

**ALGORITMO PARA PROGRAMAÇÃO  
INTEGRADA DE VEÍCULOS E TRIPULAÇÕES  
NO SISTEMA DE TRANSPORTE PÚBLICO POR  
ÔNIBUS**



EMILIANA MARA LOPES SIMÕES  
ORIENTADOR: GERALDO ROBSON MATEUS  
CO-ORIENTADOR: MARCONE JAMILSON FREITAS SOUZA

ALGORITMO PARA PROGRAMAÇÃO  
INTEGRADA DE VEÍCULOS E TRIPULAÇÕES  
NO SISTEMA DE TRANSPORTE PÚBLICO POR  
ÔNIBUS

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Minas Gerais como requisito parcial para a obtenção do grau de Mestre em Ciência da Computação.

Belo Horizonte

Agosto de 2009

© 2009, Emilianara Mara Lopes Simões.  
Todos os direitos reservados.

D1234p Lopes Simões, Emilianara Mara  
ALGORITMO PARA PROGRAMAÇÃO  
INTEGRADA DE VEÍCULOS E TRIPULAÇÕES  
NO SISTEMA DE TRANSPORTE PÚBLICO POR  
ÔNIBUS / Emilianara Mara Lopes Simões. — Belo  
Horizonte, 2009  
xviii, 82 f. : il. ; 29cm  
Dissertação (mestrado) — Universidade Federal  
de Minas Gerais  
Orientador: Geraldo Robson Mateus  
Co-orientador: Marcane Jamilson Freitas Souza  
1. Programação Integrada de Veículos e  
Tripulações. 2. Metaheurísticas. 3. Busca Local  
Iterada. I. Título.

CDU 519.6\*82.10

# [Folha de Aprovação]

Quando a secretaria do Curso fornecer esta folha,  
ela deve ser digitalizada e armazenada no disco em formato gráfico.

Se você estiver usando o `pdflatex`,  
armazene o arquivo preferencialmente em formato PNG  
(o formato JPEG é pior neste caso).

Se você estiver usando o `latex` (não o `pdflatex`),  
terá que converter o arquivo gráfico para o formato EPS.

Em seguida, acrescente a opção `approval={nome do arquivo}`  
ao comando `\ppgccufmg`.



*Dedico este trabalho aos meus inesquecíveis avós, Vô Lopes e Vó Mms, dos quais sempre recebi um amor incondicional, acalentador e encorajador.*



*“Tudo é do pai!  
Toda honra e toda glória,  
é dele a vitória alcançada em minha vida.”*  
(Frederico Cruz)



# Resumo

Este trabalho aborda dois dos problemas envolvidos no processo de planejamento do Sistema de Transporte Público de Ônibus Urbano: o Problema de Programação de Veículos (PPV) e o Problema de Programação de Tripulações (PPT). O PPV consiste em criar uma rotina diária de operação para uma frota de veículos, dado um conjunto de viagens a serem obrigatoriamente realizadas. Na resolução do PPT são definidas as jornadas diárias de trabalho para motoristas e cobradores, de maneira que, todos os deslocamentos dos veículos estejam sob responsabilidade de alguma tripulação. O objetivo é reduzir os custos envolvidos em tais atividades, fazendo o melhor aproveitamento da frota e da mão-de-obra, mas ao mesmo tempo obedecendo à legislação trabalhista e as regras operacionais vigentes.

Normalmente, devido à complexidade destes problemas, eles são resolvidos separadamente e seqüencialmente (abordagem seqüencial tradicional, ou seja, primeiro resolve-se PPV e logo em seguida o PPT). Porém, dada a forte dependência entre PPV e PPT, espera-se que a resolução integrada dos mesmos possa proporcionar considerável redução dos custos. Assim, dadas as expectativas atuais acerca dessa nova metodologia, neste trabalho é desenvolvida uma heurística, baseada na metaheurística Busca Local Iterada, para resolução do Problema de Programação Integrada de Veículos e Tripulações (PPVT). No PPVT as rotinas de operação dos veículos e as jornadas de trabalho das tripulações são definidas simultaneamente. Trata-se de um problema altamente combinatorial e que se enquadra na classe de problemas NP-difícil, não havendo, portanto, algoritmos em tempo polinomial para encontrar soluções ótimas.

Com o intuito de analisar o desempenho da abordagem integrada, PPV e PPT também são resolvidos de forma iterativa, seqüencial tradicional e independente (PPV e PPT são resolvidos separadamente, em qualquer ordem e de forma a não se relacionarem). Para validar os métodos propostos são apresentados experimentos com dados reais de uma empresa de transporte público da cidade de Belo Horizonte. Os resultados encontrados confirmam as expectativas e mostram que as soluções geradas pela heurística integrada são melhores do que as obtidas de forma seqüencial.



# Abstract

This work deals with two of the problems involved in the planning process of the public mass transit: the Vehicle Scheduling Problem (VSP) and the Crew Scheduling Problem (CSP). The VSP is a way to create a daily operation routine for a fleet of vehicles, given a set of trips required to be performed. In the resolution of CSP the daily work duty for the drivers and collectors are defined, so that all displacements of the vehicles are in charge of a crew. This paper aimed to investigate how to reduce costs involved in such activities for the best use of the fleet and labor force, but also comply with the public policies of labor laws and operational rules.

These problems has been usually treated separately and sequentially due to their complexity (traditional sequential approach, in other words, first VSP is solved and then CSP). However, since the strong dependence between VSP and CSP, it is expected that the integrated resolution of them could provide considerable costs reduction. Thus, given the current expectations about this new methodology, in this work heuristics are developed, based on the Iterated Local Search metaheuristic to solve the Integrated Vehicle and Crew Scheduling Programing (VCSP). In the VCSP the routines of vehicles operation and the crews work duty are defined simultaneously. This is a highly combinatorial problem, which is inserted in the class of NP-hard problems, not having so, polynomial-time algorithms to find optimal solutions.

To analyze the efficiency of the performance of the integrated approach, both VSP and CSP are solved in the traditional sequential and independent way (VSP and CSP are solved separately, in any order and in an unrelated way). The proposed methods can be validated with real data from a Public Transport Company of Belo Horizonte city. The results obtained signalize that the solutions generated by the integrated heuristics are better than those obtained in sequential order.



# Sumário

<b>1</b>	<b>Introdução</b>	<b>1</b>
1.1	Justificativa e Objetivos do Trabalho . . . . .	3
1.2	Organização do Texto . . . . .	4
<b>2</b>	<b>Revisão Bibliográfica</b>	<b>7</b>
<b>3</b>	<b>Programações Tradicional e Independente de Veículos e Tripulações</b>	<b>13</b>
3.1	Heurísticas . . . . .	14
3.1.1	Método de Descida . . . . .	14
3.1.2	Busca Local Iterada . . . . .	15
3.2	Descrição do Problema de Programação de Veículos . . . . .	16
3.3	Método de Solução do PPV . . . . .	19
3.3.1	Representação . . . . .	19
3.3.2	Estruturas de Vizinhança . . . . .	19
3.3.3	Função de Avaliação . . . . .	20
3.3.4	Procedimento de Geração da Solução Inicial . . . . .	22
3.3.5	Algoritmo de Resolução . . . . .	22
3.4	Descrição do Problema de Programação de Tripulações . . . . .	25
3.5	Método de Solução do PPT . . . . .	28
3.5.1	Representação . . . . .	29
3.5.2	Estruturas de Vizinhança . . . . .	29
3.5.3	Função de Avaliação . . . . .	31
3.5.4	Procedimento de Construção das Tarefas . . . . .	31
3.5.5	Procedimento de Geração da Solução Inicial . . . . .	32
3.5.6	Algoritmo de Resolução . . . . .	33
3.6	Programação Independente de Veículos e Tripulações . . . . .	35
<b>4</b>	<b>Programação Integrada de Veículos e Tripulações</b>	<b>37</b>
4.1	Descrição do Problema de Programação Integrada de Veículos e Tripulações . . . . .	38

4.2	Método de Solução Iterativa do PPV e PPT . . . . .	40
4.2.1	Representação . . . . .	40
4.2.2	Função de Avaliação . . . . .	41
4.2.3	Algoritmo de Resolução . . . . .	42
4.3	Método de Solução do PPVT . . . . .	44
4.3.1	Representação . . . . .	45
4.3.2	Estruturas de Vizinhaça . . . . .	49
4.3.3	Função de Avaliação . . . . .	53
4.3.4	Procedimento de Geração da Solução Inicial . . . . .	54
4.3.5	Algoritmo de Resolução . . . . .	54
<b>5</b>	<b>Experimentos Computacionais</b>	<b>59</b>
5.1	Descrição das Instâncias . . . . .	59
5.2	Resultados . . . . .	62
<b>6</b>	<b>Conclusões e Trabalhos Futuros</b>	<b>75</b>
	<b>Referências Bibliográficas</b>	<b>79</b>

# Lista de Figuras

1.1	Planejamento do sistema de transporte público. . . . .	1
3.1	Exemplo de uma solução $s^v$ para o PPV. . . . .	20
3.2	Estruturas de vizinhança do PPV. . . . .	21
3.3	Bloco de veículo particionado em tarefas. . . . .	27
3.4	Exemplo de uma solução $s^t$ para o PPT. . . . .	29
3.5	Estruturas de vizinhança do PPT. . . . .	30
4.1	Bloco de veículo particionado em tarefas. . . . .	41
4.2	Representação de um bloco de veículo. . . . .	41
4.3	Bloco de Veículo - 1 e o seu particionamento em tarefas. . . . .	46
4.4	Viagens expandidas obtidas do Bloco de Veículo - 1 da Figura 4.3. . . . .	47
4.5	Representação de uma solução para o PPVT com definição das viagens expandidas. . . . .	49
4.6	Solução para o PPVT utilizada para exemplificar as estruturas de vizinhança. . . . .	51
4.7	Exemplo do movimento realocação PPV. . . . .	51
4.8	Exemplo do movimento realocação PPT. . . . .	51
4.9	Exemplo do movimento super realocação. . . . .	52
4.10	Exemplo do movimento troca PPV. . . . .	52
4.11	Exemplo do movimento troca PPT. . . . .	52
4.12	Exemplo do movimento super troca. . . . .	53



# Lista de Tabelas

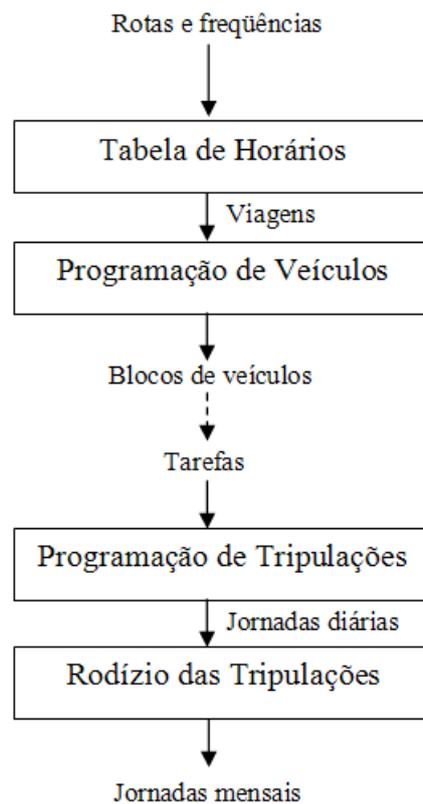
3.1	Tabela de horários. . . . .	17
5.1	Descrição das instâncias utilizadas nos experimentos. . . . .	60
5.2	Penalizações utilizadas no PPV e seus respectivos valores. . . . .	61
5.3	Penalizações utilizadas no PPT e seus respectivos valores. . . . .	61
5.4	Resultados considerando os custos das soluções obtidas pelas abordagens independente, seqüencial, iterativa e integrada. . . . .	62
5.5	Detalhamento das melhores soluções obtidas pelas abordagens independente, seqüencial, iterativa e integrada. . . . .	67



# Capítulo 1

## Introdução

Devido à sua complexidade o processo de planejamento do sistema de transporte público é normalmente decomposto em subproblemas menores, conforme mostrado na Figura 1.1.



**Figura 1.1.** Planejamento do sistema de transporte público.

O conjunto de viagens, que compõe a tabela de horários, é definido a partir das rotas a serem percorridas e a frequência com que as mesmas serão atendidas. Estas decisões devem ser tomadas com base na disponibilidade de infra-estrutura, nos serviços

requeridos pelos usuários e em aspectos de demanda.

Dada a tabela de horários passa-se à resolução do Problema de Programação de Veículos (PPV), que consiste em determinar uma rotina diária de operação para uma frota de veículos, tendo por objetivo fazer o seu melhor aproveitamento. Como resultado, obtém-se os chamados “blocos de veículos”. Cada bloco corresponde ao conjunto de viagens realizadas por um mesmo veículo ao longo do dia e mantém informações relativas à seqüência de execução das viagens, seus horários e locais de início e término, os tempos destinados para embarque e desembarque de passageiros e os tempos de espera e de deslocamento fora de operação do veículo.

Uma vez que deve haver uma tripulação (motorista e cobrador) responsável por cada deslocamento realizado por um veículo, o Problema de Programação de Tripulações (PPT) trata da alocação das atividades a serem realizadas por cada tripulação no período de um dia, definindo assim as chamadas jornadas de trabalho das tripulações. Para resolução do PPT, inicialmente os blocos de veículos provenientes da solução do PPV são simplificados em tarefas. Cada tarefa engloba viagens consecutivas de um mesmo veículo e entre as quais não pode ocorrer a troca de tripulações devido a limitações de tempo e espaço. Portanto, a solução do PPT é um conjunto de jornadas diárias de trabalho, em que cada jornada consiste em uma seqüência de tarefas a serem realizadas por uma mesma tripulação e que satisfaz a legislação trabalhista e as regras operacionais da empresa.

Resolvido o PPT para cada dia, o problema seguinte é o da Rotação de Tripulações (PRT). Esse problema consiste em atribuir uma jornada de trabalho mensal para cada tripulação, considerando certas regras trabalhistas relativas a longos períodos de tempo e que, portanto, não foram contempladas na programação diária.

Este trabalho trata especificamente da definição das programações de veículos e tripulações, que juntas, concentram a maior parte dos gastos das empresas do setor de transporte público por ônibus. Comumente PPV e PPT são resolvidos de forma seqüencial tradicional, isto é, primeiro é definido o modo de operação da frota e, em seguida, as jornadas de trabalho das tripulações. Nesta metodologia a resolução do PPT torna-se totalmente dependente da programação de veículos a partir da qual é obtida e isto pode ser visto como uma desvantagem, pois em muitas situações práticas os gastos das empresas com tripulantes superam aqueles relativos aos veículos.

Outras abordagens para tratar PPV e PPT encontradas na literatura são:

1. Seqüencial Inversa: primeiro é resolvido o PPT e em seguida, utilizando os resultados obtidos, é solucionado o PPV. Nesta abordagem é priorizada a melhora da qualidade da programação de tripulações e restringe-se a definição da programação dos veículos. Os resultados encontrados normalmente não são bons, pois

o número de veículos necessários costuma aumentar drasticamente;

2. Independente: PPV e PPT são resolvidos separadamente, em qualquer ordem e de forma a não se relacionarem. Geralmente os resultados obtidos são inviáveis do ponto de vista prático, pois dificilmente as soluções para PPV e PPT são compatíveis. A principal finalidade desta abordagem é a determinação de uma melhor programação para as tripulações desconsiderando os veículos e de uma melhor programação para os veículos desconsiderando as tripulações;
3. Integrada: PPV e PPT são resolvidos simultaneamente. Trata-se de uma abordagem promissora, uma vez que a forte dependência entre estes problemas sugere que uma resolução integrada possa resultar em consideráveis reduções de custo.

## 1.1 Justificativa e Objetivos do Trabalho

Segundo Sampaio et al. [2006] o transporte público é considerado uma prioridade governamental em todo o mundo, pois trata-se de uma necessidade para locomoção das pessoas, predominantemente, mas não exclusivamente, para as de renda mais baixa. Além disso, o transporte coletivo atende a um número mais amplo de passageiros que o transporte individual, sem sobrecarregar a infra-estrutura rodoviária e permite substanciais reduções nas emissões de poluentes ao meio ambiente, dada a menor quantidade de veículos por passageiros.

Mais especificamente, nas cidades brasileiras observa-se que juntamente com o crescimento da população urbana vem aumentando a demanda por um serviço de transporte público barato, eficiente e de qualidade. Conforme a ANTP [1999], para que ocorra a democratização do acesso ao transporte público, deve haver a garantia de tarifas toleráveis para os usuários, o que depende em grande parte do aumento da eficiência e redução de custos por parte das organizações do setor. Justificando assim, qualquer iniciativa que venha a proporcionar melhoras no planejamento do sistema de transporte público.

No que se refere ao Sistema de Transporte Público de Ônibus Urbano, determinar uma rotina diária de operação para uma frota de veículos de uma empresa bem como definir as jornadas de trabalho de suas tripulações (motoristas e cobradores) são tarefas muito complexas. Quando estas são realizadas de forma manual, requerem grande esforço e tempo dos profissionais e nem sempre as programações geradas são de qualidade. As dificuldades inerentes à resolução destes problemas devem-se principalmente a uma infinidade de restrições e objetivos a serem alcançados e que são fortemente de-

pendentes das políticas operacionais e das leis trabalhistas consideradas. Evidenciando assim, a necessidade da utilização de recursos computacionais.

Normalmente, no processo de planejamento do sistema de transporte público, a programação de veículos é realizada de forma independente e em uma etapa anterior à programação de tripulações. Nesta metodologia a resolução do PPT torna-se totalmente condicionada à programação de veículos a partir da qual será obtida e, portanto, este pode ser um dos fatores que limita a melhora da qualidade da programação de tripulações. Esta abordagem foi fortemente criticada por Ball et al. [1983], que ao estudar uma empresa de transporte público da América do Norte verificou que o custo associado à programação das tripulações supera o custo relacionado à programação dos veículos, e em muitos casos chega a corresponder a mais de 80% do custo operacional total.

Portanto, neste trabalho é proposta uma estratégia para resolução do Problema de Programação Integrada de Veículos e Tripulações (PPVT). No PPVT são geradas simultaneamente as programações dos veículos e dos tripulantes possibilitando encontrar soluções com custo operacional menor que aquele obtido pela solução seqüencial do PPV e PPT. Além disso, com o intuito de identificar a efetividade da abordagem integrada, PPV e PPT também são resolvidos de forma seqüencial tradicional, independente e iterativa (em que, ora a busca por melhoras se concentra especificamente no PPV e ora no PPT). O objetivo é avaliar a metodologia proposta, a fim de que seja encontrada uma maneira eficiente de especificar a forma de utilização da frota e da mão-de-obra de uma empresa de transporte público, de modo a minimizar ao máximo os custos envolvidos em tal atividade, respeitando as regras operacionais e as leis trabalhistas intrínsecas ao problema. Para tanto, são resolvidos problemas reais referentes ao Sistema de Transporte Público da cidade de Belo Horizonte, capital do estado de Minas Gerais, permitindo que seja adquirido conhecimento sobre as características e dificuldades da aplicação da metodologia integrada à realidade brasileira.

## 1.2 Organização do Texto

A seguir, no Capítulo 2, é feita uma revisão da literatura sobre o problema que norteia todo esse trabalho, o Problema de Programação Integrada de Veículos e Tripulações (PPVT). Modelos e algoritmos propostos para tratar o PPVT são apresentados.

O Capítulo 3 trata separadamente dos problemas de Programação de Veículos (PPV) e de Programação de Tripulações (PPT), considerando duas abordagens distintas, a seqüencial tradicional e a independente. O objetivo é descrever detalhadamente os problemas e a metodologia utilizada para resolvê-los.

O Capítulo 4 apresenta o Problema de Programação Integrada de Veículos e Tripulações. Inicialmente é proposto um método que aborda PPV e PPT de forma iterativa e, em um segundo passo, é apresentada uma estratégia de resolução totalmente integrada dos problemas.

