

## 4 Algoritmos para o Problema de Roteirização Dinâmica de Veículos com Janelas de Tempo

Este capítulo apresenta as definições do Problema de Roteirização Dinâmica de Veículos com Janelas de Tempo abordado e seis algoritmos propostos para resolvê-lo. São apresentados os resultados computacionais utilizando instâncias utilizadas com *benchmark* na literatura.

### 4.1 Introdução

Os algoritmos propostos a seguir se baseiam nos algoritmos Inserção (PFIH) proposto por Solomon (1987), no algoritmo Descida em Vizinhança Variável (*Variable Neighborhood Descent - VND*) proposto por Mladenović e Hansen (1997), no algoritmo MACS-RVND proposto neste trabalho e no algoritmo ACS-DVRP proposto por Montemanni et al. (2002) para o DVRP.

#### 4.1.1 Definições do DVRPTW

No contexto da taxonomia proposta por Psaraftis (1995) e adaptada por Pillac et al. (2013), o DVRPTW abordado neste trabalho trata problemas que são dinâmicos e determinísticos. Não são aceitas interrupções do destino atual dos veículos (não-preemptivo) (Gendreau et al., 1999). Novos veículos podem ser inseridos na solução, desde que estejam disponíveis no depósito. A restrição que obriga que todos os clientes sejam visitados é relaxada, podendo existir soluções factíveis com clientes não atendidos. O problema passa a ter um terceiro objetivo, que é a minimização do número de clientes não atendidos. Os objetivos passam a ser: minimizar o número de clientes não atendidos (NA), minimizar o número de veículos (NV) e minimizar a distância total (DT). Baseando-se na definição do VRPTW (item 2.2.1), a cada cliente  $i \in \{1, 2, \dots, N\}$  são associados os parâmetros:

- momento que o cliente é conhecido  $t_{ai} \geq 0$ ;
- estado do cliente  $e_i \in \{planejado, atribuído, em\_trânsito, entregue\}$

A metodologia aplicada consiste na solução de uma sequência de  $n$  problemas estáticos e determinísticos  $PE_n$  com várias soluções  $s_n$  durante a jornada de trabalho  $tempo_{total} = b_{n+1}$ . O tamanho de  $n$  depende da abordagem utilizada na fase a-posteriori. Na re-otimização *contínua*,  $n$  será igual ao número de novos clientes. Já na re-otimização *periódica*,  $n$  será igual ao número de faixas de tempo  $nts$  de tamanhos iguais  $f$ . Este é um processo iterativo, que evolui em função do tempo e novas entradas devem ser inseridas no procedimento durante sua execução. Para flexibilizar o uso das abordagens de re-roteirização contínua e periódica se propõe o algoritmo DVRPTW (Algoritmo 19).

---

**Algoritmo 19: DVRPTW**


---

```

1: Procedimento DVRPTW(estratégia, tipo_método)
2:  $tempo \leftarrow 0$ 
3:  $n \leftarrow 0$ 
4:  $PE_n \leftarrow \{\text{depósito e clientes conhecidos, com } tai \leq tempo\}$ 
5: // Obter uma solução inicial utilizando um método de roteirização estática para o VRPTW
6:  $s_n \leftarrow obter\_solução\_inicial(N)$ 
7: Para  $tempo = 1 \dots tempo_{total}$  faça
8:    $D \leftarrow D \cup \{\text{novos clientes com } tai = tempo\}$ 
9:   Se  $D \neq \{\}$  então
10:    Se estratégia = contínua OU (estratégia = periódica E  $tempo \text{ MOD } f = 0$ )
11:      $n \leftarrow n + 1$ 
12:      $PE_n \leftarrow PE_{n-1} \cup D$ 
13:      $D \leftarrow \{\}$ 
14:     Se tipo_método = inserção então
15:        $s' \leftarrow s_{n-1}$ 
16:     Senão Se tipo_método = re-roteirização então
17:       // Remover de  $s$  os clientes com  $e_i \notin \{em\_trânsito, entregue, atribuído\}$ 
18:        $s' \leftarrow extrair\_solução\_parcial(s_{n-1})$ 
19:     Fim se
20:     // Obter uma solução utilizando um método de roteirização estática adaptado
21:      $s_n \leftarrow obter\_solução\_intermediaria(s', PE_n)$ 
22:   Fim se
23: Fim se
24: // Atualizar os estados dos clientes
25: Para  $r1 = 1 \dots V$  faça
26:   // Para todos os clientes  $i$  da rota  $r1$ 
27:   Para  $i=1$  até o tamanho de  $r1$  faça
28:      $inicio\_viagem_i \leftarrow T_i - t_{i-1,i}$ 
29:     Se  $tempo \geq inicio\_viagem_i$  E  $tempo < T_i$  então
30:        $e_i \leftarrow em\_trânsito$ 
31:     Senão Se  $tempo \geq T_i$  E  $tempo < T_i + s_i$  então
32:        $e_i \leftarrow entregue$ 
33:     Senão Se estratégia = periódica E  $inicio\_viagem_i \leq n.f$  então
34:        $e_i \leftarrow atribuído$ 
35:     Senão
36:        $e_i \leftarrow planejado$ 
37:   Fim se
38: Fim para
39: Fim para
40: Retorne  $s$ 

