhoramento enfrenta. Quando comparados os tempos entre os resultados da heurística de melhoramento, com soluções iniciais das heurísticas construtivas, este é superior quando o ponto de partida são as soluções obtidas da heurística construtiva ponderada.

Assim, de acordo com esta análise a heurística construtiva ponderada apresenta os melhores resultados em média, porém a heurística construtiva aleatória obteve melhores resultados entre todos os obtidos, pelo que não é possível perceber qual das heurísticas construtivas é a melhor. Assim, os próximos testes serão sobre ambas na próxima etapa que é a meta-heurística.

4.4. Pesquisa Local iterada

Por fim, a meta-heurística escolhida nesta dissertação é testada, isto é, a pesquisa local iterada.

Nestes testes, para o número de iterações foram considerados 50 iterações. Esta meta-heurística e o seu algoritmo encontram-se explicados no capítulo anterior.

De modo a poder comparar os resultados obtidos na meta-heurística com os resultados obtidos nas restantes heurísticas, foram calculados gaps de acordo com a equação apresentada por 4.3.1. e os valores de referência utilizados foram também os anteriormente apresentados na tabela 4.3.1..

De seguida, são apresentados os resultados obtidos pela pesquisa local iterada, nas tabelas 4.4.1. e 4.4.2., com soluções iniciais de cada uma das heurísticas construtivas.

As seguintes tabelas apresentam os resultados da mesma forma que a heurística de melhoramento, explicados anteriormente. Porém, terão mais um indicador, a média das iterações onde foi encontrada a solução apresentada em cada uma das 5 runs para cada instância. Os resultados não sintetizados encontram-se em anexo C, para consulta.

Tabela 4.4.1 - Resultados da Meta-heurística, correspondentes à heurística construtiva de inserção ponderada, com 50 iterações

Instância	Mínimo Gap	Máximo Gap	Média Gap	Média CPU	Média Iterações
20nodos_n19_m3_b4	-3470%	-3106%	-3277%	0	32,4
30nodos_n29_m3_b5	-19%	-15%	-17%	1	39,8
50nodos_n49_m3_b9	-39%	-26%	-35%	2	33
70nodos_n69_m3_b12	-32%	-20%	-26%	7	14,6
100nodos_n99_m3_b17	-21%	-19%	-20%	36	36,4
136nodos_n135_m3_b23	-19%	-17%	-18%	108	26,4
152nodos_n151_m3_b26	-76%	-71%	-74%	77	27
202nodos_n201_m3_b34	-75%	-73%	-74%	1436	41,6
264nodos_n263_m3_b44	-11%	-10%	-10%	907	27,4
Média	-418%	-373%	-395%	286	31

Tabela 4.4.2 - Resultados da Meta-heurística, correspondentes à Heurística construtiva de inserção aleatória, com 50 iterações

Instância	Mínimo Gap	Máximo Gap	Média Gap	Média CPU	Média Iterações
20nodos_n19_m3_b4	-4565%	-2845%	-3500%	0	26,4
30nodos_n29_m3_b5	-19%	-17%	-17%	1	11,4
50nodos_n49_m3_b9	-37%	-14%	-29%	2	40,2
70nodos_n69_m3_b12	-41%	-24%	-33%	10	29
100nodos_n99_m3_b17	-21%	-18%	-20%	75	34,2
136nodos_n135_m3_b23	-25%	-22%	-23%	84	39
152nodos_n151_m3_b26	-88%	-77%	-82%	236	20,6
202nodos_n201_m3_b34	-74%	-70%	-72%	1454	32,4
264nodos_n263_m3_b44	-14%	-10%	-13%	7625	29,6
Média	-543%	-344%	-421%	1054	29

Na análise aos resultados apresentados nas tabelas 4.4.1. e 4.4.2., é importante como já referido perceber bem os resultados de forma a comparar:

- Os resultados obtidos entre as diferentes soluções iniciais (perceber qual será a heurística construtiva mais vantajosa a utilizar);
- Eficácia da meta-heurística face aos resultados obtidos pela heurística de melhoria;
- Eficiência de resposta da meta-heurística, quanto ao tempo médio por instância;

Começa-se por comparar os resultados apresentados nas duas tabelas anteriores, para cada instância. Nas tabelas anteriores é possível observar que os melhores valores são obtidos nas soluções da meta-heurística com início na heurística construtiva de inserção aleatória para seis das nove instâncias utilizadas, para além disso houve ainda um empate entre os resultados apresentados nas tabelas 4.4.1 e 4.4.2. para duas instâncias e por fim a meta-heurística com início na heurística construtiva de inserção ponderada conseguiu a melhor solução para a instância de 202 nodos.

Esta análise permite perceber que os melhores valores apresentados na coluna de “Mínimo Gap” estão presentes na sua maioria partindo da solução inicial obtida através heurística construtiva aleatória, pois quanto menor for o valor do gap melhor é a solução encontrada.

Ao analisar ainda, a coluna com a média dos valores de *gap* obtidos e apresentados anteriormente, as observações feitas para a coluna do *gap* mínimo voltam a repetir-se: os melhores valores são obtidos nas soluções da meta-heurística com início na heurística construtiva de inserção aleatória para cinco das nove instâncias utilizadas, houve um empate entre os resultados em duas instâncias e por fim a meta-heurística com início na heurística construtiva de inserção ponderada conseguiu em média melhor valor de lucro para a instância de 50 e ainda 202 nodos.

Assim, os valores presentes na coluna “média Gap” também se encontram na segunda tabela, apresentada por 4.4.2., ou seja, com início na heurística construtiva aleatória.

De acordo, com as médias apresentadas nas tabelas 4.4.1. e 4.4.2., é clara a análise feita anteriormente. Assim, para a meta-heurística implementada e testada, é possível concluir que apesar dos bons resultados apresentados para qualquer uma das heurísticas construtivas desenvolvidas, a solução inicial que devolveu melhores soluções é a heurística construtiva de inserção aleatória. Quanto à média

obtida de tempos de CPU apresentadas nas tabelas 4.4.3. e 4.4.4. é ainda importante referir que a eficiência de desempenho foi bastante melhor partindo da solução inicial obtida através heurística construtiva ponderada, porém, uma vez que o objetivo para a resolução do problema é a maximização do lucro, e dado que o desenho da estrutura é algo que será com vista num desenho praticamente definitivo este parâmetro apresentado não predomina quando comparado nos valores que o lucro pode alcançar.

Para comparação de resultados obtidos na heurística de melhoramento e na meta-heurística implementadas, segue-se um gráfico onde estão apresentados apenas os valores médios apresentados nas tabelas 4.3.3. e 4.4.2., uma vez que se concluiu que a melhor heurística inicial a considerar deverá ser a construtiva de inserção aleatória.

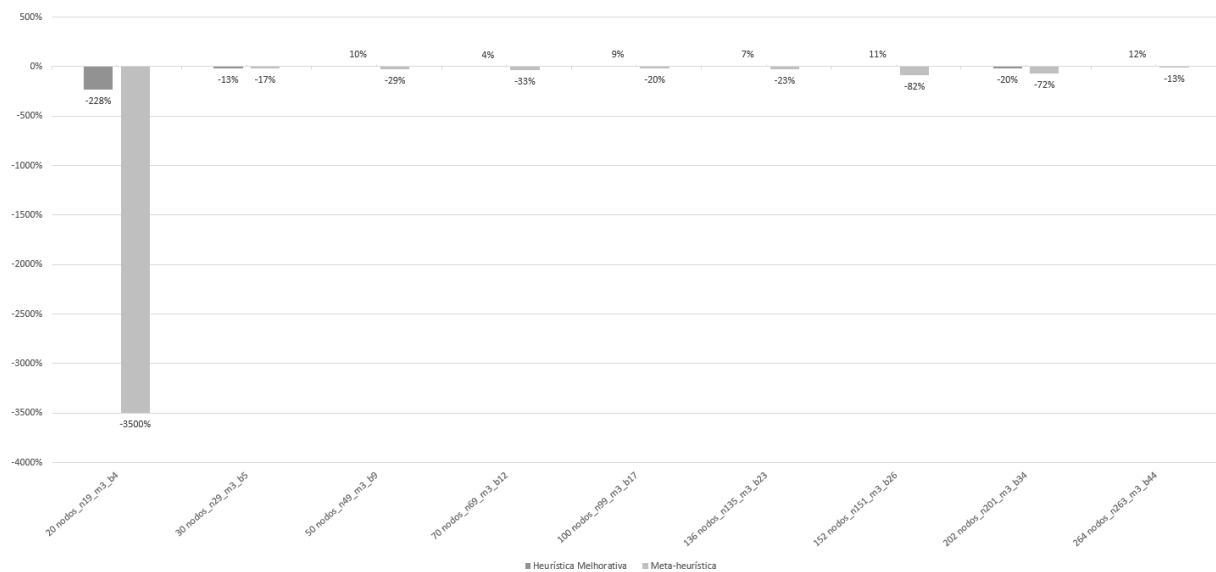


Figura 4.4.1. – Gráfico Comparativo de valores médios de gap obtidos na heurística de melhoramento e na meta-heurística, com referência inicial na construtiva de inserção aleatória

Na figura 4.4.1., os valores médios calculados com base nos valores de referência apresentados na tabela 4.3.1., são mais benéficos para a meta-heurística do que para a heurística de melhoramento implementadas, de acordo com o objetivo do problema, a maximização do valor do lucro. Uma vez mais, é possível reparar a discrepância nos valores da instância de 20 nodos para os restantes, pela mesma razão referida na análise às soluções obtidas na heurística de melhoramento, ou seja, devido ao valor de referência considerado.

A melhoria da meta-heurística face à heurística de melhoramento era espectável, porém não se esperava uma melhoria tão acentuada, pelo que prova uma vez mais, as boas referências que esta tem pela sua eficiência.

Assim, pode-se concluir que o desempenho e resolução do SmRNDP, a meta-heurística escolhida nesta dissertação consegue bons resultados. Apesar da heurística de melhoramento não conseguir tão bons resultados, é também com a sua ajuda que a eficiência desta meta-heurística é possível, uma vez que a pesquisa local utilizada na pesquisa local iterada desta dissertação tem por base o algoritmo da heurística de melhoramento, explicado no capítulo anterior.

Em relação aos tempos médios de CPU obtidos na heurística de melhoramento e na meta-heurística, o desempenho é muito mais eficiente na heurística de melhoramento, porém, dada a melhoria significativa no valor do lucro obtida na meta-heurística testada, este parâmetro não terá impacto na decisão de qual algoritmo é melhor para a resolução do problema, pelas mesmas razões anteriormente mencionadas.