No Capítulo 5 são descritos, respectivamente, as instâncias utilizadas e os resultados obtidos nos experimentos realizados, considerando as abordagens apresentadas nos capítulos 3 e 4.

Por fim, no Capítulo 6, são apresentados as conclusões e os trabalhos futuros.



## Capítulo 2

# Revisão Bibliográfica

Embora haja um entendimento de que a resolução integrada das programações de veículos e tripulações seja vantajosa, a maioria dos trabalhos encontrados na literatura segue a abordagem seqüencial na qual a programação de veículos é definida a priori e independentemente da programação de tripulações.

O Problema de Programação de Veículos há muito tempo vem sendo pesquisado e resolvido com o auxílio de computadores. Os primeiros trabalhos tratavam o PPV tanto utilizando métodos exatos [Kirkman, 1968] quanto procedimentos heurísticos [Saha, 1970; Wren, 1972]. O Problema de Programação de Tripulações também tem recebido considerável atenção na literatura e é estudado desde a década de 60, sendo Elias [1964] um dos precursores. Pesquisas mais recentes para resolução do PPV podem ser encontradas em Baita et al. [2000], Silva [2001] e Souza et al. [2007] e para o PPT em Shen e Kwan [2001], Daduna e Voss [2001], Marinho [2005] e em Santos e Mateus [2009].

O Problema de Programação Integrada de Veículos e Tripulações somente começou a ser estudado recentemente, uma vez que devido à sua grande complexidade, o custo computacional desta metodologia sempre foi restritivo. Ball et al. [1983] foram os primeiros a propor um método de resolução para o PPVT. O procedimento apresentado pelos autores envolve a definição de uma rede de programação, consistindo de vértices indicando viagens que devem ser executadas por um veículo e uma tripulação (vértices *d-trip*), e dois outros vértices *s* e *t* representando a garagem. Diversos tipos de arcos podem ser agrupados em duas categorias, aqueles que indicam que uma tripulação e veículo seguem de uma *d-trip* para outra e aqueles que indicam que apenas a tripulação realiza este deslocamento (arcos *crew-only*). O procedimento de resolução é decomposto em três componentes. Inicialmente são criadas partes de jornadas de trabalho, limitadas por uma duração máxima dada por uma constante *T*. É importante lembrar que uma jornada corresponde às atividades a serem realizadas

por uma mesma tripulação em um dia de trabalho. Em seguida, aquelas partes de jornadas obtidas que apresentam pequena duração são colocadas aos pares, formando assim, jornadas parciais. Por fim, longas partes de jornadas e jornadas parciais são combinadas resultando em um conjunto de jornadas de trabalho para as tripulações (solução do PPT). A programação dos veículos é gerada simultaneamente ignorando os arcos exclusivos das tripulações (arcos *crew-only*) presentes na solução. Heurísticas similares são propostas por Tosini e Vercellis [1988] e Patrikalakis e Xerocostas [1992].

A primeira formulação matemática para o PPVT foi proposta em Patrikalakis e Xerocostas [1992]. No entanto, este modelo era apenas para fins ilustrativos sendo computacionalmente intratável [Freling et al., 1999]. Assim, considera-se que Freling et al. [1995] foram os pioneiros em propor uma formulação de programação inteira para o problema e deram origem a uma série de pesquisas desenvolvidas por Freling e co-autores sobre o tema, tais como: Freling [1997], Freling et al. [1999] e Freling et al. [2003]. As formulações apresentadas nestes trabalhos são similares e constituídas basicamente de duas partes:

1. Uma formulação de quase atribuição baseada em uma rede  $G(V, A)$  que assegura a viabilidade do Problema de Programação de Veículos. Nesta rede  $V$  é o conjunto de todas as viagens a serem realizadas e mais dois nós,  $s$  (origem) e  $t$  (destino), ambos representando a garagem. O conjunto  $A$  corresponde aos arcos que conectam  $s$  e  $t$  a todos os demais nós em  $V$  e aos arcos que ligam viagens compatíveis. Salienta-se que duas viagens são compatíveis se elas podem ser realizadas por um mesmo veículo. Desta forma, uma solução viável para o PPV é um conjunto de caminhos disjuntos de  $s$  para  $t$  na rede  $G$ , de tal forma que cada nó viagem em  $V$  seja coberto exatamente uma vez;
2. Restrições de particionamento de conjunto que garantem que cada viagem e cada deslocamento realizado pelo veículo fora de operação, que façam parte da solução do PPV, sejam cobertos pela jornada de trabalho de alguma tripulação.

Em Haase e Friberg [1999] foi apresentada a primeira abordagem de solução exata para o Problema Integrado de Programação de Veículos e Tripulações. O modelo proposto combina as abordagens de Desrochers e Soumis [1989] e Ribeiro e Soumis [1994], em que o PPVT é formulado como um problema de particionamento de conjunto com restrições adicionais que são utilizadas para relacionar as programações de veículos e de tripulações. Na metodologia proposta uma coluna no modelo representa o bloco de um veículo ou a jornada de trabalho de uma tripulação. Estas colunas correspondem a caminhos em dois grafos, um para a programação de veículos e outro para a programação das tripulações. Destaca-se que restrições de recursos foram inseridas

---

no grafo relacionado às tripulações com o intuito de limitar o tempo de trabalho e o número de paradas em uma jornada. Soluções ótimas são obtidas por um algoritmo *branch-and-price-and-cut* em que a técnica de geração de colunas é utilizada para determinar tanto blocos de veículos, quanto jornadas de trabalho para as tripulações. Com o objetivo de melhorar os limites inferiores obtidos pela relaxação linear do problema, cortes poliédricos como propostos em Hoffman e Padberg [1993] são considerados para os veículos. A modelagem proposta é suficientemente genérica para ser estendida para múltiplas garagens e pode receber restrições adicionais, porém, somente pequenas instâncias envolvendo no máximo 20 viagens são resolvidas na otimalidade em um tempo razoável de CPU.

Em Freling et al. [2003] foram apresentadas três metodologias para resolução dos problemas de Programação de Veículos e Programação de Tripulações: A abordagem seqüencial em que primeiro resolve-se o PPV e a partir da solução obtida o PPT é solucionado; a programação independente das tripulações, isto é, o PPT é resolvido sem considerar a solução obtida no PPV; e a abordagem integrada em que PPV e PPT são resolvidos simultaneamente. Procurou-se neste trabalho identificar os benefícios da programação integrada de veículos e tripulações em comparação com a seqüencial. Para tanto, foram considerados problemas reais de uma empresa de transporte público da Holanda e avaliou-se tanto a situação em que uma tripulação pode realizar troca de veículos durante sua jornada de trabalho, quanto a situação em que esta troca de veículos não é permitida. Modelos matemáticos são apresentados para as três abordagens: Seqüencial, independente e integrada. Para resolução do PPVT foi proposta uma heurística lagrangeana que faz uso da técnica de geração de colunas no processo de resolução do problema relaxado. Nesta abordagem o subproblema lagrangeano considerado é o problema de programação de veículos com uma única garagem, que é resolvido pelo algoritmo proposto em Freling et al. [2001]. Durante o processo de resolução do problema relaxado o conjunto de jornadas de trabalho das tripulações é atualizado, de tal forma que todas as jornadas com custo reduzido negativo sejam consideradas. Como não é possível gerar todas as jornadas das tripulações de imediato (não se tem uma programação de veículos previamente definida), isto é feito dinamicamente a partir de um processo que inicialmente cria *partes de jornadas* e em seguida combina estas partes formando jornadas. Neste contexto, uma *parte de jornada* é entendida como uma seqüência de tarefas provenientes de um mesmo bloco de veículo e que podem ser realizadas por uma mesma tripulação sem interrupções. Destaca-se que cada tarefa continua sendo definida como o agrupamento de viagens de um mesmo veículo que devem ser obrigatoriamente realizadas por uma mesma tripulação. Resultados computacionais mostram a eficiência da abordagem integrada em diferentes situações,

principalmente naquelas em que uma tripulação não pode trocar de veículo.

Laurent e Hao [2007] tratam do problema particular de uma companhia de aluguel de limousines que atua na cidade de Paris (França). Dado um conjunto de viagens diárias, com seus respectivos horários e locais de início e término e número de passageiros, deseja-se alocar simultaneamente, veículos e motoristas, de tal forma que sejam atendidas tantas viagens demandadas quanto possível. Destaca-se que cada viagem, no problema abordado, requer um motorista que tenha certa aptidão e um veículo com determinadas características. Outra particularidade refere-se ao fato da carga de trabalho da companhia ser dinâmica, pois constantemente viagens são agendadas, canceladas e modificadas, conforme necessidades dos clientes. Assim, o objetivo é, ao fim de cada dia, determinar as jornadas de trabalho dos veículos e motoristas visando otimizar a utilização de recursos disponíveis, atender certas restrições e ao mesmo tempo priorizar a qualidade do serviço considerando as viagens até então agendadas para o próximo dia. Dessa forma, posteriores mudanças no quadro de viagens são tratadas manualmente. Para resolver o problema foi utilizado um algoritmo de duas fases. Inicialmente uma solução é obtida utilizando a técnica de programação por restrições [Tsang, 1993], da seguinte forma: um procedimento construtivo guloso primeiramente dispõe as viagens em ordem decrescente de duração e, em seguida, seqüencialmente determina um par motorista-veículo para realizar cada uma delas. Após cada atribuição, um procedimento prevê futuros conflitos e elimina as possibilidades dos mesmos ocorrerem. Esta fase de construção é finalizada quando há um par motorista-veículo alocado a cada viagem ou quando nenhuma atribuição pode mais ser realizada sem que haja violação de alguma restrição. Na segunda fase a solução gerada é melhorada por um procedimento baseado na metaheurística *Simulated Annealing*. Para tanto, foram elaborados cinco mecanismos de exploração da vizinhança sendo o espaço de busca composto apenas por soluções viáveis. A avaliação de uma solução é feita considerando vários objetivos, cada qual com um peso associado. Este peso é definido a partir de um parâmetro de entrada que reflete a importância do objetivo ao qual se refere em conjunto com um método de aprendizagem que, a partir de várias execuções de diferentes instâncias, estima os valores dos atributos da solução baseado no seu número de viagens. Testes foram realizados com 10 quadros de viagens diferentes da companhia de aluguel de limousines. Os resultados obtidos foram comparados com aqueles gerados manualmente e constatou-se que utilizando menos tempo o algoritmo proposto foi capaz de gerar programações de qualidade muito superior.

Outra variação do PPVT aparece quando este é resolvido considerando que a frota de veículos, disponível para realização das viagens, se encontra espalhada em diferentes garagens. Tal problema é conhecido na literatura inglesa como *Multiple-Depot Vehicle*

---

*and Crew Scheduling Problem - MDVCSP.*

Em Mesquita e Paias [2008] são propostas duas formulações para o MDVCSP. A primeira (SP-VCSP) é similar a uma apresentada em Huisman et al. [2005], porém com um menor número de restrições e variáveis de decisão. Esta formulação combina dois modelos, um de fluxos em rede multi-produto para a programação de veículos e outro de particionamento de conjunto para a programação de tripulações. A segunda formulação (SPC-VCSP) é uma extensão da anterior em que algumas restrições de particionamento de conjunto são trocadas por restrições de cobertura de conjunto. Esta modificação tornou o modelo mais flexível e viabilizou as situações em que era permitido à tripulação trocar de veículos durante a sua jornada de trabalho sem, contudo, aumentar a complexidade do problema. A metodologia de resolução proposta pelas autoras consiste em quatro etapas:

1. Definição do conjunto de tarefas: Para formação das tarefas (agrupamento de viagens que devem ser realizadas por uma mesma tripulação) primeiramente é necessário definir a programação de veículos. Desconsiderando a necessidade de um veículo retornar sempre para a garagem de origem, o Problema de Programação de Veículos com Múltiplas Garagens conforme formulado no trabalho pôde ser resolvido em tempo polinomial (problema de atribuição). Assim, cada bloco de veículo obtido é analisado separadamente de tal forma que duas viagens consecutivas são agrupadas em uma única tarefa sempre que o tempo compreendido entre elas seja inferior a um dado valor e não se viole nenhuma restrição relativa a uma jornada da tripulação;
2. Obtenção do conjunto de jornadas inicial: Um conjunto inicial de jornadas de trabalho para as tripulações se faz necessário para que seja resolvido o problema de *pricing* no processo de geração de colunas (etapa 3). Estas jornadas também foram obtidas resolvendo o Problema de Programação de Veículos com Múltiplas Garagens sem a restrição de que os veículos retornem à garagem de origem. No entanto, nesta abordagem manipulam-se tarefas ao invés de viagens e se obtém jornadas no lugar de blocos de veículos. Ressalta-se que neste processo jornadas inviáveis podem ser construídas, assim aqueles blocos que excedem a duração máxima permitida para uma jornada são particionados e os blocos que violem qualquer outra restrição são mantidos, porém atribui-se um custo elevado a eles;
3. Resolução da relaxação linear do problema utilizando a técnica de geração de colunas: É definida uma Rede Jornada, tal que, um caminho na rede corresponde a uma jornada viável. A jornada com o menor custo reduzido é um caminho mínimo na Rede Jornada. Assim, como se deseja inserir várias colunas (jorna-

das) no problema mestre a cada iteração, o problema de *pricing* é resolvido por programação dinâmica. Nesta abordagem, com o intuito de agilizar o processo de resolução, durante os estágios da programação dinâmica são eliminados alguns estados, de duas maneiras: Sempre que o custo reduzido da jornada sendo formada exceder um dado valor pré-definido e de forma randômica;

4. Aplicação de um procedimento de *branch and bound* caso a solução obtida na etapa anterior não seja inteira: Basta realizar *branching* sobre as variáveis referentes à programação de veículos ou à programação de tripulações, conforme provado neste trabalho.

## Capítulo 3

# Programações Tradicional e Independente de Veículos e Tripulações

Conforme mencionado anteriormente existem diferentes maneiras de tratar os problemas de Programação de Veículos e Programação de Tripulações. Nesse capítulo, PPV e PPT, são resolvidos seguindo as estratégias seqüencial tradicional e independente.

Na abordagem seqüencial tradicional inicialmente é definida a forma de operação dos veículos e, em seguida, baseando-se na programação obtida para o PPV, são definidas as jornadas de trabalho dos tripulantes. Essa metodologia de resolução, por si só, já é interessante visto que PPV e PPT são problemas de grande relevância, havendo inclusive, muitos trabalhos na literatura que tratam dos mesmos separadamente. Além disso, as idéias aqui apresentadas se mostram importantes uma vez que são exploradas no decorrer de todo o trabalho e permitem comparar os desempenhos das abordagens, seqüencial e integrada, para definição das programações de veículos e tripulações considerando problemas reais.

Esse capítulo está organizado conforme descrito a seguir. Inicialmente, na seção 3.1, são definidos conceitos básicos sobre o desenvolvimento de heurísticas e é apresentada a metaheurística de Busca Local Iterada, tratam-se de técnicas utilizadas nesse trabalho. Nas seções 3.2 e 3.3 são descritos, respectivamente, o Problema de Programação de Veículos e a metodologia utilizada para resolvê-lo. Na seção 3.4 é apresentado o Problema de Programação de Tripulações e, posteriormente, na seção 3.5, o método de resolução proposto para o PPT é mostrado. Por fim, a seção 3.6 trata do Problema de Programação Independente de Veículos e Tripulações, em que PPV e PPT são resolvidos separadamente e sem se relacionarem.

## 3.1 Heurísticas

Observa-se que no PPV e PPT o número de combinações, em termos de blocos de veículos e jornadas das tripulações, cresce exponencialmente com o tamanho da instância sendo considerada. Tratam-se de problemas altamente combinatoriais e que se enquadram na classe de problemas NP-difícil, não havendo, portanto, algoritmos em tempo polinomial para encontrar soluções ótimas. Dessa forma, torna-se necessária uma abordagem heurística para solução de problemas maiores.

As heurísticas são algoritmos exploratórios que visam obter uma boa solução a um custo computacional aceitável, sem contudo, garantir a otimalidade ou quão próxima dela a solução se encontra.

As heurísticas são classificadas em construtivas ou de melhoria. Uma heurística construtiva tem por objetivo determinar uma solução, elemento por elemento. A forma de escolha de cada elemento a ser inserido na solução pode variar e depende da estratégia de construção adotada. As heurísticas de melhoria, por sua vez, têm a finalidade de melhorar uma solução construída e normalmente são baseadas no conceito de *vizinhança*. Sendo  $S$  o espaço de pesquisa de um problema de otimização, a função  $N$  associa a cada elemento  $s \in S$  um conjunto  $N(s) \subseteq S$ . O conjunto  $N(s)$  é chamado de *vizinhança* de  $s$  e cada solução  $s' \in N(s)$  é chamada de *vizinha* de  $s$ . Cada solução  $s'$  é obtida de  $s$  a partir de uma operação chamada de *movimento*, que define a transição entre soluções através de pequenas modificações.

### 3.1.1 Método de Descida

O Método de Descida é uma técnica iterativa que partindo de uma solução inicial qualquer explora toda a vizinhança de diferentes soluções, até atingir um ótimo local, do qual o método não é capaz de escapar.

Inicialmente, o Método de Descida explora toda a vizinhança de sua solução corrente. Se o melhor vizinho encontrado proporcionar um aprimoramento no valor da função de avaliação (função que quantifica a qualidade de uma solução) até então conhecido, ele é aceito como a nova solução a ser explorada. Caso contrário, um ótimo local foi alcançado e a solução corrente é retornada.

O Algoritmo 1 mostra o pseudo-código do Método de Descida, cujo objetivo é minimizar uma função de avaliação  $f$ .

O Método Randômico de Descida é uma variante do Método de Descida e sua finalidade é evitar, a cada iteração, a busca exaustiva em toda a vizinhança da solução corrente.

A idéia básica dessa técnica é obter, a cada iteração, um vizinho qualquer da solução

---

**Algoritmo 1:** Método de Descida.

---

- 1 Seja  $s_0$  uma solução inicial conhecida
  - 2  $s \leftarrow s_0$ , onde  $s$  é a solução corrente
  - 3  $s' \leftarrow \text{melhorVizinho}(s)$ , onde  $s'$  é o melhor dentre todos os vizinhos de uma dada solução  $s$
  - 4 **enquanto**  $f(s') < f(s)$  **faça**
  - 5      $s \leftarrow s'$
  - 6      $s' \leftarrow \text{melhorVizinho}(s)$
  - 7 Retorna a melhor solução encontrada,  $s$
- 

corrente, admitindo-o como nova solução corrente caso ele proporcione melhoras no valor da função de avaliação. Caso contrário, a solução corrente é mantida inalterada. Esse procedimento se repete até que uma condição de parada seja satisfeita.

No Algoritmo 2 é apresentado o pseudo-código do Método Randômico de Descida aplicado à minimização de uma função de avaliação  $f$ .

---

**Algoritmo 2:** Método Randômico de Descida.

---

- 1 Seja  $s_0$  uma solução inicial conhecida
  - 2  $s \leftarrow s_0$ , onde  $s$  é a solução corrente
  - 3 **enquanto** condição de parada não for satisfeita **faça**
  - 4     Seja  $s' \in N(s)$  um vizinho qualquer da solução  $s$
  - 5     **se**  $f(s') < f(s)$  **então**
  - 6          $s \leftarrow s'$
  - 7 Retorna a melhor solução encontrada,  $s$
- 

### 3.1.2 Busca Local Iterada

As metaheurísticas podem ser vistas como sendo heurísticas genéricas mais sofisticadas, que visam explorar inteligentemente as instâncias do problema e o seu espaço de soluções. Elas se caracterizam pela habilidade de escapar de ótimos locais.

A metaheurística Busca Local Iterada - *ILS* (*Iterated Local Search*) tem como idéia principal focar a pesquisa por melhoras apenas no subespaço definido pelas soluções que são localmente ótimas segundo um dado procedimento de otimização, e não em todo o espaço de soluções. Para tanto é proposto o uso de um método de busca local combinado com a aplicação de perturbações (modificações realizadas em uma solução).