```

---

O Algoritmo 19 é iniciado com uma *estratégia* de re-otimização *contínua* ou *periódica* e com um *tipo\_método* que pode ser de *inserção* ou *re-roteirização*. No problema  $PE_0$  são inseridos o depósito e os clientes já conhecidos e uma solução inicial  $s_0$  é obtida utilizando um método desejado para resolver o VRPTW. Um procedimento iterativo em função do tempo é iniciado com  $tempo=1$  até o fim da jornada de trabalho ( $tempo\_total$ ). Novos clientes dinâmicos são conhecidos e inseridos no conjunto D. Se o conjunto D possuir algum cliente, então é verificada a abordagem da re-otimização, caso contrário são atualizados os estados dos clientes. Se a abordagem for *contínua* ou *periódica* com tempo igual ao início de uma faixa de tempo, então é incrementado o contador  $n$ . Em seguida é criado um novo problema estático  $PE_n$  com a união dos nós de  $PE_{n-1}$  e dos nós do conjunto D. Uma solução parcial  $s'$  é criada. Se o *tipo\_método* for *inserção*, então é utilizada  $s_{n-1}$  como solução parcial. Caso contrário é extraída uma solução parcial de  $s_{n-1}$  com apenas os clientes comprometidos, ou seja, aqueles que possuem o estado *em\_trânsito*, *entregue* ou *atribuído*. Uma nova solução intermediária  $s_n$  é obtida para o problema estático  $PE_n$ , utilizando a solução parcial  $s'$  para iniciar as novas rotas. Se necessário, novos veículos  $v$  podem ser alocados para atender o maior número de clientes possível. A atualização dos estados dos clientes é realizada em todas as rotas da solução  $s_n$ . Se o *tempo* for maior ou igual ao início da viagem do veículo  $v$  na rota  $rl$  para o cliente  $i$  e menor que o tempo de chegada  $T_i$ , então o estado  $e_i$  é atualizado para *em\_trânsito*. Caso contrário, se o *tempo* for maior ou igual o tempo de chegada  $T_i$  e menor que o tempo de processamento  $tp_i$  ( $T_i + s_i$ ), então o estado  $e_i$  é atualizado para *entregue*. Caso contrário, se a estratégia for de re-otimização *periódica* e início da viagem em direção ao cliente  $i$  for menor do que o tamanho da faixa de tempo  $f$  multiplicado pelo número da faixa  $n$ , então o estado  $e_i$  é atualizado para *atribuído*. Caso contrário, o estado  $e_i$  é atualizado para *planejado*. Ao fim da jornada de trabalho, a solução  $s_n$  será a solução final para o DVRPTW.

#### 4.1.2 Adaptações dos algoritmos para o VRPTW estático

Para obter as soluções intermediárias do DVRPTW (linha 21: do Algoritmo 19) para os problemas estáticos  $PE_1, \dots, PE_{n-1}$ , os algoritmos propostos para resolver o VRPTW estático necessitam de adaptações, as quais dependem da

classificação do algoritmo utilizado. Conforme descrito na seção 2.3.1.3.3, existem duas classificações: algoritmos de re-roteirização e de inserção.

#### 4.1.3 Adaptações para métodos de inserção

As adaptações a serem implementadas nos algoritmos de inserção são:

- Uma solução completa  $s'$  é utilizada como entrada no algoritmo. Esta solução contém todos os clientes já roteirizados.
- Um cliente  $j$  poderá ser inserido entre os clientes  $i$  e  $i+1$  se o cliente  $i+1$  não estiver comprometido,  $e_{i+1} \notin \{em\_tr\grave{a}nsito, entregue, atribu\grave{i}do\}$ .

#### 4.1.4 Adaptações para métodos de re-roteirização

As adaptações a serem implementadas nos algoritmos construtivos para realizar a re-roteirização são:

- Uma solução parcial  $s'$  é utilizada como entrada no algoritmo. Esta solução contém rotas apenas para os clientes que já foram comprometidos.
- As rotas de  $s'$  podem estar abertas ou encerradas.
  - Uma rota aberta inicia no depósito, atende os clientes comprometidos e não é finalizada com o retorno ao depósito.
  - Uma rota encerrada indica que o veículo iniciou no depósito, atendeu os clientes comprometidos e já retornou ao depósito, encerrando sua jornada de trabalho.
- A solução parcial  $s'$  é utilizada como solução atual no algoritmo.
  - As rotas encerradas não são alteradas.
  - As rotas abertas podem ser alteradas. Os últimos clientes comprometidos de cada rota são utilizados como início e os clientes não roteirizados são inseridos sequencialmente nestas rotas.

#### 4.1.5 Algoritmos propostos para o DVRPTW

Os algoritmos propostos são detalhados a seguir. A Tabela 5 mostra os nomes e os respectivos tipos de re-otimização e classificação dos algoritmos propostos para resolver o DVRPTW.

Tabela 5 – Algoritmos propostos para o DVRPTW

Nome do Algoritmo	Abordagem de re-otimização	Classificação
PFIH_PFIH	Contínua	Inserção
PFIH_PFIH-VND	Contínua	Inserção
PFIH-VND_PFIH-VND	Contínua	Inserção
MACS-RVND_PFIH-VND	Contínua	Inserção
MACS-RVND_Re-roteiriza	Contínua	Re-roteirização
MACS-RVND-Periódico	Periódica	Re-roteirização

Todos os algoritmos propostos se baseiam no Algoritmo 19, proposto para resolver o DVRPTW. A principal diferença entre estes algoritmos está nos métodos para obter a solução inicial (linha 6:) e as soluções intermediárias (linha 21:), conforme mostra a Tabela 6.

Tabela 6 – Métodos para obter as soluções iniciais e intermediárias para os algoritmos propostos

Nome do Algoritmo	Solução inicial	Solução intermediária
PFIH_PFIH	PFIH	PFIH
PFIH_PFIH-VND	PFIH	PFIH + VND
PFIH-VND_PFIH-VND	PFIH + VND	PFIH + VND
MACS-RVND_PFIH-VND	MACS-RVND	PFIH + VND
MACS-RVND_Re-roteiriza	MACS-RVND	MACS-RVND
MACS-RVND-Periódico	MACS-RVND	MACS-RVND

##### 4.1.5.1 O algoritmo PFIH\_PFIH

O algoritmo PFIH\_PFIH utiliza a abordagem de re-otimização contínua e é classificado como um método de inserção. Ele se utiliza do algoritmo I1 nas fases a-priori (para obter uma solução inicial) e a-posteriori (para inserir novos clientes dinâmicos à medida que são conhecidos).

##### 4.1.5.2 O algoritmo PFIH\_PFIH-VND

O algoritmo PFIH\_PFIH-VND utiliza a abordagem de re-otimização contínua e é classificado como um método de inserção. Ele se utiliza do algoritmo PFIH na fase a-priori. Na fase a-posteriori se utiliza do algoritmo PFIH com uma fase de busca local com o algoritmo VND. Este algoritmo busca melhorar a inserção míope do algoritmo PFIH\_PFIH, a partir da aplicação da busca local.

