



**UNIVERSIDADE
FEDERAL DE
OURO PRETO**



**PROBLEMA DE ROTEAMENTO DE MAMÓGRAFOS MÓVEIS: UMA
ABORDAGEM MULTIOBJETIVO**

Thiago Giachetto de Araujo

Orientadores: Puca Huachi Vaz Penna
Marcone Jamilson Freitas Souza

Ouro Preto
Fevereiro de 2023



**UNIVERSIDADE
FEDERAL DE
OURO PRETO**



**PROBLEMA DE ROTEAMENTO DE MAMÓGRAFOS MÓVEIS: UMA
ABORDAGEM MULTIOBJETIVO**

Thiago Giachetto de Araujo

Exame de Qualificação de Mestrado
apresentado ao Programa de Pós-graduação
em Ciência da Computação, da Universidade
Federal de Ouro Preto, como parte dos
requisitos necessários à obtenção do título de
Mestre em Ciência da Computação.

Orientadores: Puca Huachi Vaz Penna
Marccone Jamilson Freitas Souza

Ouro Preto
Fevereiro de 2023

Sumário

Lista de Figuras

Lista de Tabelas

1	Introdução	1
1.1	Objetivos	2
1.2	Organização do Trabalho	2
2	Referencial Teórico	3
2.1	Revisão da Literatura	3
2.2	Problemas de Otimização Multiobjetivo	5
2.3	Métodos Exatos de Resolução	6
2.3.1	Combinação Linear com Pesos	6
2.3.2	Método ϵ -restrito	7
2.4	Métodos Heurísticos de Resolução	7
2.4.1	Nondominated Sorting Genetic Algorithm II	7
3	Metodologia	12
3.1	Método Exato	12
3.1.1	Modelo Matemático	12
3.1.2	Modelo mono-objetivo para resolução do MMURP	14
3.2	Método Heurístico	14
3.2.1	Representação da solução	14
3.2.2	Função de avaliação	15
3.2.3	Operadores de recombinação de soluções	17
3.2.4	Determinação dos parâmetros da metaheurística	18
4	Experimentos Computacionais	19
4.1	Instâncias	19
4.2	Método Exato	20
4.2.1	Implementação	20
4.2.2	Resultados	20

SUMÁRIO

4.3	Método Heurístico	21
4.3.1	Implementação	21
4.3.2	Resultados	21
4.4	Discussão	21
5	Conclusões e Trabalhos Futuros	25
5.1	Cronograma	25
6	Rascunho	27
6.1	Objetivos	27
6.2	Revisão	27
6.2.1	Problemas de coleta de prêmios	30
6.2.2	Problema de orientação de times	31
6.2.3	Splitting para o problema de minimização de distância e de número de veículos	31
	Referências Bibliográficas	33

Lista de Figuras

1.1	Caminhão com mamógrafo. ¹	1
3.1	Exemplo de representação de um indivíduo	15
4.1	Comparação do hipervolume	22
4.2	Comparação do hipervolume	23
4.3	Comparação do hipervolume	24

¹Fonte: <https://www.campogrande.ms.gov.br/cgnoticias/noticias/carreta-do-hospital-de-amor-leva-exames-de-prevencao-ao-cancer-a-mulheres-do-distrito-de-anhandu>

Lista de Tabelas

3.1	Faixa de valores atribuídos aos parâmetros da calibração	18
4.1	Instâncias utilizadas nos testes	19
4.2	Resultados dos testes do método exato	20
4.3	Resultados dos testes usando o NSGAII	21

Capítulo 1

Introdução

Quando se exclui o câncer de pele não melanoma, o câncer de mama é o que tem a maior taxa de incidência e o que causa a maior mortalidade entre as mulheres [1]. Segundo a Organização Mundial de Saúde (OMS), o câncer de mama quando detectado em estágio inicial tem grande chance de cura e também existem evidências de que convidar mulheres entre 50 e 69 anos para realizarem a mamografia de rotina, a cada dois anos, sozinha, diminui a mortalidade por câncer de mama [2].

Porém, a desigualdade e o tempo de deslocamento da mulher até um mamógrafo são fatores que diminuem a probabilidade das mulheres realizarem o exame de mamografia [3]. Desta maneira, o problema de alocação de mamógrafos é importante para possibilitar a diminuição dessa desigualdade, pois existem equipamentos suficientes para atender toda a demanda do Brasil, mas eles não estão distribuídos de maneira que seja possível atendê-las [4]. Visto esse problema na alocação, há na literatura vários trabalhos tratando da melhor maneira de distribuir os equipamentos existentes ([5], [6], [7], [8]).



Figura 1.1: Caminhão com mamógrafo.¹

Nesses trabalhos de alocação de mamógrafos, pode-se observar que ao limitar a máxima distância que uma mulher deve percorrer para realizar uma mamografia e tendo em vista que nem toda cidade tem infraestrutura para sediar um equipamento, existem regiões que não serão atendidas por um mamógrafo. Para tentar atender essa demanda, Unidades Móveis de Mamografia (MMU, da sigla em inglês *Mobile Mammography Unit*) podem ser utilizadas [9]. MMUs são caminhões baú equipados com mamógrafo, sala para exames e unidade reveladora de raios-X, como pode ser visto na Figura 1.1. Visto a possibilidade da utilização de MMUs, surge a necessidade de planejar as rotas desses caminhões de tal forma que a atender a maior demanda possível e percorrer a menor distância. Visando estudar esse problema, Rosa *et al.* [10] introduziram o Problema de Roteamento de Unidades Móveis de Mamografia (*MMURP*, do inglês *Mobile Mammography Unit Routing Problem*).

1.1 Objetivos

Este trabalho tem como objetivo geral a formulação do *MMURP* como um problema de otimização multiobjetivo e o desenvolvimento de um método heurístico para resolvê-lo.

Os objetivos específicos são:

1. Desenvolver uma formulação de programação inteira mista para o *MMURP*.
2. Desenvolver uma heurística baseada na meta-heurística NSGAII (*Non-dominated Sorting Genetic Algorithm II*).
3. Implementar métodos exatos de resolução de problemas multiobjetivo para o *MMURP*.
4. Propor instâncias baseadas nas instâncias encontradas na literatura e realizar testes dos métodos implementados.

1.2 Organização do Trabalho

O restante deste trabalho está organizado como se segue. O Capítulo 2 traz a revisão da literatura e o referencial teórico. O Capítulo 3 traz o modelo matemático para o problema *MMURP* e como será implementado o método exato e o heurístico de resolução. No Capítulo 4 são apresentadas as instâncias utilizadas e os resultados dos experimentos realizados. Por fim, o Capítulo 5 traz as conclusões e um cronograma com as atividades que serão realizadas.

Capítulo 2

Referencial Teórico

Neste capítulo temos a revisão da literatura, algumas definições necessárias para o entendimento de problemas multiobjetivos, dois métodos exatos utilizados para a resolução destes problemas e a descrição do funcionamento do NSGA-II.

2.1 Revisão da Literatura

O câncer de mama é o câncer que tem a maior incidência entre mulheres e a que causa a maior mortalidade, excetuando-se o câncer de pele não melanoma [1]. Em 2020, a estimativa foi que 29.7% dos casos de câncer e 16.5% das mortes em mulheres por causa do câncer, se deram pelo câncer de mama (quando se exclui os casos de câncer de pele não melanoma) [11].

Brooks *et al.* [12] analisaram o resultado de exames de 3923 mulheres nos Estados Unidos da América e mostraram que a utilização de unidades móveis aumenta a chance de mulheres que não vinham realizando mamografias nos últimos anos, a realizar.

Ao analisarem o tempo de deslocamento de uma mulher até um mamógrafo, Andrade *et al.* [3] mostraram que o aumento no tempo de deslocamento diminui a probabilidade dela realizar o exame. Além disso, observaram que no Brasil a desigualdade social também afeta a probabilidade de uma mulher realizar a mamografia.

Como mostra o trabalho de Amaral *et al.* [4], existe uma desigualdade na localização dos mamógrafos no Brasil, sendo que a quantidade total de mamógrafos seria suficiente para atender a toda a demanda. Porém, quando se limita a distância de oferta do exame a 60 km de distância da instalação do equipamento, identifica-se regiões que não são atendidas. Além disso, nota-se a existência de regiões que têm mais equipamentos que o necessário.

Correa *et al.* [5] estudaram o Problema de Alocação de Mamógrafos (PAM), que visa alocar uma quantidade limitada de mamógrafos fixos e atribuir locais de atendimento para cada um, de tal forma que otimize uma função objetivo e respeite

uma distância máxima de deslocamento. Nesse trabalho, são propostas 4 diferentes formulações para o PAM: na primeira não se permite que um mamógrafo diste mais que uma distância máxima R do local de atendimento; na segunda formulação, o deslocamento superior a R é permitido, porém ele é penalizado na função objetivo; nas formulações três e quatro, o objetivo é alterado para tentar minimizar o deslocamento de mulheres, sendo que a terceira formulação não permite que o deslocamento seja superior a R Km.

Souza *et al.* [6] tratam o Problema de Localização de Mamógrafos (PLM) onde o objetivo é definir locais de instalação dentre um conjunto limitado de locais, de tal forma que se maximize a quantidade de mulheres atendidas, obedecendo-se a restrição de máxima distância percorrida. São propostas duas formulações. Na primeira restringe-se que a demanda de uma localidade deve ser integralmente atendida pelo mamógrafo designado ou não será atendida. Já na segunda formulação, uma localidade não precisa ter toda sua demanda atendida, isto é, uma localidade pode ser parcialmente atendida.

Souza *et al.* [7] também trabalham com o PLM. Porém, dado que o problema é NP-difícil [13], eles desenvolveram um algoritmo heurístico baseado na metaheurística VNS (do inglês, *Variable Neighborhood Search* e testaram o algoritmo proposto usando instâncias do estado de Minas Gerais, que possui 853 municípios.

Campos *et al.* [8] também estudam o PLM para o estado de Minas Gerais, porém em sua formulação, admite-se que um mamógrafo designado a determinada localidade não atenda integralmente sua demanda. Observando os resultados encontrados pelos autores, por causa da restrição de máximo deslocamento, não é possível atender 100% da demanda com mamógrafos fixos. Desta maneira, MMUs podem ser utilizadas para atender essa demanda que não foi atendida, tentando aumentar a equidade no acesso à mamografia [9].

Devido à essa desigualdade de localização e às características geográficas do Brasil, com algumas localidades com população pequena e muito distante de centros mais populosos, a possibilidade de mamógrafos móveis possibilita a diminuição dessa desigualdade [14]. Visto esse problema, foi introduzido em [10] o MMURP. Neste problema, o objetivo é maximizar a quantidade de mulheres atendidas e minimizar a distância total percorrida pelas MMUs.

O problema de roteamento de veículos aberto multi-depósito (MDOVRP, da sigla em inglês *Multi-depot Open Vehicle Routing Problem*) consiste em um conjunto de veículos, que saem de um conjunto de depósitos e não precisam retornar a um depósito após a entrega dos bens aos consumidores. Este problema foi proposto pela primeira vez em [15] para resolver o problema de entrega de carnes para açougues, contratando-se veículos e motoristas terceirizados para sua realização.