Para uma análise à relação entre os tempos de CPU obtidos e a instâncias consideradas, segue-se o gráfico:

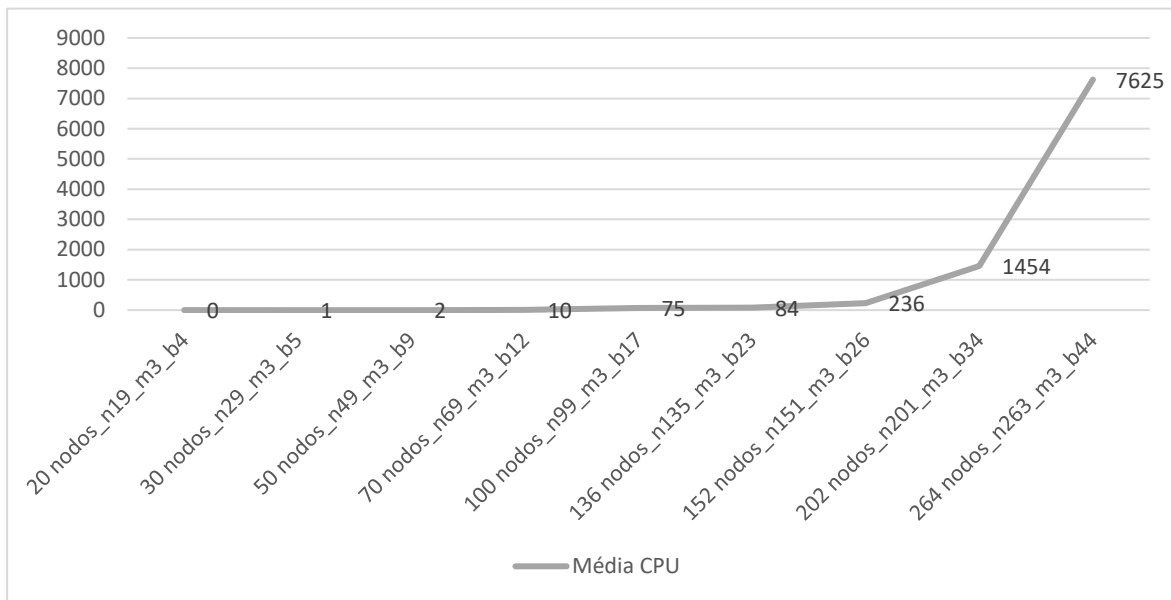


Figura 4.4.2. – Gráfico com os tempos médios de CPU obtidos na meta-heurística e apresentados na tabela 4.4.2., por cada instância considerada

De acordo com a figura 4.4.2, é possível observar que o crescimento do valor médio de CPU por instância é exponencial, o que significa que quanto maior for a instância, isto é, quantos mais nodos a instância tiver mais lenta será a resposta da meta-heurística. Contudo, por tudo o que já foi referido, essa espera valerá a pena, dado os valores e potencial que a solução encontrada irá oferecer.

4.4.1. Teste ao número de Iterações

Apesar das soluções obtidas terem sido bastante boas, decidiu-se explorar o impacto que o número de iterações teria na qualidade das mesmas. Assim, terminou-se esta dissertação com testes ao aumento do número de iterações com o objetivo de avaliar a qualidade das soluções em relação ao tempo que estas demoram a ser obtidas.

As soluções obtidas pela pesquisa local iterada, com 50 iterações, sintetizadas na tabela 4.4.2., é possível observar uma coluna correspondente à “Média de Iterações”, que corresponde à média das 5 *runs* de iterações onde a solução apresentada foi obtida. Ao observar a média geral de todas as soluções obtidas nesses testes as soluções foram obtidas, em média, na 29ª iteração.

De forma a perceber qual seria o impacto no aumento de número de iterações a considerar na meta-heurística decidiu-se começar por testar para cinco instâncias, as que têm 20, 30, 50, 70 e 100 nodos, com os seguintes valores de iterações: 200, 500, 1000, 2000 e 3000.

Estes testes tiveram como solução inicial as soluções, obtidas através das duas heurísticas construtivas, de forma a perceber se a análise feita quanto ao desempenho e resultados obtidos em cada uma delas.

Para a apresentação dos resultados seguem-se tabelas que terão o mesmo registo que as tabelas anteriormente apresentadas para as soluções da meta-heurística. Os resultados não sintetizados encontram-se em anexo para consulta D, E, F, G e H.

Tabela 4.4.1.1. - Resultados da Meta-heurística, instâncias de 20, 30, 50, 70 e 100 nodos correspondentes à heurística construtiva de inserção ponderada, com 200 iterações

Instância	Mínimo Gap	Máximo Gap	Média Gap	Média CPU	Média Iterações
20nodos_n19_m3_b4	-3763%	-3461%	-3644%	1	81,6
30nodos_n29_m3_b5	-19%	-18%	-18%	2	109,2
50nodos_n49_m3_b9	-49%	-25%	-40%	6	135
70nodos_n69_m3_b12	-33%	-26%	-30%	28	137,8
100nodos_n99_m3_b17	-22%	-19%	-21%	183	84,8
Média	-777%	-710%	-750%	44	110

Tabela 4.4.1.2 - Resultados da Meta-heurística, instâncias de 20, 30, 50, 70 e 100 nodos correspondentes à heurística construtiva de inserção aleatória, com 200 iterações

Instância	Mínimo Gap	Máximo Gap	Média Gap	Média CPU	Média Iterações
20nodos_n19_m3_b4	-4904%	-3285%	-3812%	1	91,8
30nodos_n29_m3_b5	-19%	-17%	-18%	2	71,8
50nodos_n49_m3_b9	-44%	-23%	-37%	8	98
70nodos_n69_m3_b12	-44%	-35%	-40%	45	142
100nodos_n99_m3_b17	-22%	-21%	-22%	322	143
Média	-1007%	-676%	-786%	76	109

Tabela 4.4.1.3. - Resultados da Meta-heurística, instâncias de 20, 30, 50, 70 e 100 nodos correspondentes à heurística construtiva de inserção ponderada, com 500 iterações

Instância	Mínimo Gap	Máximo Gap	Média Gap	Média CPU	Média Iterações
20nodos_n19_m3_b4	-3763%	-3762%	-3763%	3	297,6
30nodos_n29_m3_b5	-19%	-18%	-19%	8	179
50nodos_n49_m3_b9	-52%	-25%	-44%	22	321,2
70nodos_n69_m3_b12	-38%	-31%	-35%	99	307
100nodos_n99_m3_b17	-22%	-19%	-21%	624	334,6
Média	-779%	-771%	-776%	151	288

Tabela 4.4.1.4 - Resultados da Meta-heurística, instâncias de 20, 30, 50, 70 e 100 nodos correspondentes à heurística construtiva de inserção aleatória, com 500 iterações

Instância	Mínimo Gap	Máximo Gap	Média Gap	Média CPU	Média Iterações
20 nodos_n19_m3_b4	-4904%	-3763%	-3991%	4	218,2
30 nodos_n29_m3_b5	-19%	-19%	-19%	5	208,2
50 nodos_n49_m3_b9	-49%	-27%	-42%	31	233
70 nodos_n69_m3_b12	-45%	-28%	-41%	183	299,6
100 nodos_n99_m3_b17	-24%	-21%	-23%	722	402,4
Média	-1008%	-772%	-823%	189	272

Tabela 4.4.1.5. - Resultados da Meta-heurística, instâncias de 20, 30, 50, 70 e 100 nodos correspondentes à heurística construtiva de inserção ponderada, com 1000 iterações

Instância	Mínimo Gap	Máximo Gap	Média Gap	Média CPU	Média Iterações
20nodos_n19_m3_b4	-3763%	-3470%	-3704%	4	146,8
30nodos_n29_m3_b5	-20%	-18%	-19%	9	154,8
50nodos_n49_m3_b9	-50%	-29%	-44%	21	453,8
70nodos_n69_m3_b12	-38%	-30%	-34%	240	586,6
100nodos_n99_m3_b17	-22%	-20%	-21%	1328	711,4
Média	-779%	-713%	-764%	321	411

Tabela 4.4.1.6 - Resultados da Meta-heurística, instâncias de 20, 30, 50, 70 e 100 nodos à heurística construtiva de inserção aleatória, com 1000 iterações

Instância	Mínimo Gap	Máximo Gap	Média Gap	Média CPU	Média Iterações
20nodos_n19_m3_b4	-4904%	-3762%	-3991%	6	188,4
30nodos_n29_m3_b5	-19%	-19%	-19%	9	152,6
50nodos_n49_m3_b9	-50%	-27%	-42%	58	341
70nodos_n69_m3_b12	-44%	-33%	-38%	256	646,8
100nodos_n99_m3_b17	-23%	-21%	-22%	1275	728,2
Média	-1008%	-772%	-822%	321	411

Tabela 4.4.1.7. - Resultados da Meta-heurística, instâncias de 20, 30, 50, 70 e 100 nodos correspondentes à heurística construtiva de inserção ponderada, com 2000 iterações

Instância	Mínimo Gap	Máximo Gap	Média Gap	Média CPU	Média Iterações
20nodos_n19_m3_b4	-3763%	-3763%	-3763%	9	298,8
30nodos_n29_m3_b5	-23%	-19%	-20%	18	199,2
50nodos_n49_m3_b9	-60%	-50%	-54%	39	857
70nodos_n69_m3_b12	-42%	-33%	-36%	380	1293,8
100nodos_n99_m3_b17	-22%	-21%	-22%	2218	1334,8
Média	-782%	-777%	-779%	533	797

Tabela 4.4.1.8 - Resultados da Meta-heurística, instâncias de 20, 30, 50, 70 e 100 nodos correspondentes à heurística construtiva de inserção aleatória, com 2000 iterações

Instância	Mínimo Gap	Máximo Gap	Média Gap	Média CPU	Média Iterações
20nodos_n19_m3_b4	-4904%	-3762%	-3991%	12	137,6
30nodos_n29_m3_b5	-25%	-19%	-20%	18	722,6
50nodos_n49_m3_b9	-52%	-28%	-44%	54	1197,2
70nodos_n69_m3_b12	-43%	-33%	-39%	512	1053,4
100nodos_n99_m3_b17	-24%	-22%	-23%	3942	983
Média	-1010%	-773%	-823%	907	819

Tabela 4.4.1.9 - Resultados da Meta-heurística, instâncias de 20, 30, 50, 70 e 100 nodos correspondentes à heurística construtiva de inserção ponderada, com 3000 iterações

Instância	Mínimo Gap	Máximo Gap	Média Gap	Média CPU	Média Iterações
20nodos_n19_m3_b4	-3763%	-3763%	-3763%	12	674,6
30nodos_n29_m3_b5	-26%	-19%	-21%	25	2088,4
50nodos_n49_m3_b9	-54%	-35%	-49%	135	1633,6
70nodos_n69_m3_b12	-41%	-35%	-37%	334	1378,4
100nodos_n99_m3_b17	-23%	-22%	-23%	3411	1914
Média	-781%	-775%	-778%	783	1538

Tabela 4.4.1.10 - Resultados da Meta-heurística, instâncias de 20, 30, 50, 70 e 100 nodos correspondentes à heurística construtiva de inserção aleatória, com 3000 iterações

Instância	Mínimo Gap	Máximo Gap	Média Gap	Média CPU	Média Iterações
20nodos_n19_m3_b4	-4904%	-3763%	-3991%	16	359,6
30nodos_n29_m3_b5	-25%	-19%	-23%	27	1919,4
50nodos_n49_m3_b9	-75%	-31%	-50%	206	1207
70nodos_n69_m3_b12	-43%	-36%	-39%	826	1633,4
100nodos_n99_m3_b17	-25%	-21%	-23%	4678	2052,8
Média	-1014%	-774%	-825%	1151	1434

A primeira análise comparativa entre as dez tabelas anteriormente apresentadas é quanto aos valores médios dos gaps obtidos. Esses valores são maioritariamente melhores para todas as instâncias testadas, quando a heurística construtiva inicial utilizada é a heurística construtiva aleatória, como se concluiu anteriormente. Por esta razão, para testes e análises futuras será esta heurística construtiva a considerar, pelo que para a ponderada o estudo fica terminado.