#### 4.1.5.3 O algoritmo PFIH-VND\_PFIH-VND

O algoritmo PFIH-VND\_PFIH-VND utiliza a abordagem de re-otimização contínua e é classificado como um método de inserção. Ele se utiliza do algoritmo PFIH com uma fase de busca local com o algoritmo VND nas fases a-priori e a-posteriori.

#### 4.1.5.4 O algoritmo MACS-RVND\_PFIH-VND

O algoritmo MACS-RVND\_PFIH-VND utiliza a abordagem de re-otimização contínua e é classificado como um método de inserção. Ele se utiliza do algoritmo MACS-RVND na fase a-priori. Na fase a-posteriori se utiliza do algoritmo PFIH com uma fase de busca local com o algoritmo VND.

#### 4.1.5.5 O algoritmo MACS-RVND-Re-roteiriza

O algoritmo MACS-RVND-Re-roteiriza utiliza a abordagem de re-otimização contínua e é classificado como um método de re-roteirização. Ele se utiliza do algoritmo MACS-RVND na fase a-priori. Na fase a-posteriori, são eliminados os clientes não comprometidos e utiliza-se do algoritmo MACS-RVND adaptado, o qual foi denominado MACS-RVND-A (Algoritmo 20).

---

#### Algoritmo 20: MACS-RVND-A

---

9. **Procedimento** MACS-RVND-A()
  10. #quantidade\_veiculo( $s^*$ ) retorna o número de veículos da solução  $s^*$
  11.  $s^* \leftarrow$  Encontre a melhor solução inicial aplicando os métodos PFIH e NN
  12.  $nimax$  = número máximo de iterações para ACS-VEI e ACS-TIME
  13. **Enquanto** não se cumpre a condição de parada **faça**
  14.      $v \leftarrow$  #quantidade\_veiculos( $s^*$ )
  15.      $s^* \leftarrow$  Execute ACS-VEI( $v - 1$ )
  16.      $s^* \leftarrow$  Execute ACS-TIME( $v$ )
  17.     // Após  $nimax$  iterações, permite obter soluções com um veículo adicional
  18.     **Se**  $ni = nimax$  **então**
  19.          $s^* \leftarrow$  Execute ACS-CLI( $v+1$ )
  20.     **Fim se**
  21.      $ni = ni + 1$
  22. **Fim enquanto**
-

O algoritmo MACS-RVND-A é uma extensão do algoritmo MACS-RVND (item 3.4), onde a principal diferença consiste no uso de uma terceira colônia de formigas (ACS-CLI) com o objetivo de minimizar o número de clientes não atendidos. Este novo objetivo possui prioridade sobre os outros dois objetivos, resultando na seguinte hierarquia de objetivos: minimizar o número de clientes não atendidos (ACS-CLI), minimizar o número de veículos (ACS-VEI) e minimizar a distância total (ACS-TIME).

O algoritmo ACS-CLI é idêntico ao algoritmo ACS-VEI, no entanto, ele possui suas próprias variáveis IN, denominadas IN\_CLI, e que são utilizadas para contabilizar o número de vezes que os clientes  $i$  não foram inseridos nas soluções obtidas pelo ACS-CLI. Ao contrário do ACS-VEI, no ACS-CLI são construídas soluções com um veículo a mais do que o número de veículos da melhor solução atual. Isto possibilita a inclusão de novos clientes na solução atual.

Outra diferença consiste na adaptação do algoritmo construtivo SimulaFormiga (Algoritmo 4), para resolver os problemas da fase a-posteriori, conforme apresentado na seção 4.1.2.

Como visto anteriormente, o algoritmo de colônia de formigas utiliza a concentração de feromônio depositado pelas formigas para indicar as boas soluções do problema. No caso do algoritmo MACS-RVND, são utilizadas duas matrizes que armazenam os feromônios resultantes dos algoritmos ACS-TIME e ACS-VEI. No algoritmo MACS-RVND-A estas matrizes são reaproveitadas para resolver os problemas da fase a-posteriori. Uma vez que os problemas são potencialmente muito semelhantes, esta operação impede que a otimização seja reiniciada toda vez a partir do zero e contribui muito para garantir o bom desempenho do algoritmo necessário no problema dinâmico. No entanto, ao serem inseridos novos clientes dinâmicos no problema, torna-se necessária uma redução parcial da concentração atual de feromônios na matriz com objetivo de tornar atrativos os arcos com destino aos novos clientes. Esta abordagem foi denominada estratégia de conservação de feromônios.

#### 4.1.5.5.1 Estratégia de conservação de feromônios

Esta estratégia de transferência da informação dos feromônios é inspirada no trabalho de Guntch e Middendorf (2001) e Oliveira (2009), no qual definem um novo parâmetro  $\gamma_r$  que regula a conservação de feromônios. Para todo par de clientes que aparecem tanto no antigo quanto no novo problema, a concentração de feromônios do o arco correspondente é inicializada pela expressão abaixo.

$$\tau_{ij} = (1 - \gamma_r) \tau_{ij}^{old} + \gamma_r \tau_0 \quad (26)$$

Onde  $\tau_{ij}^{old}$  representa o valor do  $\tau_{ij}$  referente ao antigo problema estático.

Sendo assim, a matriz de feromônios não é reiniciada toda vez, mas é reaproveitada para obter as próximas soluções. Para cada novo cliente é criada sua entrada correspondente na matriz de feromônios com a concentração de feromônio inicial  $\tau_0$ . Para definição do valor do parâmetro  $\gamma_r$  é importante considerar que as informações antigas possuirão menor importância para as novas soluções, para que as formigas possam explorar novos caminhos.

