



# Abordagens metaheurísticas para o problema de roteamento de veículos com janelas de tempo e múltiplos entregadores

*Metaheuristic approaches for the vehicle routing problem with time windows and multiple deliverymen*

Aldair Álvarez<sup>1</sup>, Pedro Munari<sup>1</sup>

**Resumo:** Neste trabalho, aborda-se o problema de roteamento de veículos com janelas de tempo e múltiplos entregadores, uma variante do problema de roteamento de veículos que, além das decisões de programação e roteamento dos veículos, envolve a determinação do tamanho da tripulação de cada veículo de entrega. Esse problema surge na distribuição de bens em centros urbanos congestionados em que, devido aos tempos de serviço relativamente longos, pode ser difícil atender todos os clientes durante o horário de trabalho permitido. Diante dessa dificuldade, uma alternativa consiste em incluir a designação de entregadores adicionais para reduzir os tempos de serviço, o que gera custos adicionais aos custos tradicionais de deslocamento e utilização de veículos. Dessa forma, o objetivo é definir rotas para atender grupos de clientes, minimizando o número de veículos usados, o número de entregadores designados e a distância total percorrida. Para tratar o problema são propostas duas abordagens metaheurísticas baseadas em Busca Local Iterada e Busca em Vizinhança Grande. O desempenho das abordagens propostas é testado utilizando conjuntos de instâncias disponíveis na literatura.

**Palavras-chave:** Roteamento de veículos; Múltiplos entregadores; Busca Local Iterada; Busca em Vizinhança Grande.

**Abstract:** *This paper addresses the vehicle routing problem with time windows and multiple deliverymen, a variant of the vehicle routing problem which includes the decision of the crew size of each delivery vehicle, besides the usual scheduling and routing decisions. This problem arises in the distribution of goods in congested urban areas where, due to the relatively long service times, it may be difficult to serve all customers within regular working hours. Given this difficulty, an alternative consists in resorting to additional deliverymen to reduce the service times, which typically leads to extra costs in addition to travel and vehicle usage costs. The objective is to define routes for serving clusters of customers, while minimizing the number of routes, the total number of assigned deliverymen, and the distance traveled. Two metaheuristic approaches based on Iterated Local Search and Large Neighborhood Search are proposed to solve this problem. The performance of the approaches is evaluated using sets of instances from the literature.*

**Keywords:** *Vehicle routing; Multiple deliverymen; Iterated Local Search; Large Neighborhood Search.*

## 1 Introdução

Processos de transporte estão envolvidos em múltiplas formas nos sistemas de produção, especialmente naqueles que envolvem atividades de distribuição. Tais processos têm impacto na competitividade e nos níveis de serviço das indústrias. Por exemplo, processos de transporte podem representar até 20% dos custos finais dos bens produzidos por uma empresa (Toth & Vigo, 2002). Adicionalmente, estima-se que os custos de distribuição podem representar até 75% dos custos de logística de uma organização (Bräysy & Gendreau, 2005), tornando necessário investir esforços para o aprimoramento desses processos. Dentre as atividades de distribuição surge o problema

de roteamento de veículos (PRV), o qual é enfrentado diariamente por muitas empresas que lidam com o transporte de produtos e/ou pessoas. Na prática, o PRV desempenha um papel importante dentro dos sistemas de distribuição e, portanto, sua resolução é parte fundamental da gestão eficiente das operações das organizações.

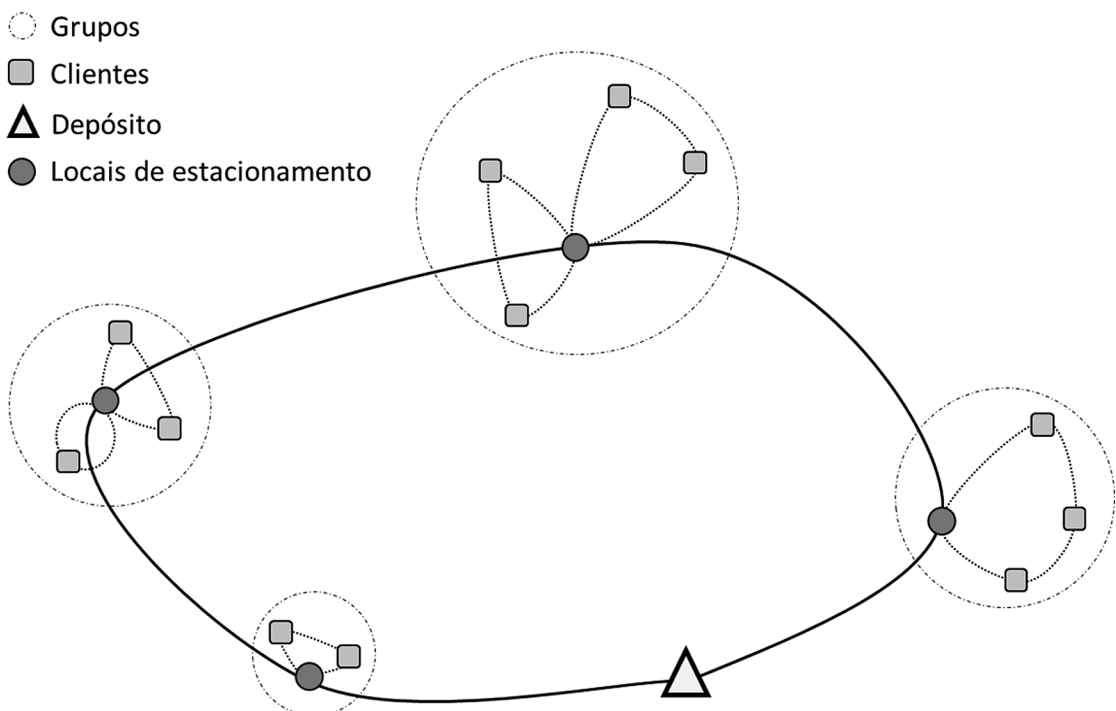
Recentemente, alguns estudos foram apresentados na literatura explorando uma nova variante do PRV, a qual envolve decisões de alocação do tamanho da tripulação nos veículos (Pureza et al., 2012; Ferreira & Pureza, 2012). Esta variante, chamada de problema de roteamento de veículos com múltiplos

<sup>1</sup> Departamento de Engenharia de Produção, Universidade Federal de São Carlos – UFSCar, Rod. Washington Luiz, Km 235, CEP 13565-905, São Carlos, SP, Brasil, e-mail: aldair@dep.ufscar.br; munari@dep.ufscar.br  
Received June 23, 2015 - Accepted Dec. 28, 2015  
Suporte financeiro: CAPES, FAPESP, CNPq.

entregadores (PRVME), tem aplicações em muitos contextos industriais, principalmente em situações nas quais se faz necessária a entrega regular de produtos em centros urbanos bastante congestionados. Exemplos típicos são fábricas de refrigerantes, laticínios e cervejas que precisam reabastecer regularmente (diariamente ou a cada poucos dias) pequenos e médios estabelecimentos como lojas de conveniência, restaurantes, mercearias, entre outros. Estes estabelecimentos se encontram tipicamente em regiões com alta concentração comercial, nas quais se torna difícil até mesmo estacionar os veículos de entrega. Assim, os veículos estacionam em pontos estratégicos de uma região que possua um grupo de clientes, e as entregas são feitas a pé até esses clientes. Ao fazer as entregas dessa forma, o tempo de serviço no grupo de clientes pode ser relativamente longo quando comparado com os tempos de viagem, o que pode inviabilizar o atendimento de todos os grupos de clientes durante o horário de trabalho permitido. Em contextos assim, o uso de múltiplos entregadores se torna importante, pois pode acelerar a entrega dos produtos, reduzindo o tempo de serviço em cada grupo de clientes. Um exemplo de uma rota típica no PRVME pode ser observado na Figura 1, na qual as entregas são feitas em duas fases. Inicialmente, o veículo chega até o ponto de parada do grupo de clientes e, posteriormente, os entregadores levam os produtos até os clientes do grupo.

Apesar da importância teórica e prática desta variante, existem poucos trabalhos na literatura abordando o PRVME, o que encoraja o desenvolvimento de métodos de solução para resolvê-lo. No presente trabalho, propomos duas abordagens metaheurísticas para resolver esse problema, as quais são baseadas em Busca Local Iterada (ILS, do inglês *Iterated Local Search*) e Busca em Vizinhança Grande (LNS, do inglês *Large Neighborhood Search*). Essas metaheurísticas foram aplicadas de forma bem-sucedida para resolver outras variantes do PRV, por exemplo: PRV com frota heterogênea (Subramanian et al., 2012); PRV com janelas de tempo (Pisinger & Ropke, 2007); PRV dinâmico (Hong, 2012); PRV com múltiplas rotas (Azi et al., 2014); PRV com entregas fracionadas (Silva et al., 2015); e PRV com coleta e entrega (Ropke & Pisinger, 2006). Dessa forma, acredita-se que ILS e LNS também possam resolver de forma eficiente o problema abordado nesta pesquisa. Usando instâncias encontradas na literatura, comparamos o desempenho das abordagens propostas entre si, bem como com o desempenho de outros métodos encontrados na literatura.