Assim, a cada iteração do *ILS* é realizada uma perturbação sobre a solução corrente seguida de um procedimento de busca local que melhorará a solução perturbada. Essa nova solução é então avaliada e caso seja satisfeito o critério de aceitação (por exemplo, se houve melhora na função de avaliação), ela será considerada a solução corrente. Caso

contrário, a solução corrente é mantida inalterada. Esse procedimento é repetido até que a condição de parada seja satisfeita.

No Algoritmo 3 é apresentada a metaheurística *ILS* básica [Lourenço et al., 2003] em que se identificam quatro componentes:

1. *geraSolucaoInicial*: que gera uma solução inicial  $s_0$  para o problema;
2. *buscaLocal*: que retorna uma solução melhorada  $s$  após geração da solução inicial e outra  $s''$  após perturbação;
3. *perturbacao*: que modifica a solução corrente  $s$  guiando a uma solução intermediária  $s'$ . Uma perturbação eficiente é aquela que não altera drasticamente a solução corrente, mas, ainda assim, permite que a busca local explore diferentes regiões;
4. *criterioAceitacao*: decisão que se refere tanto à aceitação da solução perturbada como solução corrente, quanto quão intensa será a próxima perturbação a ser aplicada (isto é, determinação do nível de perturbação).

---

**Algoritmo 3:** ILS - Busca Local Iterada.

---

```

1  $s_0 \leftarrow \text{geraSolucaoInicial}()$ 
2  $s \leftarrow \text{buscaLocal}(s_0)$ , onde  $s$  é a solução corrente
3 enquanto condição de parada não for satisfeita faça
4    $s' \leftarrow \text{perturbacao}(\text{historico}, s)$ 
5    $s'' \leftarrow \text{buscaLocal}(s')$ 
6    $s \leftarrow \text{criterioAceitacao}(s, s'', \text{historico})$ 
7 Retorna a solução  $s$ 

```

---

Destaca-se que algoritmos mais eficientes são obtidos quando tanto a perturbação quanto o critério de aceitação dependem das soluções obtidas em iterações anteriores, ou seja, utiliza-se do recurso de memória (representada pelo *historico* no Algoritmo 3).

## 3.2 Descrição do Problema de Programação de Veículos

O Problema da Programação de Veículos (PPV) consiste em criar uma rotina diária de operação para uma frota de veículos disponibilizada para uso de uma empresa, de modo que, todas as viagens a ela responsabilizadas sejam realizadas e a programação gerada não viole as regras operacionais e as limitações de tempo e espaço existentes.

A programação dos veículos é feita a partir de uma tabela de horários que é definida em uma etapa anterior a resolução do PPV e na qual se encontram todas as viagens a serem realizadas em um determinado dia por uma empresa. A Tabela 3.1 ilustra um fragmento de uma tabela de horários.

**Tabela 3.1.** Tabela de horários.

Número da Viagem	Horário de Início	Ponto Inicial	Horário de Término	Ponto Final	Linha	Tempo de Embarque	Tempo de Desembarque
1	5	30	70	30	30.4	1	1
2	5	302	38	302	302.4	1	1
3	20	30	76	30	30.4	1	1
4	240	302	265	302	302.4	1	1
5	255	30	309	30	30.3	1	2
6	260	302	281	302	302.1	1	1
7	270	305	335	305	305.1	1	2
8	270	30	331	30	30.3	1	2
9	270	326	300	326	326.1	1	2
10	280	30	340	30	30.3	1	2

Na tabela de horários, cada viagem é identificada univocamente por meio de um conjunto de características que consistem em informações relevantes no processo de resolução do problema. Organizados em colunas e aparecendo seqüencialmente na tabela, temos: o número da viagem, horário de início da viagem (em minutos), o ponto inicial que corresponde ao local de início da viagem, horário de término da viagem (em minutos), o ponto final que se refere ao local em que a viagem termina, a linha a qual a viagem pertence, e finalmente as duas últimas colunas que se referem respectivamente aos tempos destinados ao embarque e desembarque de passageiros, dados em minutos, e que não necessariamente são iguais, sendo o primeiro dependente do horário de início da viagem e o segundo do horário de fim da mesma.

Para exemplificar, considere a primeira viagem “1” apresentada na Tabela 3.1. Essa viagem começa no ponto “30” aos 5 minutos do dia, ou seja, às 00:05 horas e é finalizada no ponto “30” aos 70 minutos do dia, que corresponde às 01:10 horas. As duas últimas colunas indicam que o veículo que fará a viagem “1” permanecerá um minuto no terminal para embarque e após chegar ao destino mais um minuto de terminal deverá ser cumprido para permitir o desembarque de passageiros.

Além do conjunto de viagens, devem ser fornecidas também informações relativas aos tempos de deslocamento entre os vários pontos de parada dos veículos, incluindo a garagem. De posse dessas informações é possível determinar o tempo que um veículo gasta fora de operação, ou seja, sem passageiros, para sair de um ponto e chegar a outro. A este tipo de deslocamento é dado o nome de viagem morta.

Para que a programação de atividades da frota seja factível é necessário atender a uma série de restrições, as quais podem variar conforme as políticas operacionais que regem o Sistema de Transporte Público Urbano por Ônibus no qual a empresa atua. No problema sendo abordado é considerado que todos os veículos da frota são idênticos e que partem de uma mesma garagem. As restrições que serão levadas em consideração são as seguintes:

- (a) Todas as viagens de responsabilidade de uma empresa, sem exceções, devem ser realizadas por algum veículo da frota;
- (b) Antes de se iniciar e após finalizar uma viagem, um veículo deve obrigatoriamente cumprir um tempo mínimo no terminal, que corresponde ao tempo necessário para o embarque e desembarque dos passageiros;
- (c) Um mesmo veículo somente poderá realizar duas viagens,  $i$  e  $j$ , consecutivamente, se o intervalo de tempo existente entre elas for suficiente para que o veículo desloque do ponto final da viagem  $i$  para o ponto inicial da viagem  $j$ , respeite os tempos destinados para embarque e desembarque de passageiros e inicie a viagem  $j$  sem atrasos. Caso contrário, ocorrerá uma sobreposição de horários entre as viagens;
- (d) Um veículo não deve ficar mais de duas horas em um terminal esperando para executar a sua próxima viagem. Se o tempo de espera exceder esse limite, o veículo deverá se deslocar para a garagem. Diz-se, nesse último caso, que o veículo está realizando uma dupla pegada;
- (e) O número de duplas pegadas da programação não pode exceder 60% do tamanho da frota;
- (f) Todo veículo deve iniciar e terminar a sua jornada de trabalho na garagem;
- (g) Todo veículo deve permanecer, no mínimo, trinta minutos consecutivos na garagem, por dia, para que ocorra a sua manutenção e limpeza. Este tempo pode ser cumprido quando o veículo pára na garagem durante uma dupla pegada ou mesmo depois de completada toda a sua jornada operacional diária.

Neste trabalho o objetivo é obter uma programação de veículos em que o tamanho da frota utilizada e os deslocamentos dos veículos fora de operação (deslocamentos que não equivalem a nenhuma viagem) sejam minimizados.

A solução para o Problema de Programação de Veículos consiste, portanto, em atribuir um conjunto de viagens para um número de veículos menor ou igual à frota

disponível pela empresa, considerando as restrições apresentadas acima, de tal forma que todas as viagens sejam realizadas com o menor custo operacional possível. Ao se resolver o PPV tem-se como resultado os chamados blocos dos veículos. Cada um destes blocos corresponde a um conjunto de viagens que serão executadas por um mesmo veículo.

### 3.3 Método de Solução do PPV

Nesta seção é apresentada a metodologia proposta para resolver o PPV. Na subseção 3.3.1 mostra-se como uma solução para o problema é representada. Na subseção 3.3.2 são apresentadas as estruturas de vizinhança desenvolvidas para explorar o espaço de busca. Na subseção 3.3.3 é mostrada a função de avaliação que foi elaborada para quantificar a qualidade de uma solução. Descreve-se na subseção 3.3.4 o procedimento de geração da solução inicial. Por fim, na subseção 3.3.5, é apresentado o algoritmo proposto para resolver o problema.

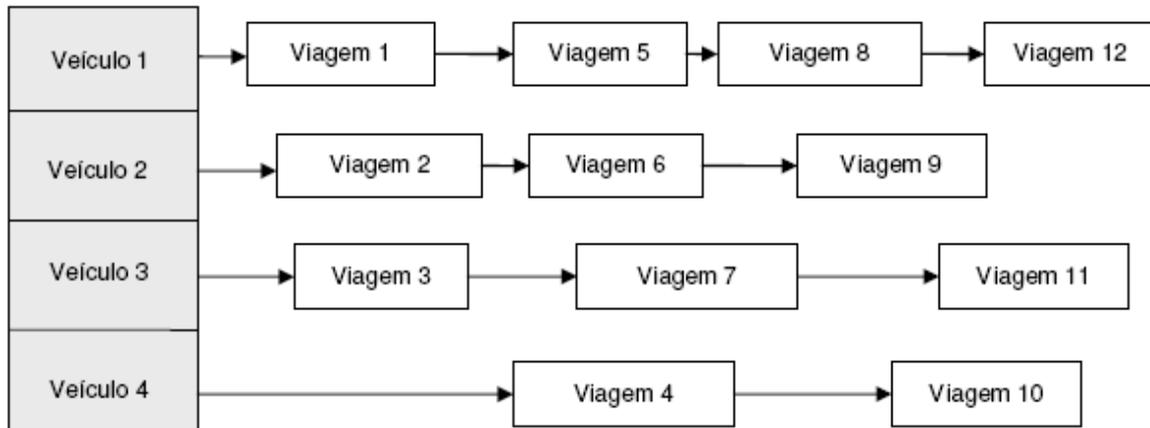
#### 3.3.1 Representação

Uma solução  $s^v$  para o PPV consiste em uma lista de veículos, sendo que a cada veículo está associada uma lista de viagens a serem por ele executadas durante um dia de trabalho. É pertinente ressaltar que as viagens realizadas por cada veículo são mantidas ordenadas por seus horários de início. Esta abordagem é importante, uma vez que através dela será possível avaliar a jornada de trabalho de um veículo sob vários aspectos, tais como: tempo de locomoção fora de operação (viagem morta), tempo de espera no terminal e total de retornos à garagem.

A Figura 3.1 ilustra uma solução  $s^v$  para o PPV, na qual uma frota de quatro veículos deve executar doze viagens a ela alocadas.

#### 3.3.2 Estruturas de Vizinhança

Para explorar o espaço de soluções do PPV foram utilizados três tipos de movimentos que definem, respectivamente, as estruturas de vizinhança  $VN^{(R)}$ ,  $VN^{(T)}$  e  $VN^{(RT)}$ , são eles: realocação de viagem, troca de viagens e realocação-troca de viagens. A realocação consiste em transferir uma viagem pertencente a um veículo para outro veículo. O movimento de troca corresponde em permutar uma viagem de um dado veículo com uma viagem pertencente a outro veículo. Já o movimento de realocação-troca de viagens envolve três veículos,  $v_1$ ,  $v_2$  e  $v_3$ , o veículo  $v_1$  tem uma de suas viagens transferida para o veículo  $v_2$  e o veículo  $v_2$ , por sua vez, cede uma de suas viagens (desde



**Figura 3.1.** Exemplo de uma solução  $s^v$  para o PPV.

que não seja a viagem proveniente de  $v_1$ ) para o veículo  $v_3$ . Portanto, neste movimento o veículo  $v_2$  troca uma de suas viagens a partir de duas realocações sucessivas.

Na Figura 3.2 estão ilustrados estes movimentos.

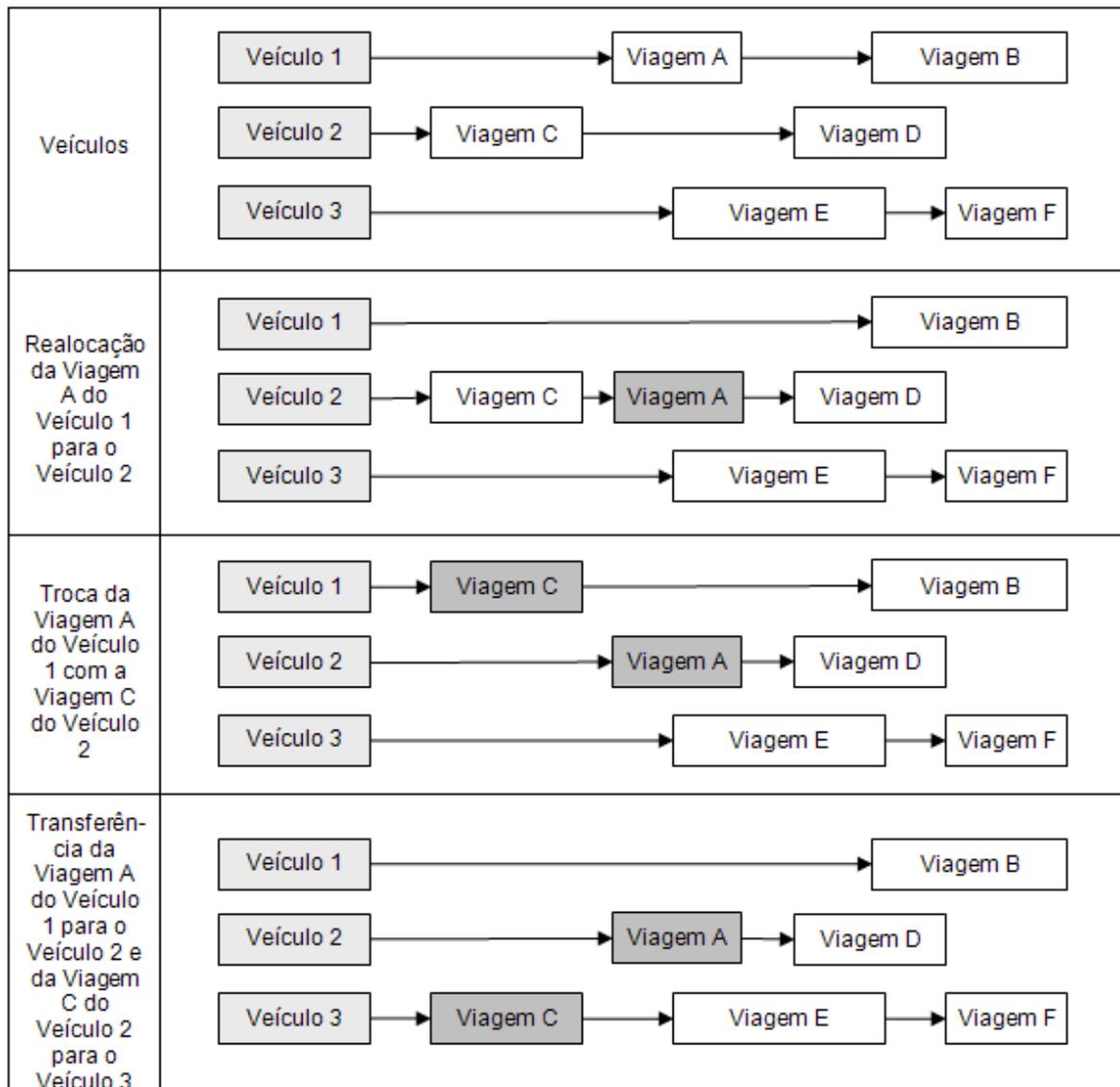
### 3.3.3 Função de Avaliação

A função elaborada para quantificar a qualidade de uma solução para o PPV é baseada na penalização dos atributos indesejáveis da programação. Portanto, uma solução  $s^v$  para o Problema de Programação de Veículos é avaliada com base na seguinte função, a qual deve ser minimizada:

$$\begin{aligned}
 f^v(s^v) = & \text{totalVeiculos} \times \text{custoUsoVeiculo} \\
 & + \text{tempoViagemMorta} \times \text{custoViagemMorta} \\
 & + \text{totalExcessoDP} \times \text{pesoInviabilidade} \\
 & + \text{tempoSobreposicao} \times \text{pesoInviabilidade}
 \end{aligned} \tag{3.1}$$

em que:

- (a)  $\text{totalVeiculos}$  é a quantidade de veículos utilizados;
- (b)  $\text{custoUsoVeiculo}$  é o custo associado ao uso de um veículo;
- (c)  $\text{tempoViagemMorta}$  é o tempo total, em minutos, que foi despendido com os deslocamentos realizados fora de operação (viagem morta), considerando todos os veículos da frota;



**Figura 3.2.** Estruturas de vizinhança do PPV.

- (d)  $custoViagemMorta$  é o custo associado a cada minuto de viagem morta realizada por um veículo;
- (e)  $totalExcessoDP$  é a quantidade de duplas pegadas da programação que supera um certo número admissível, previamente definido;
- (f)  $tempoSobreposicao$  é a soma, dada em minutos, de todos os intervalos de tempo em que há sobreposição de horários entre viagens de um mesmo veículo da programação. Também será considerado sobreposição, o tempo faltante para que sejam completados os trinta minutos diários na garagem de cada veículo que não contemple esta restrição. Destaca-se que na sobreposição de horários entre duas

viagens estão embutidos os tempos para embarque e desembarque de passageiros e de deslocamento do veículo que não foram respeitados;

(g) *pesoInviabilidade* é o peso dado a cada inviabilidade presente na solução.

### 3.3.4 Procedimento de Geração da Solução Inicial

A programação de veículos inicial é gerada por uma heurística construtiva gulosa. Em heurísticas deste tipo uma solução é construída iterativamente, elemento por elemento, tomando-se a cada passo a melhor decisão segundo um dado critério. No algoritmo proposto, inicia-se com uma frota formada por um único veículo com jornada vazia e um conjunto de viagens a serem realizadas, dispostas ordenadas crescentemente pelo horário de início. A cada passo, remove-se a primeira viagem do conjunto e a aloca na jornada de trabalho diária de algum veículo da frota. Esta combinação deve ser a que implique no menor custo possível de operação determinado pela aplicação da função de avaliação descrita anteriormente pela expressão (3.1). À medida que a solução vai sendo gerada novos veículos são criados dinamicamente, de tal maneira que, sempre seja avaliada a atribuição de uma viagem a um veículo com jornada vazia. Este processo é repetido até que não haja mais viagens a serem alocadas.

O Algoritmo 4, a seguir, mostra o pseudo-código da heurística construtiva gulosa utilizada no PPV.

### 3.3.5 Algoritmo de Resolução

Na resolução do PPV foram utilizados procedimentos de busca local baseados no Método de Descida e no Método Randômico de Descida, descritos na subseção 3.1.1 na página 14, são eles:

1. *DescidaComRealocacao\_PPV* e *DescidaRandomicaComRealocacao\_PPV* que consideram durante as buscas a estrutura de vizinhança  $VN^{(R)}$  (realocação de viagem);
2. *DescidaComTroca\_PPV* e *DescidaRandomicaComTroca\_PPV* que consideram durante as buscas a estrutura de vizinhança  $VN^{(T)}$  (troca de viagens).