De seguida, procede-se à comparação das instâncias com as iterações aplicadas na meta-heurística. Assim, para as instâncias apresentadas nas tabelas acima, analisa-se o número de iterações mais benéficas em cada uma delas:

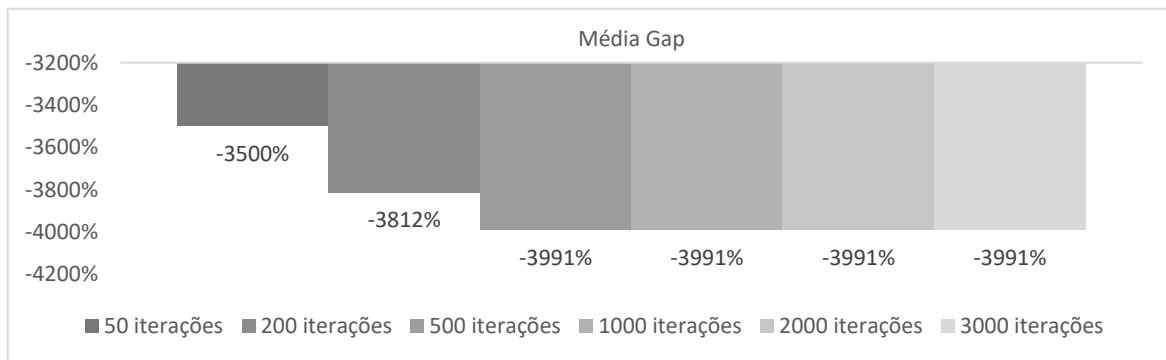


Figura 4.4.1.1. - Gráfico Comparativo de valores médios de gap obtidos pela aplicação de diferentes iterações na meta-heurística, com referência inicial na construtiva de inserção aleatória para a instância de 20 nodos

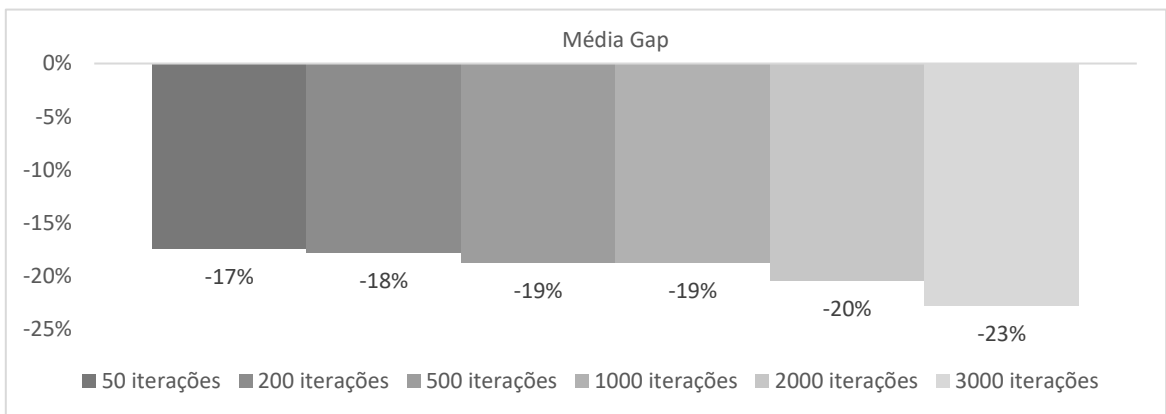


Figura 4.4.1.2. - Gráfico Comparativo de valores médios de gap obtidos pela aplicação de diferentes iterações na meta-heurística, com referência inicial na construtiva de inserção aleatória para a instância de 30 nodos

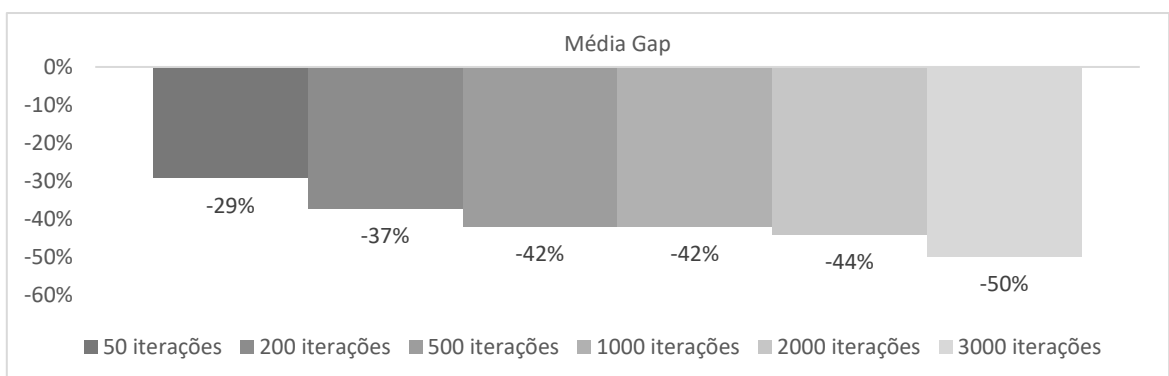


Figura 4.4.1.3. - Gráfico Comparativo de valores médios de gap obtidos pela aplicação de diferentes iterações na meta-heurística, com referência inicial na construtiva de inserção aleatória para a instância de 50 nodos

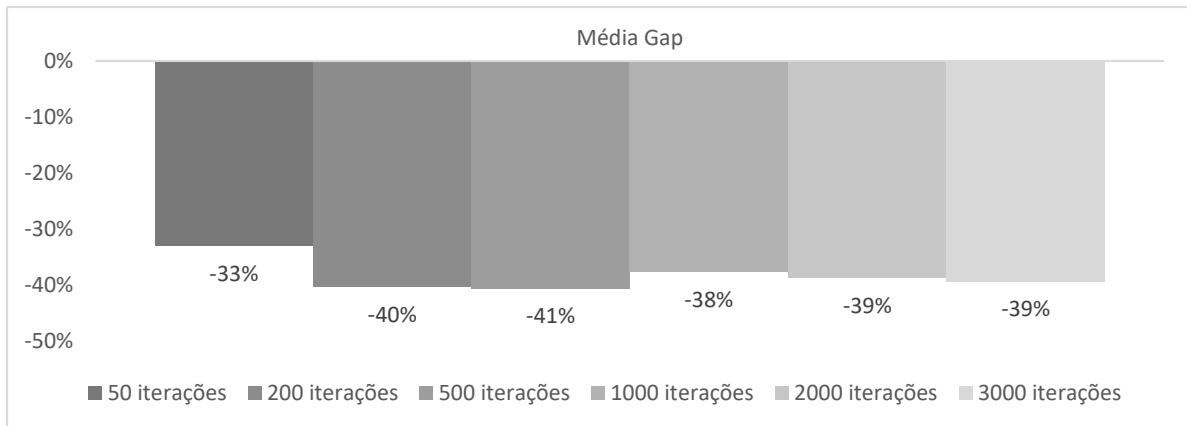


Figura 4.4.1.4. - Gráfico Comparativo de valores médios de gap obtidos pela aplicação de diferentes iterações na meta-heurística, com referência inicial na construtiva de inserção aleatória para a instância de 70 nodos

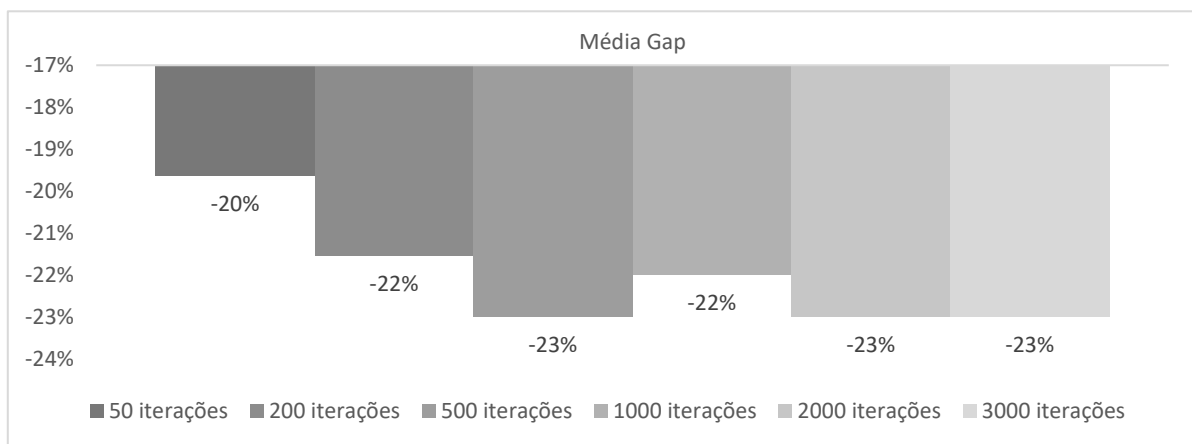


Figura 4.4.1.5. - Gráfico Comparativo de valores médios de gap obtidos pela aplicação de diferentes iterações na meta-heurística, com referência inicial na construtiva de inserção aleatória para a instância de 100 nodos

Com o auxílio dos cinco gráficos acima apresentados é possível perceber que existe uma melhoria constante dos valores médios de *gaps* obtidos para todas as instâncias testadas, até 500 iterações. Quando o número de iterações é aumentado, para 1000, 2000 e 3000, a média de *gaps* das soluções obtidas diminuem ou mantêm-se para a maioria das instâncias. Para os casos onde os *gaps* melhoram, isto é, diminuem, para além dessa melhoria não ser significativa, o tempo necessário para obter a solução aumenta exponencialmente.

De acordo com a análise feita, considera-se que um bom compromisso, entre a qualidade da solução e o tempo CPU necessário, é obtido para 500 iterações.

De seguida, a meta-heurística é testada para as instâncias com 136, 152, 202 e 264 nodos, com referência à heurística construtiva de inserção aleatória, pelas razões mencionadas e explicadas anteriormente. Nestes testes, a quantidade de iterações aplicada é de 200, 500 e 1000 iterações. Uma vez que na análise anterior concluiu-se que para as instâncias anteriormente testadas, o número de iterações a tomar é de 500 e como para as instâncias seguintes o número de nodos aumenta, optou-se por considerar ainda as 1000 iterações para estes novos testes.