As demais seções deste artigo estão organizadas da seguinte forma. Na Seção 2, é descrito o problema a ser abordado. A Seção 3 apresenta as abordagens de solução propostas. A seguir, são descritos os experimentos computacionais realizados na Seção 4. Finalmente, na Seção 5, são apresentadas as conclusões do trabalho e as perspectivas para pesquisas futuras.



**Figura 1.** Exemplo de uma rota no PRVME.

## 2 Descrição do problema

O problema a ser abordado neste trabalho é o problema de roteamento de veículos com janelas de tempo e múltiplos entregadores (PRVJTME), uma variante do clássico PRV com janelas de tempo que, além de tratar as decisões de roteirização e programação dos veículos, envolve a decisão adicional da determinação do tamanho da tripulação de cada veículo. No PRVJTME, os tempos de serviço podem ser relativamente longos quando comparados com os tempos de viagem, dado que são atendidos grupos de clientes e não clientes individuais. Além disso, o tempo de serviço necessário para o atendimento de um grupo de clientes é dependente do número de entregadores designados à rota.

Em termos práticos, o problema envolve dois estágios: inicialmente deve-se agrupar os clientes em torno de locais de estacionamento para depois estabelecer rotas para visitar esses grupos. Deve-se respeitar restrições de capacidade, janelas de tempo e número de entregadores disponíveis enquanto o custo total é minimizado (custo de utilização dos veículos, de designação de entregadores e de distância percorrida). Dada a complexidade do problema completo, os estágios de agrupamento e roteamento são tratados de forma separada (Senarclens de Grancy & Reimann, 2014). Assim, no que tange ao PRVJTME, os locais de estacionamento dos grupos são assumidos como dados de entrada, cada um dos quais tem uma demanda acumulada e um tempo de serviço que inclui o transporte dos produtos desde o local de estacionamento até os clientes do grupo.

Dado o contexto acima, o PRVJTME pode ser definido formalmente como segue. Dada uma frota homogênea de veículos disponível em um único depósito, cada um com capacidade  $Q$ , ela deve ser usada para visitar  $n$  grupos de clientes de modo a atender a suas demandas  $d_i, i = 1, \dots, n$ . O objetivo é determinar rotas que tenham custo total mínimo e satisfaçam os seguintes requisitos: cada grupo deve ser visitado exatamente uma única vez e a visita deve satisfazer a janela de tempo  $[w_i^a, w_i^b]$ , isto é, o veículo não pode chegar no grupo após o instante  $w_i^b$  e deve esperar até  $w_i^a$  para iniciar o serviço, caso chegue antes deste instante. O tempo de serviço necessário para atender o grupo  $i$  com  $l$  entregadores é conhecido *a priori* e denotado por  $s_{il}$ . O tempo de viagem entre dois grupos  $i$  e  $j$  é dado por  $t_{ij}$ . Os veículos devem retornar ao depósito após realizarem as rotas designadas.

A função que define o custo de uma solução  $S$  é definida com a Equação 1, em que  $V$  é o número de veículos usados,  $E$  é o número total de entregadores e  $D$  é a distância total percorrida na solução. Os valores  $p_1, p_2$  e  $p_3$  são os pesos de cada componente, os quais são usados para priorizar esses componentes.

$$c(S) = p_1V + p_2E + p_3D \quad (1)$$

A fim de mostrar como o uso de múltiplos entregadores ajuda a melhorar o nível de serviço, considere um caso com três grupos de clientes, um veículo com capacidade suficientemente grande para atender todos os grupos e os dados mostrados na Tabela 1. Na tabela,  $d_i, s_{i1}, s_{i2}, w_i^a, w_i^b$  representam a demanda, o tempo de serviço com um e dois entregadores e o instante de abertura e de encerramento da janela de tempo do grupo  $i$ , respectivamente. O grupo  $i = 0$  representa o depósito, no qual se encontra o veículo inicialmente. Além disso,  $t_{ij}$  representa o tempo de viagem entre os grupos  $i$  e  $j$ . Soluções para o problema descrito na Tabela 1, considerando um e dois entregadores no veículo, são apresentadas na Figura 2. Na figura,  $w_i$  representa o instante de tempo no qual o serviço começa no grupo  $i$ . Note que quando são usados dois entregadores é possível atender a todos os grupos de clientes dentro do horizonte de planejamento permitido (encerramento da janela de tempo do depósito). No entanto, quando é usado só um entregador, o grupo 3 não pode ser atendido, pois, partindo do grupo 2, só seria possível chegar a ele após o encerramento de sua janela de tempo ( $w_3^b$ ).

## 3 Abordagens metaheurísticas

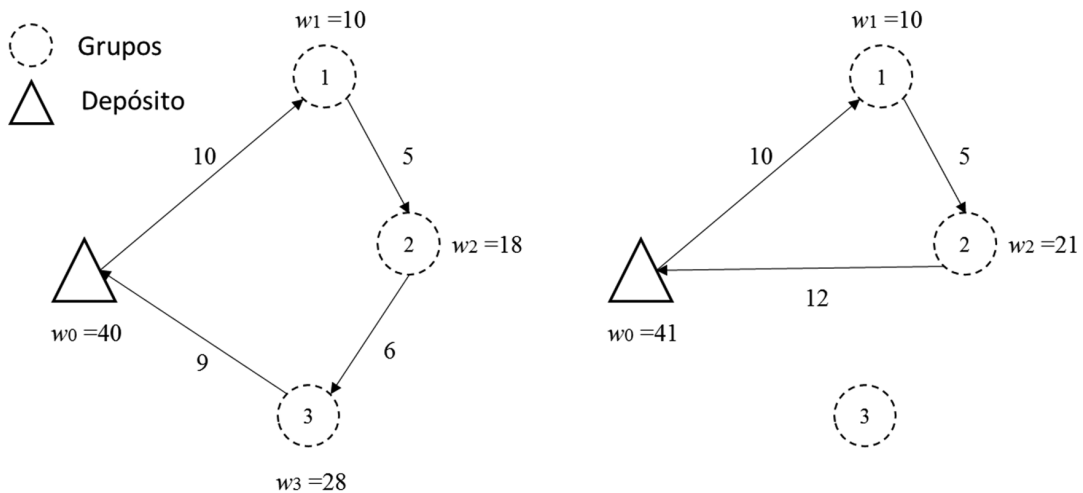
Nesta seção, são descritas as duas abordagens metaheurísticas propostas para o problema. A primeira é baseada na metaheurística ILS e é apresentada na seção 3.2. A segunda é baseada na metaheurística LNS e sua descrição é mostrada na seção 3.3. Para ambas as abordagens, foi utilizada a mesma heurística construtiva, a qual é descrita na seção 3.1.

### 3.1 Heurística construtiva

Para obter uma solução inicial para as abordagens metaheurísticas, foi desenvolvida uma heurística construtiva semelhante à usada por Senarclens de Grancy & Reimann (2014), a qual é baseada na heurística de inserção II de Solomon (1987). Em nossa

**Tabela 1.** Dados de um PRV com múltiplos entregadores.

$i$	$d_i$	$s_{i1}$	$s_{i2}$	$w_i^a$	$w_i^b$	$t_{ij}$			
						0	1	2	3
0	0	0	0	0	50	0	10	12	9
1	8	6	3	8	15	10	0	5	12
2	6	8	4	18	26	12	5	0	6
3	7	6	3	25	30	9	12	6	0



**Figura 2.** (esquerda) Solução com dois entregadores na rota; (direita) Solução com um entregador na rota.

aplicação da heurística, as rotas são construídas sequencialmente inicializando-as com o grupo mais distante em relação ao depósito e que ainda não foi atendido, definindo a tripulação inicial do veículo como a máxima tripulação possível. Em seguida, novos grupos são inseridos na rota, minimizando um critério ponderado de minimização de tempo adicional e de distância adicional percorrida ao atender o grupo. Uma vez que não possam ser inseridos mais grupos na rota atual, uma nova rota é inicializada e o processo é repetido até completar uma solução atendendo todos os grupos.

### 3.2 Abordagem metaheurística baseada em ILS

A primeira abordagem desenvolvida é baseada na metaheurística ILS (Lourenço et al., 2003). Essa metaheurística aplica uma busca local repetidamente sobre soluções obtidas da perturbação dos ótimos locais previamente visitados. Um algoritmo ILS utiliza quatro componentes básicos: (i) uma solução inicial; (ii) um procedimento de busca local; (iii) um mecanismo de perturbação; e (iv) um critério de aceitação. Para mais informações e aplicações da ILS sugere-se o trabalho de Lourenço et al. (2010).