Destaca-se que os métodos *DescidaRandomicaComRealocacao\_PPV* e *DescidaRandomicaComTroca\_PPV* foram utilizados com o intuito de introduzir uma certa aleatoriedade ao processo de busca por melhoras e, ao mesmo tempo, torná-lo menos dispendioso em relação ao tempo de processamento. Para tanto, as escolhas dos veículos e das viagens manipulados nos movimentos são feitas de forma aleatória e a solução

---

**Algoritmo 4:** Heurística para geração de uma solução inicial para o PPV.

---

- 1 Seja  $T$  o conjunto de viagens a serem realizadas, com  $T$  ordenado crescentemente por horário de início de suas viagens
  - 2 Seja  $F$  o conjunto de veículos da empresa, inicialmente  $F$  possui um único veículo sem qualquer viagem
  - 3 **enquanto**  $T \neq \emptyset$  **faça**
  - 4      $t \leftarrow$  primeira viagem de  $T$
  - 5      $melhorDelta \leftarrow \infty$ , onde  $melhorDelta$  é a melhor variação ocorrida na função de avaliação ao se alocar a viagem  $t$
  - 6     **para** cada veículo  $v \in F$  **faça**
  - 7          $custoAnterior \leftarrow$  custo do veículo  $v$  antes de receber a viagem  $t$
  - 8          $custoPosterior \leftarrow$  novo custo do veículo  $v$  caso a viagem  $t$  seja inserida em sua jornada de trabalho diária
  - 9         Calcule a variação de custo  $\Delta$  do veículo  $v$  caso ele receba a viagem  $t$ :  
 $\Delta \leftarrow custoPosterior - custoAnterior$
  - 10         **se**  $\Delta < melhorDelta$  **então**
  - 11              $melhorDelta \leftarrow \Delta$
  - 12              $melhorVeiculo \leftarrow info(v)$ , isto é, armazene uma informação que permita identificar o veículo da frota que irá contribuir menos para o valor da função de avaliação ao receber a viagem  $t$
  - 13     **se**  $melhorVeiculo$  não possui viagens **então**
  - 14         Crie um novo veículo sem viagens e adicione ao conjunto  $F$ . (criação dinâmica de veículo, mantendo sempre um veículo sem viagens em  $F$ )
  - 15     Aloque a viagem  $t$  ao veículo da frota  $F$  indicado por  $melhorVeiculo$
  - 16      $fo \leftarrow fo + melhorDelta$ , ou seja, atualize o valor da função de avaliação
  - 17     Remova  $t$  de  $T$
  - 18 Retorne, como solução, a programação da frota  $F$  da empresa
- 

corrente é atualizada sempre que a solução vizinha tenha reduzido o valor da melhor função de custo até então obtida. Além disso, a condição de parada destes procedimentos é determinada pelo número de iterações sem melhoras na solução corrente, conforme expressão (3.2) a seguir:

$$itMax_{PPV} = (total\_viagens \times (total\_veiculos - 1)) \times 0,6 \quad (3.2)$$

onde:

- (a)  $itMax_{PPV}$  é o número máximo de iterações sem melhoras na solução corrente;
- (b)  $total\_viagens$  é o número de viagens a serem realizadas;
- (c)  $total\_veiculos$  é o número de veículos sendo empregados na programação.

Dessa forma, o número máximo de iterações sem melhoras,  $itMax_{PPV}$ , será igual a 60% do total de possíveis realocações de viagens considerando a programação de veículos atual. Tal percentual foi definido empiricamente de forma a encontrar uma boa relação entre o tempo de execução da busca e a qualidade da solução obtida. Esta estratégia é interessante, uma vez que, relaciona o valor de  $itMax_{PPV}$  com o tamanho do problema sendo considerado.

O procedimento proposto neste trabalho para resolver o Problema de Programação de Veículos, cujo pseudo-código é apresentado pelo Algoritmo 5, é baseado na metaheurística Busca Local Iterada.

Uma solução inicial para o PPV é obtida pela heurística construtiva gulosa descrita anteriormente na subseção 3.3.4. Em seguida, com o objetivo de melhorar a qualidade da programação, foram utilizados os procedimentos de busca local *DescidaRandomicaComRealocacao\_PPV*, *DescidaRandomicaComTroca\_PPV*, *DescidaComRealocacao\_PPV* e *DescidaComTroca\_PPV*.

As perturbações correspondem a movimentos realizados sobre a solução corrente e que envolvem veículos e viagens aleatoriamente selecionados na programação. Há cinco níveis de perturbação, são eles: *nível 1* - um movimento de realocação de viagem, *nível 2* - um movimento de troca de viagens, *nível 3* - dois movimentos de realocação de viagem, *nível 4* - dois movimentos de troca de viagens e *nível 5* - um movimento de realocação-troca de viagens.

Os procedimentos de busca local utilizados para refinar a solução perturbada fundamentam-se no Método Randômico de Descida, são eles: *DescidaRandomicaComRealocacao\_PPV* e *DescidaRandomicaComTroca\_PPV*.

Uma nova solução para o PPV somente será aceita se ela for de qualidade melhor ou igual a da solução corrente (linhas 15 e 18). No primeiro caso, o nível de perturbação é reiniciado de forma a intensificar a busca na nova solução obtida. Nas demais situações o nível de perturbação é aumentado em um ou reiniciado (caso já se esteja no nível de perturbação mais alto).

---

**Algoritmo 5:** Algoritmo de resolução do PPV.

---

```

1 Seja  $T$  o tempo de processamento corrente do algoritmo e  $Tmax$  o tempo
  máximo admissível de processamento
2 Seja  $nivel$  o nível de perturbação corrente
3  $T \leftarrow 0$ 
4  $nivel \leftarrow 1$ 
5 Gere uma solução inicial  $s^v$  a partir da heurística construtiva gulosa descrita
  pelo Algoritmo 4, na página 23
6  $s^v \leftarrow DescidaRandomicaComRealocacao\_PPV(s^v)$ 
7  $s^v \leftarrow DescidaRandomicaComTroca\_PPV(s^v)$ 
8  $s^v \leftarrow DescidaComRealocacao\_PPV(s^v)$ 
9  $s^v \leftarrow DescidaComTroca\_PPV(s^v)$ 
10  $s_*^v \leftarrow s^v$ , onde  $s_*^v$  é a melhor solução encontrada até então
11 enquanto  $T < Tmax$  faça
12    $s^v \leftarrow perturbacao(s_*^v, nivel)$ 
13    $s^v \leftarrow DescidaRandomicaComRealocacao\_PPV(s^v)$ 
14    $s^v \leftarrow DescidaRandomicaComTroca\_PPV(s^v)$ 
15   se  $f^v(s^v) < f^v(s_*^v)$  então
16      $s_*^v \leftarrow s^v$ , ou seja, atualize a melhor solução até então encontrada
17      $nivel \leftarrow 1$ 
18   se  $f^v(s^v) = f^v(s_*^v)$  então
19      $s_*^v \leftarrow s^v$ , ou seja, atualize a melhor solução até então encontrada
20      $nivel \leftarrow nivel + 1$ 
21   se  $f^v(s^v) > f^v(s_*^v)$  então
22      $nivel \leftarrow nivel + 1$ 
23   se  $nivel > numNiveis$  então
24      $nivel \leftarrow 1$ , onde  $numNiveis$  é o número máximo de níveis de perturbação
25 Retorne a solução  $s_*^v$ 

```

---

### 3.4 Descrição do Problema de Programação de Tripulações

Em um Sistema de Transporte Público de Ônibus Urbano, todo deslocamento realizado por um veículo envolve a mão-de-obra de uma tripulação constituída por dois funci-

onários: um motorista e um cobrador. O Problema de Programação de Tripulações (PPT) consiste em criar jornadas diárias de trabalho para os motoristas e cobradores de tal forma a viabilizar a execução de todas as viagens da empresa. Além disso, esta distribuição de trabalho deve ser realizada de maneira a minimizar os custos com mão-de-obra e ao mesmo tempo obedecer à legislação trabalhista e as regras operacionais sob as quais a empresa atua.

Uma limitação para a resolução do PPT é o fato de que não é entre quaisquer viagens de um veículo que poderá haver a troca de tripulação. Esta só ocorrerá nas chamadas oportunidades de troca, ou seja, em locais e em intervalos de tempo que possibilitem a mudança de tripulação e a revisão do veículo (verificação do estado em que a tripulação anterior deixou o ônibus).

Assim, na abordagem sequencial tradicional, antes de efetivamente se gerar a programação de tripulações, os blocos de veículos são quebrados em tarefas. Uma tarefa engloba viagens consecutivas, de um mesmo veículo, entre as quais não há nenhuma oportunidade de troca e é caracterizada pelos seguintes atributos: veículo ao qual pertence, horário de início, horário de término, ponto inicial, ponto final, folga acumulada (isto é, tempo ocioso da tripulação no decorrer da tarefa) e duração.

Esta estratégia de construir tarefas facilita a resolução do PPT, pois mantém as viagens que obrigatoriamente devem ser realizadas por uma mesma tripulação juntas, evitando dessa forma que as soluções do PPV e PPT sejam incompatíveis.

Supondo que para ocorrer a troca de tripulação é necessário um tempo maior ou igual a seis minutos entre as viagens, um exemplo de um bloco de veículo particionado em tarefas é mostrado na Figura 3.3 a seguir. Além dos atributos normais de uma viagem, no exemplo, as colunas “Viagem Morta” e “Tempo de Terminal” apresentam, respectivamente, os tempos em minutos, de deslocamento do veículo fora de operação e de espera no terminal aguardando o horário de início da próxima viagem programada.

Nesta figura observa-se, por exemplo, que a “Tarefa 5” engloba as viagens “101”, “120” e “149” por não haver um tempo suficiente entre elas para que ocorra a troca de tripulação. Esta tarefa iniciará às 15:15 horas, no ponto “3055”, e será finalizada neste mesmo ponto às 20:04 horas. Assim sendo, a “Tarefa 5” possui uma duração de 4 horas e 49 minutos e sua folga acumulada é de quatro minutos.

Vale ressaltar que as tarefas são construídas de tal forma a cobrir não apenas as viagens operacionais dos veículos, mas como também os deslocamentos sem passageiros dos mesmos, conforme pode ser notado nas estruturas das tarefas “Tarefa 1”, “Tarefa 3” e “Tarefa 6” do exemplo. A “tarefa 1” considera o deslocamento do veículo da garagem para o ponto de início de sua primeira viagem, a “tarefa 3” contempla o reposicionamento do veículo do ponto “3050” (local de fim da viagem “60”) para o

ponto “3055” (local de início da próxima viagem, viagem “87”) e a “Tarefa 6” cobre o retorno do veículo à garagem ao fim de sua jornada de trabalho.

	Número da Viagem	Ponto Inicial	Horário de Início (hh:mm)	Ponto Final	Horário de Término (hh:mm)	Viagem Morta	Tempo de Terminal	Linha
Tarefa 1	-	Garagem	04:12	3050	04:30	18	0	-
	1	3050	04:30	3050	05:50	0	30	3050.1
Tarefa 2	34	3050	06:20	3050	07:55	0	5	3050.1
	47	3050	08:00	3050	09:28	0	32	3050.1
Tarefa 3	60	3050	10:00	3050	11:47	0	0	3050.1
	-	3050	11:47	3055	12:01	14	39	-
Tarefa 4	87	3055	12:40	3055	14:22	0	53	3055.1
	101	3055	15:15	3055	17:03	0	0	3055.1
Tarefa 5	120	3055	17:03	3055	18:20	0	4	3055.1
	149	3055	18:24	3055	20:04	0	8	3055.1
Tarefa 6	183	3055	20:12	3055	21:31	0	0	3055.1
	-	3055	21:31	Garagem	21:49	18	0	-

**Figura 3.3.** Bloco de veículo particionado em tarefas.

Destaca-se que na resolução do PPV a unidade manipulada é a viagem, já no PPT é a tarefa (a menor porção de uma jornada de trabalho de uma tripulação). As jornadas geradas pela resolução do PPT podem ser de dois tipos: pegada simples ou dupla pegada. Uma jornada do tipo pegada simples é aquela em que todos os intervalos entre suas tarefas consecutivas são inferiores a duas horas, ou seja, a tripulação responsável por essa jornada trabalhará um único turno. Aquelas jornadas que possuem um intervalo de tempo, entre duas de suas tarefas consecutivas, igual ou superior a duas horas, são classificadas como do tipo dupla pegada e são realizadas em dois turnos, sendo esta extensa pausa entre tarefas não remunerada pela empresa. No problema sendo abordado, a duração normal de uma jornada seja ela de pegada simples ou de dupla pegada é de 6:40 horas de efetivo trabalhado. No caso de pegada simples, a tripulação tem direito a um intervalo de 20 minutos para descanso e/ou alimentação. Assim, a duração de sua jornada estende-se a 7:00 horas.

Na formação das jornadas a legislação trabalhista e as regras operacionais devem ser obrigatoriamente satisfeitas para que uma solução para o PPT seja viável. As restrições que serão levadas em consideração são as seguintes:

- (a) Uma tripulação não pode executar duas tarefas simultaneamente, isto é, em nenhuma jornada poderá ocorrer coincidências de horário entre quaisquer de suas tarefas;

- (b) Uma tripulação somente poderá realizar a troca de veículos nas seguintes circunstâncias: nas situações em que os veículos envolvidos se encontram em um mesmo terminal no momento da troca, basta que a tripulação tenha um pequeno intervalo ocioso (cinco minutos no mínimo) para que ela abandone um veículo e desloque rapidamente para assumir o outro. Caso contrário, se os veículos envolvidos estão em terminais distintos, a troca somente poderá ocorrer no intervalo entre os turnos de uma jornada do tipo dupla pegada. Esta última condição se faz necessária devido ao fato de a distância entre os diferentes terminais ser grande. Vale ressaltar que um terminal é o local de disposição dos vários pontos de parada dos veículos e onde ocorre o embarque e desembarque de passageiros;
- (c) Uma tripulação pode extrapolar, no máximo, duas horas diárias de trabalho além da duração normal de sua jornada, sendo este tempo excedente considerado hora extra;
- (d) Toda jornada do tipo pegada simples deve ter uma folga mínima de 20 minutos, podendo esta ser fracionada em dois intervalos, tendo cada um, uma duração igual ou superior a 10 minutos;
- (e) O intervalo mínimo entre o final de uma jornada e o início da mesma no dia seguinte deverá ser de 11:00 horas;
- (f) O número de jornadas de trabalho do tipo dupla pegada não pode exceder a 20% do total de tripulações da programação.

Neste trabalho o objetivo é obter uma programação de tripulações que priorize a diminuição dos custos com mão-de-obra. É importante destacar que isto não implica em apenas reduzir ao máximo o número de tripulantes empregados, mas baseando-se nos gastos envolvidos com a contratação e extensão da jornada (hora extra) de um funcionário, obter a melhor distribuição de trabalho em termos financeiros.

Portanto, uma solução para o PPT é um conjunto de jornadas de trabalho, sendo cada uma delas realizada por uma tripulação pertencente ao quadro de funcionários da empresa. O objetivo é minimizar o custo com a remuneração das tripulações, mas ao mesmo tempo cumprir as regras de caráter operacional e jurídico envolvidos.

### 3.5 Método de Solução do PPT

A metodologia adotada neste trabalho para resolução do PPT é análoga àquela apresentada anteriormente para abordar o PPV, conforme poderá ser observado no decorrer

desta seção. Na subseção 3.5.1 mostra-se como uma solução para o PPT é representada. Em seguida, na subseção 3.5.2, são descritos os tipos de movimentos utilizados para explorar o espaço de busca. Mostra-se na subseção 3.5.3 como se avalia uma solução do problema. O procedimento de construção das tarefas é detalhado na subseção 3.5.4. Os algoritmos para geração das soluções inicial e final para o PPT são descritos, respectivamente, nas subseções 3.5.5 e 3.5.6.

### 3.5.1 Representação

Uma solução  $s^t$  para o PPT consiste em uma lista de jornadas, sendo que a cada jornada está associada uma lista de tarefas a serem executadas por uma mesma tripulação, durante um dia de trabalho, ordenadas crescentemente por seus horários de início. Apesar de ser computacionalmente custoso manter as tarefas ordenadas, esta disposição se faz necessária, pois através dela é possível determinar características relevantes da jornada, tais como: número de trocas de veículos realizadas, tempo de ociosidade, tipo de jornada (pegada simples ou dupla) e existência de sobreposição entre tarefas.

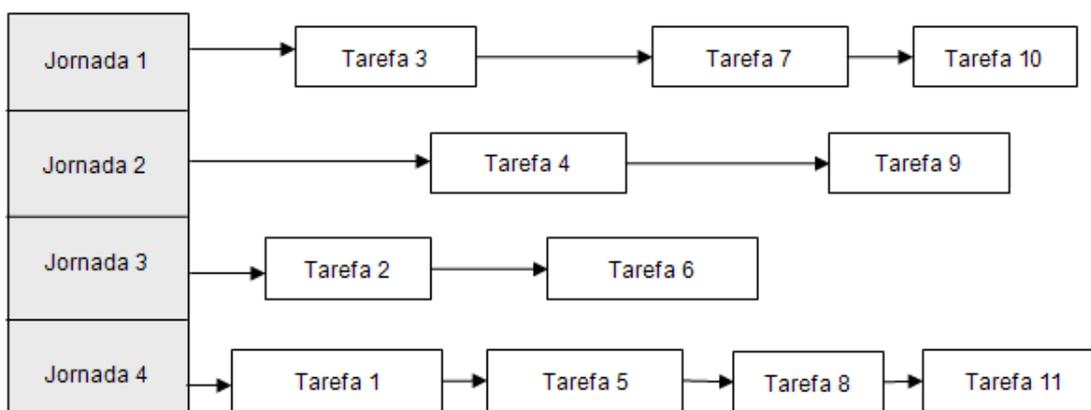


Figura 3.4. Exemplo de uma solução  $s^t$  para o PPT.

### 3.5.2 Estruturas de Vizinhança

Para exploração do espaço de soluções do PPT foram utilizados, assim como no PPV, três tipos de movimentos: realocação de tarefa, troca de tarefas e realocação-troca de tarefas. A realocação de tarefa consiste em transferir uma tarefa pertencente à jornada de trabalho de uma dada tripulação para a jornada de outra tripulação. O movimento de troca de tarefas corresponde a permutar uma tarefa de uma dada jornada com uma tarefa pertencente à outra jornada. Já o movimento de realocação-troca de tarefas envolve três jornadas,  $j_1$ ,  $j_2$  e  $j_3$ , a jornada  $j_1$  tem uma de suas tarefas transferida para

a jornada  $j_2$  e a jornada  $j_2$ , por sua vez, cede uma de suas tarefas (desde que não seja a tarefa proveniente de  $j_1$ ) para a jornada  $j_3$ .

Portanto, foram definidas para o PPT as estruturas de vizinhança  $CN^{(R)}$  que se refere ao movimento de realocação de tarefa,  $CN^{(T)}$  associada ao movimento de troca de tarefas e  $CN^{(RT)}$  relacionada ao movimento de realocação-troca de tarefas.

Os movimentos de realocação, troca e realocação-troca de tarefas estão representados na Figura 3.5.