Liu *et al.* [16] propusera pela primeira vez um modelo de programação inteira

mista para formular o problema MDOVRP, onde utiliza uma variável de decisão com três índices, x_{ijk} , que será 1 se um veículo proveniente de um depósito k , viaja de i até j .

Lalla-Ruiz *et al.* [17] propuseram uma nova formulação, também utilizando 3 índices, onde são adicionadas restrições que tentam deixar a formulação mais apertada, porém o número total de restrições é inferior por causa de mudança nas restrições de eliminação de sub-rotas.

Lalla-Ruiz e Mes [18] propuseram uma nova formulação matemática com dois índices, x_{ij} , que será 1 se um veículo transita de i até j . Esta nova formulação supera as anteriores, tanto em qualidade da solução, quanto em tempo computacional.

Para a solução de problemas discretos de otimização multiobjetivo é comum abordagens metaheurísticas, devida à dificuldade de resolver muitos desses problemas. Liu *et al.* [19] mostram em sua revisão que aproximadamente 25% dos trabalhos publicados na solução de problemas multiobjetivos discretos são baseados em algoritmos genéticos.

Um algoritmo muito popular para problemas multiobjetivos é o NSGA-II (sigla do inglês *Nondominated Sorting Genetic Algorithm II*) [20], que é um algoritmo genético baseado no sorteio por frentes de soluções não dominadas (*rank*), pelo operador de preservação da elite, pelo operador de diversidade (*crowding distance*) e pelo operador de seleção (baseado no *rank* e no *crowding distance*).

2.2 Problemas de Otimização Multiobjetivo

Um problema de otimização é um problema de busca, que visa encontrar uma solução que pertença ao conjunto factível (conjunto de todas as soluções que obedecem às restrições do problema) e que tenha o menor valor de função objetivo dentre as soluções factíveis (considerando um problema de minimização).

Um problema mono objetivo, possui uma única função objetivo e pode ser reescrito da seguinte maneira:

$$\begin{aligned}
 \min_x \quad & f(x) \\
 \text{s.a.} \quad & g_i(x) \leq 0, \quad i = 1, \dots, m \\
 & h_j(x) = 0, \quad j = 1, \dots, p \\
 & x \in \Omega
 \end{aligned} \tag{2.1}$$

Um problema de otimização multiobjetivo é um problema de otimização com múltiplas funções objetivo, onde deseja-se otimizar todas essas funções ao mesmo tempo. Neste tipo de problema não se define uma única solução, mas sim um conjunto delas, de tal maneira, que nenhuma solução do conjunto domine outra

[21]. Sem perda de generalidade, vamos considerar que todos os objetivos são de minimização e podemos definir um problema multiobjetivo usando o conjunto de equações 2.2.

$$\begin{aligned}
\min_x \quad & F(x) = (f_1(x), f_2(x), \dots, f_n(x)) \\
\text{s.a.} \quad & g_i(x) \leq 0, \quad i = 1, \dots, m \\
& h_j(x) = 0, \quad j = 1, \dots, p \\
& x \in \Omega
\end{aligned} \tag{2.2}$$

Definição 1. (*Dominância pareto [21]*) : Um vetor $v = (v_1, v_2, \dots, v_n)$ domina um outro vetor $u = (u_1, u_2, \dots, u_n)$ (representado por $v \preceq u$) se $v_i \leq u_i, \forall i \in \{1, \dots, n\}$ e $\exists k \in \{1, \dots, n\}$ tal que $v_k < u_k$.

Definição 2. (*Ótimo Pareto [21]*) : Uma solução $x \in \Omega$ é um Ótimo Pareto em Ω se e somente se $\nexists x' \in \Omega$ tal que $F(x') \preceq F(x)$.

Definição 3. (*Conjunto Ótimo de Pareto [21]*) : Dado um problema multi-objetivo $F(x)$ com $x \in \Omega$, o conjunto ótimo de Pareto, $\mathcal{P}^*$ é definido como:

$$\mathcal{P}^* = \{x \in \Omega | \nexists x' \in \Omega, F(x') \preceq F(x)\}$$

2.3 Métodos Exatos de Resolução

2.3.1 Combinação Linear com Pesos

Este método de resolução, também conhecido como Método da Soma Ponderada (ou *Weighted Sum Method*), vem da terceira condição de Kuhn-Tucker de soluções não dominadas e consiste em realizar uma combinação linear das diversas funções objetivo, atribuindo pesos $w_i \geq 0$ para cada uma, de tal forma que variando esses pesos e encontrando a solução para cada problema, são encontradas soluções não-dominadas [21]. Aplicando este método ao problema 2.2, obtemos a seguinte formulação:

$$\begin{aligned}
\min_x \quad & F(x) = \sum_{i \in [1, \dots, n]} w_i \cdot f_i(x) \\
\text{s.a.} \quad & g_i(x) \leq 0, \quad i = 1, \dots, m \\
& h_j(x) = 0, \quad j = 1, \dots, p \\
& x \in \Omega
\end{aligned} \tag{2.3}$$

2.3.2 Método ϵ -restrito

Neste método, se escolhe a função objetivo mais prioritária f_p e as outras funções objetivo são adicionadas como restrições, onde cada uma obedece a um limitante superior ϵ_i . A ideia do método consiste em variar os valores dos limitantes ϵ_i , resolver o modelo para cada variação e obter pontos não dominados [21]. Aplicando o método para o problema 2.2, obtemos o seguinte modelo:

$$\begin{aligned} \min_x \quad & F(x) = \sum_{i \in [1, \dots, n]} f_p(x) \\ \text{s.a.} \quad & f_i(x) \leq \epsilon_i, \quad i = 1, \dots, n, \quad i \neq p \\ & g_i(x) \leq 0, \quad i = 1, \dots, m \\ & h_j(x) = 0, \quad j = 1, \dots, p \\ & x \in \Omega \end{aligned} \tag{2.4}$$

2.4 Métodos Heurísticos de Resolução

Metaheurísticas são utilizadas para a resolução de problemas multiobjetivos do mundo real, visto sua capacidade de prover uma habilidade de busca do ótimo e de gerar soluções não dominadas que se aproximam do Pareto ótimo em um tempo computacional razoável. Muitos tipos de algoritmos são utilizados para a resolução destes problemas como por exemplos, algoritmos populacionais, de busca local, meméticos e evolutivos [19].

O uso de algoritmos evolutivos para a resolução de problemas de otimização multiobjetivo foi sugerido nos anos 1960 por Rosenberg [22] na sua tese de doutorado [21]. Porém nesse trabalho, ele não implementou sua sugestão para um problema multiobjetivo. A primeira implementação de um algoritmo evolutivo para a resolução de um problema multiobjetivo é creditado a David Schaffer, que propôs o VEGA (*Vector Evaluation Genetic Algorithm*) em 1984 na sua tese de doutorado [23]. Este é o primeiro dentre uma grande lista de outros algoritmos evolutivos especializados em resolver problemas multiobjetivos [21].

2.4.1 Nondominated Sorting Genetic Algorithm II

O NSGA-II é uma metaheurística populacional proposta por Deb *et al.* [24]. Trata-se de um algoritmo genético elitista, que utiliza a ordenação baseada em não dominância; combina os pais e os filhos de uma geração e escolhe-se os melhores para a próxima geração, considerando a função fitness e uma função de espalhamento [24].

O algoritmo NSGA-II é direto, como pode ser observado no Algoritmo 1. Nas linhas 1 a 4, uma população P_0 é inicializada, sorteada em frentes, usando o Al-

goritmo 2, e a partir dela uma população Q_0 é criada. Após essa inicialização, o algoritmo consistirá em um laço de *max_generations* iterações. Na linha 6 é realizada a união da população atual P_t com a população criada à partir desta, Q_t , obtendo-se uma população intermediária $R_t = P_t \cup Q_t$ de tamanho $2N$. Na linha 7, esta população ampliada é sorteada em frentes, utilizando o Algoritmo 2 e na linha 8 a próxima população, $P_{t+1} = \emptyset$, é inicializada. No laço de 10 a 13 a próxima população é construída, priorizando-se as soluções que pertençam a uma frente mais dominante. Desta forma, se a primeira frente F_1 de soluções não dominadas, tenha $|F_i| \leq N$, então todas as soluções desta frente pertencerão à próxima geração, P_{t+1} . Isto se repetirá para as outras frentes até que a união de um nova frente exceda N . Quando isso ocorrer, na linha 14 e as soluções desta nova frente serão ordenadas em ordem decrescente de acordo com a crowding distance (3) e serão selecionadas as melhores soluções (linha 15), tal que o tamanho da população da próxima geração seja N . Na linha 16, à partir da população P_{t+1} , uma nova população Q_{t+1} é criada e na linha 18, o contador de iterações é incrementado.

Algorithm 1 NSGA-II

```

1:  $t = 0$ 
2:  $P_0 = \text{initialize-population}()$ 
3:  $F = \text{fast-non-dominated-sort}(P_0)$ 
4:  $Q_0 = \text{make-new-pop}(P_0)$ 
5: while  $t < \text{max\_generations}$  do
6:    $R_t = P_t \cup Q_t$ 
7:    $F = \text{fast-non-dominated-sort}(R_t)$ 
8:    $P_{t+1} = \emptyset$ 
9:   repeat
10:    crowding-distance-assignment( $F_i$ )
11:     $P_{t+1} = P_{t+1} \cup F_i$ 
12:     $i = i + 1$ 
13:   until  $|P_{t+1}| + |F_i| \leq N$ 
14:    $\text{Sort}(F_i, \prec_n)$ 
15:    $P_{t+1} = P_{t+1} \cup F_i[1 : (N - |P_{t+1}|)]$ 
16:    $Q_{t+1} = \text{make-new-pop}(P_{t+1})$ 
17:
18:    $t = t + 1$ 
19: end while

```

Ordenação por Dominância

O algoritmo de definição do *rank* de dominância é uma evolução do utilizado no NSGA, sendo um algoritmo com custo computacional de $O(MN^2)$ operações. Para o entendimento de seu funcionamento pode-se dividir o algoritmo em duas partes:

- Para cada solução $p \in P$, calcula quantas soluções, n_p , a dominam e o conjunto S_p de soluções que p domina;
- A partir dos indicadores calculados na primeira parte, para cada solução $p \in P$ que $n_p = 0$, sabemos que ela pertence à primeira frente de dominância e devemos percorrer o conjunto $q \in S_p$, diminuindo n_q em 1 e caso $n_q = 0$, esta solução será adicionada à próxima frente. Isso é realizado até que se identifiquem todas as frentes [24].