A abordagem proposta, além dos componentes básicos de um algoritmo ILS, utiliza duas heurísticas adicionais para melhorar seu desempenho, a saber, heurística de redução de entregadores e heurística de redução de rotas, as quais foram projetadas para o PRVJTME. A estrutura da abordagem é mostrada na Figura 3. Inicialmente, uma solução inicial é gerada com a heurística construtiva (linha 2). Esta solução é melhorada por meio da busca local e definida como a melhor solução inicial em cada iteração da abordagem (linha 5). O ciclo principal do algoritmo é dado pelas linhas 7 a 23, o qual tem o

```

Entrada: Instância, parâmetros;
Saída: Melhor solução  $S^*$ ;

1 início
2    $S_0 \leftarrow$  Gerar solução inicial;
3    $S^* \leftarrow S_0$ ;
4   repita
5      $S^+ \leftarrow$  Busca local( $S_0$ );
6      $iterILS \leftarrow 0$ ;
7     enquanto  $iterILS < 2 * MaxIterILS$  faça
8        $S' \leftarrow$  Perturbar( $S^+$ );
9        $S' \leftarrow$  Busca local( $S'$ );
10       $S' \leftarrow$  Redução entregadores( $S'$ );
11      se  $c(S') < c(S^+)$  então
12         $S^+ \leftarrow S'$ ;
13        se  $iterILS < MaxIterILS$  então
14           $iterILS \leftarrow 0$ ;
15        fim
16      senão
17         $iterILS \leftarrow MaxIterILS$ ;
18      fim
19    fim
20    senão
21       $iterILS \leftarrow iterILS + 1$ ;
22    fim
23  fim
24  se  $c(S^+) < c(S^*)$  então
25     $S^* \leftarrow S^+$ ;
26  fim
27  até atingir o critério de parada;
28 fim
    
```

**Figura 3.** Abordagem baseada em Busca Local Iterada.

propósito de melhorar a melhor solução atual usando a heurística de busca local (linha 9) combinada com a heurística de redução de entregadores (linha 10) e do mecanismo de perturbação (linha 8). O critério de aceitação define que a perturbação é aplicada na solução incumbente da atual iteração da abordagem ( $S^+$ ). Este ciclo de melhoria compreende duas fases, cada uma das quais completa-se quando o algoritmo atinge  $MaxIterILS$  perturbações consecutivas sem melhoria (linhas 13-15 e 16-18, respectivamente).

Ao final do ciclo, a melhor solução global obtida pela abordagem é atualizada (linhas 24-26) e um novo ciclo é iniciado caso o critério de parada não tenha sido atingido.

As diferentes fases da abordagem são necessárias para se considerar diferentes partes da função objetivo do PRVJTME. A primeira fase foca na redução de veículos da solução e, para isso, o mecanismo de perturbação é efetuado utilizando-se a heurística de redução de rotas da Seção 3.2.3. A segunda fase da abordagem foca a redução da distância total percorrida e por isso a perturbação é realizada pelo mecanismo descrito na Seção 3.2.2. Note que na primeira fase nem sempre a heurística de redução de rotas consegue modificar a solução. Nesse caso, aplica-se o mecanismo de perturbação da segunda fase. Finalmente, note também que a redução do número de entregadores é diretamente tratada usando a heurística de redução de entregadores.

### 3.2.1 Busca Local

A busca local desempenha o papel da intensificação dentro da ILS. Em nosso algoritmo, é realizada por meio de uma heurística de busca de vizinhança variável (Mladenovic & Hansen, 1997) com Ordenação Aleatória, sendo conhecida por RVND (do inglês *Randomized Variable Neighborhood Descent*). A heurística aplica repetidamente um conjunto de buscas locais menores, chamadas de estruturas de vizinhança, para melhorar progressivamente a solução. Dado um conjunto de estruturas de vizinhança  $V = \{v^1, \dots, v^n\}$ , a heurística escolhe aleatoriamente uma delas para melhorar a solução atual. Se a estrutura de vizinhança não melhora a solução, ela é eliminada do conjunto  $V$  e a busca continua com as outras estruturas. Por outro lado, toda vez que uma estrutura melhora a solução atual, o conjunto é restabelecido à sua forma inicial. O algoritmo termina quando  $V$  se torna vazio.

Só movimentos factíveis são admitidos e a estratégia de primeira melhoria é utilizada nas estruturas de vizinhança. Além disso, sempre que uma estrutura de vizinhança consegue melhorar a solução, aplica-se a heurística de redução de rotas descrito na seção 3.2.3. O conjunto de estruturas de vizinhança utilizadas é composto pelos movimentos apresentados a seguir:

- Estruturas de vizinhança inter-rotas:
  - o **Shift( $k,0$ )**:  $k$  grupos adjacentes são transferidos de uma rota  $r_1$  para a rota  $r_2$ ,  $k = \{1,2,3\}$  (veja Figura 4).
  - o **Swap( $1,1$ )**: troca o grupo  $c_1$  da rota  $r_1$  com o grupo  $c_2$  da rota  $r_2$  (veja Figura 5).
  - o **Swap( $2,1$ )**: troca dois grupos adjacentes  $c_1$  e  $c_2$  da rota  $r_1$  com o grupo  $c_3$  da rota  $r_2$ .
  - o **Swap( $2,2$ )**: troca dois grupos adjacentes  $c_1$  e  $c_2$  da rota  $r_1$  com dois grupos adjacentes  $c_3$  e  $c_4$  da rota  $r_2$ .
- Estruturas de vizinhança intra-rotas:
  - o **Or-opt-1**: transfere um grupo para outra posição na mesma rota (veja Figura 6).
  - o **2-opt**: dois arcos não adjacentes  $(i, i^+)$  e  $(j, j^+)$  são removidos e dois arcos  $(i, j)$ ,  $(i^+, j^+)$  são adicionados de forma que uma nova rota é gerada (veja Figura 7).

### 3.2.2 Heurística de remoção e inserção

A perturbação é a responsável pela diversificação dentro da ILS, dado que ela muda o ótimo local atual. Em nossa implementação, um dos mecanismos de

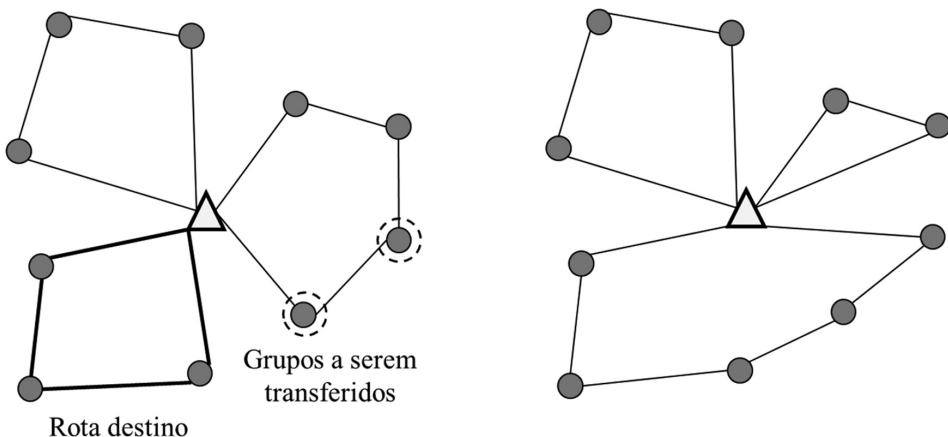


Figura 4. Movimento shift(2,0).



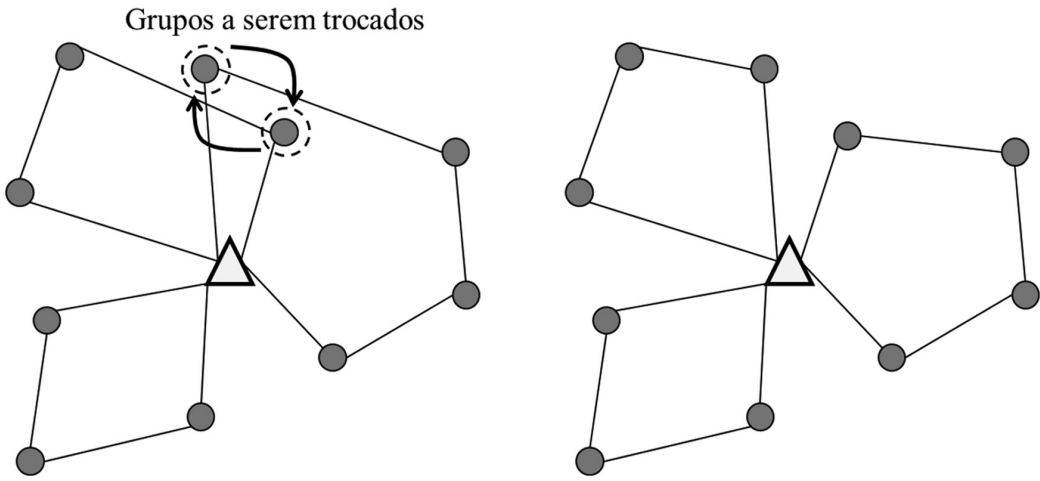


Figura 5. Movimento  $\text{swap}(1,1)$ .

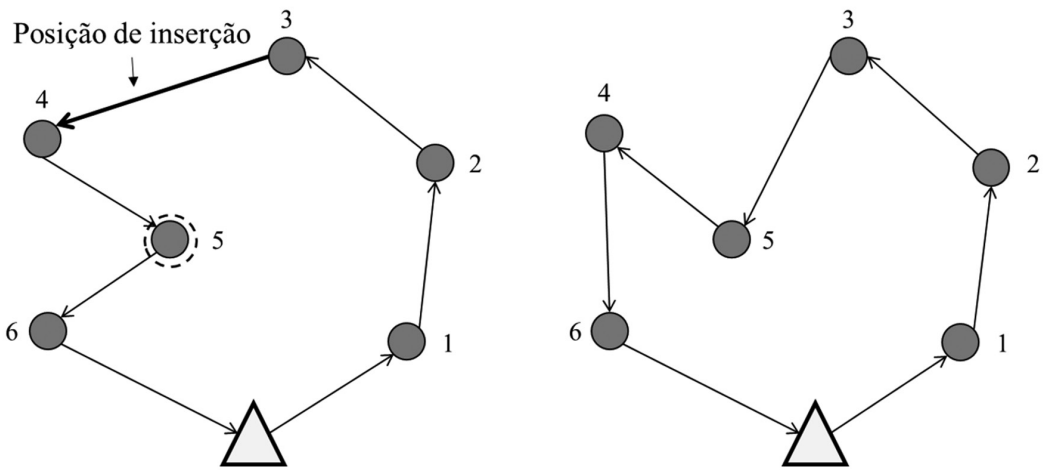


Figura 6. Movimento Or-opt-1.