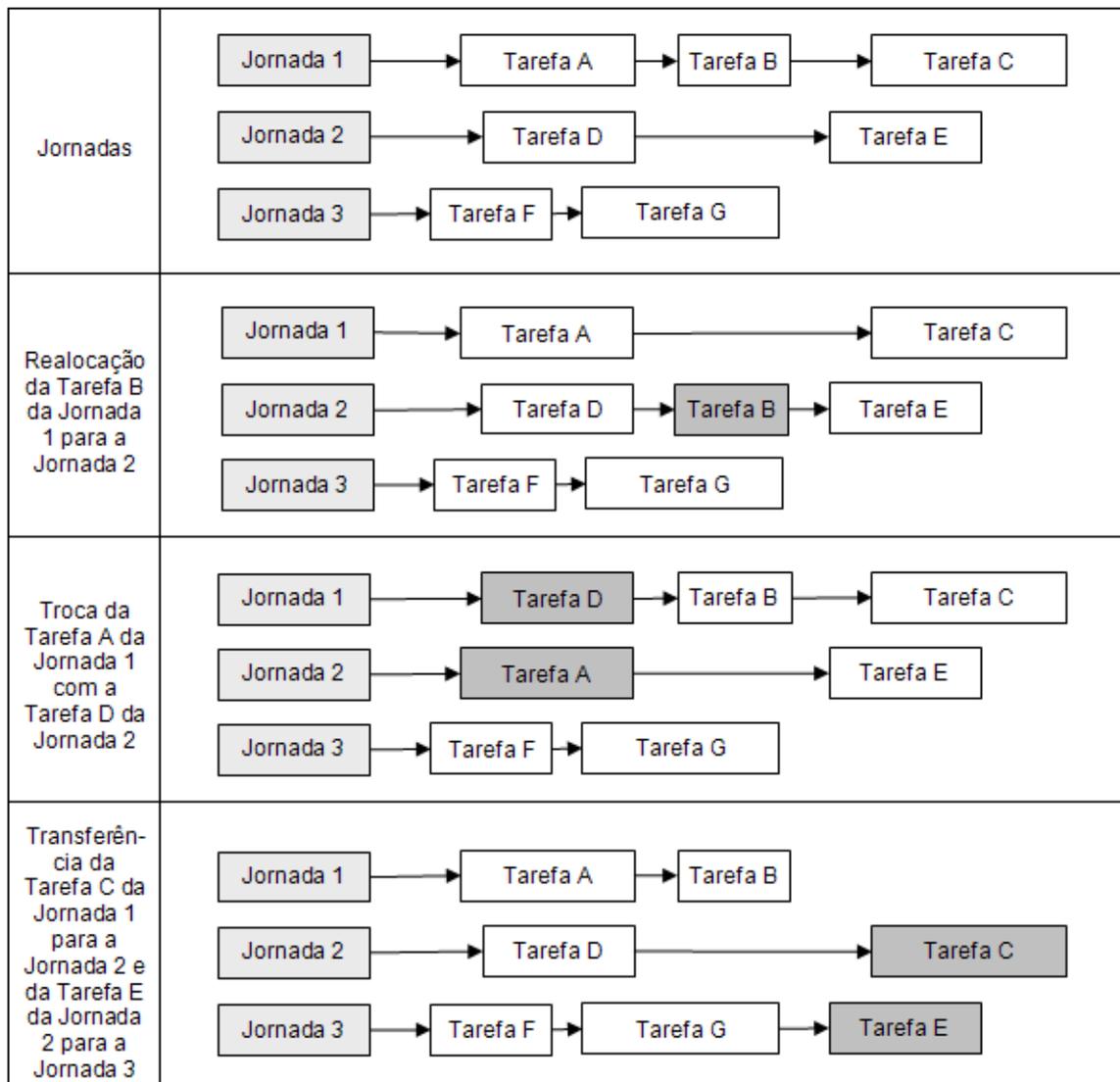


Figura 3.5. Estruturas de vizinhança do PPT.

### 3.5.3 Função de Avaliação

Uma solução  $s^t$  para o Problema de Programação de Tripulações é avaliada penalizando-se os seus atributos indesejáveis. Portanto, o PPT nesta abordagem é tratado como um problema de minimização, cujo custo associado é calculado pela função  $f^t$ , conforme apresentado a seguir pela expressão (3.3).

$$\begin{aligned}
 f^t(s^t) = & \text{totalTripulacoes} \times \text{custoUsoTripulacao} \\
 & + \text{tempoHoraExtra} \times \text{custoHoraExtra} \\
 & + \text{totalExcessoDP} \times \text{pesoInviabilidade} \\
 & + \text{tempoSobreposicao} \times \text{pesoInviabilidade}
 \end{aligned} \tag{3.3}$$

em que:

- (a) *totalTripulacoes* é o número de tripulações empregadas na programação;
- (b) *custoUsoTripulacao* é o custo associado a cada tripulação sendo utilizada;
- (c) *tempoHoraExtra* é o tempo total de horas extras, em minutos, considerando todas as tripulações da programação;
- (d) *custoHoraExtra* é o custo associado a cada minuto de hora extra realizada por uma tripulação;
- (e) *totalExcessoDP* é a quantidade de duplas pegadas da programação que supera um certo número admissível, previamente definido;
- (f) *tempoSobreposicao* é a soma, dada em minutos, de todos os intervalos de tempo em que há sobreposição de horários entre tarefas de uma mesma tripulação da programação. Destaca-se que, também será considerado sobreposição, o tempo faltante para que sejam completadas as 11:00 horas de intervalo entre o final de uma jornada e o início da mesma no dia seguinte;
- (g) *pesoInviabilidade* é o peso dado a cada inviabilidade presente na solução.

### 3.5.4 Procedimento de Construção das Tarefas

Para construção das tarefas cada bloco de veículo em uma solução para o PPV é considerado separadamente. Como as viagens em um bloco encontram-se ordenadas crescentemente pelos seus horários de início, o processo começa da primeira viagem do veículo no dia e sistematicamente vai percorrendo toda a sua jornada de operação, viagem por viagem. Neste procedimento, a cada iteração, dadas a viagem  $i$  atualmente sendo analisada e a tarefa  $t$  em formação, são executados os seguintes passos:

1. Se  $i$  é a primeira viagem do veículo, então a tarefa  $t$  inicia na garagem, cobrindo assim, o deslocamento fora de operação realizado da garagem para o ponto de início da viagem  $i$ ;
2. Se  $i$  é a última viagem do veículo, então a tarefa  $t$  é finalizada na garagem, após cobrir a viagem  $i$ , e não há mais tarefas a serem construídas. Caso contrário, se após a viagem  $i$  o veículo realiza a viagem  $j$  (que será a viagem a ser analisada na próxima iteração), tem-se que:
  - a) Se entre as viagens  $i$  e  $j$  não houver uma oportunidade de troca, ambas serão agrupadas na tarefa  $t$  e a folga acumulada da tarefa  $t$  será acrescida do tempo de ociosidade do veículo entre as viagens  $i$  e  $j$ . Neste caso, a tarefa  $t$  continua em processo de construção;
  - b) Se entre as viagens  $i$  e  $j$  o veículo retorna à garagem (ou seja, entre estas viagens houver uma dupla pegada), a tarefa  $t$  é finalizada na garagem após cobrir a viagem  $i$  e uma nova tarefa  $t'$  iniciará na garagem. Assim, a nova tarefa em formação será  $t'$  e esta cobrirá o deslocamento fora de operação realizado da garagem para o ponto de início da viagem  $j$ ;
  - c) Se entre as viagens  $i$  e  $j$  houver uma oportunidade de troca e, o veículo não realizar dupla pegada entre elas, a tarefa  $t$  cobre a viagem  $i$  e é finalizada no ponto de início da viagem  $j$ . Dessa forma, caso ocorra deslocamento fora de operação entre as viagens  $i$  e  $j$ , ele será coberto pela tarefa  $t$ . Em seguida, inicia-se uma nova tarefa,  $t'$ , no ponto de início da viagem  $j$  e esta passa a ser a tarefa em construção.

Conforme pode ser observado o procedimento de construção das tarefas contempla todos os deslocamentos realizados pelos veículos e engloba, em uma única tarefa, viagens que devem ser realizadas por uma mesma tripulação.

### 3.5.5 Procedimento de Geração da Solução Inicial

Antes de dar início ao processo de resolução do PPT, dada a programação de veículos, devem ser definidas as suas tarefas. Para tanto, foi utilizado o procedimento de construção das tarefas descrito anteriormente na subseção 3.5.4.

A programação de tripulações inicial é gerada por uma heurística construtiva gulosa (Algoritmo 6). O algoritmo proposto consiste na alocação, a cada passo, de uma nova tarefa na jornada de trabalho diária de alguma tripulação. Esta combinação (tarefa-jornada) deve ser a que implique no menor custo possível, determinado pela aplicação da função de avaliação apresentada anteriormente pela expressão (3.3).

Destaca-se que, durante a definição da programação, jornadas são criadas dinamicamente conforme demanda por mão-de-obra (linhas 13 e 14, no Algoritmo 6).

---

**Algoritmo 6:** Heurística para geração de uma solução inicial para o PPT.

---

```

1 Seja  $T$  o conjunto de tarefas obtidas pelo procedimento de construção de tarefas
  descrito na subseção 3.5.4, na página 31. Em  $T$ , tarefas de um mesmo veículo se
  encontram dispostas seqüencialmente e ordenadas crescentemente por horário de
  início
2 Seja  $J$  o conjunto de jornadas diárias de trabalho, inicialmente  $J$  possui uma
  única jornada sem qualquer tarefa
3 enquanto  $T \neq \emptyset$  faça
4    $t \leftarrow$  primeira tarefa de  $T$ 
5    $melhorDelta \leftarrow \infty$ , onde  $melhorDelta$  é a melhor variação ocorrida na
  função de avaliação ao se alocar a tarefa  $t$ 
6   para cada jornada  $j \in J$  faça
7      $custoAnterior \leftarrow$  custo da jornada  $j$  antes de receber a tarefa  $t$ 
8      $custoPosterior \leftarrow$  custo da jornada  $j$  caso a tarefa  $t$  passe a pertencer a
  ela
9     Calcule a variação de custo  $\Delta$  da jornada  $j$  após receber a tarefa  $t$ :
     $\Delta \leftarrow custoPosterior - custoAnterior$ 
10    se  $\Delta < melhorDelta$  então
11       $melhorDelta \leftarrow \Delta$ 
12       $melhorJornada \leftarrow info(j)$ , isto é, armazene uma informação que
    permita identificar a jornada que irá contribuir menos para o valor da
    função de avaliação ao receber a tarefa  $t$ 
13    se  $melhorJornada$  não possui tarefas então
14      Crie uma nova jornada sem tarefas e adicione ao conjunto  $J$ . (criação
    dinâmica de jornada, mantendo sempre uma jornada sem tarefas em  $J$ )
15    Aloque a tarefa  $t$  à jornada do conjunto  $J$  indicada por  $melhorJornada$ 
16     $fo \leftarrow fo + melhorDelta$ , ou seja, atualize o valor da função de avaliação
17    Remova  $t$  de  $T$ 
18 Retorne, como solução, o conjunto de jornadas  $J$ 

```

---

### 3.5.6 Algoritmo de Resolução

A heurística proposta para resolução do PPT utiliza as mesmas idéias apresentadas anteriormente para tratar o PPV. O seu pseudo-código é apresentado a seguir pelo Algoritmo 7. Trata-se de um procedimento fundamentado na metaheurística Busca Local Iterada em que, primeiramente, uma solução inicial é gerada pela heurística construtiva gulosa descrita na subseção 3.5.5. Com o objetivo de melhorar a qualidade das soluções obtidas, foram utilizados procedimentos de busca local baseados no Método de Descida

e no Método Randômico de Descida, são eles: *DescidaComRealocacao\_PPT* e *DescidaRandomicaComRealocacao\_PPT* que consideram durante as buscas a estrutura de vizinhança  $CN^{(R)}$  (realocação de tarefa), e *DescidaComTroca\_PPT* e *DescidaRandomicaComTroca\_PPT* que limitam suas pesquisas ao conjunto de soluções dado por  $CN^{(T)}$  (troca de tarefas).

---

**Algoritmo 7:** Algoritmo de resolução do PPT.

---

- 1 Seja  $T$  o tempo de processamento corrente do algoritmo e  $Tmax$  o tempo máximo admissível de processamento
  - 2 Seja  $nivel$  o nível de perturbação corrente
  - 3  $T \leftarrow 0$
  - 4  $nivel \leftarrow 1$
  - 5 Gere uma solução inicial  $s^t$  a partir da heurística construtiva gulosa descrita pelo Algoritmo 6, na página 33
  - 6  $s^t \leftarrow DescidaRandomicaComRealocacao\_PPT(s^t)$
  - 7  $s^t \leftarrow DescidaRandomicaComTroca\_PPT(s^t)$
  - 8  $s^t \leftarrow DescidaComRealocacao\_PPT(s^t)$
  - 9  $s^t \leftarrow DescidaComTroca\_PPT(s^t)$
  - 10  $s_*^t \leftarrow s^t$ , onde  $s_*^t$  é a melhor solução encontrada até então
  - 11 **enquanto**  $T < Tmax$  **faça**
  - 12      $s^t \leftarrow perturbacao(s_*^t, nivel)$
  - 13      $s^t \leftarrow DescidaRandomicaComRealocacao\_PPT(s^t)$
  - 14      $s^t \leftarrow DescidaRandomicaComTroca\_PPT(s^t)$
  - 15     **se**  $f^t(s^t) < f^t(s_*^t)$  **então**
  - 16          $s_*^t \leftarrow s^t$ , ou seja, atualize a melhor solução até então encontrada
  - 17          $nivel \leftarrow 1$
  - 18     **se**  $f^t(s^t) = f^t(s_*^t)$  **então**
  - 19          $s_*^t \leftarrow s^t$ , ou seja, atualize a melhor solução até então encontrada
  - 20          $nivel \leftarrow nivel + 1$
  - 21     **se**  $f^t(s^t) > f^t(s_*^t)$  **então**
  - 22          $nivel \leftarrow nivel + 1$
  - 23     **se**  $nivel > numNiveis$  **então**
  - 24          $nivel \leftarrow 1$ , onde  $numNiveis$  é o número máximo de níveis de perturbação
  - 25 Retorne a solução  $s_*^t$
- 

Destaca-se que nos métodos *DescidaRandomicaComRealocacao\_PPV* e *DescidaRandomicaComTroca\_PPV* a condição de parada é determinada pelo número de iterações sem melhoras na solução corrente, conforme expressão (3.4) a seguir:

$$itMax_{PPT} = (total\_tarefas \times (total\_jornadas - 1)) \times 0,6 \quad (3.4)$$

onde:

- (a)  $itMax_{PPT}$  é o número máximo de iterações sem melhoras na solução corrente;
- (b)  $total\_tarefas$  é o número de tarefas a serem realizadas;
- (c)  $total\_jornadas$  é o número de jornadas da programação.

Dessa forma, o número máximo de iterações sem melhoras,  $itMax_{PPT}$ , corresponderá a 60% do total de possíveis realocações de tarefas, considerando a programação de tripulações atual.

As perturbações, neste contexto, correspondem a movimentos de realocação, troca e realocação-troca envolvendo tripulações e tarefas aleatoriamente selecionadas na programação. Há cinco níveis de perturbação, são eles: *nível 1* - um movimento de realocação de tarefa, *nível 2* - um movimento de troca de tarefas, *nível 3* - dois movimentos de realocação de tarefas, *nível 4* - dois movimentos de troca de tarefas e *nível 5* - um movimento de realocação-troca de tarefas.

Uma nova solução para o PPT somente será aceita se ela possuir um custo menor ou igual ao custo da solução corrente. No primeiro caso, o nível de perturbação é reiniciado de forma a intensificar a busca na nova solução obtida. Nas demais situações o nível de perturbação é aumentado em um ou reiniciado (caso já se esteja no nível de perturbação mais alto).

## 3.6 Programação Independente de Veículos e Tripulações

Verifica-se que na abordagem seqüencial tradicional a solução do PPV impõe restrições adicionais à resolução do PPT. Este fato é claramente evidenciado na formação das tarefas, pois estas contemplam os deslocamentos fora de operação dos veículos e aglomeram viagens que obrigatoriamente deverão ser realizadas pela mesma tripulação, devido à impossibilidade dos veículos de efetivarem a troca entre algumas de suas viagens.

Na abordagem independente PPV e PPT são resolvidos separadamente, em qualquer ordem e de forma a não se relacionarem. Para tanto, a estratégia de resolução adotada é igual à apresentada anteriormente, na abordagem seqüencial, exceto pelo fato de que cada tarefa no PPT passa a corresponder a exatamente uma viagem. Além disso, supõe-se que todas as tarefas são realizadas por um mesmo veículo para que não haja a necessidade de contemplar trocas de veículos na programação de tripulações. Assim, são eliminados todos os vínculos entre as programações de veículos e tripulações e ambas passam a ser definidas a partir do conjunto de viagens.

Difícilmente a programação independente de veículos e tripulações gera soluções para PPV e PPT que sejam compatíveis e factíveis na prática. A principal finalidade desta metodologia é a determinação de uma melhor programação para as tripulações desconsiderando, para tanto, a programação dos veículos. Desta forma, uma comparação entre as soluções obtidas nessa abordagem e na seqüencial tradicional permite identificar os possíveis benefícios de se resolver PPV e PPT de forma integrada.

Segundo Freling et al. [2003], se ao comparar os custos das soluções obtidas nas abordagens seqüencial tradicional e independente houver uma grande diferença entre estes valores (o custo da solução independente é muito menor) espera-se que a qualidade da programação de tripulações seja melhorada significativamente quando esta for definida de forma integrada ao invés de seqüencial. Por outro lado, se os custos das soluções estão próximos, presume-se que a abordagem integrada não será promissora. Não havendo, portanto, a necessidade de aplicá-la.

## Capítulo 4

# Programação Integrada de Veículos e Tripulações

Apesar dos dois problemas, PPV e PPT, normalmente serem abordados separadamente, há uma relação de dependência entre eles. Os custos das programações podem ser inclusive antagônicos, ou seja, uma característica favorável a uma solução para o PPV nem sempre reflete de forma satisfatória no PPT e vice-versa. Assim, conclui-se que na abordagem seqüencial tradicional é priorizada a resolução do PPV uma vez que este é tratado em uma etapa anterior ao PPT.

Como no contexto do Sistema de Transporte Público de Ônibus Urbano comumente os gastos das empresas com tripulantes superam os gastos com veículos, presume-se que a metodologia de resolução integrada dos problemas poderá produzir resultados melhores do que aqueles gerados pela abordagem seqüencial. Além disso, destaca-se que o PPT é um problema mais restrito do que o PPV, por envolver limitações tanto de caráter operacional quanto trabalhista, o que justifica ainda mais a relevância da integração.

Portanto, a finalidade desse capítulo é apresentar estratégias para resolução do Problema de Programação Integrada de Veículos e Tripulações (PPVT) de maneira que, posteriormente, possa-se verificar a efetividade dessa abordagem comparando-a com a seqüencial tradicional.

Esse capítulo está organizado conforme descrito a seguir. Na seção 4.1 o Problema de Programação Integrada de Veículos e Tripulações é detalhado. Na seção 4.2 é apresentado um método iterativo em que as buscas por melhoras são alternadamente realizadas nos espaços de soluções do PPV e PPT. Por fim, na seção 4.3, um procedimento fundamentado na definição de uma nova estrutura (a viagem expandida) é proposto para resolução do PPVT.

## 4.1 Descrição do Problema de Programação Integrada de Veículos e Tripulações

O Problema de Programação Integrada de Veículos e Tripulações (PPVT) consiste em definir simultaneamente, as rotinas de operação dos veículos e as jornadas de trabalho das tripulações, de tal forma a minimizar os custos envolvidos e ao mesmo tempo viabilizar a execução de todas as viagens sob responsabilidade da empresa.

Destaca-se que os mesmos conceitos considerados para definir PPV e PPT, separadamente, são válidos no PPVT. Ou seja, “bloco de veículo”, “veículo com dupla pegada”, “viagem morta”, “tarefa”, “jornada com pegada simples”, “jornada com dupla pegada”, dentre outros, são termos que continuarão sendo usados na abordagem integrada e sem alteração de seus significados.

Nesse trabalho, na resolução do PPVT foram consideradas conjuntamente as restrições do PPV e PPT. Estas se encontram exatamente como foram descritas anteriormente no capítulo 3 e são reapresentadas a seguir.

*Restrições para a programação de veículos:*

- (a) Todas as viagens de responsabilidade de uma empresa, sem exceções, devem ser realizadas por algum veículo da frota;
- (b) Antes de se iniciar e após finalizar uma viagem, um veículo deve obrigatoriamente cumprir um tempo mínimo no terminal, que corresponde ao tempo necessário para o embarque e desembarque dos passageiros;
- (c) Um mesmo veículo somente poderá realizar duas viagens,  $i$  e  $j$ , consecutivamente, se o intervalo de tempo existente entre elas for suficiente para que o veículo desloque do ponto final da viagem  $i$  para o ponto inicial da viagem  $j$ , respeite os tempos destinados para embarque e desembarque de passageiros e inicie a viagem  $j$  sem atrasos. Caso contrário, ocorrerá uma sobreposição de horários entre as viagens;
- (d) Um veículo não deve ficar mais de duas horas em um terminal esperando para executar a sua próxima viagem. Se o tempo de espera exceder esse limite, o veículo deverá se deslocar para a garagem. Diz-se, nesse último caso, que o veículo está realizando uma dupla pegada;
- (e) O número de duplas pegadas da programação não pode exceder 60% do tamanho da frota;

- (f) Todo veículo deve iniciar e terminar a sua jornada de trabalho na garagem;
- (g) Todo veículo deve permanecer, no mínimo, trinta minutos consecutivos na garagem, por dia, para que ocorra a sua manutenção e limpeza. Este tempo pode ser cumprido quando o veículo pára na garagem durante uma dupla pegada ou mesmo depois de completada toda a sua jornada operacional diária.