Os resultados serão apresentados da mesma forma que as tabelas anteriores, para cada uma das instâncias. Os resultados não sintetizados encontram-se em anexo para consulta D, E e F.

Tabela 4.4.1.11 - Resultados da Meta-heurística, instâncias de 136, 152, 202 e 264 nodos correspondentes à heurística construtiva de inserção aleatória, com 200 iterações

Instância	Mínimo Gap	Máximo Gap	Média Gap	Média CPU	Média Iterações
136 nodos_n135_m3_b23	-27%	-19%	-24%	352	146,6
152 nodos_n151_m3_b26	-90%	-78%	-83%	865	119,4
202 nodos_n201_m3_b34	-75%	-71%	-73%	4807	128,6
264 nodos_n263_m3_b44	-15%	-11%	-13%	12260	180,6
Média	-52%	-45%	-48%	4571	144

Tabela 4.4.1.12 - Resultados da Meta-heurística, instâncias de 136, 152, 202 e 264 nodos correspondentes à heurística construtiva de inserção aleatória, com 500 iterações

Instância	Mínimo Gap	Máximo Gap	Média Gap	Média CPU	Média Iterações
136 nodos_n135_m3_b23	-31%	-24%	-26%	1449	332
152 nodos_n151_m3_b26	-86%	-79%	-83%	1418	363
202 nodos_n201_m3_b34	-75%	-71%	-74%	11894	339,8
264 nodos_n263_m3_b44	-15%	-11%	-13%	42400	254,2
Média	-52%	-46%	-49%	14290	322

Tabela 4.4.1.13 - Resultados da Meta-heurística, instâncias de 136, 152, 202 e 264 nodos correspondentes à heurística construtiva de inserção aleatória, com 1000 iterações

Instância	Mínimo Gap	Máximo Gap	Média Gap	Média CPU	Média Iterações
136 nodos_n135_m3_b23	-27%	-23%	-24%	2510	505,2
152 nodos_n151_m3_b26	-91%	-74%	-83%	5721	424,8
202 nodos_n201_m3_b34	-75%	-74%	-75%	34027	609,4
264 nodos_n263_m3_b44	-15%	-12%	-14%	73467	727,2
Média	-52%	-46%	-49%	28931	567

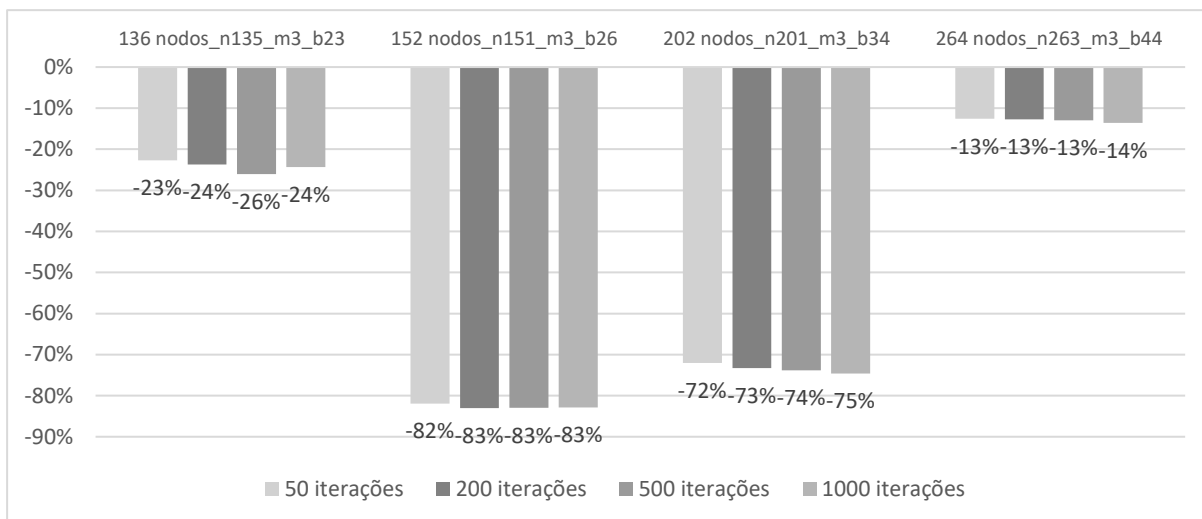


Figura 4.4.1.6. - Gráfico Comparativo de valores médios de gap obtidos pela aplicação de diferentes iterações na meta-heurística, com referência inicial na construtiva de inserção aleatória para as instâncias de 136, 152, 202 e 164 nodos

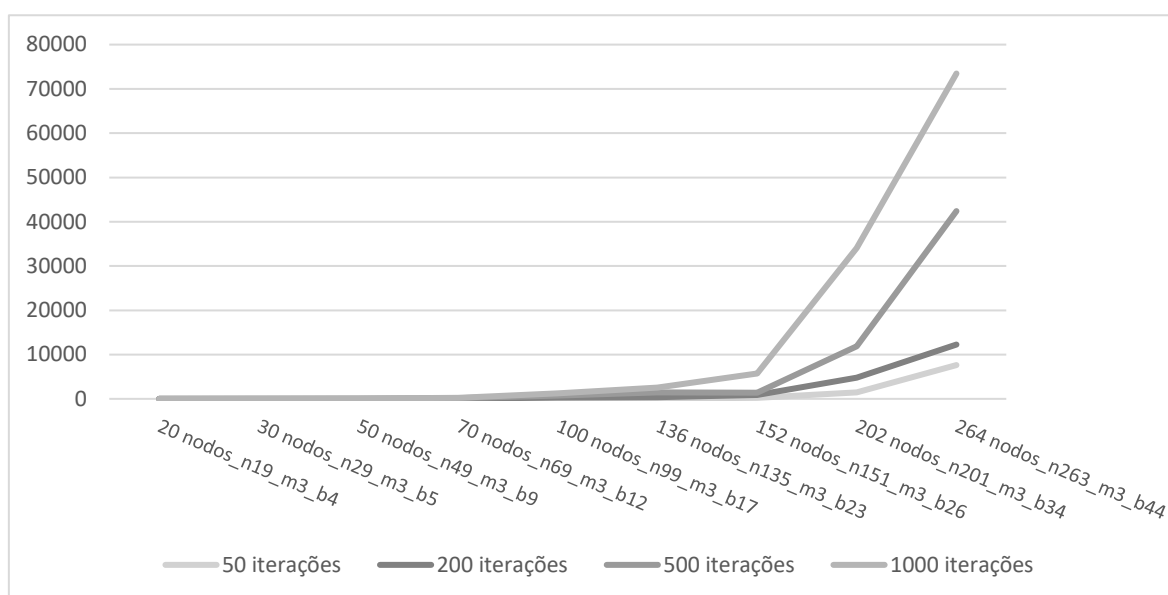


Figura 4.4.1.7. - Gráfico Comparativo de valores médios de CPU obtidos pela aplicação de diferentes iterações na meta-heurística, com referência inicial na construtiva de inserção aleatória para todas as instâncias

De acordo com as últimas três tabelas apresentadas e o gráfico 4.4.1.6., é possível fazer uma análise quanto à qualidade das soluções obtidas em relação ao número de iterações.

Para as instâncias de 136 nodos, o melhor valor médio apresentado foi conseguido com 500 iterações, enquanto que na instância de 152 nodos o valor médio apesar de não piorar à medida que se aumenta o número de iterações, também não melhora. Por fim nas últimas duas instâncias testadas, com 202 e 264 nodos, consegue-se uma melhoria quando se aumenta o número de iterações até à quantidade 1000.

De acordo com a média apresentada em cada uma das tabelas, 4.4.1.11., 4.4.1.12., conclui-se que o aumento de iterações de 200 para 500 compensa, pois para além dos valores da média melhorarem,

os valores médios de cada uma das instâncias também melhora, como referido na análise anterior. Desta forma, a escolha das 500 iterações face às 200 é benéfica.

Por fim, ao comparar os valores da média apresentada em cada uma das tabelas, 4.4.1.12., 4.4.1.13., não se consegue uma melhoria nos gaps apresentados. Para além disso, o tempo apresentado no gráfico 4.4.1.7. revela que o tempo aumenta exponencialmente com o aumento das iterações. Assim, para 1000 iterações o tempo apresentado é muito maior quando comparado com o tempo para 500 iterações.

Por estas razões conclui-se que apesar das instâncias neste estudo terem dimensões maiores, o melhor número de iterações a aplicar será 500, tal como nas instâncias anteriores analisadas.

5. Conclusões

No último do capítulo desta dissertação, apresenta-se um resumo de todo o trabalho desenvolvido e ainda as conclusões finais do estudo feito ao SmRNDP.

O problema desenvolvido nesta dissertação, consiste no desenho de uma estrutura em anel. Este desenho tem definido o número de anéis que este deve ter e ainda um conjunto de clientes que é dividido em dois conjuntos, de acordo com a prioridade atribuída a cada um, alta prioridade ou baixa prioridade, respeitando a capacidade máxima de cada anel.

Esta dissertação tem por objetivo a resolução do SmRNDP para instâncias de dimensões maiores, uma vez que até hoje, de acordo com os trabalhos publicados, só era possível para instâncias com um máximo de 41 clientes. Para além disso, procura-se alcançar a maximização do lucro obtido no desenho, que é obtido pelos custos associados a cada ligação assim como o proveito gerado pela combinação dos clientes presentes em cada anel.

Para a resolução do problema em estudo, começou-se pela implementação de duas heurísticas construtivas, que se intitulam de: heurística construtiva ponderada e heurística construtiva aleatória. O que difere as duas heurísticas construtivas é a forma como se escolhe os clientes obrigatórios, isto é, clientes com alta prioridade.

De acordo com as soluções obtidas e que se encontram em anexo em A, cada uma das heurísticas construtivas apresentam benefícios quando comparadas entre si: a heurística construtiva ponderada apresenta melhores valores nas médias calculadas a partir das suas soluções, porém a heurística construtiva aleatória apresenta melhores valores entre todas as soluções obtidas das cinco *runs* efetuadas para cada uma das heurísticas construtivas. As heurísticas construtivas mostraram ser bastante eficientes na resposta ao problema. Apesar da heurística construtiva ponderada ser menos rápida do que a heurística construtiva aleatória, ambas apresentam valores médios do tempo necessário na procura das soluções bastante satisfatórias.

Depois de obtidas as soluções das heurísticas construtivas, estas serão utilizadas nesta dissertação como pontos de partida, para seguinte heurística a implementar, a heurística de melhoramento. Nesta heurística o objetivo será explorar as melhorias possíveis ao desenho obtido anteriormente, quanto à disposição dos clientes em cada um dos anéis e nos anéis do desenho inicial. Para além disso, é feito um estudo na escolha, ou falta dela, dos melhores clientes facultativos, isto é, de baixa prioridade, que são mais benéficos à solução.

As soluções obtidas na heurística de melhoramento, estão apresentadas no anexo B. De acordo com estas soluções, as conclusões a retirar são parecidas com as apresentadas nas heurísticas construtivas. Na avaliação dos valores médios é possível perceber que na sua maioria, a heurística construtiva ponderada, apresenta melhores valores para a heurística construtiva aleatória, porém os melhores resultados obtidos na heurística construtiva aleatória testada. Quanto ao desempenho desta heurística, como se esperava, conseguiu-se uma melhoria das soluções iniciais, e a rapidez de resposta é ainda bastante rápida face às dimensões das instâncias testadas.