Na linha 2 inicializa-se a primeira frente F_1 . O laço das linhas 3 a 17 itera por todos os elementos de P e em 4 e 5 inicializa-se S_p e n_p , que respectivamente armazenam todos os elementos que são dominados por p e contam quantos elementos dominam p . Nas linhas 6 a 12 itera-se por todas as outras soluções de $q \in P$, acrescentando-se q ao conjunto S_p , caso $p \prec q$ (linha 8) e caso $q \prec p$, incrementa-se n_p (linha 10). Caso, $n_p = 0$ (linha 13) então p pertencerá à primeira frente e seu rank será 1 (linhas 14 e 15). Na linha 18 i , que indica a frente pela qual se está iterando é inicializado. O laço que se inicia na linha 19 se repetirá enquanto houver elementos na frente F_i . Na linha 20 inicializa-se o conjunto Q , que receberá as soluções da próxima frente. Para cada elemento $p \in F_i$ (linha 21) itera-se pelos elementos $q \in S_p$, ou seja, são dominados por p (linha 22) e na linha 23 decrementa-se n_q . Caso n_q seja zero (linha 24), o rank de q será $i + 1$ e q será adicionado ao conjunto Q . Na linha 30 o contador de frentes é incrementado e na linha 31 a frente F_i recebe Q .

Preservação de Diversidade

Outra evolução em relação ao NSGA é a mudança na forma de preservar-se a diversidade, não sendo mais necessária a definição de um fator de compartilhamento. Para manter a diversidade é utilizada uma função de densidade, que estima a densidade dos pontos. Na linha 2 inicializa-se l com a quantidade de elementos de L e na linha 3 a distância de cada elemento de L é inicializada com 0. Para cada objetivo m (linha 5) é realizado a ordenação dos elementos de L , considerando o objetivo m (linha 6) e o primeiro e o último elemento, nessa ordenação, tem sua distância igual a infinito (linhas 7 e 8). Para cada elemento intermediário de L , calcula-se a diferença normalizada dos valores adjacentes e soma-se à sua distância acumulada (linhas 9 a 10) [24].

Algorithm 2 Ordenação por dominância

```
1: function FAST-NON-DOMINATED-SORT( $P$ )
2:    $F_1 = \emptyset$ 
3:   for  $p \in P$  do
4:      $S_p = \emptyset$ 
5:      $n_p = 0$ 
6:     for  $q \in P$  do
7:       if  $p \prec q$  then
8:          $S_p = S_p \cup \{q\}$ 
9:       else if  $q \prec p$  then
10:         $n_p = n_p + 1$ 
11:      end if
12:    end for
13:    if  $n_p = 0$  then
14:       $p_{rank} = 1$ 
15:       $F_1 = F_1 \cup \{p\}$ 
16:    end if
17:  end for
18:   $i = 1$ 
19:  while  $F_i \neq \emptyset$  do
20:     $Q = \emptyset$ 
21:    for  $p \in F_i$  do
22:      for  $q \in S_p$  do
23:         $n_q = n_q - 1$ 
24:        if  $n_q = 0$  then
25:           $q_{rank} = i + 1$ 
26:           $Q = Q \cup \{q\}$ 
27:        end if
28:      end for
29:    end for
30:     $i = i + 1$ 
31:     $F_i = Q$ 
32:  end while
33: end function
```

Algorithm 3 Crowding Distance

```
1: function CROWDING-DISTANCE-ASSIGNMENT( $L$ )
2:    $l = |L|$ 
3:   for  $i \in [1, l]$  do  $L[i]_{distance} = 0$ 
4:   end for
5:   for all objective  $m$  do
6:      $L = \text{sort}(L, m)$ 
7:      $L[1]_{distance} = \infty$ 
8:      $L[l]_{distance} = \infty$ 
9:     for  $i \in [2, l - 1]$  do
10:       $L[i]_{distance} = L[i]_{distance} + (L[i + 1].m - L[i - 1].m) / (f_m^{max} - f_m^{min})$ 
11:    end for
12:   end for
13: end function
```

Capítulo 3

Metodologia

Este capítulo é dividido em duas seções: na primeira, descrevemos o modelo matemático e como ele é utilizado para a resolução do MMURP e na segunda, descrevemos como o método heurístico para sua resolução foi implementado.

3.1 Método Exato

3.1.1 Modelo Matemático

Para modelar o MMURP, Rosa *et al.* [10] consideraram as informações sobre o funcionamento da MMU localizada no município de Patrocínio, Minas Gerais, fornecidas pelo Hospital do Câncer de Patrocínio Dr. José Figueiredo. O problema consiste em definir a rota de um conjunto de MMUs, onde a localização de um depósito $k \in D$ é conhecido e para cada depósito $k \in D$ existe a disponibilidade de no máximo N_{MMUs}^k MMUs, sendo Q a capacidade de uma MMU realizar exames durante sua rota.

Existe um conjunto D de depósitos, que é a base das MMUs e um conjunto de N localidades, cada um com demanda q_j . Cada localidade pode ser visitada no máximo uma única vez e toda a sua demanda deve ser atendida. Entre duas localidades há uma distância d_{ij} a ser percorrida e existe uma distância máxima (*distMax*) permitida para uma MMU viajar entre duas localidades, exceto para o primeiro trecho, saindo de um depósito, onde não existe esse limite. O roteamento é aberto, ou seja, as MMUs não precisam retornar a nenhum depósito após realizar a sua rota [10]. O conjunto $V = N \cup D$ reúne o conjunto de todas as localidades candidatas a serem atendidas e os depósitos das MMUs.

O modelo proposto é uma adaptação da formulação do MDOVRP_{2i-flv} proposto por Lalla-Ruiz e Mes [18] para o problema MDOVRP. Foram realizadas nessa formulação modificações para possibilitar que algumas localidades não fossem atendidas e que se restringisse o número de veículos por depósito, assim como adicionada

a restrição para limitar a distância viajada entre duas localidades atendidas. As variáveis de decisão do modelo proposto são as seguintes:

x_{ij} 1 se um veículo viaja do nó $i \in V$ até o nó $j \in V$, 0 caso contrário.

w_j 1 se a demanda da cidade da cidade $j \in V$ é atendida, 0 caso contrário.

$u_{ij} > 0$ define um limitante superior na “carga” que sai de $i \in V$ para $j \in V$

O modelo que representa a versão multiobjetivo do MMURP é dado pelas Equações (1) a (15):

$$Min \sum_{i \in V} \sum_{j \in V} c_{ij} x_{ij} \quad (1)$$

$$Max \sum_{j \in N} q_j w_j \quad (2)$$

$$\sum_{i \in V, i \neq j} x_{ij} = w_j, \quad \forall j \in N \quad (3)$$

$$\sum_{j \in N} x_{kj} \leq N_{MMU_s}^k, \quad \forall k \in D \quad (4)$$

$$\sum_{j \in N} q_j \cdot w_j \leq \sum_{k \in D} N_{MMU_s}^k \cdot Q \quad (5)$$

$$c_{ij} \cdot x_{ij} \leq dist_{max}, \quad \forall i \in V, j \in V, i \neq j \quad (6)$$

$$\sum_{i \in V, i \neq j} x_{ij} - \sum_{i \in N, i \neq j} x_{ji} \geq 0, \quad \forall j \in N \quad (7)$$

$$x_{ij} + x_{ji} \leq 1, \quad \forall i, j \in V, i \neq j \quad (8)$$

$$\sum_{k \in D} \sum_{j \in V} x_{jk} = 0 \quad (9)$$

$$\left(\sum_{i \in V, i \neq j} u_{ij} - \sum_{i \in V, i \neq j} u_{ji} \right) - q_j \geq -Q \cdot (1 - w_j), \quad \forall j \in N \quad (10)$$

$$(Q - q_i) \cdot x_{ij} \geq u_{ij}, \quad \forall i, j \in N \quad (11)$$

$$Q \cdot x_{kj} \geq u_{kj}, \quad \forall k \in D, j \in N \quad (12)$$

$$x_{ij} \in \{0, 1\}, \quad \forall i, j \in V \quad (13)$$

$$u_{ij} \geq 0, \quad \forall i, j \in V \quad (14)$$

$$w_j \in \{0, 1\}, \quad \forall j \in N \quad (15)$$

A equação (1) traz a função objetivo, que tenta minimizar a distância total percorrida e a (2) tenta maximizar a demanda total atendida. A restrição (3) diz que a demanda de uma localidade j será atendida somente se existir um veículo que chegue em j . As restrições (4) e (5) estabelecem um limite superior para a quantidade de veículos que saem de cada depósito e da demanda que pode ser atendida. A restrição (6) limita a máxima distância que pode ser percorrida entre duas localidades, exceto no primeiro trecho, saindo de algum depósito. A restrição (7) define que quando uma localidade é visitada, ela pode ser o fim da rota ou ir para outra localidade, desde que não seja um depósito. A restrição (8) proíbe que seja utilizado ao mesmo tempo o caminho ij e o ji e a restrição (9) restringe o retorno de qualquer veículo a algum depósito. Pode-se dividir a restrição (10) em dois casos: se w_j é igual a 0 a restrição não é relevante; caso w_j seja igual a 1, ela exige que exista capacidade suficiente para se atender a demanda da localidade j . As restrições (11) e (12) são limitantes superiores para as variáveis u .

3.1.2 Modelo mono-objetivo para resolução do MMURP

Utilizaremos esta formulação e o método de combinação linear com pesos para a busca dos pontos da fronteira de Pareto. Para realizarmos um tipo de normalização das funções objetivos originais, adicionamos os parâmetros $A^{-1} = \sum_{i \in V} \sum_{j \in V, j > i} c_{ij}$ e $B^{-1} = \sum_{j \in N} q_j$ e um fator $\alpha \in [0, 1]$ que será modificado para podermos calcular a aproximação da fronteira de Pareto do problema.

$$\begin{aligned} \min_x \quad & \alpha A \sum_{i \in V} \sum_{j \in V} c_{ij} x_{ij} - (1 - \alpha) B \sum_{j \in N} q_j w_j \\ \text{s.a.} \quad & (3) \text{ a } (15) \end{aligned} \quad (3.1)$$

3.2 Método Heurístico

3.2.1 Representação da solução

Para a representação de uma solução, será utilizado a representação por ordem de visitação, de tal forma que qualquer solução será representada como uma permutação