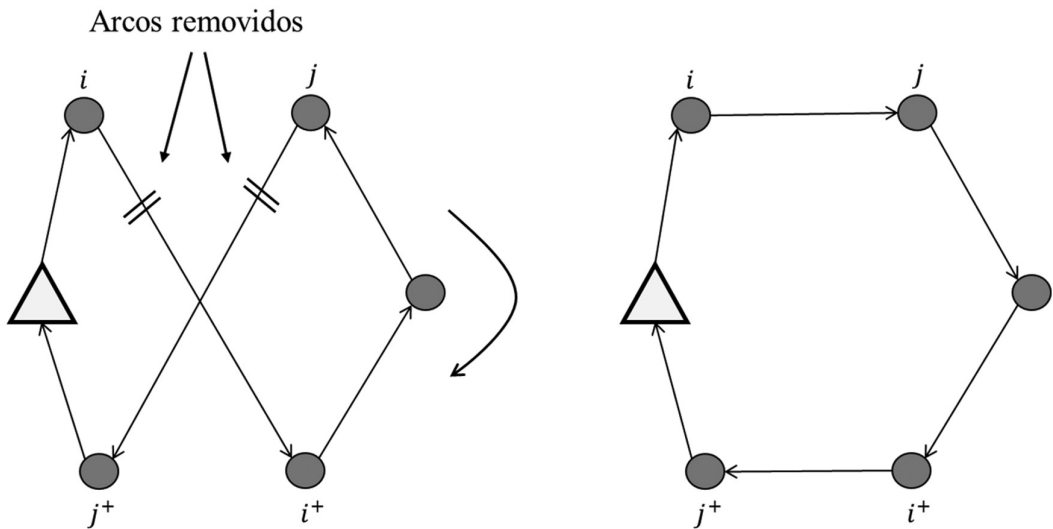


Figura 7. Movimento 2-opt.

perturbação consiste em uma heurística de remoção e inserção, baseada no mecanismo proposto por Melechovsky (2012). Para uma rota dada, escolhe-se aleatoriamente até  $nP$  grupos, os quais são removidos da rota. Para esses grupos removidos, testa-se sua inserção nas rotas restantes, realizando a primeira inserção factível encontrada e inserindo-os na mesma ordem que foram removidos. Se no final existem grupos sem inserção factível, uma nova rota atendendo só a esses grupos é criada.

### 3.2.3 Heurística de redução de rotas

Considerando que diminuir a quantidade de rotas pode também reduzir o número de entregadores designados, a heurística de redução de rotas utilizada por Senarclens de Grancy & Reimann (2014) foi estendida e utilizada na abordagem. Para uma solução dada, a heurística toma uma rota por vez, remove todos os grupos dela e tenta reinseri-los em sua melhor posição factível nas rotas restantes. No caso de não poder realocar algum grupo, as tripulações das rotas destino são temporariamente aumentadas em uma unidade (se a tripulação for menor que a máxima possível), a fim de aumentar suas folgas e facilitar a inserção factível desses grupos. Se, aumentando a tripulação das rotas torna-se possível inserir os grupos, então as tripulações temporárias das rotas que receberam grupos são mantidas. A heurística continua até alocar todos os grupos de clientes da rota eliminada ou até todas as rotas terem experimentado o tamanho máximo de tripulação e ainda ter grupos não atendidos. Vale lembrar que esta heurística além de ser usada diretamente para diminuir o número de rotas de uma solução, também é utilizada como mecanismo de perturbação durante a primeira fase da abordagem ILS.

### 3.2.4 Heurística de redução de entregadores

As rotas de uma solução podem ter mais entregadores que o necessário, seja por construção ou devido aos movimentos de melhoria. Para melhorar isso, uma heurística foi projetada para reduzir o número total de entregadores de uma solução dada. Seja  $S = \{r_1, r_2, \dots, r_n\}$  uma solução factível composta pelas rotas  $r_1, r_2, \dots, r_n$ . Define-se  $crew_i$  como a tripulação da rota  $r_i, \forall r_i \in S$ . Para cada rota da solução, o tamanho da tripulação é reduzido em uma unidade (se  $crew_i > 1$ ). Se a redução da tripulação tornou a rota infactível (em termos de janelas de tempo), remove-se o primeiro grupo da rota que se tornou infactível com a intenção de aumentar a folga da rota a partir do ponto de remoção e facilitar o atendimento dos grupos posteriores a ele. Este processo é repetido até que a rota se torne factível. Finalmente, a heurística tenta inserir os grupos que

foram removidos na melhor posição possível nas rotas e, caso alguns grupos não possuam inserção factível, uma nova rota é criada para atender esses grupos. A solução resultante é denotada por  $s'$  e, caso ela possua o mesmo número de rotas da solução inicial  $S$ , então  $s'$  substitui a solução  $S$ .

### 3.3 Abordagem metaheurística baseada em LNS

A segunda abordagem proposta é baseada na metaheurística LNS, introduzida por Shaw (1998). LNS foi projetada visando superar as dificuldades das buscas locais tradicionais, as quais só fazem pequenas modificações nas soluções e consequentemente não são capazes de se deslocar entre áreas promissoras do espaço de soluções. Na metaheurística LNS, a solução incumbente é iterativamente melhorada por meio de operações alternadas de destruição e reparação. Para uma descrição detalhada da LNS, sugere-se o trabalho de Pisinger & Ropke (2010).

Na abordagem proposta, LNS guia a busca por meio dos operadores de destruição e reparação, acompanhada das heurísticas de melhoria adicionais usadas na abordagem baseada em ILS (RVND, heurística de redução de rotas e heurística de redução de entregadores). A estrutura da abordagem é mostrada na Figura 8. Inicialmente, é gerada uma solução inicial com a heurística construtiva (linha 2), que, no começo de cada iteração da abordagem, é armazenada como a melhor solução inicial (linha 5). Em seguida, a abordagem entra em um ciclo de melhoria (linhas 7-16), em que inicialmente são aplicados os operadores de destruição e reparação (linha 8). Primeiro aplica-se um dos operadores de destruição da Seção 3.3.1 e, logo depois, um dos dois operadores de reparação da Seção 3.3.2. Em ambos os casos, os operadores

```

Entrada: Instância, parâmetros;
Saída: Melhor solução  $S^*$ ;
1 início
2  $S_0 \leftarrow$  Gerar solução inicial;
3  $S^* \leftarrow S_0$ ;
4 repita
5    $S^+ \leftarrow S_0$ ;
6    $iterLNS \leftarrow 0$ ;
7   enquanto  $iterLNS < MaxIterLNS$  faça
8      $S' \leftarrow$  Aplicar operador de remoção e inserção em  $S^+$ ;
9      $S' \leftarrow$  Redução rotas( $S'$ );
10     $S' \leftarrow$  Redução entregadores( $S'$ );
11     $S' \leftarrow$  RVND( $S'$ );
12    se  $c(S') < c(S^+)$  então
13       $S^+ \leftarrow S'$ ;
14    fim
15     $iterLNS \leftarrow iterLNS+1$ ;
16  fim
17  se  $c(S^+) < c(S^*)$  então
18     $S^* \leftarrow S^+$ ;
19  fim
20  até atingir o critério de parada;
21 fim

```

Figura 8. Abordagem baseada em LNS.

são escolhidos aleatoriamente. Depois, são aplicadas as heurísticas de redução de rotas, de redução de entregadores e a heurística RVND (linhas 9-11). Ao final do ciclo, é atualizada a melhor solução da execução atual da abordagem, aceitando só soluções melhoradas (linhas 12-14). Finalmente, toda vez que a metaheurística executa *MaxIterLNS* iterações, a melhor solução global é atualizada (linhas 17-19) e a abordagem metaheurística é reiniciada.

### 3.3.1 Operadores de destruição

Para a abordagem proposta, foram usados quatro operadores de remoção diferentes. Cada um deles toma como entrada uma solução completa e retorna uma solução da qual foram removidos  $q$  grupos de clientes. Os operadores são descritos a seguir.