*Restrições para a programação de tripulações:*

- (a) Uma tripulação não pode executar duas tarefas simultaneamente, isto é, em nenhuma jornada poderá ocorrer coincidências de horário entre quaisquer de suas tarefas;
- (b) Uma tripulação somente poderá realizar a troca de veículos nas seguintes circunstâncias: nas situações em que os veículos envolvidos se encontram em um mesmo terminal no momento da troca, basta que a tripulação tenha um pequeno intervalo ocioso (cinco minutos no mínimo) para que ela abandone um veículo e desloque rapidamente para assumir o outro. Caso contrário, se os veículos envolvidos estão em terminais distintos, a troca somente poderá ocorrer no intervalo entre os turnos de uma jornada do tipo dupla pegada. Esta última condição se faz necessária devido ao fato de a distância entre os diferentes terminais ser grande. Vale ressaltar que um terminal é o local de disposição dos vários pontos de parada dos veículos e onde ocorre o embarque e desembarque de passageiros;
- (c) Uma tripulação pode extrapolar, no máximo, duas horas diárias de trabalho além da duração normal de sua jornada, sendo este tempo excedente considerado hora extra;
- (d) Toda jornada do tipo pegada simples deve ter uma folga mínima de 20 minutos, podendo esta ser fracionada em dois intervalos, tendo cada um, uma duração igual ou superior a 10 minutos;
- (e) O intervalo mínimo entre o final de uma jornada e o início da mesma no dia seguinte deverá ser de 11:00 horas;
- (f) O número de jornadas de trabalho do tipo dupla pegada não pode exceder a 20% do total de tripulações da programação.

Atenta-se ao fato de que, além da preocupação em manter as programações de veículos e de tripulações separadamente viáveis, é necessário que elas sejam mutuamente compatíveis para que a resolução do PPVT torna-se aplicável na prática.

O objetivo ao resolver o PPVT é obter programações em que o número de veículos e de tripulações seja minimizado. Além disso, devem-se reduzir os deslocamentos fora de operação realizados pelos veículos e as horas extras das tripulações.

## 4.2 Método de Solução Iterativa do PPV e PPT

Como primeiro passo para o estudo do PPVT, nessa seção será apresentada uma estratégia iterativa de resolução em que, ora a busca por melhoras se concentra especificamente no PPV e ora no PPT. Porém, para permitir a avaliação das programações dos veículos e das tripulações conjuntamente, durante o processo a compatibilidade entre elas é mantida.

Na subseção 4.2.1 é descrito como as soluções para PPV e PPT são representadas. A função usada para avaliar simultaneamente as programações de veículos e de tripulações é mostrada na subseção 4.2.2. O algoritmo de resolução do PPV e PPT iterativamente é apresentado na subseção 4.2.3.

### 4.2.1 Representação

Assim como na abordagem sequencial, nesta metodologia, a cada jornada de trabalho está associada uma lista de tarefas.

Conforme mencionado anteriormente, uma tarefa é a associação de viagens de um mesmo veículo entre as quais não há nenhuma oportunidade de troca e que devem ser obrigatoriamente realizadas por uma mesma tripulação. Portanto, conclui-se que a formação das tarefas é totalmente dependente de uma solução para o Problema de Programação de Veículos. Na Figura 4.1 é mostrado o particionamento de um bloco de veículo em três tarefas, supondo que para ocorrer a troca de tripulação é necessário um tempo maior ou igual a seis minutos. Nota-se que remoções e/ou inserções de viagens na jornada de trabalho do veículo provavelmente irão alterar as tarefas obtidas a partir dele. Por exemplo, se a viagem “149” for removida do bloco de veículo, a tarefa “Tarefa 2” será suprimida e surgirá uma nova tarefa que consistirá apenas das viagens “101” e “120”. Por outro lado, se uma viagem com os seguintes atributos (número da viagem: “90”, Ponto inicial: “3055”, Horário de Início: 14:28, Ponto Final: “3055”, Horário de Término: 15:09, Linha: “3055.1”) for inserida neste bloco de veículo, uma nova tarefa será criada entre as tarefas “Tarefa 1” e “Tarefa 2” e compreenderá apenas a viagem “90”.

Dessa forma, com o intuito de preservar a compatibilidade entre as programações de veículos e de tripulações sendo obtidas simultaneamente, foi associado a cada bloco de veículo, além do seu conjunto de viagens, uma lista com as tarefas obtidas a partir dele.

	Número da Viagem	Ponto Inicial	Horário de Início (hh:mm)	Ponto Final	Horário de Término (hh:mm)	Viagem Morta	Tempo de Terminal	Linha
Tarefa 1	-	Garagem	12:22	3055	12:40	18	0	-
	87	3055	12:40	3055	14:22	0	53	3055.1
Tarefa 2	101	3055	15:15	3055	17:03	0	0	3055.1
	120	3055	17:03	3055	18:20	0	4	3055.1
	149	3055	18:24	3055	20:04	0	8	3055.1
Tarefa 3	183	3055	20:12	3055	21:31	0	0	3055.1
	-	3055	21:31	Garagem	21:49	18	0	-

Figura 4.1. Bloco de veículo particionado em tarefas.

Na Figura 4.2 está representado o bloco de veículo dado na Figura 4.1. Com esta nova representação as tarefas que surgem ou são suprimidas após modificações (remoções e/ou inserções de viagens) no bloco de um veículo serão determinadas da seguinte forma: a lista de viagens do veículo é percorrida e suas tarefas são reconstruídas. Em seguida, compara-se o seu conjunto de tarefas anterior com o novo conjunto encontrado, identificando assim, as alterações ocorridas.

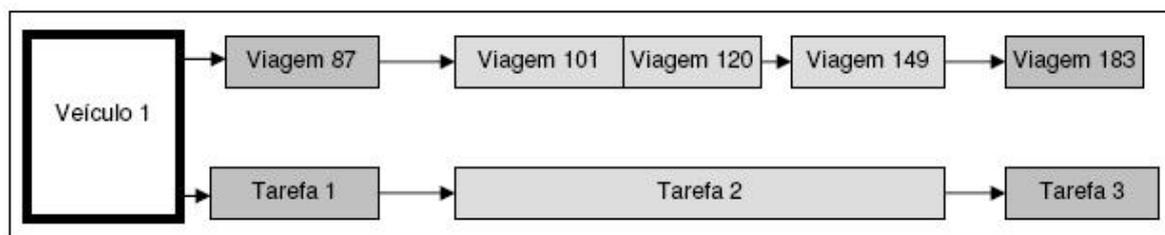


Figura 4.2. Representação de um bloco de veículo.

### 4.2.2 Função de Avaliação

Durante a resolução iterativa as soluções do PPV ( $s^v$ ) e PPT ( $s^t$ ) são consideradas conjuntamente. Portanto, para avaliar as programações obtidas foi definida a função  $f^{vt}$  dada a seguir pela expressão (4.1).

$$f^{vt}(s^v, s^t) = f^v(s^v) + f^t(s^t) \quad (4.1)$$

Nessa expressão, os valores retornados pelas funções  $f^v$  e  $f^t$  estão sendo somados e estes são calculados exatamente como descrito anteriormente, na abordagem sequencial, pelas expressões (3.1) e (3.3).

### 4.2.3 Algoritmo de Resolução

A seguir no Algoritmo 8 é apresentado o pseudo-código da heurística desenvolvida para resolver iterativamente PPV e PPT. Destaca-se que este algoritmo também é fundamentado na metaheurística Busca Local Iterada. Além disso, as estruturas de vizinhança e os procedimentos de busca local utilizados, são os mesmos que foram descritos anteriormente para tratar PPV e PPT separadamente.

As soluções iniciais para PPV e PPT são geradas pelos mesmos procedimentos descritos na abordagem seqüencial (subseções 3.3.4 e 3.5.5, nas páginas 22 e 32). Em seguida, antes de se iniciar as iterações do algoritmo, procura-se melhorar estas soluções a partir de buscas locais realizadas considerando primeiramente as estruturas de vizinhança do PPT, e depois as do PPV (passos de 9 a 12 no Algoritmo 8). Salienta-se que nem sempre se alcança um aprimoramento das soluções após esta etapa, pois como as buscas são realizadas independentemente (procuras por melhoras são realizadas ora no PPT, ora no PPV), pode ocorrer de melhoras na programação de veículos ocasionarem pioras na programação de tripulações e, ao final, ocorrer um aumento na função de custo considerando ambas as programações.

Foram definidos quatro níveis de perturbação, consecutivamente tem-se: *nível 1* - uma realocação de tarefa, *nível 2* - uma realocação de viagem, *nível 3* - uma troca de tarefas e *nível 4* - uma troca de viagens.

É importante ressaltar que sempre que a programação de veículos for modificada é necessário compatibilizar a programação de tripulações com esta nova solução (função *compatibilizar*( $s^v$ ,  $s_*^t$ ) no Algoritmo 8 e cujo pseudo-código é dado pelo Algoritmo 9), isto é, atualizações serão realizadas de duas maneiras:

1. Remoção na solução do PPT de tarefas que deixaram de existir, devido mudanças no PPV;
2. Inserção na solução do PPT de tarefas que surgiram, devido mudanças no PPV. Neste caso, esta inserção é realizada de forma “gulosa”, isto é, cada tarefa é alocada na jornada da tripulação que ao recebê-la irá contribuir menos com o custo da programação de tripulações corrente.

Assim sendo, sempre que a perturbação for realizada sobre a programação de veículos, executam-se buscas locais nas soluções do PPV e do PPT, pois ambas serão modificadas. Caso contrário, restringem-se as buscas à programação de tripulações.

Uma nova solução (programações de veículos e tripulações) somente será aceita se ela possuir um custo menor ou igual ao custo da solução corrente. No primeiro caso, o nível de perturbação é reiniciado de forma a intensificar a busca na nova solução

---

**Algoritmo 8:** Algoritmo de resolução iterativa do PPV e PPT.

---

```

1  Seja  $T$  o tempo de processamento corrente do algoritmo e  $Tmax$  o tempo
   máximo admissível de processamento
2  Seja  $nivel$  o nível de perturbação corrente
3   $T \leftarrow 0$ 
4   $nivel \leftarrow 1$ 
5  Gere uma solução inicial  $s^v$ , para o PPV, a partir da heurística construtiva
   gulosa descrita pelo Algoritmo 4, na página 23
6  Dado  $s^v$  gere uma solução  $s^t$  para o PPT. Esta solução inicial é obtida a partir
   do método guloso apresentado pelo Algoritmo 6, na página 33
7   $s_*^v \leftarrow s^v$ , onde  $s_*^v$  é a melhor solução encontrada até então para o PPV
8   $s_*^t \leftarrow s^v$ , onde  $s_*^t$  é a melhor solução encontrada até então para o PPT
9   $s^t \leftarrow DescidaComRealocacao\_PPT(s^t)$ 
10  $s^t \leftarrow DescidaComTroca\_PPT(s^t)$ 
11  $s^v \leftarrow DescidaComRealocacao\_PPV(s^v)$ 
12  $s^v \leftarrow DescidaComTroca\_PPV(s^v)$ 
13  $s^t \leftarrow compatibilizar(s^v, s^t)$ , isto é, atualiza a solução para o PPT a partir das
   modificações realizadas no PPV
14 enquanto  $T < Tmax$  faça
15     se  $f^{vt}(s^v, s^t) \leq f^{vt}(s_*^v, s_*^t)$  então
16          $s_*^v \leftarrow s^v$ 
17          $s_*^t \leftarrow s^t$ 
18         se  $f^{vt}(s^v, s^t) < f^{vt}(s_*^v, s_*^t)$  então
19              $nivel \leftarrow 1$ 
20         senão
21              $nivel \leftarrow nivel + 1$ 
22     senão
23          $nivel \leftarrow nivel + 1$ 
24     se  $nivel > numNiveis$  então
25          $nivel \leftarrow 1$ , onde  $numNiveis$  é o número máximo de níveis de perturbação
26      $s^v \leftarrow perturbacao(s_*^v, nivel)$  ou  $s^t \leftarrow perturbacao(s_*^t, nivel)$ , conforme o nível
   de perturbação modifique PPV ou PPT
27     se a perturbação realizada envolve apenas realocação ou troca de tarefas no
   PPT então
28          $s^t \leftarrow DescidaRandomicaComRealocacao\_PPT(s^t)$ 
29          $s^t \leftarrow DescidaRandomicaComTroca\_PPT(s^t)$ 
30     senão
31          $s^v \leftarrow DescidaRandomicaComRealocacao\_PPV(s^v)$ 
32          $s^v \leftarrow DescidaRandomicaComTroca\_PPV(s^v)$ 
33          $s^t \leftarrow compatibilizar(s^v, s_*^t)$ , isto é, atualiza a solução para o PPT a
   partir das modificações realizadas no PPV
34          $s^t \leftarrow DescidaRandomicaComRealocacao\_PPT(s^t)$ 
35          $s^t \leftarrow DescidaRandomicaComTroca\_PPT(s^t)$ 
36 Retorne as soluções  $s_*^v$  e  $s_*^t$ 

```

---

---

**Algoritmo 9:** Função compatibilizar( $s_2^v, s_1^t$ ).

---

- 1 **Entrada:** Uma solução  $s_2^v$  para o PPV obtida a partir de modificações em  $s_1^v$   
Uma solução  $s_1^t$  para o PPT que foi definida a partir da solução  $s_1^v$   
para o PPV
  - 2 **Saída:** Uma solução  $s_2^t$  para o PPT compatível com a solução  $s_2^v$  para o PPV
  - 3  $s_2^t \leftarrow s_1^t$
  - 4 Seja *tarefasCriadas* uma lista com todas as tarefas que surgiram a partir das mudanças no PPV. Inicialmente esta lista está vazia
  - 5 Seja  $F$  a frota de veículos
  - 6 **para** cada veículo  $v \in F$  **faça**
  - 7     **se**  $v$  foi modificado na transição de  $s_1^v$  para  $s_2^v$  **então**
  - 8     |     Percorra a lista de viagens de  $v$  reconstruindo as suas tarefas e atualize  
as listas *tarefasSuprimidas* e *tarefasCriadas*. Sendo *tarefasSuprimidas* o  
conjunto de tarefas que deixaram de existir em  $v$  devido às mudanças
  - 9     |     **para** cada tarefa  $t \in$  *tarefasSuprimidas* **faça**
  - 10     |     |      $j \leftarrow$  *retornarJornada\_Tarefa*( $s_2^t, t$ ), isto é, encontre a jornada  $j$  da  
solução  $s_2^t$  que executa a tarefa  $t$
  - 11     |     |     Remova a tarefa  $t$  da jornada  $j$  em  $s_2^t$
  - 12 **para** cada tarefa  $t \in$  *tarefasCriadas* **faça**
  - 13     |     *insercao\_Gulosa*( $s_2^t, t$ ), isto é, alocar a tarefa  $t$  à jornada do tripulante que  
ao recebê-la irá contribuir menos com o custo da programação
  - 14 Retorne a solução  $s_2^t$
- 

obtida. Nas demais situações o nível de perturbação é aumentado em um ou reiniciado (caso já se esteja no nível de perturbação mais alto).

### 4.3 Método de Solução do PPVT

Nesta seção é proposta uma metodologia de resolução do PPVT com uma idéia de integração mais forte do que a apresentada anteriormente. Em tal estratégia não há mais a formação explícita de tarefas para definição de uma solução para o PPT, porém, é sugerida a criação de uma nova estrutura denominada viagem expandida e, tanto a programação de veículos quanto a programação de tripulações, passam a ser determinadas a partir dela.

Essa seção está organizada como segue. Uma descrição de como o problema é representado é dada na subseção 4.3.1. Na subseção 4.3.2 são apresentadas as estruturas de vizinhança desenvolvidas para explorar o espaço de busca. Descreve-se na subseção 4.3.4 o procedimento de geração da solução inicial. Por fim, na subseção 4.3.5, é apresentado o algoritmo proposto para resolver o problema.

### 4.3.1 Representação

O elemento básico manipulado nesta estratégia é a viagem expandida. Esta corresponde a uma viagem convencional com propriedades adicionais que definem como ela deve ser considerada no PPT, dada a sua disposição no PPV. Assim, além dos atributos normais de uma viagem (número da viagem, horário de início, ponto inicial, horário de término, ponto final, linha, tempo de embarque e tempo de desembarque) a viagem expandida agrega as seguintes informações: identificador do veículo, horário de início expandido, ponto inicial expandido, horário de término expandido, ponto final expandido, dependência à esquerda e dependência à direita.

Os atributos exclusivos das viagens expandidas consideram os deslocamentos fora de operação dos veículos e mantêm um vínculo entre as viagens que obrigatoriamente deverão ser realizadas pela mesma tripulação (devido à impossibilidade dos veículos de efetivarem a troca entre elas). As regras gerais para definição destes atributos são descritas a seguir:

1. Não é entre quaisquer viagens que uma tripulação poderá realizar a troca de veículos, pois esta operação requer um tempo. Dessa forma, cada viagem expandida deve indicar o veículo que a executa (identificador do veículo);
2. A tripulação responsável pela viagem realizada por um veículo logo após a sua saída da garagem, também deverá conduzi-lo em seu deslocamento fora de operação da garagem para o ponto de início da viagem. Assim, o horário de início expandido, da viagem, será igual ao seu horário de início decrescido do tempo necessário para posicionamento do veículo e, o seu ponto inicial expandido, será a garagem;
3. A tripulação responsável pela viagem realizada por um veículo imediatamente antes de seu retorno à garagem, também deverá conduzi-lo em seu deslocamento fora de operação do ponto de fim da viagem para a garagem. Portanto, o horário de término expandido, da viagem, será igual ao seu horário de término acrescido do tempo necessário para retornar com o veículo para a garagem e, o seu ponto final expandido, será a garagem;
4. Sempre que houver troca de pontos entre duas viagens consecutivas, de um mesmo veículo, a tripulação que executar a primeira viagem deverá posicionar o veículo no ponto de início da segunda. Logo, o horário de término expandido, da primeira viagem, será igual ao seu horário de término acrescido do tempo de deslocamento para reposicionamento do veículo e, o seu ponto final expandido, corresponderá ao ponto inicial da segunda viagem;

5. Sempre que entre duas viagens consecutivas de um veículo não houver tempo suficiente para a troca de tripulação, a dependência à direita da primeira viagem será a segunda e a dependência à esquerda da segunda viagem será a primeira. Isto é, as dependências à esquerda e à direita de uma viagem expandida determinam, respectivamente, quais viagens deverão imediatamente precedê-la e sucedê-la em uma mesma jornada no PPT.

Destaca-se que, a princípio, para uma viagem expandida tem-se que os atributos horário de início expandido, ponto inicial expandido, horário de término expandido e ponto final expandido assumem os mesmos valores, respectivamente, dos seus atributos horário de início, ponto inicial, horário de término e ponto final e, é considerado que não há dependências à esquerda e à direita. Assim, será a aplicação das regras acima que modificará estes valores conforme a disposição da viagem expandida no bloco do veículo que a executa.

A seguir, na Figura 4.3, é mostrado como seria o particionamento, em tarefas, do primeiro bloco de veículo de uma solução para o PPV. Em seguida, na Figura 4.4, são apresentadas as viagens expandidas definidas a partir deste mesmo bloco de veículo. Considera-se, nestes exemplos, que para ocorrer a troca de tripulação é necessário um tempo maior ou igual a seis minutos e que os tempos destinados para embarque e desembarque de passageiros são nulos para todas as viagens.