Os tempos médios de ambas as heurísticas, são bastante baixos até à instância de 152 nodos incluída. A partir dessa instância, os tempos aumentam, mas igualmente eficientes face à dimensão das instâncias utilizadas. Quando comparados os tempos entre os resultados da heurística de melhoramento, com soluções iniciais das heurísticas construtivas, este é superior quando o ponto de partida são as soluções obtidas da heurística construtiva ponderada.

Por fim, termina-se esta dissertação com o estudo e implementação das meta-heurística escolhida, pesquisa local iterada. Esta meta-heurística passa pela destruição da admissibilidade da solução a considerar como inicial, ao adicionar os clientes facultativos que ainda não integram a solução. Depois de todos os clientes estarem na solução, devolve-se a admissibilidade da solução, ao escolher aleatoriamente os clientes facultativos a retirar, até que esta seja reposta. Termina-se o ciclo ao aplicar a pesquisa local, que nesta dissertação escolheu-se a heurística de melhoramento e repete-se todo o processo novamente, o número de vezes que se atribuir.

Nas primeiras soluções apresentadas no anexo C, optou-se por aplicar 50 iterações, ou seja, o processo da meta-heurística foi repetido 50 vezes. As soluções obtidas permitem concluir que a eficiência e qualidade da meta-heurística considerada nesta dissertação retorna bons resultados, acima do esperado. Apesar da rapidez a obter soluções ser menos eficiente que na heurística de melhoramento, os valores dos lucros são bastantes melhores. Assim, uma vez que para o desenho da estrutura em estudo é muito mais importante o lucro que este pode gerar do que a rapidez na resposta, a meta-heurística revela ser a melhor resolução a aplicar a este problema.

Para além disso, a conclusão em relação à melhor heurística construtiva a considerar inicialmente é a heurística construtiva aleatória.

Com o objetivo de melhorar ainda mais as soluções encontradas nos testes com 50 iterações, decidiu-se explorar o impacto que o aumento das iterações teria na qualidade da solução.

De acordo com toda a experiência computacional apresentada no capítulo anterior, concluiu-se que um bom equilíbrio entre a qualidade da solução e o tempo necessário é obtido para 500 iterações.

5.1. Trabalho Futuro

Nesta dissertação foram explorados alguns cenários para o SmRNDP, porém não se explorou o impacto que alguns elementos teriam na solução, como por exemplo o número de anéis ou a proporção de clientes obrigatórios e a proporção de clientes por anel. Seria interessante fazer uma análise comparativa de qual o número de anéis devia ser considerado para cada uma das instâncias consideradas.

Para além disso, seria ainda interessante comparar a meta-heurística escolhida, com outras que tenham uma boa referência como é o caso da pesquisa local iterada, com o objetivo de aprimorar a resolução ao problema em estudo.

6. Referências

- [1] Baldacci R., Dell'Amico M., Salazar González J. *The Capacitated M-Ring-Star Problem*. Vol. 55, 1147-1162. *Operations Research*, 2007.
- [2] Baldacci R., Dell'Amico M. *Heuristic Algorithms for the Multi-Depot Ring-Star Problem*. Vol. 203, 270-281. *European Journal of Operational Research*, 2010.
- [3] Bautzer A., Gouveia L., Paiais A., Pires J. *Models for a Steiner multi-ring network design problem with revenues*. Vol. 24, 360-380. *TOP*, 2016.
- [4] Feillet D., Dejax P., Gendreau M. *Traveling Salesman Problems with Profits*. Vol. 39, 188-205. *Transportation Science*, 2005.
- [5] Fink A., Schneidereit G., Voß S. *Solving General Ring Network Design Problems by meta-heuristics*, Vol. 12, 91-92. *Operations Research/Computer Science Interfaces Series book series*, 1993 - 2020.
- [6] Gouveia L., Pires J. *Models for a Steiner Ring Network Design Problem with Revenues*. Vol. 133, 21-31. *European Journal of Operational Research*, 2001.
- [7] Hill A., e Voß S. *Generalized local branching heuristics and the capacitated ring tree problem*. Vol. 242, 34-52. *Discrete Applied Mathematics*, 2018.
- [8] Lourenço H., Martin O., Stützle T. *Iterated Local Search*. Vol. 57, 321-353. *International Series in Operations Research & Management Science book series*, 2003.
- [9] Naji-Azimi Z., Salari M., Toth P. *A Heuristic Procedure for the Capacitated M-Ring-Star Problem*. Vol. 207, 1227-1234. *European Journal of Operational Research*, 2010.
- [10] Naji-Azimi Z., Salari M., Toth P. *An Integer Linear Programming Based Heuristic for the Capacitated M-Ring-Star Problem*. Vol. 217, 17-25. *European Journal of Operational Research*, 2012.
- [11] “The Family Traveling Salesman Problem.” s.d. <http://familytsp.rd.ciencias.ulisboa.pt/instances/>.
- [12] Breunig U., Schmid V., Hartl R., Vidal T. *A large neighbourhood based heuristic for two-echelon routing problems*. Vol. 76, 208-225. *Computers & Operations Research*, 2016.

Anexos

A. Resultados das cinco *runs* da Heurística construtiva ponderada

A.1. - Resultados das Heurísticas Construtivas para a instância de 20 nodos

Heurística Construtiva de Inserção Ponderada			Heurística Construtiva de Inserção Aleatória		
Teste	Valor Lucro	Tempo CPU	Teste	Valor Lucro	Tempo CPU
1	-1147	0,09	1	-1511	0,14
2	-503	0,07	2	-2239	0,10
3	-1244	0,06	3	-150	0,05
4	-877	0,08	4	-2482	0,07
5	-1244	0,08	5	-1444	0,07
Média	-1003	0	Média	-1565,2	0

A.2. - Resultados das Heurísticas Construtivas para a instância de 30 nodos

Heurística Construtiva de Inserção Ponderada			Heurística Construtiva de Inserção Aleatória		
Teste	Valor Lucro	Tempo CPU	Teste	Valor Lucro	Tempo CPU
1	21961	0,15	1	22829	0,09
2	24701	0,13	2	26323	0,10
3	24811	0,08	3	20963	0,15
4	23323	0,11	4	22972	0,16
5	21679	0,09	5	25438	0,15
Média	23295	0	Média	23705	0

A.3. - Resultados das Heurísticas Construtivas para a instância de 50 nodos

Heurística Construtiva de Inserção Ponderada			Heurística Construtiva de Inserção Aleatória		
Teste	Valor Lucro	Tempo CPU	Teste	Valor Lucro	Tempo CPU
1	27002	0,24	1	21096	0,20
2	27002	0,18	2	15543	0,22
3	23550	0,20	3	16460	0,21
4	27002	0,18	4	9945	0,22
5	27002	0,16	5	24074	0,20
Média	26311,6	0	Média	17423,6	0

A.4. - Resultados das Heurísticas Construtivas para a instância de 70 nodos

Heurística Construtiva de Inserção Ponderada			Heurística Construtiva de Inserção Aleatória		
Teste	Valor Lucro	Tempo CPU	Teste	Valor Lucro	Tempo CPU
1	46948	0,36	1	55813	0,38
2	44457	0,36	2	34330	0,34
3	66110	0,47	3	71231	0,38
4	38416	0,34	4	45159	0,33
5	69691	0,34	5	46700	0,31
Média	53124,4	0	Média	50646,6	0

A.5. - Resultados das Heurísticas Construtivas para a instância de 100 nodos

Heurística Construtiva de Inserção Ponderada			Heurística Construtiva de Inserção Aleatória		
Teste	Valor Lucro	Tempo CPU	Teste	Valor Lucro	Tempo CPU
1	210099	0,75	1	134783	0,46
2	216238	0,70	2	140953	0,51
3	223327	0,72	3	141547	0,41
4	226553	0,72	4	145675	0,54
5	215258	0,79	5	144348	0,56
Média	218295	1	Média	141461,2	0

A.6. - Resultados das Heurísticas Construtivas para a instância de 136 nodos

Heurística Construtiva de Inserção Ponderada			Heurística Construtiva de Inserção Aleatória		
Teste	Valor Lucro	Tempo CPU	Teste	Valor Lucro	Tempo CPU
1	164474	1,34	1	198816	1,08
2	163141	1,41	2	307679	1,80
3	161182	1,38	3	205945	1,22
4	168496	1,25	4	179042	1,17
5	179345	1,32	5	310477	1,84
Média	167327,6	1	Média	240391,8	1

A.7. - Resultados das Heurísticas Construtivas para a instância de 152 nodos

Heurística Construtiva de Inserção Ponderada			Heurística Construtiva de Inserção Aleatória		
Teste	Valor Lucro	Tempo CPU	Teste	Valor Lucro	Tempo CPU
1	251994	2,11	1	169606	1,65
2	256169	2,07	2	167393	1,82
3	238925	2,42	3	180517	1,54
4	240461	2,12	4	275031	1,72
5	231987	2,02	5	277271	1,89
Média	243907,2	2	Média	213963,6	2

A.8. - Resultados das Heurísticas Construtivas para a instância de 202 nodos

Heurística Construtiva de Inserção Ponderada			Heurística Construtiva de Inserção Aleatória		
Teste	Valor Lucro	Tempo CPU	Teste	Valor Lucro	Tempo CPU
1	574279	5,20	1	572325	4,35
2	568050	5,20	2	585544	3,97
3	569902	5,02	3	579561	3,87
4	575549	4,98	4	571810	4,24
5	559053	4,95	5	561249	3,89
Média	569366,6	5	Média	574097,8	4

A.9. - Resultados das Heurísticas Construtivas para a instância de 264 nodos

Heurística Construtiva de Inserção Ponderada			Heurística Construtiva de Inserção Aleatória		
Teste	Valor Lucro	Tempo CPU	Teste	Valor Lucro	Tempo CPU
1	1426501	22,55	1	950752	11,80
2	1435747	23,78	2	940484	11,01
3	1438618	21,19	3	1410905	18,67
4	1415302	20,82	4	945282	11,57
5	1446708	20,65	5	941703	11,45
Média	1432575,2	22	Média	1037825,2	13

B. Resultados das cinco *runs* da Heurística de melhoramento

B.1. - Resultados da Heurística de melhoramento, onde as soluções iniciais das heurísticas construtivas para a instância de 20 nodos

Heurística Construtiva de Inserção Ponderada				Heurística Construtiva de Inserção Aleatória			
Teste	Valor Lucro	Tempo CPU	Gap (%)	Teste	Valor Lucro	Tempo CPU	Gap (%)
1	615	0,05	-510%	1	615	0,04	-510%
2	615	0,05	-510%	2	615	0,04	-510%
3	615	0,05	-510%	3	615	0,04	-510%
4	615	0,04	-510%	4	-1498	0,04	899%
5	615	0,04	-510%	5	615	0,04	-510%
Média	615	0	-510%	Média	192,4	0	-228%