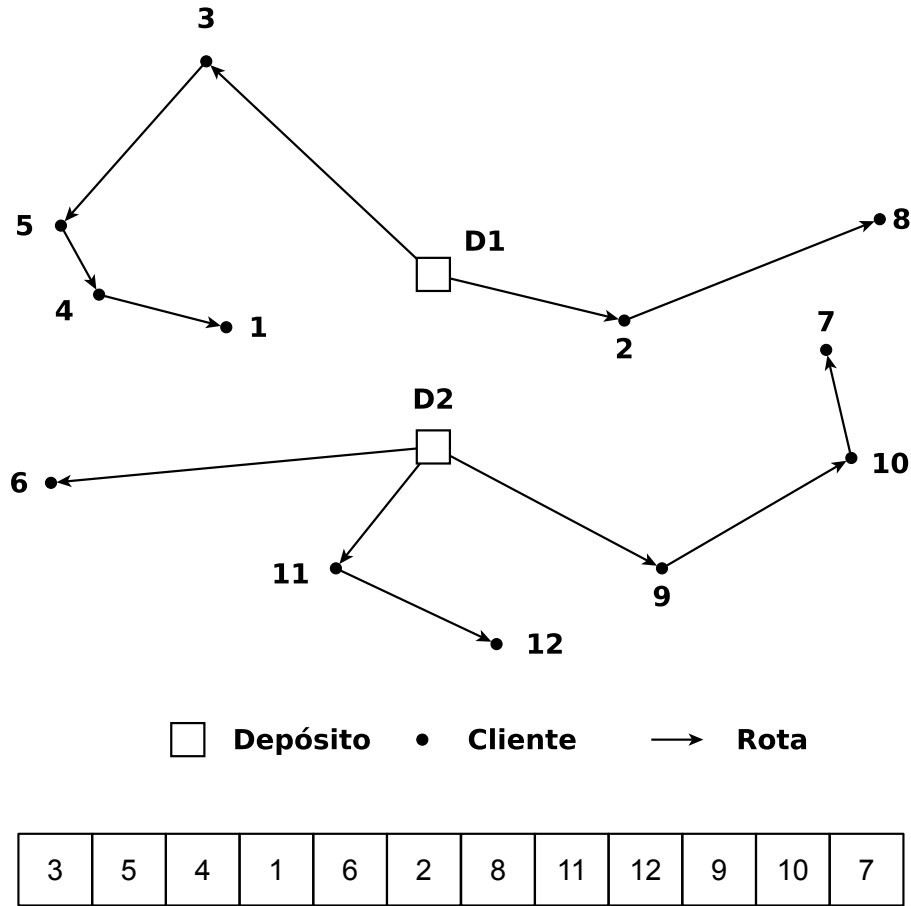


Figura 3.1: Exemplo de representação de um indivíduo

dos n clientes, tal qual a solução de um Problema do Caixeiro Viajante (PCV). Prins [25] utilizou essa representação para resolver o problema de roteamento com limitação de distância (*DVRP*). Associada à representação, ele utilizou um método de determinação da rota ótima (*splitting*), dada uma permutação qualquer, isso foi realizado utilizando um algoritmo de Bellman. Na figura 3.1 pode-se observar como uma solução do MMURP poderia ser representada.

3.2.2 Função de avaliação

Para se calcular o valor das funções objetivos, foi utilizado um algoritmo de divisão, baseado no algoritmo de splitting para o problema de roteamento com frota fixa heterogênea (*HVRP*) presente na revisão de [26].

A partir do algoritmo encontrado em Prins *et al.* [26], foi construído o Algoritmo 4 para realizar o splitting de uma solução do MMURP.

Dada uma sequência de cidades a serem atendidas, de tamanho n , o Algoritmo 4 cria um subgrafo de tamanho $n + 1$ e em cada nó i obtêm-se o conjunto de

Algorithm 4 Algoritmo de splitting para o MOMMURP

```
1:  $\Lambda(0) \leftarrow \{(0, 0|0, \dots, 0)\}$ 
2: for  $i \leftarrow 1$  to  $n$  do  $\Lambda(i) = \emptyset$ 
3: end for
4: for  $i \leftarrow 1$  to  $n$  do
5:   for all depot  $d_k$  do
6:     for all label  $V = (\phi, \pi|r_1, r_2, \dots, r_p) \in \Lambda(i-1)$  do
7:        $j \leftarrow i$ 
8:       tour_demand  $\leftarrow 0$ 
9:       repeat
10:        tour_demand  $\leftarrow$  tour_demand +  $q(T_j)$ 
11:        if  $i = j$  then
12:          if  $r_k + 1 \leq a_k$  then
13:            tour_distance  $\leftarrow c(d_k, T_i)$ 
14:             $W \leftarrow (\phi + \text{tour\_distance}, \pi + \text{tour\_demand}|r_1, \dots, r_k +$ 
15:               $1, \dots, r_p)$ 
16:            if no label in  $\Lambda(j)$  dominates  $W$  then
17:              delete in  $\Lambda(j)$  all labels dominated by  $W$ 
18:               $\Lambda(j) \leftarrow \Lambda(j) \cup \{W\}$ 
19:            end if
20:          else
21:            stop  $\leftarrow false$ 
22:          end if
23:        else
24:          if  $c(T_{j-1}, T_j) \leq \text{dist}_{\text{Max}} \wedge (\text{tour\_demand} \leq Q_{\text{Max}})$  then
25:            stop  $\leftarrow true$ 
26:            tour_distance  $\leftarrow$  tour_distance +  $c(T_{j-1}, T_j)$ 
27:             $W \leftarrow (\phi + \text{tour\_distance}, \pi + \text{tour\_demand}|r_1, \dots, r_p)$ 
28:            if no label in  $\Lambda(j)$  dominates  $W$  then
29:              delete in  $\Lambda(j)$  all labels dominated by  $W$ 
30:               $\Lambda(j) \leftarrow \Lambda(j) \cup \{W\}$ 
31:            end if
32:          else
33:            stop  $\leftarrow false$ 
34:          end if
35:        until  $(j > n) \vee (\text{stop} = false)$ 
36:      end for
37:    end for
38:  end for
39: end for
```

soluções não dominadas, que atendam até os $i - 1$ primeiros clientes da sequência. Na linha 1 inicia-se o nó 0 e na linha 2 inicia-se com vazio o conjunto de soluções não dominadas, $\Lambda(i)$. Para cada nó i (linha 4) itera-se por todos os depósitos, d_k , e para cada label de $\Lambda(i - 1)$ (linha 6), inicia-se uma nova rota que será controlada pela variável j (linhas 7 e 8). Na linha 9 inicia-se um laço que será finalizado quando $j > n$ ou $stop = false$. Na linha 10 incrementa-se a demanda acumulada na rota atual com a demanda da j -ésima cidade. Caso a cidade atual seja a primeira da rota (linha 11), verifica-se se é possível utilizar um novo veículo do depósito d_k : caso não seja, $stop \leftarrow false$ e caso seja, inicia-se a distância percorrida nessa rota (linha 13), incrementa-se a distância e a demanda atendida do label atual V (linha 14), obtendo-se W . Se nenhum label em $\Lambda(j)$ domina W , então todos os labels em $\Lambda(j)$ que sejam dominados por W serão removidos e W será inserido em $\Lambda(j)$ (linhas 15 a 17). Caso a cidade atual não seja a primeira da rota (linha 22) verifica se a distância entre a cidade anterior e essa, não é superior à $dist_{Max}$ e se a demanda acumulada na rota atual não é superior à capacidade máxima. Caso a condicional da linha 23 seja falsa, $stop \leftarrow false$ e caso seja verdadeira, incrementa-se a distância percorrida na rota atual (linha 25), atualiza-se a demanda atendida e a distância percorrida do label atual (linha 26). Na linha 27 verifica se W não é dominado por nenhum label de $\Lambda(j)$, caso não seja, remove-se todos os labels dominados por W (linha 28) e insere-se W em $\Lambda(j)$. Na linha 35 incrementa-se j e na 36 verifica se a rota atual foi finalizada.

3.2.3 Operadores de recombinação de soluções

Como a nossa representação é a mesma do PCV, pode-se utilizar operadores clássicos já usados neste problema. Para o operador mutação, foi escolhido realizar a troca de duas posições quaisquer de um indivíduo. Como uma solução do nosso problema não possui extremidades, para se realizar a combinação de duas soluções, pode-se escolher o operador OX, como proposto por Prins [25], que constrói um filho mantendo uma subsequência de um dos pais e mantendo a ordem relativa das localidades do outro pai.

Neste operador de crossover, o filho é construído mantendo-se uma subsequência de uma rota associada a um dos pais e a sequência relativa das cidades do outro pai [27]. Considere dois pais com as seguintes sequências e com dois cortes:

$$p_1 = (5 \ 3 \ 0 \mid 8 \ 7 \ 1 \ 6 \mid 2 \ 4)$$

$$p_2 = (3 \ 6 \ 0 \mid 4 \ 1 \ 2 \ 5 \mid 7 \ 8)$$

Entre os dois cortes a subsequência de um dos pais será mantida, como se segue:

$$O_1 = (X \ X \ X \mid 8 \ 7 \ 1 \ 6 \mid X \ X)$$

$$O_2 = (X \ X \ X \mid 4 \ 1 \ 2 \ 5 \mid X \ X)$$

Para a construção do filho O_1 , a sequência de cidades à partir do segundo corte de p_2 (7 – 8 – 3 – 6 – 0 – 4 – 1 – 2 – 5) serão adicionadas nessa sequência, excluindo-se aquelas que já estejam presentes em O_1 . Desta maneira, serão removidas as cidades 8, 7, 1 e 6 desta sequência e as cidades que sobraram serão adicionadas à O_1 , obtendo-se:

$$O_1 = (4 \ 2 \ 5 \mid 8 \ 7 \ 1 \ 6 \mid 3 \ 0)$$

Realizando o mesmo procedimento para o filho O_2 , temos:

$$O_2 = (8 \ 7 \ 6 \mid 4 \ 1 \ 2 \ 5 \mid 3 \ 0)$$

3.2.4 Determinação dos parâmetros da metaheurística

Para a utilização do NSGA-II é necessário determinar o tamanho da população, a quantidade de gerações (n é o número de cidades da instância) e a taxa de mutação. Após alguns testes preliminares, foi utilizado o pacote irace [28] para calibrar os valores desses parâmetros, partindo da faixa de valores atribuídas aos parâmetros conforme Tabela 3.1.

Tabela 3.1: Faixa de valores atribuídos aos parâmetros da calibração

# população	{50, 75, 100}
# gerações	{5n, 10n, 15n, 20n}
taxa de mutação	{0.1, 0.2, 0.4}

Após a execução do irace, que demorou aproximadamente 50 horas, encontrou-se o seguinte conjunto de valores dos parâmetros que dominaram as outras configurações: {população: 100, geração: 20n, μ : 0.4}. Com esses parâmetros do NSGAII definidos, o algoritmo foi executado 10 vezes para cada instância de teste.

Capítulo 4

Experimentos Computacionais

Neste capítulo temos a descrição das instâncias utilizadas para comparação dos métodos exato e heurístico; detalhes do ambiente onde os testes foram realizados; os resultados encontrados e uma breve discussão.