Bloco de Veículo - 1								
	Número da Viagem	Ponto Inicial	Horário de Início (hh:mm)	Ponto Final	Horário de Término (hh:mm)	Viagem Morta	Tempo de Terminal	Linha
Tarefa 1	-	Garagem	12:22	3055	12:40	18	0	-
	87	3055	12:40	3055	14:22	0	53	3055.1
Tarefa 2	101	3055	15:15	3055	17:03	0	0	3055.1
	120	3055	17:03	3055	18:20	0	4	3055.1
	149	3055	18:24	3055	20:04	0	8	3055.1
Tarefa 3	183	3055	20:12	3055	21:31	0	0	3055.1
	-	3055	21:31	2200	21:36	5	4	-
	210	2200	21:40	2200	22:25	0	0	2200.1
	-	2200	22:25	Garagem	22:45	20	0	-

**Figura 4.3.** Bloco de Veículo - 1 e o seu particionamento em tarefas.

Observa-se que a “Viagem expandida 87” deverá cobrir a viagem morta realizada pelo veículo ao sair da garagem (regra 2). Logo, o seu ponto inicial expandido é a garagem e o seu horário de início expandido antecede em 18 minutos o seu horário de início. Destaca-se que esta viagem não possui dependências à esquerda e à direita e ela

Viagem expandida 87	
Ponto inicial: 3055	Ponto inicial expandido: Garagem
Horário de início: 12:40 horas	Horário de início expandido: 12:22 horas
Ponto final: 3055	Ponto final expandido: 3055
Horário de término: 14:22 horas	Horário de término expandido: 14:22 horas
Linha: 3055.1	Dependência à esquerda: não há
Tempo de embarque: 0	Dependência à direita: não há
Tempo de desembarque: 0	Identificador do veículo: 1
Viagem expandida 101	
Ponto inicial: 3055	Ponto inicial expandido: 3055
Horário de início: 15:15 horas	Horário de início expandido: 15:15 horas
Ponto final: 3055	Ponto final expandido: 3055
Horário de término: 17:03 horas	Horário de término expandido: 17:03 horas
Linha: 3055.1	Dependência à esquerda: não há
Tempo de embarque: 0	Dependência à direita: Viagem 120
Tempo de desembarque: 0	Identificador do veículo: 1
Viagem expandida 120	
Ponto inicial: 3055	Ponto inicial expandido: 3055
Horário de início: 17:03 horas	Horário de início expandido: 17:03 horas
Ponto final: 3055	Ponto final expandido: 3055
Horário de término: 18:20 horas	Horário de término expandido: 18:20 horas
Linha: 3055.1	Dependência à esquerda: Viagem 101
Tempo de embarque: 0	Dependência à direita: Viagem 149
Tempo de desembarque: 0	Identificador do veículo: 1
Viagem expandida 149	
Ponto inicial: 3055	Ponto inicial expandido: 3055
Horário de início: 18:24 horas	Horário de início expandido: 18:24 horas
Ponto final: 3055	Ponto final expandido: 3055
Horário de término: 20:04 horas	Horário de término expandido: 20:04 horas
Linha: 3055.1	Dependência à esquerda: Viagem 120
Tempo de embarque: 0	Dependência à direita: não há
Tempo de desembarque: 0	Identificador do veículo: 1
Viagem expandida 183	
Ponto inicial: 3055	Ponto inicial expandido: 3055
Horário de início: 20:12 horas	Horário de início expandido: 20:12 horas
Ponto final: 3055	Ponto final expandido: 2200
Horário de término: 21:31 horas	Horário de término expandido: 21:36 horas
Linha: 3055.1	Dependência à esquerda: não há
Tempo de embarque: 0	Dependência à direita: Viagem 210
Tempo de desembarque: 0	Identificador do veículo: 1
Viagem expandida 210	
Ponto inicial: 2200	Ponto inicial expandido: 2200
Horário de início: 21:40 horas	Horário de início expandido: 21:40 horas
Ponto final: 2200	Ponto final expandido: Garagem
Horário de término: 22:25 horas	Horário de término expandido: 22:45 horas
Linha: 2200.1	Dependência à esquerda: Viagem 183
Tempo de embarque: 0	Dependência à direita: não há
Tempo de desembarque: 0	Identificador do veículo: 1

Figura 4.4. Viagens expandidas obtidas do Bloco de Veículo - 1 da Figura 4.3.

é realizada pelo bloco de veículo com identificador igual a 1, exatamente como descrito na Figura 4.4.

Entre o horário de término da viagem “101” e o horário de início da viagem “120” não há tempo ocioso do veículo, logo não será possível ocorrer a troca de tripulação entre elas e a dependência à direita, da viagem “101”, será a viagem “120” e a dependência

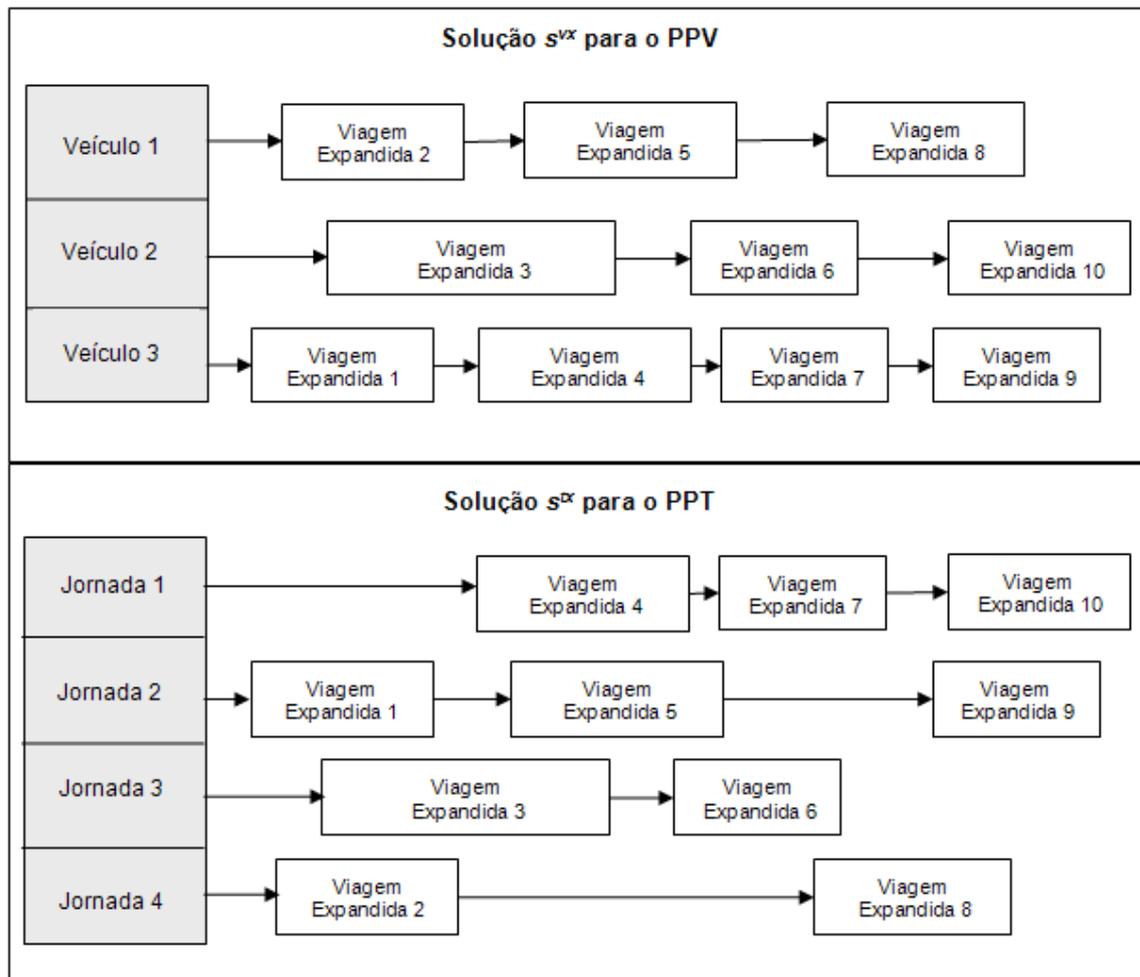
à esquerda, da viagem “120”, será a viagem “101” (regra 5). O mesmo ocorrerá entre as viagens “120” e “149”, pois o intervalo ocioso do veículo entre elas é menor que o tempo mínimo necessário para a troca da tripulação. Dessa forma, percebe-se que será criado um vínculo entre as viagens “101”, “120” e “149” que, obrigatoriamente, deverão ser realizadas pela mesma tripulação (definição implícita da tarefa “Tarefa 2” da Figura 4.3).

No bloco de veículo sendo considerado, as viagens “183” e “210” são realizadas consecutivamente. Porém, como o ponto final da primeira viagem (ponto “3055”) é diferente do ponto inicial da segunda (ponto “2200”), existe um deslocamento fora de operação do veículo entre elas (viagem morta de 5 minutos). Assim, o horário de término expandido da viagem “183” será o seu horário de término normal incrementado do tempo de viagem morta do veículo entre os pontos “3055” e “2200” e, o seu ponto final expandido será o ponto “2200” (regra 4). Como o tempo do veículo parado no terminal, entre as viagens “183” e “210”, também é insuficiente para a troca de tripulação, estabelece-se uma relação de dependência entre elas (definição implícita da tarefa “Tarefa 3” da Figura 4.3). Além disso, uma vez que após a viagem “210” o veículo retornará para a garagem, o horário de término expandido da viagem “210” corresponderá ao seu horário de término aumentado do tempo de deslocamento do veículo, do ponto “2200” para a garagem, que será de 20 minutos (regra 3). Dessa forma, o ponto final expandido desta viagem será a garagem.

Pelo exposto, nota-se que os novos atributos das viagens permitem contemplar as mesmas condições asseguradas ao se formar tarefas na resolução do PPT. Portanto, nesta abordagem, tanto os blocos dos veículos quanto as jornadas de trabalho das tripulações serão definidos a partir das viagens expandidas.

A seguir, na Figura 4.5, está ilustrada uma solução para o PPVT com definição das viagens expandidas. Esta solução é composta pela programação dos veículos ( $s^{vx}$ ) e pela programação das tripulações ( $s^{tx}$ ), sendo estas mutuamente compatíveis.

Pela Figura 4.5 observa-se que há dez viagens programadas. Estas estão alocadas a três veículos e empregará a mão-de-obra de quatro tripulações. A “Jornada 1” cobrirá as viagens “Viagem Expandida 4” e “Viagem Expandida 7” do “Veículo 3”. Em seguida, a tripulação responsável por esta jornada trocará de veículo e a “Viagem Expandida 10”, do “Veículo 2”, será executada. A “Jornada 2” iniciará suas atividades com a “Viagem Expandida 1”, do “Veículo 3”, posteriormente será realizada a “Viagem Expandida 5”, do “Veículo 1”, e por fim retorna-se ao “Veículo 3” para executar a “Viagem Expandida 9”. A “Jornada 3” será responsável pelas viagens “Viagem Expandida 3” e “Viagem Expandida 6”, ambas do “Veículo 2”, e a “Jornada 4” pelas viagens “Viagem Expandida 2” e “Viagem Expandida 8” do “Veículo 1”.



**Figura 4.5.** Representação de uma solução para o PPVT com definição das viagens expandidas.

### 4.3.2 Estruturas de Vizinhança

Para exploração do espaço de soluções do PPVT foram utilizados seis tipos de movimentos. Estes e suas respectivas estruturas de vizinhança são: realocação PPV ( $VN_x^{(R)}$ ), realocação PPT ( $CN_x^{(R)}$ ), troca PPV ( $VN_x^{(T)}$ ), troca PPT ( $CN_x^{(T)}$ ), super realocação ( $VCN_x^{(R)}$ ) e super troca ( $VCN_x^{(T)}$ ).

Os movimentos de realocação manipulam uma única viagem expandida modificando o seu veículo e/ou a sua tripulação:

- Realocação PPV: Modifica o veículo que realizará a viagem;

- Realocação PPT: Modifica a tripulação responsável pela viagem;
- Super realocação: Modifica simultaneamente o veículo e a tripulação alocados à viagem.

Os movimentos de troca manipulam duas viagens expandidas permutando os seus veículos e/ou as suas tripulações:

- Troca PPV: Permuta o veículo de uma viagem com o veículo de outra viagem;
- Troca PPT: Permuta a tripulação de uma viagem com a tripulação de outra viagem;
- Super troca: Permuta simultaneamente o veículo e a tripulação alocados a uma viagem com o veículo e a tripulação alocados a outra viagem.

Conforme exposto anteriormente, uma viagem expandida corresponde a uma viagem convencional com atributos adicionais que definem como ela deve ser considerada no PPT, dada a sua disposição no PPV. Portanto, é importante destacar que movimentos que alteram a programação de veículos podem acarretar mudanças nos atributos exclusivos de algumas viagens expandidas. Assim, sempre que uma viagem expandida é transferida de um veículo para outro, os seus atributos exclusivos, bem como os daquelas viagens que imediatamente a precedem e sucedem, no veículo antigo e no novo veículo, poderão estar sendo modificados. Salienta-se que essas mudanças afetarão a solução do PPT e, portanto, é necessário calcular as variações de custo ocorridas nas jornadas das tripulações que realizam as viagens expandidas que foram alteradas.

Nas figuras de 4.7 a 4.12 estão ilustrados os movimentos considerados no PPVT, estando estes aplicados a uma solução dada na Figura 4.6. Nestas ilustrações as viagens que participam diretamente dos movimentos estão destacadas em cinza e as hachuradas são aquelas, cujos atributos, provavelmente foram modificados em decorrência dos movimentos.

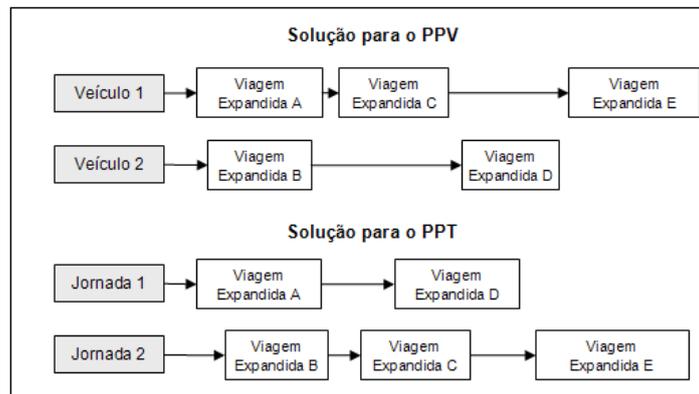


Figura 4.6. Solução para o PPVT utilizada para exemplificar as estruturas de vizinhança.

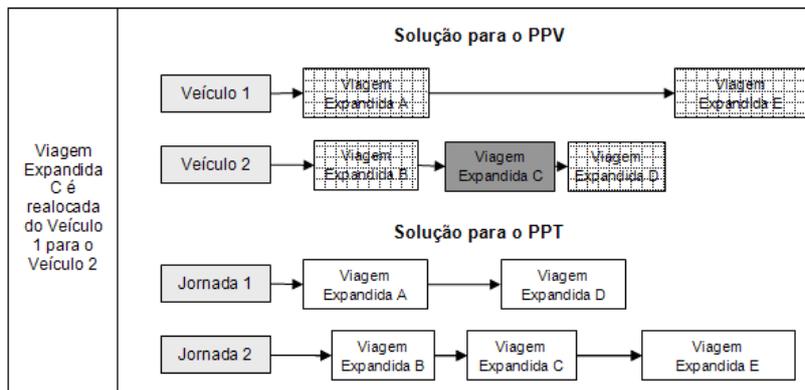


Figura 4.7. Exemplo do movimento realocação PPV.

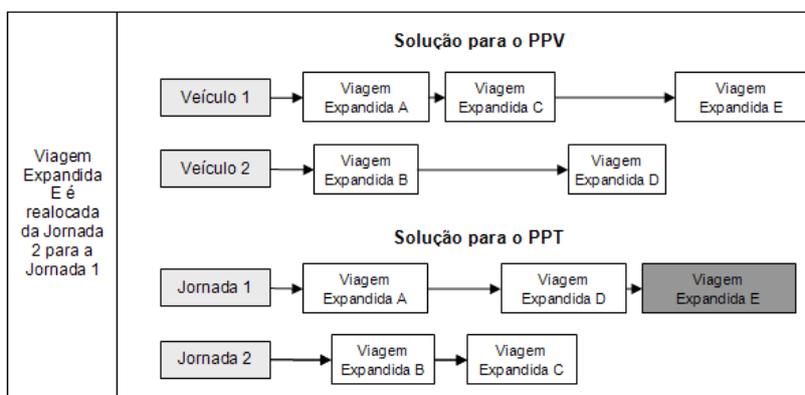


Figura 4.8. Exemplo do movimento realocação PPT.

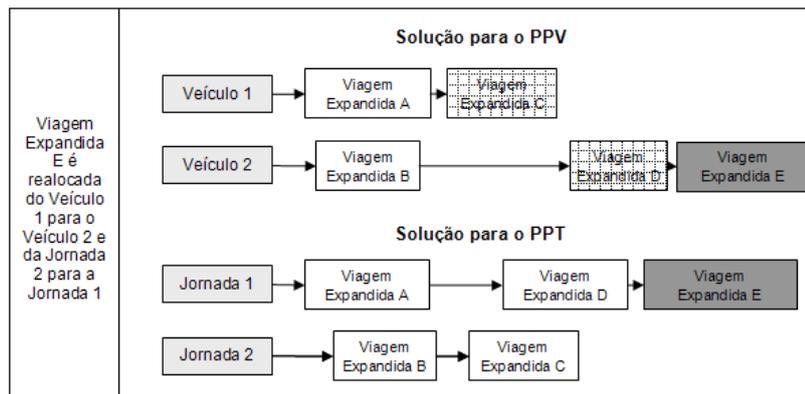


Figura 4.9. Exemplo do movimento super realocação.

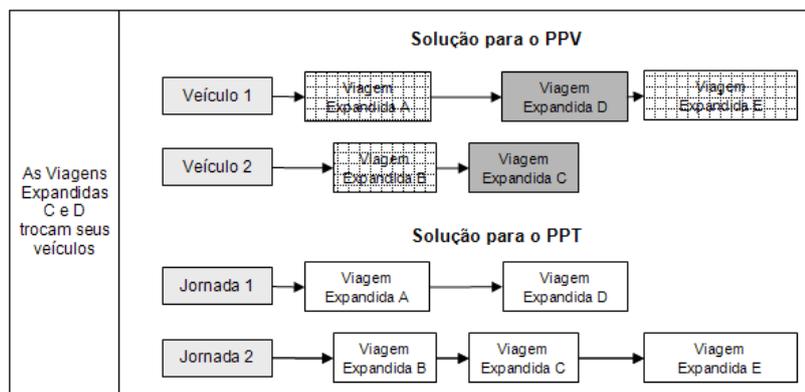


Figura 4.10. Exemplo do movimento troca PPV.

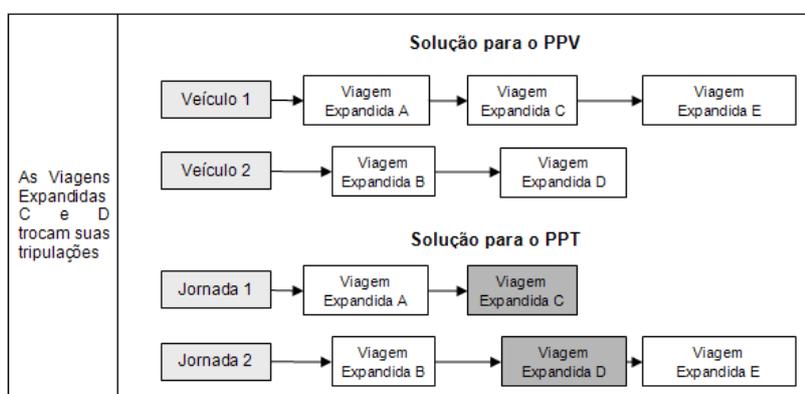


Figura 4.11. Exemplo do movimento troca PPT.