B.2. - Resultados da Heurística de melhoramento, onde as soluções iniciais das heurísticas construtivas para a instância de 30 nodos

Heurística Construtiva de Inserção Ponderada				Heurística Construtiva de Inserção Aleatória			
Teste	Valor Lucro	Tempo CPU	Gap (%)	Teste	Valor Lucro	Tempo CPU	Gap (%)
1	28000	0,06	-6%	1	29592	0,08	-12%
2	28447	0,08	-8%	2	28637	0,07	-9%
3	29052	0,06	-10%	3	30466	0,08	-16%
4	28823	0,08	-9%	4	30667	0,08	-17%
5	28356	0,07	-8%	5	29594	0,09	-12%
Média	28535,6	0	-8%	Média	29791,2	0	-13%

B.3. - Resultados da Heurística de melhoramento, onde as soluções iniciais das heurísticas construtivas para a instância de 50 nodos

Heurística Construtiva de Inserção Ponderada				Heurística Construtiva de Inserção Aleatória			
Teste	Valor Lucro	Tempo CPU	Gap (%)	Teste	Valor Lucro	Tempo CPU	Gap (%)
1	27958	0,06	-4%	1	27660	0,10	-2%
2	27958	0,07	-4%	2	19191	0,07	29%
3	24653	0,07	9%	3	28768	0,12	-7%
4	27958	0,08	-4%	4	17488	0,07	35%
5	27958	0,06	-4%	5	28768	0,10	-7%
Média	27297	0	-1%	Média	24375	0	10%

B.4. Resultados da Heurística de melhoramento, onde as soluções iniciais das heurísticas construtivas para a instância de 70 nodos

Heurística Construtiva de Inserção Ponderada				Heurística Construtiva de Inserção Aleatória			
Teste	Valor Lucro	Tempo CPU	Gap (%)	Teste	Valor Lucro	Tempo CPU	Gap (%)
1	58070	0,16	18%	1	86611	0,67	-22%
2	51947	0,26	27%	2	57439	0,16	19%
3	85761	0,40	-20%	3	93866	0,23	-32%
4	47495	0,33	33%	4	50441	0,11	29%
5	81647	0,18	-15%	5	54720	0,09	23%
Média	64984	0	9%	Média	68615,4	0	4%

B.5. - Resultados da Heurística de melhoramento, onde as soluções iniciais das heurísticas construtivas para a instância de 100 nodos

Heurística Construtiva de Inserção Ponderada				Heurística Construtiva de Inserção Aleatória			
Teste	Valor Lucro	Tempo CPU	Gap (%)	Teste	Valor Lucro	Tempo CPU	Gap (%)
1	260156	0,80	-15%	1	170025	1,26	25%
2	255636	1,71	-13%	2	169835	0,67	25%
3	265665	1,17	-17%	3	265174	3,49	-17%
4	256758	0,94	-13%	4	162050	0,39	28%
5	258288	0,70	-14%	5	267472	2,92	-18%
Média	259300,6	1	-14%	Média	206911,2	2	9%

B.6. - Resultados da Heurística de melhoramento, onde as soluções iniciais das heurísticas construtivas para a instância de 136 nodos

Heurística Construtiva de Inserção Ponderada				Heurística Construtiva de Inserção Aleatória			
Teste	Valor Lucro	Tempo CPU	Gap (%)	Teste	Valor Lucro	Tempo CPU	Gap (%)
1	224044	5,32	28%	1	211000	0,15	32%
2	224830	2,50	28%	2	362485	3,48	-17%
3	233401	1,88	25%	3	254257	1,34	18%
4	225969	4,33	27%	4	252465	0,72	19%
5	222176	1,71	28%	5	359049	1,10	-16%
Média	226084	3	27%	Média	287851,2	1	7%

B.7. - Resultados da Heurística de melhoramento, onde as soluções iniciais das heurísticas construtivas para a instância de 152 nodos

Heurística Construtiva de Inserção Ponderada				Heurística Construtiva de Inserção Aleatória			
Teste	Valor Lucro	Tempo CPU	Gap (%)	Teste	Valor Lucro	Tempo CPU	Gap (%)
1	319862	7,03	-15%	1	194660	0,51	30%
2	318720	8,80	-15%	2	192531	0,61	31%
3	305021	7,80	-10%	3	199179	1,50	28%
4	308065	3,97	-11%	4	302605	0,29	-9%
5	304550	6,71	-10%	5	347319	1,17	-25%
Média	311243,6	7	-12%	Média	247258,8	1	11%

B.8. - Resultados da Heurística de melhoramento, onde as soluções iniciais das heurísticas construtivas para a instância de 202 nodos

Heurística Construtiva de Inserção Ponderada				Heurística Construtiva de Inserção Aleatória			
Teste	Valor Lucro	Tempo CPU	Gap (%)	Teste	Valor Lucro	Tempo CPU	Gap (%)
1	990130	27,61	-69%	1	651076	11,07	-11%
2	985453	29,35	-68%	2	622856	0,81	-6%
3	998023	48,70	-70%	3	1007414	49,47	-72%
4	986645	56,21	-69%	4	635450	25,67	-9%
5	605935	0,52	-3%	5	603120	0,51	-3%
Média	913237,2	32	-56%	Média	703983,2	18	-20%

B.9. Resultados da Heurística de melhoramento, onde as soluções iniciais das heurísticas construtivas para a instância de 264 nodos

Heurística Construtiva de Inserção Ponderada				Heurística Construtiva de Inserção Aleatória			
Teste	Valor Lucro	Tempo CPU	Gap (%)	Teste	Valor Lucro	Tempo CPU	Gap (%)
1	990130	27,61	-69%	1	651076	11,07	-11%
2	985453	29,35	-68%	2	622856	0,81	-6%
3	998023	48,70	-70%	3	1007414	49,47	-72%
4	986645	56,21	-69%	4	635450	25,67	-9%
5	605935	0,52	-3%	5	603120	0,51	-3%
Média	1557935	46	-8%	Média	1270733	37	12%

C. Resultados das cinco *runs* da Meta-heurística com 50 iterações

C.1. Resultados da Meta-Heurística, onde as soluções iniciais das heurísticas construtivas para a instância de 20 nodos, com 50 iterações

Heurística Construtiva de Inserção Ponderada					Heurística Construtiva de Inserção Aleatória				
Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações	Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações
1	4620	0,37	-3180%	46	1	4118	0,34	-2845%	18
2	5055	0,34	-3470%	16	2	5055	0,36	-3470%	24
3	4778	0,32	-3285%	33	3	5494	0,42	-3763%	24
4	4509	0,38	-3106%	45	4	6698	0,44	-4565%	44
5	4864	0,36	-3343%	22	5	4136	0,32	-2857%	22
Média	4765	0	-3277%	32,4	Média	5100	0	-3500%	26,4

C.2. Resultados da Meta-Heurística, onde as soluções iniciais das heurísticas construtivas para a instância de 30 nodos, com 50 iterações

Heurística Construtiva de Inserção Ponderada					Heurística Construtiva de Inserção Aleatória				
Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações	Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações
1	30671	0,69	-17%	27	1	30755	0,57	-17%	11
2	30601	0,72	-16%	48	2	31083	0,72	-18%	16
3	31138	0,82	-18%	50	3	31266	0,68	-19%	18
4	30297	0,69	-15%	41	4	30667	0,64	-17%	6
5	31266	0,76	-19%	33	5	30816	0,60	-17%	6
Média	30795	1	-17%	39,8	Média	30917	1	-17%	11,4

C.3. Resultados da Meta-Heurística, onde as soluções iniciais das heurísticas construtivas para a instância de 50 nodos, com 50 iterações

Heurística Construtiva de Inserção Ponderada					Heurística Construtiva de Inserção Aleatória				
Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações	Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações
1	37664	1,65	-39%	47	1	34261	1,60	-27%	46
2	34003	1,72	-26%	44	2	35527	3,25	-32%	46
3	36891	1,37	-37%	28	3	37081	2,10	-37%	36
4	36858	1,87	-37%	17	4	30725	2,62	-14%	45
5	36425	1,63	-35%	29	5	36863	1,62	-37%	28
Média	36368	2	-35%	33	Média	34891	2	-29%	40,2

C.4. Resultados da Meta-Heurística, onde as soluções iniciais das heurísticas construtivas para a instância de 70 nodos, com 50 iterações

Heurística Construtiva de Inserção Ponderada					Heurística Construtiva de Inserção Aleatória				
Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações	Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações
1	85277	8,29	-20%	3	1	96021	9,02	-35%	21
2	89475	5,95	-26%	37	2	88051	7,72	-24%	11
3	93800	6,40	-32%	2	3	100234	8,99	-41%	46
4	87376	7,06	-23%	16	4	97728	12,60	-37%	40
5	91443	6,99	-28%	15	5	91737	10,31	-29%	27
Média	89474	7	-26%	14,6	Média	94754	10	-33%	29

C.5. Resultados da Meta-Heurística, onde as soluções iniciais das heurísticas construtivas para a instância de 100 nodos, com 50 iterações

Heurística Construtiva de Inserção Ponderada					Heurística Construtiva de Inserção Aleatória				
Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações	Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações
1	273417	42,25	-21%	50	1	273582	73,74	-21%	25
2	271615	39,14	-20%	19	2	266461	58,50	-18%	48
3	273495	70,58	-21%	47	3	273148	85,29	-21%	43
4	268702	15,01	-19%	18	4	271496	65,06	-20%	14
5	269789	15,24	-19%	48	5	270454	91,80	-19%	41
Média	271404	36	-20%	36,4	Média	271028	75	-20%	34,2

C.6. Resultados da Meta-Heurística, onde as soluções iniciais das heurísticas construtivas para a instância de 136 nodos, com 50 iterações

Heurística Construtiva de Inserção Ponderada					Heurística Construtiva de Inserção Aleatória				
Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações	Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações
1	363311	79,72	-17%	12	1	378142	106,19	-22%	42
2	365274	124,38	-18%	34	2	378675	69,49	-22%	46
3	370797	93,32	-19%	10	3	377303	70,31	-22%	45
4	368564	125,19	-19%	44	4	383140	76,57	-23%	25
5	369230	115,27	-19%	32	5	387708	95,88	-25%	37
Média	367435	108	-18%	26,4	Média	380994	84	-23%	39

C.7. Resultados da Meta-Heurística, onde as soluções iniciais das heurísticas construtivas para a instância de 152 nodos, com 50 iterações

Heurística Construtiva de Inserção Ponderada					Heurística Construtiva de Inserção Aleatória				
Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações	Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações
1	482169	72,54	-74%	22	1	507669	264,25	-83%	25
2	485360	74,82	-75%	30	2	503014	197,89	-81%	7
3	482861	94,80	-74%	47	3	489384	117,19	-77%	25
4	486695	74,28	-76%	17	4	520351	368,92	-88%	20
5	475276	70,77	-71%	19	5	501799	230,10	-81%	26
Média	482472	77	-74%	27	Média	504443	236	-82%	20,6