## 4.2 O algoritmo MACS-RVND-Periódico

O algoritmo MACS-RVND-Periódico utiliza a abordagem de re-otimização periódica e é classificado como um método de re-roteirização. Ele foi baseado no modelo de roteirização periódica proposto por Kilby et al. (1998) e que também foi estudado por Montemanni et al. (2002). Este modelo consiste no conceito de dividir o dia de trabalho em faixas de tempo de tamanhos iguais. Portanto, para um dia de trabalho que possui um total  $JT$  segundos é dividido em  $m$  faixas de tempo, cada uma de tamanho  $f$  segundos, com  $f=JT/m$ .

Todos os novos clientes recebidos antes de um dado tempo limite  $tco$  (*time of cutoff*) são aceitos e processados durante o dia de trabalho. Os clientes recebidos depois do tempo limite  $tco$  não são atendidos no dia atual e são atribuídos ao próximo dia de trabalho. Por exemplo, se o um dia de trabalho inicia às 7 horas e termina às 17 horas (total de 10 horas de trabalho) e a política da empresa consiste em aceitar novos clientes às 12h, então o valor de  $tco$  será 18.000 segundos (5 horas).



No algoritmo MACS-RVND-Periódico o problema de roteirização da fase a-priori é resolvido utilizando o algoritmo MACS-RVND. Na fase a-posteriori são resolvidos os problemas de cada faixa de tempo utilizando o algoritmo MACS-RVND-A. Em ambas as fases, após ser encontrada uma solução, atribuem-se aos veículos aqueles clientes que possuem o tempo de processamento  $tp_i$  dentro da faixa de tempo, ou seja, os clientes que possuem o horário de término do atendimento maior que  $f.(n-1)$  e menor que  $f.(n)$ . O estado dos clientes atribuídos  $e_i$  é atualizado para o estado *atribuído* e uma solução parcial é obtida.

Um problema que surge desta consideração anterior é que ao atender o último cliente, o veículo deverá aguardar o recebimento de seus próximos destinos. Para contornar este problema, Kilby et al. (1998) propuseram uma estratégia denominada tempo de compromisso avançado *tac*, a qual consiste em comprometer os clientes com tempo de processamento iniciando nos próximos  $f.n+tac$  segundos, conforme ilustra a Figura 6. Segundo o autor, esta estratégia permite que seja comprometido ao menos o próximo destino do veículo.

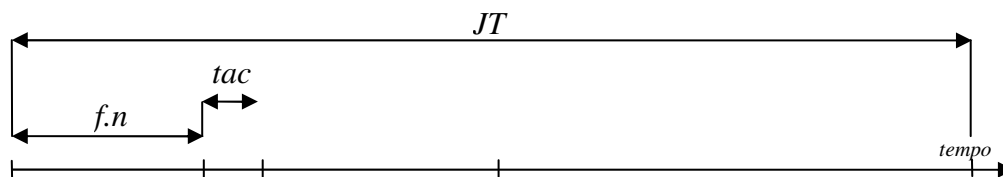


Figura 6 - Conceito da divisão do dia em faixas de tempo

Uma exceção desta estratégia é o retorno do veículo ao depósito, que ocorre apenas em duas condições, quando a capacidade do veículo  $K$  foi totalmente utilizada ou quando o término da jornada de trabalho  $b_0$  está próximo do fim, conforme a expressão abaixo. A variável  $p$  representa o número de clientes já atendidos pelo veículo  $v$ .

$$retorna\_depósito(v) = \begin{cases} 1, se & K = \sum_{i=1}^p d_i \text{ ou } tempo + c_{p0} \geq b_0 \\ 0, caso \text{ contrário} \end{cases} \quad (27)$$

Quando nenhuma destas condições ocorrerem e o veículo não conhecer seu próximo destino, então ele deverá aguardar na posição atual por novas instruções.

Isto ocorre porque as rotas podem ser replanejadas e novos clientes podem ser atribuídos aos veículos.

A Figura 7 ilustra as rotas de dois veículos obtidas na fase a-priori ( $n=0$ ) e em duas faixas de tempo da fase a-posteriori, com  $n=1$ . Os quadrados representam os depósitos. Os círculos e losangos representam o tempo de processamento  $tp_i$  dos clientes estáticos e dinâmicos, respectivamente. O tempo de processamento é utilizado para definir o momento que um cliente é atribuído a um veículo. Após ser resolvido o primeiro problema estático, com  $n=0$ , os clientes com  $tp_i$  menor do que  $tac$  são atribuídos aos veículos que iniciam suas rotas. Quando não são atribuídos clientes a um veículo, como ocorreu no caso da rota 1, o veículo deverá aguardar por instruções na sua posição atual.

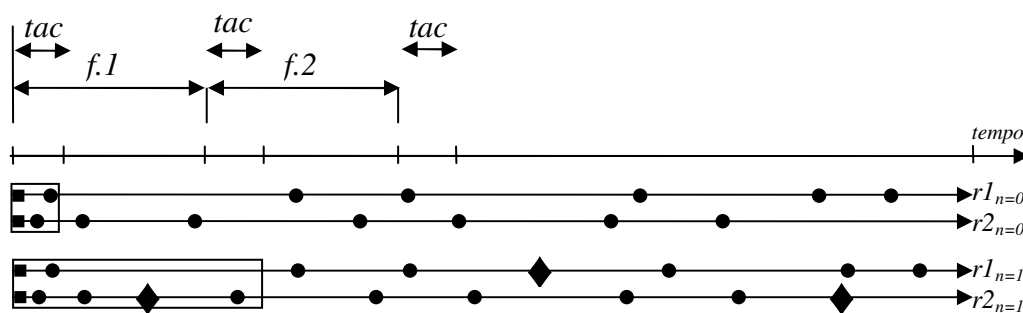


Figura 7 - Funcionamento da estratégia de divisão do dia em faixas de tempo

Com o passar do tempo, durante a primeira faixa de tempo, novos clientes são conhecidos. No fim da primeira faixa de tempo, com  $n=1$ , são atualizadas as informações sobre a posição atual dos veículos e capacidades atuais. Um novo problema estático com os clientes não atendidos e com os novos clientes é resolvido. Os clientes com tempo de processamento  $f.0 \leq tp_i \leq f.1 + tca$  são atribuídos aos veículos e as rotas são iniciadas a partir dos últimos clientes atendidos ou comprometidos pelos veículos. Comparando com as rotas em  $n=0$ , nota-se que foi adicionado um cliente dinâmico na rota  $r1$  e dois clientes dinâmicos na rota  $r2$ . Este processo continua até que seja finalizada a jornada de trabalho e todos os veículos tenham retornado ao depósito.