4.1 Instâncias

Para a realização dos experimentos computacionais foram adaptadas as instâncias utilizadas por Liu *et al.* [16], por Lalla-Ruiz *et al.* [17] e por Lalla-Ruiz e Mes [18] para o problema MDOVRP. O MMURP tem uma limitação no número de veículos por depósito, o que não acontece no MDOVRP. Além disso, no MMURP, existe a possibilidade de não se atender a toda demanda, desta forma foram criadas instâncias onde o número de veículos disponíveis não conseguem atender toda a demanda.

id	n	d	Q	# MMUs	# MMU/depósito
1	50	4	80	4	1, 1, 1, 1
2	50	4	80	6	2, 2, 1, 1
3	50	4	80	10	3, 3, 2, 2
4	50	4	160	4	1, 1, 1, 1
5	50	4	160	5	2, 1, 1, 1
6	100	2	100	3	2, 1
7	100	2	100	6	3, 3
8	100	2	100	9	5, 4
9	100	2	100	15	8, 7
10	100	4	100	6	2, 2, 1, 1
11	100	4	100	9	3, 2, 2, 2
17	249	2	500	10	5, 5

Tabela 4.1: Instâncias utilizadas nos testes

Para definir o número de veículos por depósito, foi calculado o limitante inferior do número de veículos suficiente para atender à toda demanda dada por

$\lceil \sum_{j \in N} q_j / N_{MMU} \rceil$ e à partir desse valor foram criadas instâncias onde o número total de veículos era uma fração desse limitante. Esses veículos foram distribuídos de tal forma que todos os depósitos tivessem ao mínimo um veículo. As instâncias utilizadas nos experimentos encontram-se na Tabela 4.1.

4.2 Método Exato

4.2.1 Implementação

Os modelos matemáticos foram implementados utilizando python3, o pacote Python-MIP [29] e como resolvidor de uso geral, foi utilizado o Gurobi [30]. Todos os experimentos foram realizados em um computador com processador Intel Core i7-4790 CPU 3.60GHz x 4.

Para a resolução com o método exato foi utilizado o método da combinação linear com pesos e estabelecido um limite de 120 segundos por modelo MIP a ser resolvido.

4.2.2 Resultados

O modelo definido em 3.1 foi resolvido usando 30 diferentes valores de α igualmente espaçados entre 0 a 1. Os resultados encontram-se na Tabela 4.2.

Tabela 4.2: Resultados dos testes do método exato

id	n	m	Q	v	Soluções Não Dominadas	Hipervolume	Tempo (s)
1	50	4	80	4	5	1,48E+06	3.096,73
2	50	4	80	6	5	2,18E+06	3.418,89
3	50	4	80	10	6	3,43E+06	3.320,78
4	50	4	160	4	8	2,86E+06	3.483,10
5	50	4	160	5	5	3,41E+06	2.680,61
6	100	2	100	3	4	3,01E+06	1.272,24
7	100	2	100	6	6	5,97E+06	3.490,14
8	100	2	100	9	5	8,83E+06	3.492,16
9	100	2	100	15	5	1,39E+07	3.456,93
10	100	4	100	6	5	5,98E+06	3.489,89
11	100	4	100	9	8	8,84E+06	3.494,75
17	249	2	500	10	7	1,49E+08	3.532,16

4.3 Método Heurístico

4.3.1 Implementação

A implementação do algoritmo NSGAIi foi realizada utilizando C++, usando o compilador gcc 9.4.0, com as flags -Wall -O3 -march=native. Para realizar o cálculo do hipervolume foi utilizada a biblioteca python pygmo.

4.3.2 Resultados

Na Tabela 4.3 temos os resultados obtidos após 10 execuções do algoritmo. Para cada execução do algoritmo é obtida uma população de tamanho fixo, onde cada indivíduo desta população representa uma ordem de visitação, de tal forma que cada indivíduo pode gerar um conjunto de possíveis soluções, onde varia-se a taxa de atendimento. Desta forma, após a execução do algoritmo, foi utilizado o Algoritmo de splitting (4) para a geração das soluções com menor taxa de atendimento.

Tabela 4.3: Resultados dos testes usando o NSGAIi

id	n	m	Q	v	Soluções Não Dominadas	Hipervolume	Tempo (s)
1	50	4	80	4	53,70	1,49E+06	26,99
2	50	4	80	6	76,90	2,20E+06	117,24
3	50	4	80	10	117,10	3,46E+06	2954,78
4	50	4	160	4	80,50	2,89E+06	83,15
5	50	4	160	5	105,30	3,42E+06	238,90
6	100	2	100	3	37,80	3,01E+06	20,42
7	100	2	100	6	73,10	5,97E+06	90,57
8	100	2	100	9	121,60	8,84E+06	314,19
9	100	2	100	15	183,20	1,39E+07	2835,97
10	100	4	100	6	79,30	5,98E+06	390,99
11	100	4	100	9	123,60	8,86E+06	3902,23
17	249	2	500	10	192,38	1,75E+08	2002,38

4.4 Discussão

Observando as figuras 4.1, 4.2 e 4.3 é possível ver que o processo de geração de soluções com menor taxa de atendimento, aumenta consideravelmente o hipervolume, o que é uma consequência óbvia, já que estamos gerando novas soluções não dominadas. Também é possível observar que após o pós-processamento o NSGAIi supera o hipervolume do método exato para a maioria das instâncias; o que não acontece antes do pós-processamento, onde a metaheurística superou o método exato somente na instância com 249 cidades.

Comparando-se as tabelas 4.2 e 4.3 podemos observar que a quantidade de soluções não dominadas para o NSGAI (após o pós-processamento) é bastante superior em relação ao método exato.