- **Remoção aleatória:** este operador escolhe  $q$  grupos de forma aleatória e os remove das rotas. Como apontado por Pisinger & Ropke (2007), este operador claramente tem o efeito de diversificar a busca.
- **Remoção da pior posição:** a ideia deste operador é remover grupos que são muito caros, ou que, de alguma forma, aumentam o custo da solução. Seja  $i$  um grupo qualquer,  $i^-$  seu predecessor e  $i^+$  seu sucessor na rota. O custo  $c_i$  do grupo  $i$  é computado conforme a Equação 2.

$$c_i = d_{i^-,i} + d_{i,i^+} - d_{i^-,i^+} \quad (2)$$

em que  $d_{ij}$  é a distância entre dois grupos  $i$  e  $j$ . Depois, o operador de remoção escolhe repetidamente um grupo  $i$  com maior custo até remover  $q$  grupos. A escolha do grupo  $i$  tem um componente aleatório controlado por um parâmetro  $p$ , como descrito a seguir. Seja  $L$  o número de grupos na solução. Toda vez que um novo grupo vai ser removido, é gerado um número aleatório  $y$  no intervalo  $(0,1]$  e calcula-se  $k$ , tal que  $k = \lceil y^p L \rceil$ . Em seguida, o  $k$ -ésimo grupo com maior custo é removido e o valor de  $L$  é atualizado. Esse procedimento é repetido até remover  $q$  grupos. Se  $p$  for grande, grupos de maior custo têm maior probabilidade de serem removidos, enquanto grupos de menor custo podem ser escolhidos para menores valores de  $p$ . Essa estratégia é usada para evitar situações nas quais os mesmos grupos sejam removidos com frequência muito mais elevada que os demais (Pisinger & Ropke, 2007).

- **Remoção relacionada:** o propósito deste operador é remover grupos que, de alguma maneira, estão relacionados e conseqüentemente podem ser fáceis de trocar. A medida de relação entre dois

grupos  $i$  e  $j$  foi tomada como a distância  $d_{ij}$  entre eles, realizando a remoção da seguinte forma. O algoritmo inicialmente seleciona um grupo  $i$  de forma aleatória. Em seguida, escolhe um dos grupos previamente selecionados e seleciona um novo grupo que esteja mais relacionado com ele. O procedimento se repete até selecionar  $q$  grupos, os quais são removidos. Semelhante ao operador de remoção anterior, o processo de seleção dos grupos relacionados tem um componente aleatório dado pelo parâmetro  $p$ .

- **Remoção orientada por tempo:** este operador é uma variação do operador de remoção relacionada, na qual a medida de relação entre dois grupos é baseada no tempo de início do serviço. Assim, tenta-se remover grupos que são atendidos aproximadamente no mesmo tempo, pois espera-se que eles sejam fáceis de trocar. O operador funciona como segue. Um grupo  $r$  é escolhido aleatoriamente e os  $2q$  grupos mais próximos são marcados como grupos potenciais. A medida de proximidade em tempo entre os grupos  $r$  e  $i$  é medida de acordo com a Equação 3.

$$\Delta_{ir} = |w_r - w_i| \quad (3)$$

em que  $w_r$  e  $w_i$  são os instantes de início do serviço nos grupos  $r$  e  $i$ , respectivamente. Dentre os grupos marcados, são escolhidos os  $q - 1$  grupos mais relacionados com  $r$ . Esses grupos são removidos junto com  $r$ . Semelhante ao operador de remoção relacionada, este operador também utiliza o componente aleatório controlado pelo parâmetro  $p$ .

### 3.3.2 Operadores de reparação

Uma vez que a solução foi destruída, deve-se efetuar sua reparação para torná-la factível novamente. Para isso, foram utilizados dois operadores de inserção. Cada um deles toma como entrada uma solução incompleta e retorna uma solução completa, inserindo os  $q$  grupos que foram extraídos na fase de remoção. Os dois operadores utilizados são descritos a seguir.

- **Inserção gulosa:** como indicado por seu nome, o operador de inserção gulosa insere os grupos na posição mais barata possível. Formalmente pode ser definido da seguinte forma. Seja  $\Delta_{ir}$  o incremento na função objetivo ao inserir o grupo  $i$  na melhor posição possível na rota  $r$ . Se o grupo  $i$  não pode ser inserido na rota  $r$ ,  $\Delta_{ir} = \infty$ . Depois, aplica-se a Equação 4.



$$(i', r') = \arg \min_{i, r} \Delta f_{ir} \quad (4)$$

e insere-se o grupo  $i'$  na melhor posição da rota  $r'$ . Esta operação é realizada até inserir todos os grupos removidos ou não existir inserção factível para algum deles. Nesse último caso, novas rotas são criadas (inicializadas com a máxima tripulação possível) para esses grupos.

- **Inserção tipo *regret*:** este operador é motivado pelos inconvenientes do operador de inserção gulosa, o qual tende a deixar para o final a alocação de grupos de difícil inserção, quando não se tem muitas possibilidades de inserção. A inserção tipo *regret* tenta contornar essa dificuldade com a incorporação de um componente de antecipação no processo de seleção do grupo a ser inserido. Seja  $\Delta f_i^q$  o aumento na função objetivo ao inserir o grupo  $i$  na  $q$ -ésima melhor inserção possível na solução. Em cada iteração é escolhido um grupo  $i'$  conforme a Equação 5.

$$i' = \arg \max_i \Delta f_i^2 - \Delta f_i^1 \quad (5)$$

e o grupo é inserido na posição correspondente à rota de menor custo de inserção. Em outras palavras, o operador maximiza a diferença do custo de inserção entre a segunda melhor e a melhor inserção do grupo  $i'$ , o que significa que grupos com menos posições factíveis de inserção tendem a ser inseridos primeiro. O operador continua até inserir todos os grupos ou não existir inserção factível restante para algum grupo. Como ocorre no operador de inserção gulosa, se no final ainda restarem grupos sem inserção factível, são criadas novas rotas para alocar esses grupos.

## 4 Experimentos e resultados computacionais

Nesta seção, apresenta-se uma comparação entre os resultados obtidos pelas abordagens propostas, como também uma comparação de seus desempenhos com métodos da literatura. Além disso, uma análise estatística é realizada a fim de calibrar as abordagens metaheurísticas propostas. Todos os algoritmos foram implementados em linguagem C++.

### 4.1 Instâncias utilizadas

Nos experimentos, foram usadas as instâncias propostas por Pureza et al. (2012), as quais são baseadas nas conhecidas instâncias de Solomon (1987) para o PRV com janelas de tempo. O conjunto contém 56 instâncias, cada uma delas possui 100 grupos de clientes. Elas são agrupadas em 6 classes baseadas no tamanho do horizonte de planejamento, capacidade

dos veículos e tamanho das janelas de tempo, a saber: R1, R2, C1, C2, RC1 e RC2. As classes R1, C1 e RC1 (R2, C2 e RC2) contêm instâncias com horizontes de planejamento de custos (longos), janelas de tempo estreitas (folgadas) e veículos com capacidade pequena (grande). Por outro lado, as classes R1/R2, C1/C2 e RC1/RC2 têm grupos de clientes espalhados aleatoriamente no espaço geográfico, grupos aglomerados e uma mistura dessas características (aleatória e aglomerada), respectivamente. Note que as características das instâncias R2, C2 e RC2 permitem que mais grupos possam ser atendidos por cada rota do que as rotas nas instâncias R1, C1 e RC1.

Nas instâncias originais de Solomon, não há diferenciação dos tempos de serviço de acordo com o número de entregadores. Assim, Pureza et al. (2012) propuseram uma modificação para representar o tempo de entrega da demanda acumulada dos clientes dos grupos, definida na Equação 6.

$$s_{il} = \frac{\min\{rs * q_i, T - \max\{w_i^a, d_{0i}\} - d_{i0}\}}{l}, \quad i=1, \dots, n, \quad l=1, 2, 3. \quad (6)$$

em que  $q_i$  é a demanda do grupo  $i$ ,  $rs$  é a taxa de serviço, a qual em nossos experimentos foi definida com valor 2,  $T$  é o instante de encerramento da janela de tempo do depósito e  $d_{0i} = d_{i0}$  é a distância entre o depósito e o grupo  $i$ , enquanto o segundo termo na operação *min* da equação garante a factibilidade da instância.

### 4.2 Ajuste de parâmetros

As abordagens metaheurísticas foram calibradas de acordo com a técnica de planejamento de experimentos (Montgomery, 2012), de forma similar ao apresentado em Naderi et al. (2010). Para isso, foi realizado um experimento fatorial completo de modo a testar os parâmetros em diferentes níveis, os quais são mostrados na Tabela 2. As combinações resultaram em 9 configurações diferentes para cada algoritmo. Os níveis testados para cada fator foram determinados realizando testes preliminares. Um conjunto de 20 instâncias foi utilizado para calibrar os algoritmos, sendo que as instâncias foram escolhidas aleatoriamente dentre as 56 disponíveis. Foram realizadas 5 réplicas para cada combinação configuração-instância, resultando em um total de 900 observações. Um tempo limite de 5 minutos foi imposto para cada execução dos algoritmos, os quais

**Tabela 2.** Fatores e níveis do planejamento experimental.

Algoritmo	Parâmetro/fator	Níveis
ILS	<i>MaxIter</i> ILS	100; 150; 200
	<i>nP</i>	1; 3; 5
LNS	<i>MaxIter</i> LNS	500; 1000; 1500
	<i>q</i>	Até 0,2n; 0,3n; 0,4n

foram realizados em um computador Dell Precision T7600 CPU E5-6280 2.70 GHz com 192 GB RAM, usando-se um único núcleo do processador. O desempenho das abordagens em cada instância foi medido como o *gap* do custo da solução encontrada ( $Alg_{sol}$ ) em relação ao custo da melhor solução encontrada por todas as configurações ( $Min_{sol}$ ), conforme definido na Equação 7.

$$gap = \frac{Alg_{sol} - Min_{sol}}{Min_{sol}} \tag{7}$$

Os resultados foram analisados por meio de uma análise de variância (ANOVA), verificando todos os pressupostos (normalidade, homocedasticidade e independência dos erros) utilizando as técnicas correspondentes, não se encontrando nenhuma evidência para questionar a validade do experimento. Os gráficos de médias e intervalos de Tukey com 95% de confiança são mostrados nas Figuras 9-10, cabendo ressaltar que a sobreposição entre intervalos indica que não existem diferenças estatisticamente significativas entre as médias.