#### 4.2.1.1 Modificação Proposta

Dado o modelo proposto por Kilby et al. (1998), após resolver um VRPTW estático, devem ser comprometidos apenas os clientes que possuem o tempo de processamento dentro da faixa de tempo. Caso não seja atribuído nenhum cliente ao veículo, então o mesmo deverá aguardar em sua posição atual. A partir da Figura 8 é possível analisar diferentes situações que devem ser tratadas.

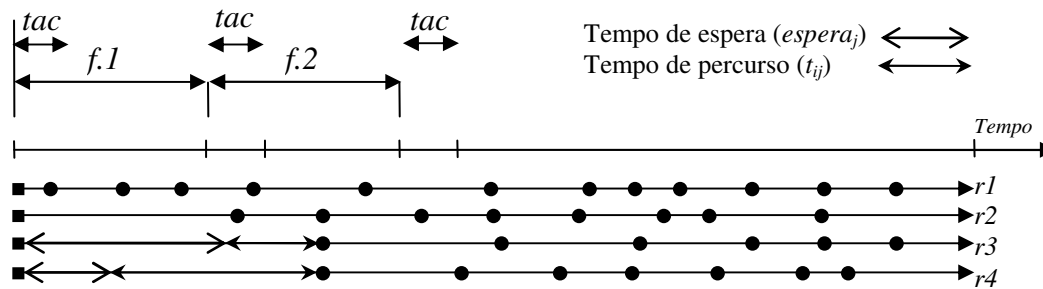


Figura 8 - Exemplos de diferentes situações

- Rota  $r1$ : na primeira faixa de tempo são atribuídos quatro clientes, onde um deles foi comprometido com o uso do  $tac$ . O veículo parte do depósito no início da jornada de trabalho.
- Rota  $r2$ : na primeira faixa de tempo foi atribuído apenas um cliente, obtido com o uso do  $tac$ . O veículo parte do depósito no início da jornada de trabalho.
- Rota  $r3$ : não são atribuídos clientes para a primeira faixa. O veículo deve aguardar na sua posição atual, neste caso no depósito. Para a segunda faixa de tempo é atribuído um cliente. Nota-se que o tempo de espera para atender o primeiro cliente desta rota é maior do que o tamanho da faixa de tempo. Desta forma, o veículo inicia sua rota e consegue chegar no mesmo horário definido na primeira faixa de tempo.
- Rota  $r4$ : não são atribuídos clientes para a primeira faixa e o veículo aguarda na sua posição atual. Para a segunda faixa de tempo é atribuído um cliente. O veículo inicia sua rota, mas devido ao tempo de percurso até seu primeiro destino, não consegue chegar no mesmo horário que foi definido na primeira faixa de tempo. Se as janelas de tempo desta rota já estão justas e não possui tempos de esperas, acarretará na inviabilidade da solução inicial já otimizada, aumentando o custo da solução e o número de veículos.

A modificação proposta consiste em deixar de utilizar o *tac* e utilizar a seguinte estratégia de compromisso avançado. A expressão (28) retorna 1 se o cliente  $i$  deve ser atribuído ao veículo de sua rota na faixa de tempo  $n$  se o estado  $e_i$  indica que o cliente ainda não foi comprometido e se início da viagem em direção ao cliente  $i$  é menor do que o índice  $n$  da faixa de tempo multiplicada pelo tamanho  $f$ . Caso contrário, o cliente não será atribuído.

$$atribui(i, n) = \begin{cases} 1, & \text{se } e_i \notin \{em\_tr\grave{a}nsito, entregue, atribuído\} \wedge T_i - c_{i-1,i} \geq n.f \\ 0, & \text{caso contrário} \end{cases} \quad (28)$$

### 4.3 Resultados Computacionais

O objetivo desta seção é mostrar o desempenho do algoritmo proposto para o DVRPTW quanto à qualidade da solução. Os algoritmos foram implementados utilizando o *framework* proposto para problemas de roteirização e executado numa máquina com as mesmas configurações citadas na seção 3.5.

Os algoritmos foram testados com as instâncias propostas por Lackner (2004), que são extensões das instâncias de Solomon (1987) para o DVRPTW. Estas extensões utilizaram os graus de dinamismo iguais a 10, 30, 50, 70 e 90. Para cada instância e para cada grau de dinamismo, o autor criou um novo parâmetro que indica o momento em que o cliente é conhecido ( $ta_i$ ), totalizando 280 instâncias. Nas versões adaptadas dos algoritmos PFIH, VND e MACS-RVND para resolver os problemas da fase a-posteriori foram utilizados os mesmo valores para os parâmetros dos algoritmos originais. O parâmetro de conservação de feromônios foi definido como  $\gamma_r=0,3$  para os algoritmos MACS-RVND\_Reroteiriza e MACS-RVND\_Períodico, conforme proposto por Guntsch e Middendorf (2001).

As tabelas Tabela 7 e Tabela 8 apresentam os valores médios por classe e por grau de dinamismo das melhores soluções obtidas pelos algoritmos propostos para o DVRPTW. A primeira coluna indica o grupo de instâncias, que é combinado com os graus de dinamismo  $\delta$  da segunda coluna. As colunas NV, DT e NA representam os valores médios do número de veículos, distância média total e percentual de clientes não atendidos para cada grupo de instâncias. A última linha TM representa a soma dos totais médios obtidos em cada um dos algoritmos e considerando os cinco graus de dinamismo.