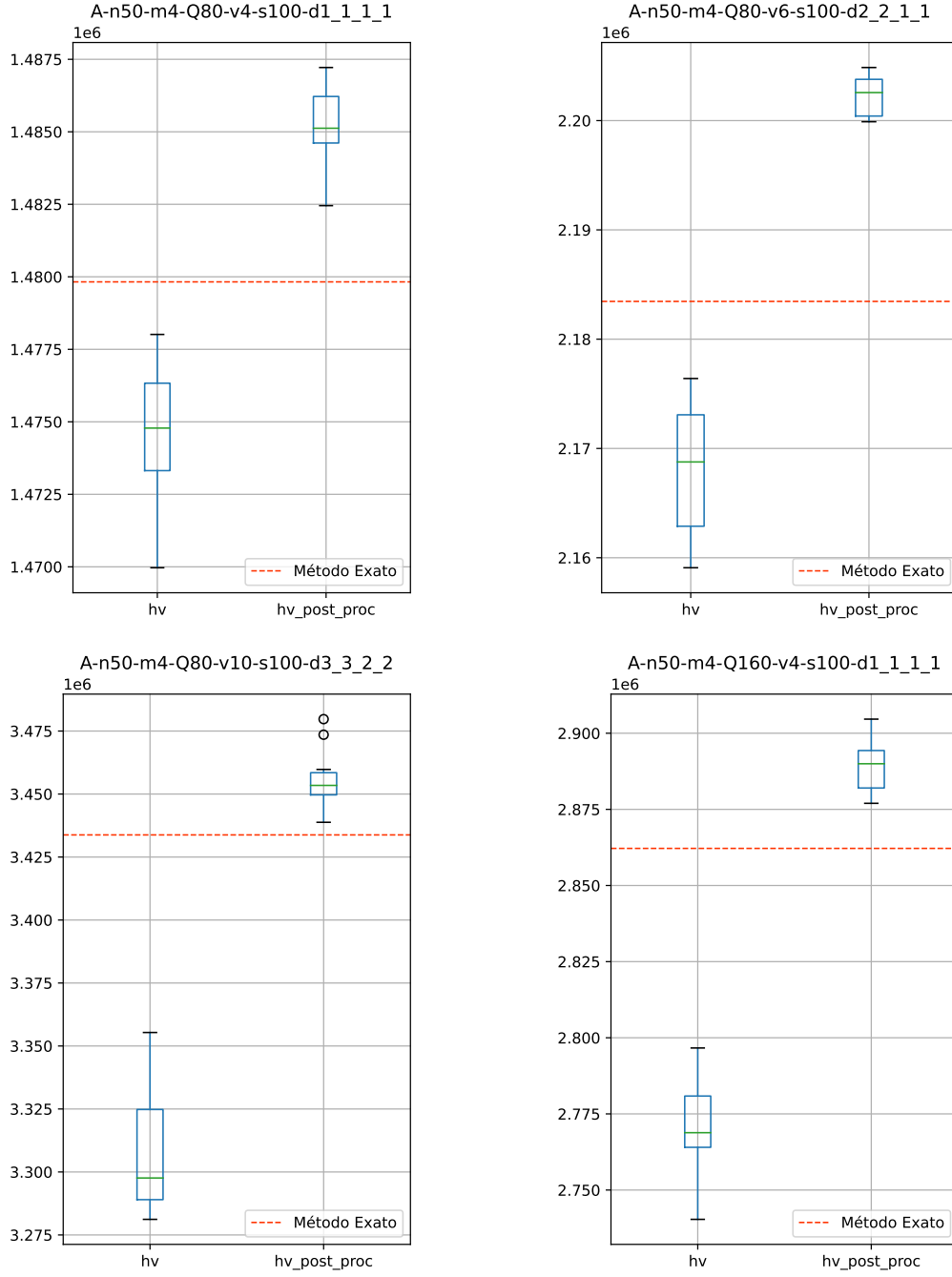


Figura 4.1: Comparação do hipervolume

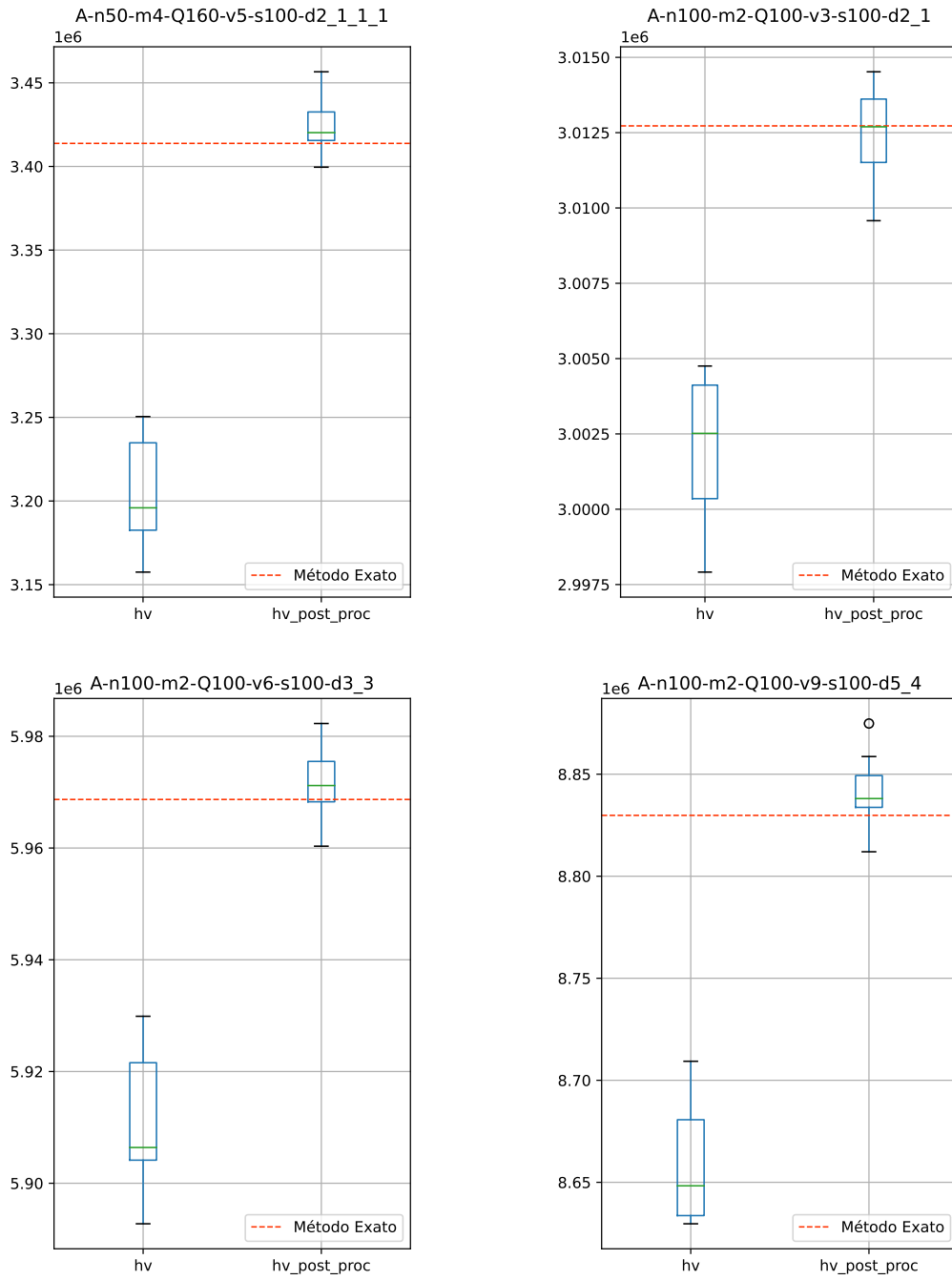


Figura 4.2: Comparação do hipervolume

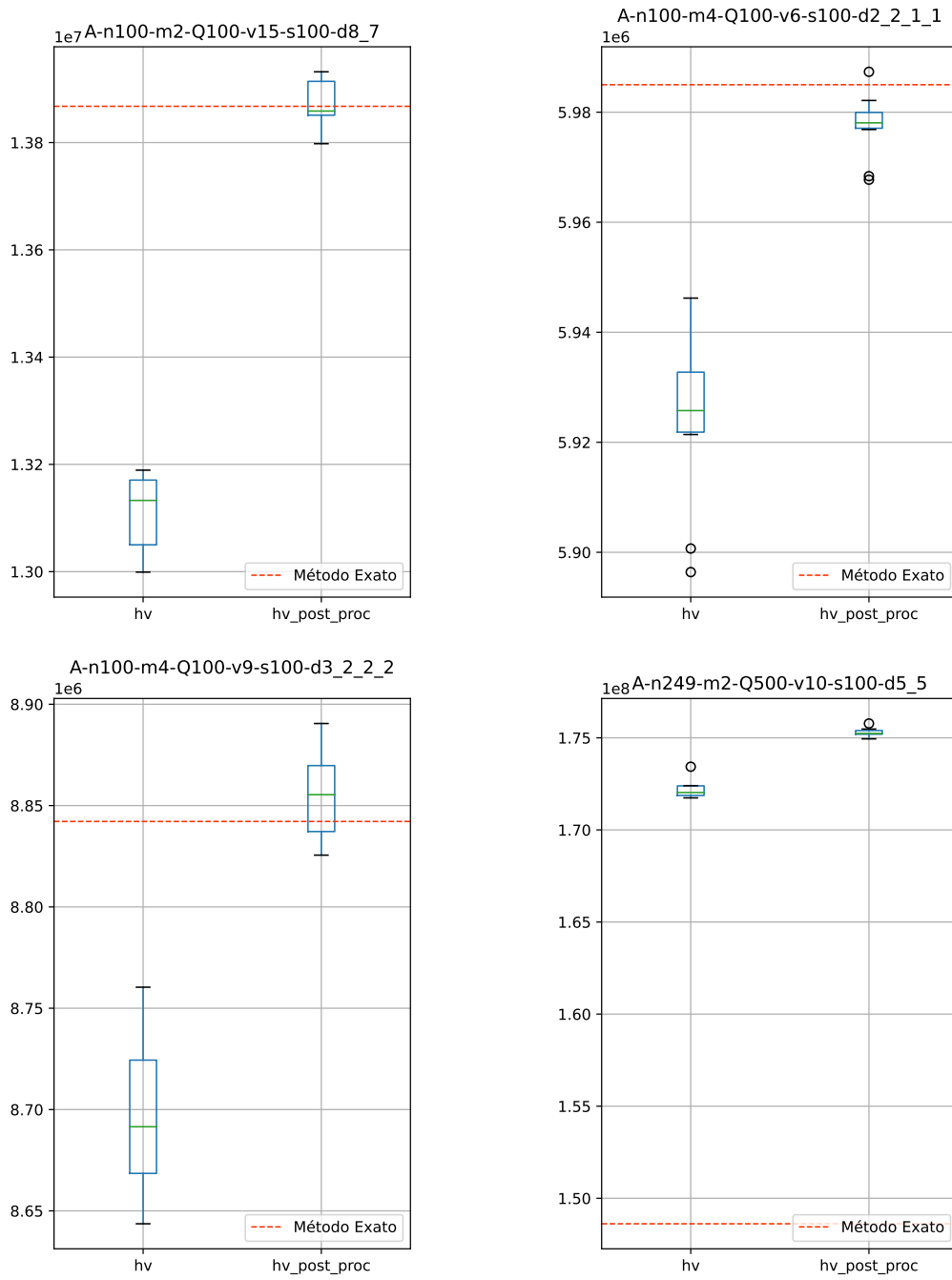


Figura 4.3: Comparação do hipervolume

Capítulo 5

Conclusões e Trabalhos Futuros

Neste trabalho fizemos uma primeira aproximação na utilização de um método exato e um heurístico para a resolução do MMURP. Foi implementado a versão pura do NSGAIL, sem nenhum processo de busca local e com população inicial sendo gerada de maneira aleatória, chegando em resultados competitivos em relação ao método exato. Como método exato, somente utilizamos o método de combinação linear com pesos e não estudamos variações na formulação que possam melhorar a performance do método exato.

As instâncias que foram utilizadas neste trabalho, são pequenas quando comparadas com o problema para o estado de Minas Gerais, que possui 853 municípios. Desta forma, será necessário avaliar melhorias no algoritmo de splitting; a utilização de outros operadores de mutação, que acelerem o processo de busca; e a utilização de pós processamento.

A seguir segue o cronograma com atividades que serão realizadas até a data da defesa.

5.1 Cronograma

1. Aperfeiçoar a implementação.
2. Avaliar a utilização de heurísticas para geração da população inicial.
3. Implementar versão $O(nm)$ do algoritmo de splitting presente em [31].
4. Testar versões do algoritmo de splitting que gere soluções sub-ótimas.
5. Testar novos operadores de mutação.
6. Avaliar a utilização de heurística de pós-processamento.

7. Realizar testes quanto à formulação do problema e testar se as restrições que deixam a formulação mais apertada, na formulação do [18], podem ser adaptadas para o problema multiobjetivo e ajudam na sua resolução.
8. Implementar o método da ϵ -restrição.
9. Fazer revisão de métricas de qualidade de algoritmos multiobjetivos e implementá-las.
10. Escrever e submeter artigos.
11. Escrever a dissertação.

	2022		2023						
	Nov	Dez	Jan	Fev	Mar	Abr	Mai	Jun	Jul
1									
2									
3									
4									
5									
6									
7									
8									
9									
10									
11									

Capítulo 6

Rascunho

6.1 Objetivos

O problema de roteamento de unidades móveis de mamografia (Mobile Mammography Unit Routing Problem MMURP) consiste em determinar quais cidades serão atendidas e qual será a ordem de visita, de tal forma a maximizar o atendimento à demanda por exames de mamografia e minimizar a distância total percorrida, obedecendo às restrições de capacidade de cada unidade e máxima distância percorrida entre duas cidades.

Este trabalho trata do problema de roteamento de unidades móveis de mamografia (MMURP) para o estado de Minas Gerais. Neste problema, temos um conjunto de cidades, cada uma com uma quantidade limitada de caminhões com mamógrafos; e um conjunto de cidades com uma demanda por exames de mamografia para serem realizados. O objetivo é decidir quais cidades atender e quais trajetos cada caminhão deve percorrer, de tal forma a maximizar o atendimento à demanda existente e minimizar a distância total percorrida.

6.2 Revisão

O câncer de mama é o câncer que tem a maior incidência entre mulheres e a que causa a maior mortalidade, excetuando-se o câncer de pele não melanoma [1]. Em 2020, a estimativa foi que 29.7% dos casos de câncer e 16.5% das mortes em mulheres por causa do câncer, se deram pelo câncer de mama (quando se exclui os casos de câncer de pele não melanoma) [11].

A falta de conhecimentos sobre a doença e as dificuldades de acesso das mulheres aos métodos diagnósticos estão relacionados com a chegada das pacientes em estágios mais avançados do câncer de mama, que pioram o prognóstico. Logo, políticas públicas que possibilitem o acesso das mulheres ao diagnóstico precoce são essenciais

diminuir a mortalidade por causa do câncer de mama, já que quanto antes um tumor invasivo é identificado e tratado, maior a probabilidade de cura [1].

Como o diagnóstico precoce é essencial para a diminuição da mortalidade por câncer de mama, Andrade et al. realizaram um estudo em Minas Gerais, relacionando o não acesso ao exame de mamografia e a localização espacial desses equipamentos. Nesse estudo, observaram que o aumento em 1 hora no tempo de deslocamento até um equipamento diminui, em média, 28% a probabilidade da realização do exame.

Amaral et al. mostram no seu trabalho que existe uma desigualdade na localização dos mamógrafos no Brasil, sendo que a quantidade total de mamógrafos seria suficiente para atender toda a demanda no país. Porém, quando se limita a distância de oferta do exame a 60 km ou 60 minutos de deslocamento (conforme indicado pelo Ministério da Saúde [32]) identificam-se regiões que não são atendidas e regiões que tem mais equipamentos que o necessário.

Observando o diagnóstico desses trabalhos, de que existe um problema na alocação de aparelhos de mamografia na rede pública de saúde, Corrêa et al. estudaram o Problema de Alocação de Mamógrafos (PAM), buscando verificar a possibilidade de uma distribuição mais racional dos mamógrafos existentes. Neste trabalho eles realizaram um estudo de caso com os mamógrafos disponíveis na rede pública e nas cidades que compõem a rede de saúde de Minas Gerais, limitando-se às cidades que distam de até 100km de Ouro Preto, exceto a cidade de Belo Horizonte, totalizando 142 cidades [5].

Souza et al. tratam o Problema de Localização de Mamógrafos (PLM) onde o objetivo é definir locais de instalação dentre um conjunto limitado, de tal forma que se maximize a quantidade de mulheres atendidas, obedecendo-se a restrição de máxima distância percorrida. Foram propostas duas formulações: na primeira restringe-se a demanda de uma localidade, tal que a demanda é integralmente atendida pelo mamógrafo designado ou não é atendida; e na segunda formulação, uma localidade pode ter sua demanda parcialmente atendida.