C.8. Resultados da Meta-Heurística, onde as soluções iniciais das heurísticas construtivas para a instância de 202 nodos, com 50 iterações

Heurística Construtiva de Inserção Ponderada					Heurística Construtiva de Inserção Aleatória				
Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações	Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações
1	1026825	1220,07	-75%	49	1	1002632	1067,05	-71%	36
2	1021402	1673,45	-74%	42	2	1010152	2036,08	-73%	16
3	1019142	1463,40	-74%	33	3	1012795	1993,91	-73%	40
4	1017000	1346,06	-74%	40	4	992509	454,48	-70%	37
5	1013816	1479,09	-73%	44	5	1017835	1718,65	-74%	33
Média	1019637	1436	-74%	41,6	Média	1007185	1454	-72%	32,4

C.9. Resultados da Meta-Heurística, onde as soluções iniciais das heurísticas construtivas para a instância de 264 nodos, com 50 iterações

Heurística Construtiva de Inserção Ponderada					Heurística Construtiva de Inserção Aleatória				
Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações	Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações
1	1593200	1014,25	-10%	7	1	1596777	687,59	-10%	15
2	1597399	947,28	-10%	23	2	1644667	2295,95	-14%	42
3	1587169	554,57	-10%	22	3	1619731	881,82	-12%	31
4	1592178	846,10	-10%	41	4	1653171	30493,79	-14%	19
5	1600397	1172,15	-11%	44	5	1629566	3765,84	-13%	41
Média	1594069	907	-10%	27,4	Média	1628782	7625	-13%	29,6

D. Resultados das cinco *runs* da Meta-heurística com 200 iterações

D.1 Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 20 nodos, com 200 iterações

Heurística Construtiva de Inserção Ponderada					Heurística Construtiva de Inserção Aleatória				
Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações	Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações
1	5494	0,66	-3763%	28	1	5323	1,13	-3649%	163
2	5493	0,90	-3762%	145	2	4778	0,77	-3285%	115
3	5494	0,86	-3763%	88	3	5494	1,37	-3763%	6
4	5055	0,76	-3470%	80	4	7206	1,38	-4904%	36
5	5041	0,95	-3461%	67	5	5041	0,94	-3461%	139
Média	5315	1	-3644%	81,6	Média	5568	1	-3812%	91,8

D.2. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 30 nodos, com 200 iterações

Heurística Construtiva de Inserção Ponderada					Heurística Construtiva de Inserção Aleatória				
Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações	Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações
1	31083	2,22	-18%	82	1	30835	2,39	-17%	34
2	31083	2,54	-18%	29	2	31266	2,45	-19%	97
3	31083	2,30	-18%	91	3	30969	2,48	-18%	139
4	31266	2,38	-19%	194	4	30773	2,39	-17%	35
5	31083	2,21	-18%	150	5	31266	2,72	-19%	54
Média	31120	2	-18%	109,2	Média	31022	2	-18%	71,8

D.3. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 50 nodos, com 200 iterações

Heurística Construtiva de Inserção Ponderada					Heurística Construtiva de Inserção Aleatória				
Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações	Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações
1	38098	5,45	-41%	128	1	37596	7,08	-39%	10
2	38481	4,63	-43%	171	2	37892	12,43	-40%	104
3	33648	11,32	-25%	188	3	37692	4,85	-40%	137
4	38125	4,92	-41%	163	4	33158	8,38	-23%	84
5	40172	4,89	-49%	25	5	38875	5,14	-44%	155
Média	37705	6	-40%	135	Média	37043	8	-37%	98

D.4. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 70 nodos, com 200 iterações

Heurística Construtiva de Inserção Ponderada					Heurística Construtiva de Inserção Aleatória				
Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações	Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações
1	94664	38,10	-33%	140	1	100959	41,18	-42%	166
2	89440	21,83	-26%	114	2	96225	24,16	-35%	90
3	92863	26,15	-30%	70	3	98712	41,29	-39%	87
4	91652	21,15	-29%	176	4	102906	62,55	-44%	191
5	94450	30,94	-33%	189	5	101155	57,16	-42%	176
Média	92614	28	-30%	137,8	Média	99991	45	-40%	142

D.5. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 100 nodos, com 200 iterações

Heurística Construtiva de Inserção Ponderada					Heurística Construtiva de Inserção Aleatória				
Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações	Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações
1	272848	263,90	-20%	70	1	276507	237,73	-22%	147
2	270097	127,22	-19%	106	2	275505	311,73	-22%	167
3	277109	240,32	-22%	143	3	274765	356,28	-21%	103
4	272511	163,69	-20%	82	4	276183	317,10	-22%	115
5	274564	120,19	-21%	23	5	273824	386,37	-21%	183
Média	273426	183	-21%	84,8	Média	275357	322	-22%	143

D.6. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 136 nodos, com 200 iterações

Heurística Construtiva de Inserção Aleatória				
Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações
1	395490	429,21	-27%	182
2	384709	347,00	-24%	157
3	369322	292,04	-19%	195
4	382737	246,15	-23%	47
5	388739	443,59	-25%	152
Média	384199	352	-24%	146,6

D.7. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 152 nodos, com 200 iterações

Heurística Construtiva de Inserção Aleatória				
Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações
1	523333	1096,04	-89%	125
2	492696	440,77	-78%	42
3	492676	710,00	-78%	174
4	526108	1353,92	-90%	90
5	502538	725,32	-81%	166
Média	507470	865	-83%	119,4

D.8. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 202 nodos, com 200 iterações

Heurística Construtiva de Inserção Aleatória				
Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações
1	1017760	7378,49	-74%	122
2	1011058	3083,85	-73%	147
3	1020030	5695,80	-74%	145
4	1001749	2883,22	-71%	93
5	1021998	4992,60	-75%	136
Média	1014519	4807	-73%	128,6

D.9. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 264 nodos, com 200 iterações

Heurística Construtiva de Inserção Aleatória				
Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações
1	1606218	4369,72	-11%	149
2	1652901	12214,43	-14%	189
3	1603521	4974,48	-11%	171
4	1657224	25342,82	-15%	198
5	1634834	14399,91	-13%	196
Média	1630940	12260	-13%	180,6

E. Resultados das cinco *runs* da Meta-heurística com 500 iterações

E.1. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 20 nodos, com 500 iterações

Heurística Construtiva de Inserção Ponderada					Heurística Construtiva de Inserção Aleatória				
Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações	Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações
1	5494	2,92	-3763%	371	1	5494	3,01	-3763%	403
2	5494	2,88	-3763%	356	2	5494	3,09	-3763%	419
3	5494	2,62	-3763%	222	3	5494	5,43	-3763%	47
4	5493	3,00	-3762%	347	4	7206	5,42	-4904%	15
5	5494	2,92	-3763%	192	5	5494	3,01	-3763%	207
Média	5494	3	-3763%	297,6	Média	5836	4	-3991%	218,2

E.2. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 30 nodos, com 500 iterações

Heurística Construtiva de Inserção Ponderada					Heurística Construtiva de Inserção Aleatória				
Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações	Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações
1	31266	8,66	-19%	68	1	31266	7,63	-19%	73
2	31083	7,40	-18%	155	2	31266	5,05	-19%	289
3	31266	8,81	-19%	217	3	31266	4,85	-19%	81
4	31266	8,25	-19%	101	4	31266	4,74	-19%	303
5	31266	7,97	-19%	354	5	31266	4,99	-19%	295
Média	31229	8	-19%	179	Média	31266	5	-19%	208,2

E.3. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 50 nodos, com 500 iterações

Heurística Construtiva de Inserção Ponderada					Heurística Construtiva de Inserção Aleatória				
Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações	Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações
1	39049	16,55	-45%	486	1	39370	18,63	-46%	300
2	41102	20,80	-52%	240	2	38534	62,17	-43%	12
3	33845	29,99	-25%	199	3	39574	21,86	-47%	48
4	40468	21,91	-50%	312	4	34214	32,14	-27%	421
5	40066	19,93	-48%	369	5	40237	19,97	-49%	384
Média	38906	22	-44%	321,2	Média	38386	31	-42%	233

E.4. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 70 nodos, com 500 iterações

Heurística Construtiva de Inserção Ponderada					Heurística Construtiva de Inserção Aleatória				
Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações	Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações
1	96344	81,42	-35%	273	1	103098	220,36	-45%	328
2	93023	93,61	-31%	397	2	91474	98,11	-28%	205
3	95371	96,91	-34%	319	3	102862	171,88	-44%	373
4	96527	108,20	-36%	290	4	102662	203,34	-44%	419
5	98053	113,08	-38%	256	5	100802	221,20	-42%	173
Média	95864	99	-35%	307	Média	100180	183	-41%	299,6

E.5. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 100 nodos, com 500 iterações

Heurística Construtiva de Inserção Ponderada					Heurística Construtiva de Inserção Aleatória				
Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações	Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações
1	274097	665,11	-21%	19	1	279787	655,75	-23%	483
2	273559	484,58	-21%	488	2	277257	739,16	-22%	493
3	276042	850,16	-22%	445	3	281911	729,73	-24%	312
4	270217	421,06	-19%	356	4	274078	831,83	-21%	315
5	276864	701,27	-22%	365	5	276289	651,07	-22%	409
Média	274156	624	-21%	334,6	Média	277864	722	-23%	402,4

E.6. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 136 nodos, com 500 iterações

Heurística Construtiva de Inserção Aleatória				
Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações
1	389219	1971,83	-25%	336
2	389852	1154,82	-26%	339
3	388154	1297,75	-25%	469
4	383783	1298,30	-24%	75
5	405960	1523,10	-31%	441
Média	391394	1449	-26%	332

E.7. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 152 nodos, com 500 iterações

Heurística Construtiva de Inserção Aleatória				
Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações
1	516690	1245,09	-86%	500
2	503027	1359,07	-81%	374
3	496180	695,67	-79%	470
4	514588	1530,33	-86%	192
5	505477	2259,92	-82%	279
Média	507192	1418	-83%	363

E.8. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 202 nodos, com 500 iterações

Heurística Construtiva de Inserção Aleatória				
Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações
1	1027310	15948,18	-75%	190
2	1013080	12360,51	-73%	466
3	1022444	12615,91	-75%	357
4	1002768	6597,53	-71%	322
5	1023330	11949,28	-75%	364
Média	1017786	11894	-74%	339,8

E.9. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 264 nodos, com 500 iterações

Heurística Construtiva de Inserção Aleatória				
Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações
1	1614179	9449,97	-12%	296
2	1630878	24245,50	-13%	25
3	1612531	32007,77	-11%	53
4	1665978	69532,32	-15%	488
5	1647497	76766,78	-14%	409
Média	1634213	42400	-13%	254,2

F. Resultados das cinco *runs* da Meta-heurística com 1000 iterações

F.1. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 20 nodos, com 1000 iterações