Tabela 7 – Comparação dos resultados dos algoritmos determinísticos propostos para o DVRPTW

Grupo	$\delta$	PFIH_PFIH			PFIH_PFIH-VND			PFIH-VND_PFIH-VND		
		NV	DT	NA	NV	DT	NA	NV	DT	NA
C1	10	12,33	2.264,16	0,00	11,00	993,53	0,00	10,56	933,18	0,00
	30	12,89	2.600,51	0,00	10,89	959,62	0,11	10,67	961,56	0,00
	50	13,56	2.608,25	0,00	10,78	1.015,44	0,00	11,11	993,46	0,00
	70	13,44	2.565,63	0,00	11,00	1.016,48	0,00	10,89	1.021,57	0,00
	90	13,22	2.491,04	0,00	10,56	1.002,95	0,00	10,56	977,87	0,00
C2	10	4,00	1.632,69	0,00	3,88	679,45	0,00	3,13	616,80	0,00
	30	4,25	1.955,03	0,00	3,25	651,03	0,00	3,38	614,19	0,00
	50	4,25	1.998,55	0,00	3,38	649,28	0,00	3,50	644,43	0,00
	70	4,25	2.086,51	0,00	3,25	626,72	0,00	3,38	624,60	0,00
	90	4,25	1.764,57	0,00	3,25	636,85	0,00	3,25	630,14	0,00
R1	10	17,55	2.172,21	0,09	14,64	1.298,54	0,09	13,91	1.275,57	0,09
	30	18,18	2.241,16	0,55	14,64	1.310,28	0,45	14,55	1.303,59	0,64
	50	18,45	2.303,89	0,45	14,64	1.315,83	0,73	14,55	1.306,27	0,82
	70	19,27	2.416,82	0,91	14,27	1.314,38	1,27	14,18	1.309,90	1,36
	90	19,55	2.441,90	1,18	14,09	1.296,48	1,55	14,18	1.298,04	1,73
R2	10	4,18	1.783,33	0,00	3,82	1.053,88	0,00	3,36	1.025,34	0,00
	30	4,45	1.988,77	0,00	3,73	1.033,23	0,00	3,64	1.023,16	0,00
	50	4,45	1.966,04	0,00	3,64	1.053,75	0,00	3,55	1.039,34	0,00
	70	4,64	2.044,29	0,00	4,00	1.055,00	0,00	3,64	1.068,37	0,00
	90	4,82	2.021,03	0,00	3,64	1.073,34	0,00	3,73	1.057,31	0,00
RC1	10	18,50	2.609,40	0,25	14,88	1.497,62	0,25	14,13	1.466,53	0,25
	30	18,38	2.620,53	0,38	14,25	1.507,03	0,88	14,63	1.494,37	0,00
	50	19,13	2.705,25	0,50	14,00	1.498,53	0,88	14,88	1.554,42	0,50
	70	20,63	3.050,07	0,50	14,38	1.514,10	0,75	14,63	1.550,17	0,63
	90	20,00	2.893,55	0,75	14,25	1.548,00	1,13	14,25	1.554,05	0,38
RC2	10	5,13	2.304,74	0,00	4,75	1.175,75	0,00	4,00	1.202,04	0,00
	30	5,25	2.599,27	0,00	4,50	1.188,84	0,00	4,25	1.197,43	0,00
	50	6,00	2.746,79	0,00	4,38	1.212,21	0,00	4,13	1.223,40	0,00
	70	6,25	2.900,69	0,00	4,38	1.232,08	0,00	4,13	1.266,23	0,00
	90	5,63	2.795,29	0,00	4,50	1.243,96	0,00	4,63	1.217,16	0,00
<b>TM</b>		<b>327</b>	<b>70.572</b>	<b>5,56</b>	<b>257</b>	<b>33.654</b>	<b>8,08</b>	<b>253</b>	<b>33.450</b>	<b>6,39</b>

A Tabela 7 apresenta os algoritmos baseados nos algoritmos PFIH e VND. Observa-se o algoritmo PFIH\_PFIH obteve soluções com o menor número médio de clientes não atendidos, porém, foram necessários mais veículos. A distância total média ficou acima do dobro da distância total média obtida pelos outros algoritmos propostos. O algoritmo PFIH\_PFIH-VND obteve boas soluções, no entanto, o algoritmo PFIH-VND\_PFIH-VND, que se utiliza do algoritmo VND nas fases a-priori e a-posteriori, apresentou os menores valores médios para o número de veículos, distância total e número de clientes não atendidos.

A Tabela 8 apresenta os resultados dos algoritmos que foram baseados nos algoritmos MACS-RVND, PFIH e VND. Observa-se o algoritmo MACS-RVND\_PFIH-VND obteve boas soluções, com o menor número médio de veículos e com a menor distância total média. No entanto, o número médio de clientes não atendidos foi maior do que os valores obtidos pela maior parte dos

algoritmos propostos. O algoritmo MACS-RVND\_Re-roteiriza obteve soluções muito boas, com o menor número médio de clientes não atendidos. O número médio de veículos e a distância total média ficaram bem próximos dos valores obtidos pelos outros algoritmos propostos. Finalmente, o algoritmo MACS-RVND\_Períodico apresentou resultados razoáveis, porém, abaixo dos resultados apresentados pelo MACS-RVND\_Re-roteiriza.

Tabela 8 – Comparação dos resultados dos algoritmos probabilísticos propostos para o DVRPTW