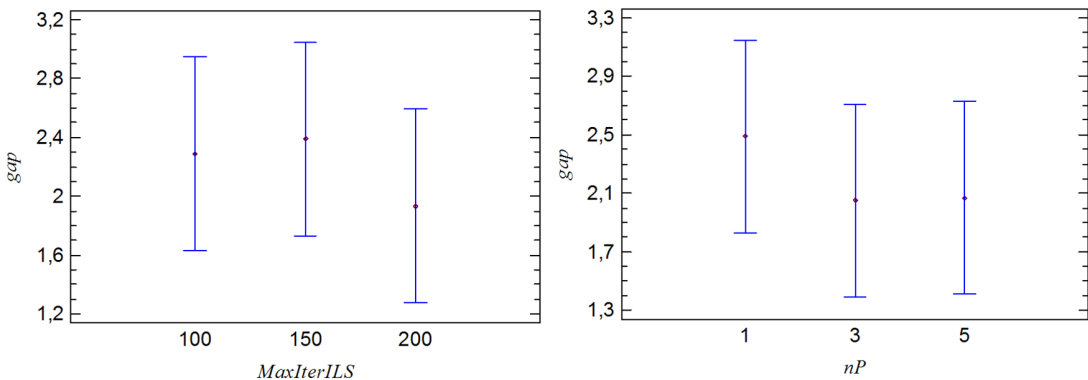
Conforme pode ser observado nas Figuras 9 e 10, os resultados mostraram que não existem diferenças estatisticamente significativas no desempenho de nenhum dos métodos, nos diferentes níveis de seus

parâmetros. Esse resultado pode ser explicado pela grande dependência que as abordagens metaheurísticas têm das heurísticas adicionais (redução de rotas e de entregadores), devido às características específicas do problema. Note que as heurísticas adicionais não possuem parâmetros e, conseqüentemente, fazem com que a sensibilidade das abordagens em relação aos parâmetros seja baixa.

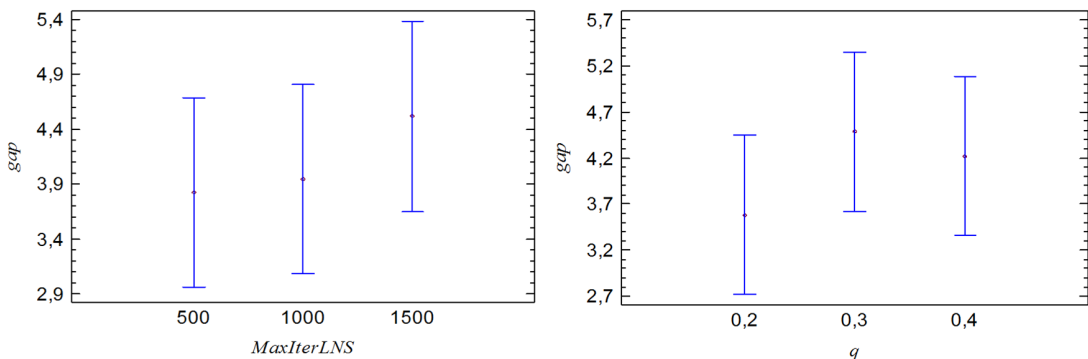
Dado que não existem diferenças estatisticamente significativas, os valores dos parâmetros foram definidos com base nos melhores resultados dos testes, ficando fixados como segue:  $MaxIterILS = 200$ ,  $nP = 3$ ,  $MaxIterLNS = 1000$  e  $q = rand(0,1n; 0,2n)$ . Por outro lado, o valor do componente de aleatorização nos operadores de remoção foi  $p = 3$ , conforme a proposta de Ropke & Pisinger (2006). Este valor é suficientemente grande para permitir a remoção de grupos de maior custo toda vez que os operadores são usados e, ainda, permite evitar as situações nas quais os mesmos grupos são removidos frequentemente.

### 4.3 Avaliação das abordagens metaheurísticas

Nesta seção, são apresentados os experimentos computacionais realizados após a fase de calibração da seção anterior. Nestes experimentos, os custos



**Figura 9.** Gráfico de médias e intervalos de confiança de Tukey para os parâmetros da ILS.



**Figura 10.** Gráfico de médias e intervalos de confiança de Tukey para os parâmetros da LNS.

da função objetivo foram definidos com os mesmos valores usados por Pureza et al. (2012), sendo eles:  $p_1=1, p_2=0,1$  e  $p_3=0,0001$ ; os quais priorizam a minimização do número de veículos seguido do número de entregadores e finalmente da distância total. Os experimentos são executados em um computador Intel Core i7 3.40 GHz com 16 GB RAM, usando-se um único núcleo do processador, com um limite de execução de 600 segundos e resolvendo cada instância 5 vezes.

Os resultados obtidos pelas abordagens são comparados com os resultados reportados em Pureza et al. (2012), para suas abordagens de Busca Tabu (TS-PMR) e de Otimização por Colônia de Formigas (ACO-PMR), as quais foram executadas em um computador com processador Intel Core2 2.40 GHz com 2 GB RAM. Também, foram usados os resultados da abordagem de Otimização por Colônia de Formigas (ACO-SR) exatamente como reportados em Senarclens de Grancy & Reimann (2014), os quais foram obtidos em um computador com processador Intel E8400. Nas tabelas a seguir, os rótulos *Custo*, *Veic*, *Dist*, *Entr* e *Tempo* indicam o custo total das soluções, o número de veículos usados, a distância total percorrida, o número de entregadores designados e o tempo de execução dos métodos (em segundos), respectivamente. Os melhores resultados para cada instância/classe, em termos de custo, são destacados em negrito. Como os custos são apresentados com precisão de duas casas decimais apenas, o desempate

entre esses valores é feito baseando-se na distância total correspondente, já que esse termo possui peso igual a 0,0001 na função objetivo.

Inicialmente, analisamos e comparamos o desempenho global das abordagens em todas as classes de instâncias em termos de qualidade das soluções obtidas. Nesse sentido, só é possível comparar com os resultados obtidos por Pureza et al. (2012), já que no artigo de Senarclens de Grancy & Reimann (2014) só se reportam os resultados para a classe R1. A Tabela 3 mostra os melhores resultados (agrupados) obtidos pelas abordagens metaheurísticas para cada uma das classes de instâncias, considerando as cinco execuções de cada abordagem. Observa-se que as abordagens propostas obtiveram resultados iguais ou superiores em todas as classes de instâncias em relação às abordagens TS-PMR e ACO-PMR, em termos da qualidade das soluções obtidas. Mais detalhadamente, comparando só o desempenho das duas abordagens metaheurísticas propostas, obteve-se que a ILS encontrou a melhor solução em 9 das 12 instâncias da classe R1, em 5 das 8 instâncias da classe RC1, em 9 das 11 instâncias da classe R2 e em 4 das 8 instâncias da classe RC2. Em relação às classes C1 e C2, ambas abordagens obtiveram as mesmas soluções para todas as instâncias. Contudo, apesar do domínio da ILS em relação ao número de melhores soluções, em termos de resultados médios agrupados, a ILS domina a LNS somente em uma das seis classes de instância, é dominada em três

**Tabela 3.** Melhores resultados (agrupados) obtidos pelas abordagens.

Método		Classe de instâncias					
		R1	RC1	C1	R2	RC2	C2
TS-PMR (melhor de 5)	Custo	15,70	16,64	11,08	3,75	4,45	3,36
	Veic	12,33	13,00	10,00	2,90	3,40	3,00
	Dist	1258,00	1527,90	830,70	1034,00	1230,40	597,20
	Entr	32,42	34,90	10,00	7,50	9,30	3,00
	Tempo	640,10	677,10	265,10	425,40	419,10	246,80
ACO-PMR (melhor de 5)	Custo	15,77	16,70	11,08	3,86	4,58	3,36
	Veic	12,50	13,00	10,00	3,10	3,60	3,00
	Dist	1261,50	1480,10	833,60	1064,20	1296,00	609,30
	Entr	31,40	35,50	10,00	6,50	8,50	3,00
	Tempo	575,80	508,60	375,20	600,60	462,00	243,30
ILS (melhor de 5)	Custo	15,34	16,59	<b>11,08</b>	<b>3,63</b>	4,31	<b>3,36</b>
	Veic	12,17	13,00	<b>10,00</b>	<b>2,91</b>	3,38	<b>3,00</b>
	Dist	1271,71	1482,46	<b>827,64</b>	<b>993,17</b>	1186,61	<b>587,51</b>
	Entr	30,50	34,38	<b>10,00</b>	<b>6,18</b>	8,13	<b>3,00</b>
	Tempo	600,75	601,25	<b>600,22</b>	<b>604,09</b>	603,25	<b>601,50</b>
LNS (melhor de 5)	Custo	<b>15,32</b>	<b>16,50</b>	<b>11,08</b>	3,64	<b>4,29</b>	<b>3,36</b>
	Veic	<b>12,08</b>	<b>12,88</b>	<b>10,00</b>	2,91	<b>3,38</b>	<b>3,00</b>
	Dist	<b>1271,64</b>	<b>1492,29</b>	<b>827,64</b>	998,00	<b>1197,49</b>	<b>587,51</b>
	Entr	<b>31,08</b>	<b>34,75</b>	<b>10,00</b>	6,27	<b>8,00</b>	<b>3,00</b>
	Tempo	<b>602,42</b>	<b>603,25</b>	<b>600,67</b>	610,09	<b>609,50</b>	<b>602,75</b>

classes e empatam para as outras duas, indicando que quando a LNS consegue superar a ILS em uma instância, o faz com diferenças significativas de forma que permite dominá-la em termos de resultados médios.