Souza et al. trabalham com o PLM, realizando o estudo para o estado de Minas Gerais, que possui 853 municípios. Dado que o problema é NP-difícil [13] e a quantidade de cidades é grande, eles desenvolveram um algoritmo heurístico baseado na metaheurística VNS (do inglês, *Variable Neighborhood Search*). Em [8] o PLM é formulado admitindo-se que um mamógrafo designado a determinada localização, não atenda integralmente a sua demanda. Foi observado que as restrições de máxima distância percorrida para realização do exame e de mínima demanda para instalação de um equipamento, impossibilitam o atendimento integral à demanda existente no estado. Assim como nos trabalhos [6, 7], de Campos et al. sugerem a utilização de caminhões de mamografia para atender essa demanda residual.

Dada a possibilidade da utilização de unidades móveis de mamografia para a realização de exames em mulheres que estejam distantes dos grandes centros, Rosa et al. introduzem o problema MMURP, cujo objetivo é maximizar a quantidade de mulheres atendidas e minimizar a distância total percorrida pelas MMUs. Neste trabalho, foi utilizado o modelo matemático presente em [7] para simular a distribuição de mamógrafos no estado de Minas Gerais e identificar onde existia demanda que não seria atendida por mamógrafos fixos. Após essa primeira etapa, foram identificadas 579 cidades não atendidas, sendo que 13 dessas possuíam demanda superior a 5069 exames, que é a capacidade anual de um mamógrafo [33]. Para essas cidades, os autores consideraram a instalação de mamógrafos fixos que atendessem a demanda, tal que as MMUs somente atenderiam demandas inferiores a 5069 exames, desta forma uma unidade de mamógrafo móvel não ficaria fixo em uma única cidade [10].

Para a resolução do problema, Rosa et al. propõem uma heurística construtiva e em [14] são propostas duas metaheurísticas para a resolução do MMURP, o *Smart IGS-VND* e o *Smart IGS-RVND*, sendo ambas variações do IGS ([34]), que usam a fase de perturbação do *Smart ILS* de Reinsma et al..

O MMURP é uma variante de um problema de roteamento de veículos (VRP, sigla do inglês *Vehicle Routing Problem*), que são problemas de otimização combinatória onde um conjunto de clientes são servidos por um conjunto de veículos. Dantzig and Ramser formularam pela primeira vez um modelo matemático para o VRP, onde o problema de despacho de caminhões (*Truck dispatching problem*) foi formulado como uma generalização do problema do caixeiro viajante. Desde então, os problemas de roteamento são amplamente estudados, possuindo uma ampla gama de variantes (para detalhes sobre as variantes do VRP leia [37–40]).

O MMURP tem semelhanças com o Problema de Roteamento de Veículos Aberto com Múltiplos Depósitos (MDOVRP, sigla do inglês *Multi-Depot Open Vehicle Routing Problem*), porém tem suas particularidades, como o limite de distância que pode ser percorrido entre duas cidades atendidas, os objetivos do problema são maximizar o atendimento e minimizar a distância percorrida e tem um tamanho de frota limitada. Para criarmos um modelo de programação matemática para o problema, estudamos a literatura que trata sobre a formulação do MDOVRP.

Este problema é proposto pela primeira vez em [15] para resolver o problema de entrega de carnes para açougues por uma frota terceirizada, onde não é necessário o retorno dos veículos ao depósito e a empresa possui mais de um centro de distribuição. Desta maneira o problema foi formulado como a combinação de dois problemas já presentes na literatura: o problema de roteamento aberto (OVRP, sigla do inglês *Open VRP*) e o problema de roteamento com múltiplos depósitos (MDVRP, sigla do inglês *Multi-depot VRP*). Neste trabalho, Tarantilis and Kiranoudis propõem uma nova metaheurística LBTA (sigla do inglês *List-based Threshold Ac-*

cepting), porém um modelo de programação inteira mista para esse problema só será proposto por Liu et al.. Nesta formulação, a variável de decisão possuem três índices, x_{ijk} , que será 1 se um veículo proveniente de um depósito k , viaja de i até j e 0 caso contrário. Nesta formulação, são utilizadas restrições de eliminação de sub-rotas baseadas nas restrições de Miller-Tucker-Zemlin (MTZ) do VRP clássico [41, 42].

Lalla-Ruiz et al. propuseram uma nova formulação onde aprimoram as restrições de eliminação de sub-rotas e adicionam restrições que tornam a formulação mais forte que a presente em [16]. Além disso, com esta nova formulação, conseguiram provar a otimalidade de 4 novas instâncias além das presentes em [16]. Em [18] foi proposta uma nova formulação matemática com dois índices, onde x_{ij} será 1 se um veículo transita de i até j e 0 caso contrário. Esta nova formulação de Lalla-Ruiz and Mes é mais forte que as descritas em [16, 17], requerem um tempo computacional muito menor e conseguiram resolver na otimalidade, pela primeira vez, 8 novas instâncias.

O MMURP possui dois objetivos conflitantes: minimizar a distância percorrida e maximizar a demanda atendida. Na abordagem presente em [10, 14] esses objetivos foram avaliados de maneira hierárquica, sendo o objetivo principal maximizar a demanda, ou seja, para comparar duas soluções, a que atender a maior demanda, será melhor e caso duas soluções atendam a mesma demanda, aquela que percorrer a menor distância será a melhor. Neste trabalho, vamos tratar o problema como multiobjetivo e avaliar as implicações desta abordagem.

Na literatura de otimização multiobjetivo, pode-se observar a divisão dos métodos de resolução em métodos clássicos e metaheurísticas [19]. Nos métodos clássicos, temos métodos baseados em escalarização (como o método da soma ponderada, ϵ -restrito, programação por metas), branch and bound multiobjetivo, etc [19, 43]. Os métodos metaheurísticos são amplamente utilizados, pois possuem a capacidade de prover boas heurísticas de busca de ótimos e gerar soluções não dominadas em tempo computacional razoável. Os métodos metaheurísticos geralmente são divididos pelo tipo de busca: baseados em dominância [20, 24], em decomposição [44], em indicadores e os híbridos [19, 21, 45].

6.2.1 Problemas de coleta de prêmios

O problema do caixeiro viajante com coleta de prêmios (PCTSP, sigla do inglês *Prize Collecting TSP*) é o problema de achar uma rota que minimize o custo total da viagem, com a restrição de que deve ser coletado no mínimo um lucro de p_{min} . Existe a versão deste problema com múltiplos veículos e com limitação de capacidade, que se chama, Capacitated Prize Collecting VRP (CPCVRP). A cada cidade está associada uma demanda, os veículos tem capacidade máxima e um lucro mínimo

deve ser coletado, ou por todas as rotas ou individualmente [46].

6.2.2 Problema de orientação de times

Dado um conjunto K veículos, o problema de orientação de times (TOP, sigla do inglês *Team Orienteering Problem*) propõem definir no máximo $|K|$ rotas de veículos, tal que o lucro coletado por todas as rotas seja maximizado e a restrição de máxima duração de cada rota seja obdecida. Existe a versão deste problema, onde cada cliente tem uma demanda associada a ele e cada veículo possui uma capacidade máxima. O objetivo é maximizar o retorno, sujeito às restrições de máxima duração e da capacidade em cada rota [46].

6.2.3 Splitting para o problema de minimização de distância e de número de veículos

Algorithm 5 Algoritmo de splitting linear para o problema de frota limitada com m veículos

```

1: for  $k \leftarrow 1$  to  $m$  do
2:   for  $t \leftarrow 1$  to  $n$  do
3:      $p[k, t] \leftarrow \infty$ 
4:   end for
5: end for
6:  $p[0, 0] \leftarrow 0$ 
7: for  $k \leftarrow 0$  to  $m - 1$  do
8:    $clear(\Lambda)$ 
9:    $\Lambda \leftarrow (k)$ 
10:  for  $t \leftarrow k + 1$  to  $n$  s.t.  $|\Lambda| > 0$  do
11:     $p[k + 1, t] \leftarrow p[k, front] + f(front, t)$ 
12:     $pred[k + 1][t] \leftarrow front$ 
13:    if  $t < n$  then
14:      if not  $dominates(k, back, t)$  then
15:        while  $|\Lambda| > 0$  and  $dominates(k, t, back)$  do
16:           $popBack()$ 
17:        end while
18:         $pushBack(t)$ 
19:      end if
20:      while  $|\Lambda| > 0$  and  $Q[t + 1] - Q[front] > Q$  do
21:         $popFront()$ 
22:      end while
23:    end if
24:  end for
25: end for

```

Algorithm 6 Algoritmo de splitting linear que devolve a menor quantidade de veículos e a menor distância possível da rota

```

1: for  $k \leftarrow 1$  to  $m$  do
2:   for  $t \leftarrow 1$  to  $n$  do
3:      $p[k, t] \leftarrow \infty$ 
4:   end for
5: end for
6:  $p[0, 0] \leftarrow 0$ 
7: for  $k \leftarrow 0$  to  $m - 1$  do
8:    $clear(\Lambda)$ 
9:    $\Lambda \leftarrow (k)$ 
10:  for  $t \leftarrow k + 1$  to  $n$  s.t.  $|\Lambda| > 0$  do
11:     $p[k + 1, t] \leftarrow p[k, front] + f(front, t)$ 
12:     $pred[k + 1][t] \leftarrow front$ 
13:    if  $t < n$  then
14:      if not  $dominates(k, back, t)$  then
15:        while  $|\Lambda| > 0$  and  $dominates(k, t, back)$  do
16:           $popBack()$ 
17:        end while
18:         $pushBack(t)$ 
19:      end if
20:      while  $|\Lambda| > 0$  and  $Q[t + 1] - Q[front] > Q$  do
21:         $popFront()$ 
22:      end while
23:    end if
24:  end for
25: end for

```

Referências Bibliográficas

- [1] Instituto Nacional de Câncer José Alencar Gomes da Silva INCA. A situação do câncer de mama no brasil: síntese de dados dos sistemas de informação, 2019.
- [2] World Health Organization et al. Cancer control: knowledge into action. who guide for effective programmes. early detection. *Cancer control: knowledge into action. WHO guide for effective programmes. Early detection.*, 2007.
- [3] Mônica Viegas Andrade, Kenya Micaela de Souza Noronha, Pedro Vasconcelos Maia do Amaral, Allan Claudius Queiroz Barbosa, Lucas Resende de Carvalho, Júlia Almeida Calazans, Michelle Nepomuceno Souza, Aline de Souza, Thiago Augusto Hernandez Rocha, and Núbia Cristina da Silva. Distribuição espacial dos mamógrafos em minas gerais e o efeito na probabilidade da realização do exame. *Anais do XX Encontro Nacional de Estudos Populacionais*, pages 1–21, 2016.
- [4] Pedro Amaral, Luciana Luz, Francisco Cardoso, and Rosiene Freitas. Distribuição espacial de equipamentos de mamografia no brasil. *Revista Brasileira de Estudos Urbanos e Regionais (RBEUR)*, 19(2):326–341, 2017.
- [5] Victor Hugo Vidigal Corrêa, Bruno Jhônatan Costa Lima, Pedro Henrique Silva-e Souza, Puca Huachi Vaz Penna, and Marcone Jamilson Freitas Souza. LocalizaÇÃo de mamÓgrafos: Um estudo de caso na rede pÚblica de saÚde. *ANAIS DO SIMPÓCIO BRASILEIRO DE PESQUISA OPERACIONAL*, pages 11–2022, 2018.
URL <https://proceedings.science/sbpo/sbpo-2018/papers/localizacao-de-mamografos--um-estudo-de-caso-na-rede-publica-de-saude>.
- [6] M.J.F. Souza, P.H.V. Penna, M.V.S.M. Sá, P.M. Rosa, J.C. Monteiro, and M.R. Lisboa. Localização de mamógrafos: formulações e estudo preliminar de caso de Rondônia. In *Anais do LI Simpósio Brasileiro de Pesquisa Operacional*, volume 51, page 107698, Limeira (SP), 2019. SOBRAPO, Galoá. Available at: <https://bityli.com/HAEBCTEkv>. Accessed on 08/31/2022.