Grupo	MACS-RVND_PFIH-VND				MACS-RVND_Re-roteiriza			MACS-RVND-Períodico		
	$\delta$	NV	DT	NA	NV	DT	NA	NV	DT	NA
C1	10	10,67	881,34	0,00	10,67	891,17	0,00	10,67	930,69	0,00
	30	10,89	940,32	0,00	10,78	919,83	0,00	10,89	984,03	0,00
	50	10,78	944,06	0,00	10,89	958,85	0,00	11,22	1.079,62	0,00
	70	10,67	963,60	0,00	10,78	998,75	0,00	11,22	1.093,60	0,00
	90	10,67	1.008,43	0,00	10,89	1.012,84	0,00	11,00	1.087,12	0,00
C2	10	3,13	597,76	0,00	3,13	598,85	0,00	3,13	600,92	0,00
	30	3,38	606,09	0,00	3,13	605,39	0,00	3,38	613,75	0,00
	50	3,38	616,71	0,00	3,38	628,88	0,00	3,63	661,96	0,00
	70	3,38	625,96	0,00	3,38	621,16	0,00	3,50	654,36	0,00
	90	3,25	630,14	0,00	3,25	637,45	0,00	3,63	709,77	0,00
R1	10	14,00	1.297,92	0,18	14,00	1.276,10	0,18	14,00	1.305,17	0,18
	30	14,27	1.262,53	0,73	14,00	1.273,95	0,27	14,64	1.333,40	0,36
	50	14,36	1.285,75	1,09	14,55	1.299,77	0,64	14,82	1.375,85	0,55
	70	14,09	1.292,54	1,55	14,09	1.318,73	1,00	14,45	1.388,91	1,82
	90	14,36	1.305,85	1,45	14,55	1.349,23	1,09	14,82	1.436,93	1,09
R2	10	3,45	986,91	0,00	3,45	973,98	0,00	3,45	1.019,90	0,00
	30	3,64	993,52	0,00	3,73	985,30	0,00	3,55	1.026,22	0,00
	50	3,82	1.018,85	0,00	3,64	1.021,71	0,00	3,55	1.070,02	0,00
	70	3,73	1.036,57	0,00	4,00	1.023,12	0,00	3,73	1.089,26	0,00
	90	3,73	1.061,42	0,00	3,82	1.021,69	0,00	3,5	1.103,91	0,00
RC1	10	13,88	1.442,25	0,50	14,13	1.438,97	0,25	14,50	1.489,95	0,00
	30	13,88	1.439,34	0,88	14,13	1.480,79	0,25	14,25	1.531,75	0,50
	50	14,38	1.475,77	0,63	14,13	1.511,98	0,50	15,38	1.619,36	0,50
	70	13,75	1.471,87	1,00	13,88	1.507,0	0,63	15,00	1.633,76	0,25
	90	14,00	1.538,07	0,63	13,88	1.555,77	1,13	14,88	1.652,40	1,00
RC2	10	4,13	1.122,48	0,00	4,38	1.122,49	0,00	3,88	1.185,86	0,00
	30	4,25	1.159,00	0,00	4,38	1.151,20	0,00	4,25	1.217,08	0,00
	50	4,38	1.188,85	0,00	4,50	1.170,26	0,00	4,50	1.253,34	0,00
	70	4,50	1.254,64	0,00	4,38	1.194,84	0,00	4,38	1.283,99	0,00
	90	4,38	1.224,47	0,00	4,63	1.212,11	0,00	4,13	1.344,19	0,00
<b>TM</b>		<b>251</b>	<b>32.673</b>	<b>8,63</b>	<b>252</b>	<b>32.762</b>	<b>5,93</b>	<b>258</b>	<b>34.777</b>	<b>6,25</b>

Nos algoritmos PFIH\_PFIH, PFIH\_PFIH-VND e PFIH-VND\_PFIH-VND o tempo de execução na fase a-priori não ultrapassa de 10 segundos. Na fase a-posteriori, o tempo de execução não ultrapassa dos 2 segundos. Por serem algoritmos determinísticos, que retornam sempre a mesma solução, cada um deles foi executado apenas uma vez para cada instância. Já para os algoritmos MACS-RVND\_PFIH-VND, MACS-RVND\_Re-roteiriza e MACS-RVND-Períodico, o critério de parada para a solução da fase a-priori foi definido como o tempo

máximo de execução de 5 minutos. Na fase a-posteriori, cada subproblema é resolvido com o tempo máximo de execução de 10 segundos. Por serem algoritmos probabilísticos, que podem retornar diferentes soluções, foram realizadas cinco execuções para cada instância e foi obtida a melhor solução.

Na Tabela 9 são apresentados os tempos de execução dos algoritmos propostos. Foram utilizadas 280 instâncias, com diferentes graus de dinamismo. A coluna CPU indica a soma do tempo total das execuções para todas as instâncias por classe e por grau de dinamismo, em segundos.

Tabela 9 – Tempos de execução dos algoritmos propostos para o DVRPTW

Grupo	$\delta$	PFIH	PFIH_PFIH	PFIH-VND	MACS-RVND	MACS-RVND	MACS-RVND
		PFIH	VND	PFIH-VND	PFIH-VND	Re-roteiriza	Periódico
		CPU (s)	CPU (s)	CPU (s)	CPU (s)	CPU (s)	CPU (s)
C1	10	0,2	1,2	2,0	95,5	421,3	294,3
	30	0,2	1,6	2,1	80,7	829,0	418,6
	50	0,2	1,9	2,1	39,9	1.125,8	383,0
	70	0,2	1,8	1,8	19,5	998,8	363,7
	90	0,2	1,6	1,6	2,8	986,5	388,2
C2	10	0,5	1,8	3,3	240,8	574,5	441,1
	30	0,3	1,8	2,8	82,1	762,3	376,8
	50	0,3	2,1	2,5	67,4	816,4	463,3
	70	0,3	1,8	2,8	82,1	762,3	376,8
	90	0,3	1,6	1,5	2,4	701,3	317,1
R1	10	0,2	3,0	6,0	86,1	1.172,3	714,4
	30	0,2	4,3	5,6	82,0	1.896,9	842,9
	50	0,2	4,2	4,7	46,3	1.830,0	812,0
	70	0,2	3,6	3,9	17,0	1.402,4	714,9
	90	0,2	3,3	3,3	4,8	1.310,7	690,1
R2	10	0,2	3,0	6,0	86,1	1.172,3	714,4
	30	0,2	4,5	6,1	150,3	1.673,8	898,0
	50	0,2	4,8	5,0	74,8	1.926,7	697,1
	70	0,3	4,7	4,7	24,1	1.844,0	737,7
	90	0,3	3,7	3,8	5,3	1.556,7	616,2
RC1	10	0,1	1,7	3,0	32,2	556,4	414,8
	30	0,1	2,3	3,1	36,7	1.054,6	546,5
	50	0,1	2,4	2,6	19,8	1.050,8	432,0
	70	0,1	2,3	2,3	8,0	993,0	427,7
	90	0,1	1,9	1,9	2,9	871,0	378,6
RC2	10	0,2	2,2	5,0	119,8	909,6	666,1
	30	0,2	3,6	4,8	84,9	1.613,3	711,3
	50	0,2	3,5	3,9	43,2	1.624,5	710,8
	70	0,2	3,3	3,4	15,5	1.443,1	627,6
	90	0,2	2,8	2,8	3,7	1.260,6	670,9
<b>T</b>		<b>6,2</b>	<b>82,2</b>	<b>104,5</b>	<b>1.657,1</b>	<b>35.140,9</b>	<b>16.846,9</b>

Avaliando os seis algoritmos propostos e considerando a hierarquia dos objetivos de minimizar o NA, NV e DT, conclui-se que o algoritmo que apresentou os melhores resultados é o MACS-RVND\_Re-roteiriza. Realizou-se uma comparação deste algoritmo com outras abordagens propostas na literatura.