A Tabela 4 apresenta os resultados médios (agrupados) para cada classe de instâncias, considerando as cinco execuções de cada abordagem. Semelhante aos resultados mostrados na Tabela 3, constata-se que as abordagens metaheurísticas baseadas em ILS e LNS superaram em todas as classes os resultados médios das abordagens ACO-PMR e TS-PMR.

Os resultados encontrados para a classe R1 serão detalhados para cada instância, já que suas características fazem com que elas representem mais acertadamente a importância dos tempos de serviço. Esta classe de instância foi detalhadamente abordada nos trabalhos encontrados sobre o PRVJTME na literatura. A Tabela 5 mostra as melhores soluções encontradas pelas abordagens ILS e LNS, como também as melhores soluções reportadas em Pureza et al. (2012) e Senarclens de Grancy & Reimann (2014). Neste último, foram propostas duas metaheurísticas, uma baseada em ACO e outra baseada em GRASP. Na presente pesquisa, se comparou apenas com a ACO (ACO-SR), pois esta apresentou melhores resultados para a classe R1. Os resultados mostram que as abordagens ILS e LNS foram capazes de encontrar a melhor solução em 6 das 12 instâncias

da classe. As abordagens propostas conseguiram superar o desempenho geral da TS-PMR, sendo que a ILS diminuiu em 1,35% e 5,91% o número médio de veículos e entregadores utilizados, respectivamente, enquanto a LNS conseguiu reduzir 2,03% e 4,11% o número médio de veículos e entregadores utilizados, respectivamente. Comparando com ACO-SR, a ILS obteve resultados com o mesmo número de veículos e reduzindo 0,81% o número médio de entregadores, enquanto a LNS diminuiu em 0,68% o número de veículos utilizando 1,08% mais entregadores. Vale ressaltar que para a instância R101, cuja solução ótima é conhecida, as abordagens metaheurísticas encontraram soluções usando a quantidade ótima de veículos e entregadores, com um incremento de 2,17% e 2,03% na distância total percorrida, respectivamente.

Finalmente, com o intuito de mostrar uma visão da robustez das abordagens propostas e, considerando que ambas têm componentes aleatórios, na Tabela 6 são mostrados os resultados médios e os desvios padrão (entre parênteses) do custo da solução, número de veículos, distância total percorrida, número de entregadores e os tempos de execução (em segundos), considerando as 5 execuções de cada abordagem para a classe R1. Note que os desvios padrão são relativamente pequenos quando comparados com os valores das médias.

**Tabela 4.** Resultados médios (agrupados) obtidos pelas abordagens.

Método		Classe de instâncias					
		R1	RC1	C1	R2	RC2	C2
TS-PMR (média de 5)	Custo	16,20	17,12	11,08	3,89	4,50	3,37
	Veic	12,80	13,40	10,00	3,00	3,40	3,00
	Dist	1272,40	1511,40	847,70	1046,80	1251,30	653,50
	Entr	32,70	35,70	10,00	7,90	9,70	3,00
	Tempo	658,20	686,50	245,70	393,00	400,50	261,90
ACO- PMR (média de 5)	Custo	15,90	17,08	11,08	3,89	4,69	3,36
	Veic	12,60	13,40	10,00	3,10	3,70	3,00
	Dist	1263,20	1496,30	838,80	1070,30	1307,90	623,70
	Entr	31,70	35,30	10,00	6,80	8,60	3,00
	Tempo	508,80	472,80	389,10	527,50	455,60	263,80
ILS (média de 5)	Custo	<b>15,45</b>	16,68	<b>11,08</b>	3,76	4,35	<b>3,36</b>
	Veic	<b>12,25</b>	13,10	<b>10,00</b>	3,05	3,43	<b>3,00</b>
	Dist	<b>1260,58</b>	1478,90	<b>827,64</b>	988,71	1203,95	<b>587,51</b>
	Entr	<b>30,73</b>	34,35	<b>10,00</b>	6,07	8,03	<b>3,00</b>
	Tempo	<b>601,03</b>	600,98	<b>600,44</b>	606,27	604,20	<b>603,29</b>
LNS (média de 5)	Custo	15,51	<b>16,60</b>	<b>11,08</b>	<b>3,73</b>	<b>4,31</b>	<b>3,36</b>
	Veic	12,30	<b>12,98</b>	<b>10,00</b>	<b>3,02</b>	<b>3,38</b>	<b>3,00</b>
	Dist	1267,65	<b>1488,90</b>	<b>827,64</b>	<b>997,41</b>	<b>1217,48</b>	<b>587,51</b>
	Entr	30,87	<b>34,80</b>	<b>10,00</b>	<b>6,15</b>	<b>8,10</b>	<b>3,00</b>
	Tempo	602,45	<b>602,65</b>	<b>601,89</b>	<b>613,05</b>	<b>608,95</b>	<b>606,42</b>

Tabela 5. Melhores resultados obtidos para a classe R1.

Método	Instância											Média	Soma		
	R101	R102	R103	R104	R105	R106	R107	R108	R109	R110	R111			R112	
TS-PMR (melhor de 5)	Custo	23,67	21,05	16,33	12,91	17,84	15,23	13,01	12,80	15,43	14,12	13,11	12,90	15,70	188,41
	Veic	19	17	13	10	14	12	10	10	12	11	10	10	12,33	148
	Dist	1740,00	1520,00	1285,00	1057,00	1446,00	1323,00	1112,00	967,00	1296,00	1217,00	1137,00	996,00	1258,00	15096,00
	Entr	45	39	32	28	37	31	29	27	33	30	30	28	32,42	389
	Tempo	645	655	959	692	463	492	473	953	428	620	616	686	640,17	7682
ACO-SR (melhor de 5)	Custo	<b>23,67</b>	20,95	15,94	<b>12,70</b>	<b>17,64</b>	15,13	<b>12,81</b>	11,70	15,12	<b>13,92</b>	<b>13,01</b>	11,80	15,37	184,40
	Veic	<b>19</b>	17	13	<b>10</b>	<b>14</b>	12	<b>10</b>	9	12	<b>11</b>	<b>10</b>	9	12,17	146
	Dist	<b>1725,46</b>	1533,40	1371,63	<b>1045,68</b>	<b>1412,52</b>	1301,34	<b>1108,92</b>	967,18	1229,72	<b>1154,95</b>	<b>1134,16</b>	996,32	1248,44	14981,28
	Entr	<b>45</b>	38	28	<b>26</b>	<b>35</b>	30	<b>27</b>	26	30	<b>28</b>	<b>29</b>	27	30,75	369
	Tempo	<b>960</b>	960	960	<b>960</b>	<b>960</b>	960	<b>960</b>	960	960	<b>960</b>	<b>960</b>	960	960,00	11520
ILS (melhor de 5)	Custo	23,68	<b>20,95</b>	<b>15,84</b>	12,71	<b>17,64</b>	15,04	12,81	<b>11,70</b>	15,03	13,92	13,02	<b>11,80</b>	15,34	184,13
	Veic	19	17	<b>13</b>	10	<b>14</b>	12	10	9	12	11	10	<b>9</b>	12,17	146
	Dist	1757,40	<b>1530,92</b>	<b>1410,73</b>	1062,58	<b>1412,52</b>	1352,09	1140,79	<b>965,31</b>	1291,62	1187,49	1161,96	<b>987,14</b>	1271,71	15260,57
	Entr	45	<b>38</b>	<b>27</b>	26	<b>35</b>	29	<b>27</b>	<b>26</b>	29	28	29	<b>27</b>	30,50	366
	Tempo	600	<b>601</b>	<b>601</b>	601	<b>600</b>	600	603	<b>601</b>	600	600	602	<b>600</b>	600,75	7209
LNS (melhor de 5)	Custo	23,68	20,95	15,94	12,71	<b>17,64</b>	<b>15,03</b>	12,91	11,70	<b>14,43</b>	13,92	13,11	11,80	15,32	183,83
	Veic	19	17	13	10	<b>14</b>	12	10	9	<b>11</b>	11	10	9	12,08	145
	Dist	1755,04	1531,36	1369,12	1072,41	<b>1412,52</b>	<b>1328,21</b>	1139,96	1011,37	<b>1281,84</b>	1219,12	1149,03	989,70	1271,64	15259,70
	Entr	45	38	28	26	<b>35</b>	<b>29</b>	28	26	<b>33</b>	28	30	27	31,08	373
	Tempo	600	603	603	603	<b>600</b>	<b>601</b>	600	601	<b>605</b>	603	607	603	602,42	7229