- [7] Marcone Jamilson Freitas Souza, Puca Huachi Vaz Penna, Manoel Victor Stilpen Moreira de Sá, and Patrick Moreira Rosa. A vns-based algorithm for the mammography unit location problem. In *International Conference on Variable Neighborhood Search*, pages 37–52. Springer, 2020.
- [8] Marcos Vinícius Andrade de Campos, Manoel Victor Stilpen Moreira de Sá, Patrick Moreira Rosa, Puca Huachi Vaz Penna, Sérgio Ricardo de Souza, and Marcone Jamilson Freitas Souza. A mixed linear integer programming formulation and a simulated annealing algorithm for the mammography unit location problem. In *ICEIS (1)*, pages 428–439, 2020.
- [9] Décio Valente Renck, Fernando Barros, Marlos Rodrigues Domingues, Maria Cristina Gonzalez, Marcelo Leal Sclowitz, Eduardo Lucia Caputo, and Laura de Moraes Gomes. Equidade no acesso ao rastreamento mamográfico do câncer de mama com intervenção de mamógrafo móvel no sul do rio grande do sul, brasil. *Cadernos de Saúde Pública*, 30:88–96, 2014.
- [10] O Rosa, Patrick Rosa, ; Moreira, Puca Penna, ; Vaz, and M Souza. Um algoritmo construtivo para o problema de roteamento de unidades móveis de mamografia. *Simpósio Brasileiro de Pesquisa Operacional*, 2020.
- [11] Instituto Nacional de Câncer José Alencar Gomes da Silva INCA. Estatísticas de câncer, April 2022. URL <https://www.inca.gov.br/numeros-de-cancer>. Acessado: 3 jul. 2022.
- [12] Sandra E Brooks, Tina M Hembree, Brent J Shelton, Sydney C Beache, Greta Aschbacher, Philip H Schervish, and Mark B Dignan. Mobile mammography in underserved populations: analysis of outcomes of 3,923 women. *Journal of community health*, 38(5):900–906, 2013.
- [13] Jan Karel Lenstra and AHG Rinnooy Kan. Complexity of vehicle routing and scheduling problems. *Networks*, 11(2):221–227, 1981.
- [14] Otávio Augusto Souza ROSA. Algoritmos heurísticos para o problema de roteamento de unidades móveis de mamografia. 2021. 48 f. dissertação. Mestrado em ciência da computação, Instituto de Ciências Exatas e Biológicas, Universidade Federal de Ouro Preto, Ouro Preto, 2021.
- [15] CD Tarantilis and CT Kiranoudis. Distribution of fresh meat. *Journal of Food Engineering*, 51(1):85–91, 2002.
- [16] Ran Liu, Zhibin Jiang, and Na Geng. A hybrid genetic algorithm for the multi-depot open vehicle routing problem. *OR spectrum*, 36(2):401–421, 2014.

- [17] Eduardo Lalla-Ruiz, Christopher Expósito-Izquierdo, Shervin Taheripour, and Stefan Voß. An improved formulation for the multi-depot open vehicle routing problem. *OR spectrum*, 38(1):175–187, 2016.
- [18] Eduardo Lalla-Ruiz and Martijn Mes. Mathematical formulations and improvements for the multi-depot open vehicle routing problem. *Optimization Letters*, 15(1):271–286, jun 2020. doi: 10.1007/s11590-020-01594-z.
- [19] Qi Liu, Xiaofeng Li, Haitao Liu, and Zhaoxia Guo. Multi-objective metaheuristics for discrete optimization problems: A review of the state-of-the-art. *Applied Soft Computing*, 93:106382, 2020.
- [20] Shanu Verma, Millie Pant, and Vaclav Snasel. A comprehensive review on nsga-ii for multi-objective combinatorial optimization problems. *IEEE Access*, 9:57757–57791, 2021.
- [21] Carlos A Coello Coello, Gary B Lamont, David A Van Veldhuizen, et al. *Evolutionary algorithms for solving multi-objective problems*, volume 5. Springer, New York, 2007.
- [22] Richard Rosenberg. *Simulation of genetic populations with biochemical properties*. PhD thesis, University of Michigan, 1967.
- [23] J D Schaffer. Some experiments in machine learning using vector evaluated genetic algorithms. 1 1985. URL <https://www.osti.gov/biblio/5673304>.
- [24] Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and TAMT Meyarivan. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE transactions on evolutionary computation*, 6(2):182–197, 2002.
- [25] Christian Prins. A simple and effective evolutionary algorithm for the vehicle routing problem. *Computers & operations research*, 31(12):1985–2002, 2004.
- [26] Christian Prins, Philippe Lacomme, and Caroline Prodhon. Order-first split-second methods for vehicle routing problems: A review. *Transportation Research Part C: Emerging Technologies*, 40:179–200, 2014.
- [27] Marcone Jamilson Freitas Souza. *Inteligência Computacional para Otimização*. Departamento de Computação, Universidade Federal de Ouro Preto, 2011. URL <http://www.decom.ufop.br/prof/marcone>.
- [28] Manuel López-Ibáñez, Jérémie Dubois-Lacoste, Leslie Pérez Cáceres, Mauro Birattari, and Thomas Stützle. The irace package: Iterated racing for

- automatic algorithm configuration. *Operations Research Perspectives*, 3: 43–58, 2016.
- [29] Haroldo G Santos and Túlio AM Toffolo. Mixed integer linear programming with python. 2020.
- [30] Gurobi Optimization, LLC. Gurobi Optimizer Reference Manual, 2022. URL <https://www.gurobi.com>.
- [31] Thibaut Vidal. Split algorithm in $O(n)$ for the capacitated vehicle routing problem. *Computers & Operations Research*, 69:40–47, 2016.
- [32] Brasil and Ministério Da Saúde. Série parâmetros sus - volume 1 - caderno 1 - republicado 1, 2017.
- [33] Instituto Nacional do Câncer. Revisão do parâmetro para cálculo da capacidade de produção do mamógrafo simples, 2015. URL <http://twixar.me/bLfm>. Accessed on 02.06.2023.
- [34] Rubén Ruiz and Thomas Stützle. An iterated greedy heuristic for the sequence dependent setup times flowshop problem with makespan and weighted tardiness objectives. *European Journal of Operational Research*, 187(3):1143–1159, 2008. ISSN 0377-2217. doi: <https://doi.org/10.1016/j.ejor.2006.07.029>. URL <https://www.sciencedirect.com/science/article/pii/S0377221706008277>.
- [35] JA Reinsma, PHV Penna, and MJF Souza. Um algoritmo simples e eficiente para resolução do problema do caixeiro viajante generalizado. *Anais do L Simpósio Brasileiro de Pesquisa Operacional, Rio de Janeiro, RJ*, 2018.
- [36] G Dantzig and J Ramser. The truck dispatching problem. *Management Science*, 6(1):80–91, 10 1959. doi: 10.1287/mnsc.6.1.80.
- [37] Rajeev Goel and Raman Maini. Vehicle routing problem and its solution methodologies: a survey. *International Journal of Logistics Systems and Management*, 28(4):419–435, 2017.
- [38] Jose Caceres-Cruz, Pol Arias, Daniel Guimarans, Daniel Riera, and Angel A. Juan. Rich vehicle routing problem: Survey. *ACM Comput. Surv.*, 47(2), dec 2014. ISSN 0360-0300. doi: 10.1145/2666003. URL <https://doi.org/10.1145/2666003>.
- [39] Haifei Zhang, Hongwei Ge, Jinlong Yang, and Yubing Tong. Review of vehicle routing problems: Models, classification and solving algorithms. *Archives of Computational Methods in Engineering*, 29(1):195–221, 2022.

- [40] Shi-Yi Tan and Wei-Chang Yeh. The vehicle routing problem: State-of-the-art classification and review. *Applied Sciences*, 11(21):10295, 2021.
- [41] Clair E Miller, Albert W Tucker, and Richard A Zemlin. Integer programming formulation of traveling salesman problems. *Journal of the ACM (JACM)*, 7(4):326–329, 1960.
- [42] Imdat Kara, Gilbert Laporte, and Tolga Bektas. A note on the lifted miller–tucker–zemlin subtour elimination constraints for the capacitated vehicle routing problem. *European Journal of Operational Research*, 158(3):793–795, 2004. ISSN 0377-2217. doi: [https://doi.org/10.1016/S0377-2217\(03\)00377-1](https://doi.org/10.1016/S0377-2217(03)00377-1). URL <https://www.sciencedirect.com/science/article/pii/S0377221703003771>.
- [43] Yann Collette and Patrick Siarry. *Multiobjective optimization: principles and case studies*. Springer Science & Business Media, 2004.
- [44] Yutao Qi, Zhanting Hou, He Li, Jianbin Huang, and Xiaodong Li. A decomposition based memetic algorithm for multi-objective vehicle routing problem with time windows. *Computers Operations Research*, 62:61–77, 2015. ISSN 0305-0548. doi: <https://doi.org/10.1016/j.cor.2015.04.009>. URL <https://www.sciencedirect.com/science/article/pii/S0305054815000891>.
- [45] E-G. Talbi. Hybrid metaheuristics for multi-objective optimization. *Journal of Algorithms & Computational Technology*, 9(1):41–63, 2015. doi: 10.1260/1748-3018.9.1.41. URL <https://doi.org/10.1260/1748-3018.9.1.41>.
- [46] Claudia Archetti, M. Grazia Speranza, and Daniele Vigo. *Chapter 10: Vehicle Routing Problems with Profits*, pages 273–297. doi: 10.1137/1.9781611973594.ch10. URL <https://epubs.siam.org/doi/abs/10.1137/1.9781611973594.ch10>.