Tabela 10 – Comparação dos resultados do MACS-RVND\_Re-roteiriza com as abordagens propostas na literatura para o DVRPTW

Grupo	MACS-RVND_Re-roteiriza			Pillac et al., 2012		Hong, 2012		Lackner, 2004	
	$\delta$	DT	NA	DT	NA	DT	NA	DT	NA
C1	10	891,17	0,00	850,60	0,11	895,80	0,22	996,40	0,00
	30	919,83	0,00	874,90	0,11	962,10	0,33	1.066,90	0,00
	50	958,85	0,00	903,40	0,11	1.001,20	0,22	1.236,10	0,00
	70	998,75	0,00	919,10	0,11	1.031,70	0,22	1.261,30	0,00
	90	1.012,84	0,00	929,90	0,11	1.039,80	0,22	1.479,60	0,00
C2	10	598,85	0,00	597,20	0,00	594,70	0,00	629,10	0,00
	30	605,39	0,00	597,60	0,00	651,40	0,00	632,30	0,04
	50	628,88	0,00	604,00	0,00	605,00	0,00	689,30	0,13
	70	621,16	0,00	619,20	0,00	636,50	0,00	743,80	0,21
	90	637,45	0,00	625,70	0,00	636,80	0,00	792,50	0,29
R1	10	1.276,10	0,18	1.197,40	0,25	1.257,10	0,17	1.278,10	0,47
	30	1.273,95	0,27	1.212,90	0,80	1.286,60	0,58	1.337,90	0,72
	50	1.299,77	0,64	1.225,00	1,25	1.295,80	0,67	1.330,00	0,78
	70	1.318,73	1,00	1.237,30	1,71	1.331,30	1,75	1.336,10	0,94
	90	1.349,23	1,09	1.230,10	2,55	1.335,90	2,33	1.278,30	0,75
R2	10	973,98	0,00	893,00	0,00	950,00	0,09	1.052,90	0,03
	30	985,30	0,00	915,60	0,00	985,50	0,00	1.085,40	0,15
	50	1.021,71	0,00	948,60	0,00	1.016,50	0,00	1.138,80	0,21
	70	1.023,12	0,00	967,70	0,00	1.032,00	0,09	1.116,90	0,30
	90	1.021,69	0,00	981,70	0,00	1.047,80	0,09	1.193,30	0,52
RC1	10	1.438,97	0,25	1.389,40	0,04	1.436,20	1,13	1.426,90	0,46
	30	1.480,79	0,25	1.421,50	0,28	1.492,20	1,13	1.439,70	0,42
	50	1.511,98	0,50	1.463,40	0,23	1.514,70	1,38	1.448,10	0,46
	70	1.507,05	0,63	1.470,10	0,58	1.511,30	1,88	1.488,40	0,58
	90	1.555,77	1,13	1.495,50	0,51	1.513,90	2,00	1.475,20	0,42
RC2	10	1.122,49	0,00	1.024,40	0,00	1.103,30	0,00	1.220,90	0,00
	30	1.151,20	0,00	1.053,10	0,00	1.166,00	0,25	1.244,90	0,04
	50	1.170,26	0,00	1.060,50	0,00	1.190,50	0,13	1.244,90	0,00
	70	1.194,84	0,00	1.091,40	0,00	1.239,50	0,00	1.269,30	0,00
	90	1.212,11	0,00	1.130,30	0,00	1.257,20	0,13	1.346,80	0,13
TM		32.762,21	5,93	30.930,50	8,75	33.018,30	15,01	35.280,10	8,05
GAP (%)				5,92	<b>-32,21</b>	<b>-0,78</b>	<b>-60,48</b>	<b>-7,14</b>	<b>-26,31</b>

Tabela 10 apresenta os melhores resultados obtidos pelo algoritmo MACS-RVND\_Re-roteiriza e pelo estado da arte da literatura para o DVRPTW, que até o presente momento é composto pelos trabalhos de Lackner (2004), Hong (2012), Pillac et al. (2012). Estes autores não publicaram a solução obtida para cada uma das 280 instâncias. Foram publicados apenas os valores médios por classe de instância e grau de dinamismo. Com base nestes valores, foram comparados os resultados com o algoritmo MACS-RVND\_Re-roteiriza. Observa-se o algoritmo proposto obteve melhores soluções em termos de distância total média do que as abordagens propostas Lackner (2004) e Hong (2012). Quanto ao percentual total médio de clientes não atendidos, o algoritmo proposto apresentou melhores resultados do que Pillac et al., 2012, Hong, 2012 e Lackner, 2004.



#### 4.4 Considerações Finais

Neste capítulo foi proposto um algoritmo genérico para resolver o Problema Dinâmico de Roteirização de Veículos (DVRPTW). Com base neste algoritmo, foram propostos seis algoritmos que se utilizam do algoritmo de Inserção (PFIH), do método de Descida em Vizinhança Variável (VND) e do Sistema de Múltiplas Colônias de Formigas com Descida em Vizinhança Variável Aleatória (MACS-RVND). Através da aplicação do algoritmo a 280 instâncias usadas como *benchmarking* na literatura é possível concluir que o algoritmo foi capaz de encontrar soluções competitivas com o estado da arte.