**Tabela 6.** Resultados médios (desvios padrão) das abordagens para a classe R.I.

Instância	Resultados médios (desvio padrão) ILS				Resultados médios (desvio padrão) LNS					
	Custo	Veic	Dist	Entr	Tempo	Custo	Veic	Dist	Entr	Tempo
R101	23,75(0,04)	19,00(0,00)	1737,37(11,99)	45,80(0,45)	600,20(0,45)	23,73(0,05)	19,00(0,00)	1739,67(14,54)	45,60(0,55)	600,60(0,89)
R102	20,95(0,00)	17,00(0,00)	1531,27(0,19)	38,00(0,00)	600,80(0,84)	20,95(0,00)	17,00(0,00)	1531,66(0,34)	38,00(0,00)	601,40(1,14)
R103	15,94(0,07)	13,00(0,00)	1375,83(23,83)	28,00(0,71)	601,60(1,95)	15,94(0,00)	13,00(0,00)	1380,77(13,08)	28,00(0,00)	602,60(1,14)
R104	12,73(0,04)	10,00(0,00)	1063,26(13,97)	26,20(0,45)	602,20(1,30)	12,79(0,04)	10,00(0,00)	1061,47(6,51)	26,80(0,45)	605,60(2,51)
R105	17,64(0,00)	14,00(0,00)	1412,52(0,00)	35,00(0,00)	600,00(0,00)	17,64(0,00)	14,00(0,00)	1413,21(1,02)	35,00(0,00)	600,40(0,55)
R106	15,11(0,04)	12,00(0,00)	1307,63(25,17)	29,80(0,45)	601,40(1,14)	15,07(0,05)	12,00(0,00)	1334,82(43,79)	29,40(0,55)	600,40(0,55)
R107	12,95(0,11)	10,00(0,00)	1127,93(15,21)	28,40(1,14)	601,20(1,10)	12,97(0,05)	10,00(0,00)	1137,38(21,06)	28,60(0,55)	600,80(1,79)
R108	11,72(0,04)	9,00(0,00)	975,94(9,06)	26,20(0,45)	601,40(0,89)	11,94(0,37)	9,20(0,45)	989,26(16,40)	26,40(0,89)	602,60(2,07)
R109	15,09(0,05)	12,00(0,00)	1261,06(35,47)	29,60(0,55)	600,40(0,55)	14,99(0,31)	11,80(0,45)	1278,42(6,91)	30,60(1,34)	602,80(1,48)
R110	13,94(0,04)	11,00(0,00)	1184,00(26,59)	28,20(0,45)	601,00(1,00)	13,96(0,05)	11,00(0,00)	1203,71(34,82)	28,40(0,55)	602,40(2,30)
R111	13,61(0,34)	10,80(0,45)	1138,48(25,28)	27,00(1,22)	601,40(1,34)	13,65(0,30)	10,80(0,45)	1135,86(9,86)	27,40(1,52)	604,40(2,70)
R112	11,96(0,36)	9,20(0,45)	1011,68(15,77)	26,60(0,89)	600,80(1,10)	12,52(0,40)	9,80(0,45)	1005,54(15,27)	26,20(0,45)	605,40(4,51)

## 5 Conclusões e perspectivas

Neste trabalho, abordou-se o problema de roteamento de veículos com janelas de tempo e múltiplos entregadores por meio de duas abordagens metaheurísticas baseadas nas metaheurísticas Busca Local Iterada (ILS) e Busca em Grande Vizinhança (LNS), acompanhadas de um conjunto adicional de heurísticas de melhoria. Ambas as abordagens foram testadas utilizando seis classes de instâncias e seus desempenhos foram comparados entre si e com outras abordagens propostas na literatura. Os testes computacionais mostraram que dentre as abordagens propostas não houve uma abordagem superior para todas as instâncias utilizadas. A LNS dominou a ILS em três das seis classes, enquanto a ILS dominou em uma das classes. Além disso, ambas as abordagens propostas produziram bons resultados sendo capazes de obter soluções com qualidade igual ou superior às obtidas por abordagens propostas por outros autores, os quais eram até então os melhores resultados disponíveis na literatura. Perspectivas para pesquisas futuras incluem a combinação destas abordagens com métodos exatos como a geração de colunas ou com *softwares* de otimização comerciais para gerar métodos híbridos para o PRVJTME. Também, pode-se estender o problema para incorporar características práticas ao problema como frota heterogênea, múltiplos depósitos ou entregas fracionadas.

## Agradecimentos

Esta pesquisa contou com apoio financeiro da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES); da Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP), processo número 2014/00939-8; e do Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), processo número 482664/2013-4.

## Referências

- Azi, N., Gendreau, M., & Potvin, J. Y. (2014). An adaptive large neighborhood search for a vehicle routing problem with multiple routes. *Computers & Operations Research*, 41(1), 167-173. <http://dx.doi.org/10.1016/j.cor.2013.08.016>.
- Bräysy, O., & Gendreau, M. (2005). Vehicle Routing Problem with Time Windows, Part I: Route Construction and Local Search Algorithms. *Transportation Science*, 39(1), 104-118. <http://dx.doi.org/10.1287/trsc.1030.0056>.
- Ferreira, V., & Pureza, V. (2012). Some experiments with a savings heuristic and a tabu search approach for the vehicle routing problem with multiple deliverymen. *Pesquisa Operacional*, 32(2), 443-463. <http://dx.doi.org/10.1590/S0101-74382012005000016>.
- Hong, L. (2012). An improved LNS algorithm for real-time vehicle routing problem with time windows. *Computers & Operations Research*, 39(2), 151-163. <http://dx.doi.org/10.1016/j.cor.2011.03.006>.
- Lourenço, H., Martin, O., & Stützle, T. (2010). Iterated local search: framework and applications. In M. Gendreau & J.-Y. Potvin (Eds.), *Handbook of metaheuristics* (pp. 363-397). Boston, MA: Springer US.
- Lourenço, H. R., Martin, O. C., & Stützle, T. (2003). Iterated local search. In F. Glover & G. A. Kochenberger (Eds.), *Handbook of metaheuristics* (pp. 321-353). Boston: Springer US.
- Melechovsky, J. (2012). Evolutionary local search algorithm to solve the multi-compartment vehicle routing problem with time windows. In *Proceedings of the XXX International Conference Mathematical Methods in Economics* (pp. 564-568). Karviná: Silesian University.
- Mladenovic, N., & Hansen, P. (1997). Variable neighborhood search. *Computers & Operations Research*, 24(11), 1097-1100. [http://dx.doi.org/10.1016/S0305-0548\(97\)00031-2](http://dx.doi.org/10.1016/S0305-0548(97)00031-2).
- Montgomery, D. C. (2012). *Design and analysis of experiments* (Vol. 2). Hoboken: Wiley.
- Naderi, B., Ruiz, R., & Zandieh, M. (2010). Algorithms for a realistic variant of flowshop scheduling. *Computers & Operations Research*, 37(2), 236-246. <http://dx.doi.org/10.1016/j.cor.2009.04.017>.
- Pisinger, D., & Ropke, S. (2007). A general heuristic for vehicle routing problems. *Computers & Operations Research*, 34(8), 2403-2435. <http://dx.doi.org/10.1016/j.cor.2005.09.012>.
- Pisinger, D., & Ropke, S. (2010). Large neighborhood search. In M. Gendreau & J.-Y. Potvin. *Handbook of metaheuristics* (Vol. 146, pp. 1-22). Boston: Springer US.
- Pureza, V., Morabito, R., & Reimann, M. (2012). Vehicle routing with multiple deliverymen: Modeling and heuristic approaches for the VRPTW. *European Journal of Operational Research*, 218(3), 636-647. <http://dx.doi.org/10.1016/j.ejor.2011.12.005>.
- Ropke, S., & Pisinger, D. (2006). An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows. *Transportation Science*, 40(4), 455-472. <http://dx.doi.org/10.1287/trsc.1050.0135>.
- Senarclens de Grancy, G., & Reimann, M. (2014). Vehicle routing problems with time windows and multiple service workers: a systematic comparison between ACO and GRASP. *Central European Journal of Operations Research*, 24(1): 29-48.
- Shaw, P. (1998). Using constraint programming and local search methods to solve vehicle routing problems. In M. Maher & J.-F. Puget. *Principles and practice of constraint programming - CP98* (pp. 417-431). Berlin: Springer Berlin.
- Silva, M. M., Subramanian, A., & Ochi, L. S. (2015). An iterated local search heuristic for the split delivery vehicle routing problem. *Computers & Operations Research*, 53, 234-249. <http://dx.doi.org/10.1016/j.cor.2014.08.005>.
- Solomon, M. (1987). Algorithms for the vehicle routing and scheduling problems with time window constraints. *Operations Research*, 35(2), 254-265. <http://dx.doi.org/10.1287/opre.35.2.254>.
- Subramanian, A., Penna, P. H. V., Uchoa, E., & Ochi, L. S. (2012). A hybrid algorithm for the Heterogeneous Fleet Vehicle Routing Problem. *European Journal of Operational Research*, 221(2), 285-295. <http://dx.doi.org/10.1016/j.ejor.2012.03.016>.
- Toth, P., & Vigo, D. (2002). *The vehicle routing problem: the vehicle routing problem*. Canada: Society for Industrial and Applied Mathematics. <http://dx.doi.org/10.1137/1.9780898718515>.