Heurística Construtiva de Inserção Ponderada					Heurística Construtiva de Inserção Aleatória				
Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações	Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações
1	5055	4,09	-3470%	28	1	5493	5,09	-3762%	126
2	5494	3,98	-3763%	248	2	5493	5,38	-3762%	146
3	5494	4,10	-3763%	157	3	5494	7,89	-3763%	5
4	5493	4,65	-3762%	218	4	7206	8,00	-4904%	79
5	5494	3,90	-3763%	83	5	5493	5,39	-3762%	586
Média	5406	4	-3704%	146,8	Média	5836	6	-3991%	188,4

F.2. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 30 nodos, com 1000 iterações

Heurística Construtiva de Inserção Ponderada					Heurística Construtiva de Inserção Aleatória				
Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações	Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações
1	31635	9,19	-20%	366	1	31266	9,49	-19%	6
2	31266	9,61	-19%	20	2	31266	8,62	-19%	450
3	31083	8,40	-18%	223	3	31266	9,44	-19%	26
4	31266	9,25	-19%	60	4	31266	8,38	-19%	236
5	31266	9,85	-19%	105	5	31266	9,80	-19%	45
Média	31303	9	-19%	154,8	Média	31266	9	-19%	152,6

F.3. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 50 nodos, com 1000 iterações

Heurística Construtiva de Inserção Ponderada					Heurística Construtiva de Inserção Aleatória				
Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações	Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações
1	40540	23,03	-50%	611	1	37234	40,38	-38%	7
2	40011	18,27	-48%	421	2	39533	114,49	-46%	553
3	34719	28,38	-29%	937	3	39866	34,87	-48%	121
4	39229	17,37	-45%	157	4	34225	57,56	-27%	804
5	39650	17,32	-47%	143	5	40544	41,85	-50%	220
Média	38830	21	-44%	453,8	Média	38280	58	-42%	341

F.4. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 70 nodos, com 1000 iterações

Heurística Construtiva de Inserção Ponderada					Heurística Construtiva de Inserção Aleatória				
Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações	Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações
1	98247	211,83	-38%	535	1	97647	201,68	-37%	768
2	93520	264,55	-31%	632	2	98537	231,63	-38%	857
3	96819	250,47	-36%	360	3	102490	358,03	-44%	496
4	92679	256,63	-30%	533	4	96958	199,26	-36%	879
5	95134	216,97	-34%	873	5	94598	288,91	-33%	234
Média	95280	240	-34%	586,6	Média	98046	256	-38%	646,8

F.5. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 100 nodos, com 1000 iterações

Heurística Construtiva de Inserção Ponderada					Heurística Construtiva de Inserção Aleatória				
Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações	Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações
1	273270	1261,88	-21%	812	1	277608	1513,98	-23%	550
2	272889	812,61	-20%	320	2	276688	1070,34	-22%	791
3	274518	1064,82	-21%	704	3	275070	1246,46	-21%	854
4	276531	1185,34	-22%	911	4	276439	960,00	-22%	741
5	274004	2317,36	-21%	810	5	275869	1584,67	-22%	705
Média	274242	1328	-21%	711,4	Média	276335	1275	-22%	728,2

F.6. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 136 nodos, com 1000 iterações

Heurística Construtiva de Inserção Aleatória				
Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações
1	383888	3991,75	-24%	365
2	388176	2346,74	-25%	409
3	381736	3464,80	-23%	22
4	395454	1466,08	-27%	973
5	381183	1278,85	-23%	757
Média	386087	2510	-24%	505,2

F.7. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 152 nodos, com 1000 iterações

Heurística Construtiva de Inserção Aleatória				
Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações
1	508666	3228,70	-83%	428
2	508867	4168,75	-84%	333
3	483459	2532,91	-74%	650
4	528556	15525,46	-91%	618
5	505257	3149,35	-82%	95
Média	506961	5721	-83%	424,8

F.8. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 202 nodos, com 1000 iterações

Heurística Construtiva de Inserção Aleatória				
Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações
1	1020925	36508,66	-74%	656
2	1021382	54637,11	-74%	727
3	1025669	41595,71	-75%	422
4	1019221	16031,07	-74%	692
5	1025023	21364,11	-75%	550
Média	1022444	34027	-75%	609,4

F.9. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 264 nodos, com 1000 iterações

Heurística Construtiva de Inserção Aleatória				
Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações
1	1623733	26465,56	-12%	685
2	1658846	90611,28	-15%	965
3	1617644	22201,55	-12%	480
4	1660584	146155,80	-15%	823
5	1655362	81899,04	-14%	683
Média	1643234	73467	-14%	727,2

G. Resultados das cinco *runs* da Meta-heurística com 2000 iterações

G.1. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 20 nodos, com 2000 iterações

Heurística Construtiva de Inserção Ponderada					Heurística Construtiva de Inserção Aleatória				
Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações	Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações
1	5494	6,75	-3763%	117	1	5494	8,29	-3763%	124
2	5494	6,96	-3763%	301	2	5494	8,50	-3763%	190
3	5494	8,70	-3763%	165	3	5494	15,00	-3763%	5
4	5494	10,62	-3763%	322	4	7206	15,28	-4904%	1
5	5494	10,95	-3763%	589	5	5493	10,45	-3762%	368
Média	5494	9	-3763%	298,8	Média	5836	12	-3991%	137,6

G.2. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 30 nodos, com 2000 iterações

Heurística Construtiva de Inserção Ponderada					Heurística Construtiva de Inserção Aleatória				
Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações	Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações
1	32294	17,93	-23%	553	1	31266	17,72	-19%	121
2	31266	17,30	-19%	44	2	31266	16,76	-19%	927
3	31266	18,14	-19%	319	3	31635	17,92	-20%	756
4	31266	18,55	-19%	62	4	31635	17,29	-20%	1090
5	31266	19,39	-19%	18	5	32773	18,81	-25%	719
Média	31472	18	-20%	199,2	Média	31715	18	-20%	722,6

G.3. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 50 nodos, com 2000 iterações

Heurística Construtiva de Inserção Ponderada					Heurística Construtiva de Inserção Aleatória				
Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações	Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações
1	41089	35,80	-52%	321	1	38728	36,02	-43%	1511
2	43076	45,70	-60%	1339	2	39015	90,52	-44%	516
3	40970	38,29	-52%	1319	3	41154	44,94	-52%	1721
4	40591	33,39	-50%	937	4	34498	60,61	-28%	1972
5	42654	42,87	-58%	369	5	41167	39,10	-52%	266
Média	41676	39	-54%	857	Média	38912	54	-44%	1197,2

G.4. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 70 nodos, com 2000 iterações

Heurística Construtiva de Inserção Ponderada					Heurística Construtiva de Inserção Aleatória				
Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações	Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações
1	94811	337,58	-33%	963	1	100971	536,91	-42%	1972
2	95585	418,90	-34%	1825	2	94579	327,72	-33%	1701
3	96017	427,10	-35%	1057	3	100589	550,06	-41%	752
4	97535	312,05	-37%	731	4	101819	793,91	-43%	600
5	100860	404,75	-42%	1893	5	96113	351,68	-35%	242
Média	96962	380	-36%	1293,8	Média	98814	512	-39%	1053,4

G.5. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 100 nodos, com 2000 iterações

Heurística Construtiva de Inserção Ponderada					Heurística Construtiva de Inserção Aleatória				
Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações	Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações
1	277044	3757,93	-22%	1880	1	280522	2117,46	-24%	1404
2	273143	1573,96	-21%	1127	2	279575	2184,70	-23%	655
3	276687	2094,59	-22%	1289	3	280824	2103,77	-24%	1191
4	274048	1962,97	-21%	765	4	276600	1757,94	-22%	405
5	276241	1700,87	-22%	1613	5	276552	11543,76	-22%	1260
Média	275433	2218	-22%	1334,8	Média	278815	3942	-23%	983

H. Resultados das cinco *runs* da Meta-heurística com 3000 iterações

H.1. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 20 nodos, com 3000 iterações

Heurística Construtiva de Inserção Ponderada					Heurística Construtiva de Inserção Aleatória				
Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações	Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações
1	5494	12,58	-3763%	2286	1	5494	12,28	-3763%	640
2	5494	11,95	-3763%	718	2	5494	11,80	-3763%	886
3	5494	11,60	-3763%	153	3	5494	22,15	-3763%	2
4	5494	12,09	-3763%	16	4	7206	21,51	-4904%	58
5	5494	11,78	-3763%	200	5	5494	13,76	-3763%	212
Média	5494	12	-3763%	674,6	Média	5836	16	-3991%	359,6

H.2. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 30 nodos, com 3000 iterações

Heurística Construtiva de Inserção Ponderada					Heurística Construtiva de Inserção Aleatória				
Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações	Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações
1	31635	27,34	-20%	1906	1	33032	26,37	-25%	1395
2	32004	27,27	-22%	2988	2	32719	27,73	-24%	2957
3	31266	21,99	-19%	729	3	31266	26,36	-19%	11
4	33114	24,55	-26%	2907	4	33032	25,88	-25%	2442
5	31266	23,92	-19%	1912	5	31635	27,17	-20%	2792
Média	31857	25	-21%	2088,4	Média	32337	27	-23%	1919,4

H.3. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 50 nodos, com 3000 iterações

Heurística Construtiva de Inserção Ponderada					Heurística Construtiva de Inserção Aleatória				
Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações	Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações
1	41610	111,03	-54%	500	1	38132	121,19	-41%	434
2	40793	133,34	-51%	1538	2	47141	482,36	-75%	1431
3	36362	185,69	-35%	2739	3	41079	124,83	-52%	153
4	41567	121,78	-54%	836	4	35454	196,28	-31%	2546
5	40642	124,96	-51%	2555	5	40540	105,11	-50%	1471
Média	40195	135	-49%	1633,6	Média	40469	206	-50%	1207

H.4. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 70 nodos, com 3000 iterações

Heurística Construtiva de Inserção Ponderada					Heurística Construtiva de Inserção Aleatória				
Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações	Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações
1	96105	296,71	-35%	1141	1	101985	1664,44	-43%	1898
2	97008	298,97	-36%	2760	2	99348	687,31	-39%	2058
3	100464	363,07	-41%	2333	3	100509	464,09	-41%	1873
4	96049	394,84	-35%	362	4	96657	638,78	-36%	1662
5	97037	317,01	-36%	296	5	97965	674,48	-38%	676
Média	97333	334	-37%	1378,4	Média	99293	826	-39%	1633,4

H.5. Resultados da Meta-Heurística, com soluções iniciais dadas pelas heurísticas construtivas para a instância de 100 nodos, com 3000 iterações

Heurística Construtiva de Inserção Ponderada					Heurística Construtiva de Inserção Aleatória				
Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações	Teste	Valor Lucro	Tempo CPU	Gap (%)	Iterações
1	278797	3860,97	-23%	2268	1	279557	6878,74	-23%	2967
2	276597	6305,99	-22%	2985	2	278984	4742,87	-23%	2834
3	277474	2529,64	-22%	1394	3	281796	4726,70	-24%	2451
4	277175	1849,85	-22%	706	4	274832	1887,35	-21%	280
5	279513	2508,28	-23%	2217	5	282513	5153,11	-25%	1732
Média	277911	3411	-23%	1914	Média	279536	4678	-23%	2052,8