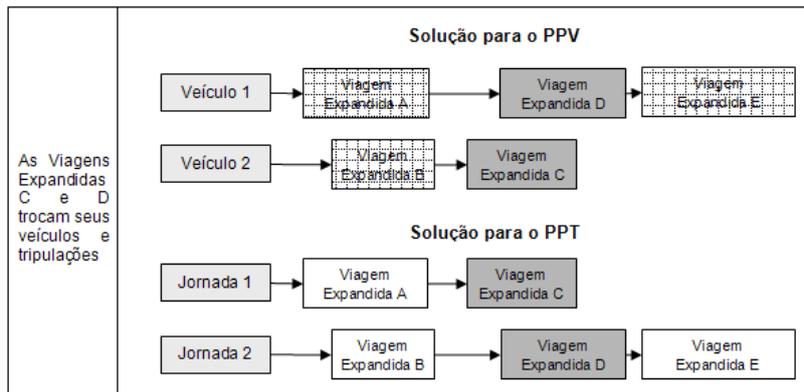


Figura 4.12. Exemplo do movimento super troca.

### 4.3.3 Função de Avaliação

Como na abordagem integrada a menor porção de trabalho que pode ser atribuída a uma tripulação é a viagem expandida, e não mais a tarefa, surge uma nova possibilidade de construção de jornadas inviáveis no PPT. Trata-se das situações em que ocorre a separação (alocação em diferentes jornadas) das viagens que implicitamente estariam compreendidas em uma mesma tarefa, conforme a especificação de suas dependências à esquerda e à direita.

Portanto, para avaliar as programações obtidas, foi definida a função  $f^{vtx}$  dada a seguir pela expressão (4.2). Nessa expressão, os valores retornados pelas funções  $f^v$  e  $f^t$  estão sendo somados e estes são calculados conforme descrito anteriormente, na abordagem seqüencial. Porém, ao invés de viagens e tarefas, estão sendo consideradas as viagens expandidas, ou seja, as soluções  $s^{vx}$  e  $s^{tx}$ .

$$f^{vtx}(s^{vx}, s^{tx}) = f^v(s^{vx}) + f^t(s^{tx}) + quebrasTarefas \times pesoInviabilidade \quad (4.2)$$

em que:

- (a) *quebrasTarefas* é o número de vezes que, na solução do PPT, uma viagem expandida encontra-se separada da viagem com a qual mantém uma dependência à esquerda ou à direita;
- (b) *pesoInviabilidade* é o peso dado a cada inviabilidade presente na solução.

### 4.3.4 Procedimento de Geração da Solução Inicial

A solução inicial para o PPVT é obtida por uma heurística construtiva gulosa, em que as programações dos veículos e das tripulações vão sendo definidas simultaneamente, a cada iteração do processo de construção.

No algoritmo proposto inicia-se com uma tripulação (um motorista e um cobrador), uma frota formada por um único veículo e um conjunto de viagens a serem realizadas (dispostas ordenadas crescentemente pelo horário de início). A cada passo, remove-se a primeira viagem do conjunto e são especificados o veículo e a tripulação que irão realizá-la. Essa combinação deve ser a que implique no menor custo possível determinado pela aplicação da função de avaliação descrita anteriormente pela expressão (4.2). Portanto, a cada iteração são avaliadas todas as possibilidades de alocação da viagem, considerando conjuntamente a mão-de-obra e a frota disponíveis. À medida que a solução vai sendo gerada novos veículos e tripulações são criados dinamicamente, de tal maneira que, sempre haja um veículo e uma tripulação cujas jornadas estão vazias. Esse processo é repetido até que todas as viagens estejam alocadas.

O pseudo-código da heurística construtiva gulosa utilizada no PPVT é descrita a seguir no Algoritmo 10.

### 4.3.5 Algoritmo de Resolução

A seguir é apresentado o pseudo-código do algoritmo desenvolvido para resolver o PPVT (Algoritmo 11).

Neste algoritmo foram utilizados seis procedimentos de busca local baseados no Método Randômico de Descida, são eles:

1. *DescidaRandomicaComRealocacaoX\_PPV*: Que considera durante as buscas a estrutura de vizinhança  $VN_x^{(R)}$  (realocação PPV);
2. *DescidaRandomicaComRealocacaoX\_PPT*: Que considera durante as buscas a estrutura de vizinhança  $CN_x^{(R)}$  (realocação PPT);
3. *DescidaRandomicaComTrocaX\_PPV*: Que considera durante as buscas a estrutura de vizinhança  $VN_x^{(T)}$  (troca PPV);
4. *DescidaRandomicaComTrocaX\_PPT*: Que considera durante as buscas a estrutura de vizinhança  $CN_x^{(T)}$  (troca PPT);
5. *DescidaRandomicaComRealocacaoX*: Que considera durante as buscas as estruturas de vizinhança  $VN_x^{(R)}$ ,  $CN_x^{(R)}$  e  $VCN_x^{(R)}$ . Assim, a cada iteração do método

---

**Algoritmo 10:** Heurística para geração de uma solução inicial para o PPVT.

---

- 1 Seja  $T_x$  o conjunto de viagens a serem realizadas, com  $T_x$  ordenado crescentemente por horário de início de suas viagens
  - 2 Seja  $F$  o conjunto de veículos da empresa, inicialmente  $F$  possui um único veículo sem qualquer viagem
  - 3 Seja  $J$  o conjunto de jornadas diárias de trabalho das tripulações, inicialmente  $J$  possui uma única jornada sem qualquer viagem
  - 4 **enquanto**  $T_x \neq \emptyset$  **faça**
  - 5      $t_x \leftarrow$  primeira viagem de  $T_x$
  - 6      $melhorDelta \leftarrow \infty$ , onde  $melhorDelta$  é a melhor variação ocorrida na função de avaliação ao se alocar a viagem  $t_x$
  - 7     **para** cada veículo  $v \in F$  **faça**
  - 8          $custoAnteriorVeic \leftarrow$  custo do veículo  $v$  antes de receber a viagem  $t_x$
  - 9          $custoPosteriorVeic \leftarrow$  novo custo do veículo  $v$  caso a viagem  $t_x$  seja inserida em sua jornada de trabalho diária
  - 10        **para** cada jornada  $j \in J$  **faça**
  - 11              $custoAnteriorJorn \leftarrow$  custo da jornada  $j$  antes de receber a viagem  $t_x$
  - 12              $custoPosteriorJorn \leftarrow$  custo da jornada  $j$  caso a viagem  $t_x$  (supostamente já alocada ao veículo  $v$ ) passe a pertencer a ela
  - 13             Calcule a variação de custo  $\Delta$  ocorrida caso a viagem  $t_x$  seja alocada ao veículo  $v$  e à jornada  $j$ :
  - 14                  $\Delta = (custoPosteriorVeic + custoPosteriorJorn)$
  - 15                  $- (custoAnteriorVeic + custoAnteriorJorn)$
  - 16             **se**  $\Delta < melhorDelta$  **então**
  - 17                  $melhorDelta \leftarrow \Delta$
  - 18                  $melhorVeiculo \leftarrow info(v)$  e  $melhorJornada \leftarrow info(j)$ , isto é, armazene as informações que permitam identificar o veículo da frota e a jornada que irão contribuir menos para o valor da função de avaliação ao receberem a viagem  $t_x$
  - 19        **se**  $melhorVeiculo$  não possui viagens **então**
  - 20             Crie um novo veículo sem viagens e adicione ao conjunto  $F$ . (criação dinâmica de veículo, mantendo sempre um veículo sem viagens em  $F$ )
  - 21        **se**  $melhorJornada$  não possui viagens **então**
  - 22             Crie uma nova jornada sem viagens e adicione ao conjunto  $J$ . (criação dinâmica de jornada, mantendo sempre uma jornada sem viagens em  $J$ )
  - 23        Aloque a viagem  $t_x$  ao veículo da frota  $F$  indicado por  $melhorVeiculo$  e à jornada do conjunto  $J$  indicada por  $melhorJornada$
  - 24         $fo \leftarrow fo + melhorDelta$ , ou seja, atualize o valor da função de avaliação
  - 25        Remova  $t_x$  de  $T_x$
  - 26 Retorne a frota  $F$  e o conjunto de jornadas  $J$  da empresa
-

é escolhido aleatoriamente um dos três movimentos: realocação PPV, realocação PPT e super realocação;

6. *DescidaRandomicaComTrocaX*: Que considera durante as buscas as estruturas de vizinhança  $VN_x^{(T)}$ ,  $CN_x^{(T)}$  e  $VCN_x^{(T)}$ . Assim, a cada iteração do método é escolhido aleatoriamente um dos três movimentos: troca PPV, troca PPT e super troca;

As perturbações correspondem a movimentos aleatórios realizados sobre a solução corrente. Foram definidos quatro níveis de perturbação, consecutivamente tem-se: *nível 1* - uma realocação PPT, *nível 2* - uma realocação PPV, *nível 3* - uma troca PPT e *nível 4* - uma troca PPV.

**Algoritmo 11:** Algoritmo de resolução do PPVT.

- 
- 1 Seja  $T$  o tempo de processamento corrente do algoritmo e  $Tmax$  o tempo máximo admissível de processamento
  - 2 Seja  $nivel$  o nível de perturbação corrente
  - 3  $T \leftarrow 0$
  - 4  $nivel \leftarrow 1$
  - 5 Gere as soluções iniciais  $s^{vx}$  para o PPV e  $s^{tx}$  para o PPT a partir do método guloso descrito pelo Algoritmo 10, na página 55
  - 6  $s_*^{vx} \leftarrow s^{vx}$ , onde  $s_*^{vx}$  é a melhor solução encontrada até então para o PPV
  - 7  $s_*^{tx} \leftarrow s^{tx}$ , onde  $s_*^{tx}$  é a melhor solução encontrada até então para o PPT
  - 8  $s_*^{tx} \leftarrow DescidaRandomicaComRealocacaoX\_PPT(s_*^{tx})$
  - 9  $s_*^{tx} \leftarrow DescidaRandomicaComTroca\_PPT(s_*^{tx})$
  - 10  $(s_*^{vx}, s_*^{tx}) \leftarrow DescidaRandomicaComRealocacaoX\_PPV(s_*^{vx}, s_*^{tx})$
  - 11  $(s_*^{vx}, s_*^{tx}) \leftarrow DescidaRandomicaComTrocaX\_PPV(s_*^{vx}, s_*^{tx})$
  - 12  $(s_*^{vx}, s_*^{tx}) \leftarrow DescidaRandomicaComRealocacaoX(s_*^{vx}, s_*^{tx})$
  - 13  $(s_*^{vx}, s_*^{tx}) \leftarrow DescidaRandomicaComTrocaX(s_*^{vx}, s_*^{tx})$
  - 14 **enquanto**  $T < Tmax$  **faça**
  - 15      $(s^{vx}, s^{tx}) \leftarrow perturbacao(s_*^{vx}, s_*^{tx}, nivel)$  ou  $s^{tx} \leftarrow perturbacao(s_*^{tx}, nivel)$ , conforme o nível de perturbação modifique PPV e/ou PPT
  - 16     **se** a perturbação realizada envolve apenas realocação ou troca de viagens no PPT **então**
  - 17          $s^{tx} \leftarrow DescidaRandomicaComRealocacaoX\_PPT(s^{tx})$
  - 18          $s^{tx} \leftarrow DescidaRandomicaComTrocaX\_PPT(s^{tx})$
  - 19     **senão**
  - 20          $(s^{vx}, s^{tx}) \leftarrow DescidaRandomicaComRealocacaoX\_PPV(s^{vx}, s^{tx})$
  - 21          $(s^{vx}, s^{tx}) \leftarrow DescidaRandomicaComTrocaX\_PPV(s^{vx}, s^{tx})$
  - 22          $(s^{vx}, s^{tx}) \leftarrow DescidaRandomicaComRealocacaoX(s^{vx}, s^{tx})$
  - 23          $(s^{vx}, s^{tx}) \leftarrow DescidaRandomicaComTrocaX(s^{vx}, s^{tx})$
  - 24     **se**  $f^{vt}(s^{vx}, s^{tx}) \leq f^{vt}(s_*^{vx}, s_*^{tx})$  **então**
  - 25          $s_*^{vx} \leftarrow s^{vx}$
  - 26          $s_*^{tx} \leftarrow s^{tx}$
  - 27         **se**  $f^{vt}(s^{vx}, s^{tx}) < f^{vt}(s_*^{vx}, s_*^{tx})$  **então**
  - 28              $nivel \leftarrow 1$
  - 29         **senão**
  - 30              $nivel \leftarrow nivel + 1$
  - 31     **senão**
  - 32          $nivel \leftarrow nivel + 1$
  - 33     **se**  $nivel > numNiveis$  **então**
  - 34          $nivel \leftarrow 1$ , onde  $numNiveis$  é o número máximo de níveis de perturbação
  - 35 Retorne as soluções  $s_*^{vx}$  e  $s_*^{tx}$
-



# Capítulo 5

## Experimentos Computacionais

### 5.1 Descrição das Instâncias

Para validar a metodologia proposta foram disponibilizadas pela empresa responsável pelo gerenciamento do Sistema de Transporte Público da cidade de Belo Horizonte, capital do estado de Minas Gerais, problemas reais de uma dada região da cidade. As tabelas de horários das viagens aqui consideradas são relativas ao mês de maio de 2005.

Há 8 empresas responsáveis pela realização de todas as viagens desta região. Juntamente elas prestam serviços em 40 linhas de ônibus classificadas em dois tipos: troncais e alimentadoras. As linhas troncais operam em áreas onde há alta demanda e têm como objetivo realizar o deslocamento de passageiros entre a estação de ônibus e o centro da cidade. As linhas alimentadoras, que possuem uma abrangência local, têm a função primordial de captar e distribuir a demanda atendida pelas linhas troncais, realizando a ligação bairro-estação-bairro.

A seguir, na Tabela 5.1, é apresentada uma descrição das instâncias utilizadas nos experimentos. Para cada empresa foram considerados separadamente os quatro dias da semana que apresentam quadros de horários distintos: segunda-feira, sexta-feira, sábado e domingo. Assim, cada instância é identificada univocamente através do dia da semana e da empresa aos quais se refere. As empresas serão aqui tratadas pelos seguintes nomes fictícios: G01, G02, G03, G04, G05, G06, G07 e G08. Nesta tabela, após o nome de cada instância estão listados, respectivamente, o seu número de viagens, o tempo total despendido com a execução das mesmas (em horas e minutos) e a duração média de suas viagens (em horas e minutos). Observa-se que as empresas G01, G02 e G08 são responsáveis por poucas viagens, porém sendo estas de longa duração. Já as empresas G03, G04, G05, G06, G07 atendem muitas viagens, no entanto, estas são viagens menores.

Por fim, é bom ressaltar que o tempo total em viagem dado na Tabela 5.1 é sempre

menor do que o tempo real de operação da frota, pois o primeiro não inclui as viagens mortas e as esperas nos terminais realizadas pelos veículos.

**Tabela 5.1.** Descrição das instâncias utilizadas nos experimentos.

Empresa	Instância	Número de Viagens	Tempo Total em Viagem (hh:mm)	Duração Média das Viagens (hh:mm)
G01	SEG-G01	260	348:10	1:20
	SEX-G01	260	362:17	1:24
	SAB-G01	172	217:16	1:16
	DOM-G01	90	124:27	1:23
G02	SEG-G02	98	141:20	1:27
	SEX-G02	98	147:05	1:30
	SAB-G02	69	91:13	1:19
	DOM-G02	52	64:33	1 14
G03	SEG-G03	505	516:56	1:01
	SEX-G03	498	539:49	0:43
	SAB-G03	372	379:55	1:01
	DOM-G03	286	281:17	0:59
G04	SEG-G04	468	387:05	0:50
	SEX-G04	468	397:35	0:51
	SAB-G04	359	282:50	0:47
	DOM-G04	298	221:04	0:45
G05	SEG-G05	1038	773:58	0:45
	SEX-G05	1036	803:53	0:47
	SAB-G05	769	541:49	0:42
	DOM-G05	538	336:21	0:38
G06	SEG-G06	639	622:30	0:58
	SEX-G06	639	660:44	1:02
	SAB-G06	441	410:20	0:56
	DOM-G06	332	316:15	0:57
G07	SEG-G07	293	253:34	0:52
	SEX-G07	292	267:55	0:55
	SAB-G07	191	147:36	0:46
	DOM-G07	115	60:13	0:31
G08	SEG-G08	206	296:41	1:26
	SEX-G08	203	295:18	1:27
	SAB-G08	130	182:50	1:24
	DOM-G08	108	147:14	1:22

Todos os algoritmos propostos neste trabalho foram desenvolvidos na linguagem C++ e compilados com a versão 4.3.2 do gcc. Os experimentos foram realizados em um microcomputador PC Intel Core 2 Quad, com processador de 2.5 GHz, 4 GB de memória RAM e sob o sistema operacional Linux Ubuntu Hardy Heron 8.04.

Objetivando estabelecer uma proporção entre os gastos da empresa relativos à programação de veículos e à programação de tripulações, foram criados dois custos fundamentais a partir dos quais são definidos todos os demais, são eles:

1. Custo da Unidade de Tempo do Veículo - CUTV: que representa o custo para a empresa do minuto de operação do veículo;
2. Custo da Unidade de Tempo da Tripulação - CUTT: que representa o custo para a empresa do minuto de trabalho de uma tripulação.

Destaca-se que custo nesse contexto, não necessariamente se refere a valores monetários, embora estes possam ser usados. Por exemplo, se em uma dada situação o minuto de trabalho de uma tripulação é duas vezes mais caro que o minuto de operação de um veículo, as atribuições  $CUTV = 1$  e  $CUTT = 2$  poderiam ser realizadas. Observe que neste caso não foi considerado o valor real gasto pela empresa com uma tripulação ou veículo, mas foi mantida uma proporcionalidade entre estes custos. Nos experimentos realizados considerou-se  $CUTT = 5$  e  $CUTV = 1$ .

Na Tabela 5.2 são apresentados os valores dos pesos utilizados para o cálculo da função de avaliação referente à programação de veículos, dada pela expressão 3.1 apresentada na subseção 3.3.3. Nessa tabela *tempoMaxOperacao* corresponde ao tempo máximo de operação diária do veículo, em minutos. Nos testes, *tempoMaxOperacao* é igual a 1410 minutos, ou seja, 23 horas e 30 minutos.

**Tabela 5.2.** Penalizações utilizadas no PPV e seus respectivos valores.

Penalização	Valor
<i>custoUsoVeiculo</i>	$tempoMaxOperacao \times CUTV$
<i>custoViagemMorta</i>	$CUTV$
<i>pesoInviabilidade</i>	$1,5 \times CUTV$

Os valores dos pesos utilizados na composição da função de avaliação para a programação de tripulações, apresentada na subseção 3.5.3 e dada pela expressão 3.3, são mostrados na Tabela 5.3. Nessa tabela *duracaoMaxJornada* corresponde ao tempo máximo, sem horas extras, efetivamente trabalhado por uma tripulação (em minutos). Nos testes, *duracaoMaxJornada* é igual a 400 minutos, isto é, 6 horas e 40 minutos.

**Tabela 5.3.** Penalizações utilizadas no PPT e seus respectivos valores.

Penalização	Valor
<i>custoUsoTripulacao</i>	$duracaoMaxJornada \times CUTT$
<i>custoHoraExtra</i>	$0,75 \times CUTT$
<i>pesoInviabilidade</i>	$1,5 \times CUTT$

O tempo de execução dos algoritmos foi fixado em 10 minutos para todas as instâncias. Nas abordagens independente e seqüencial tradicional foram despendidos tempos iguais para resolver separadamente, PPV e PPT, ou seja, 5 minutos para cada.

## 5.2 Resultados

A Tabela 5.4 mostra uma comparação entre os custos das soluções obtidas pelas quatro metodologias apresentadas nesse trabalho para definição das programações de veículos e tripulações, são elas: independente, seqüencial tradicional, iterativa e integrada. Nesta tabela, “Melhor FO” e “FO Médio” são, respectivamente, o menor custo e a média dos custos obtidos pelos métodos desenvolvidos em 10 execuções, considerando PPV e PPT. A “Melhoria” representa o aprimoramento, em percentual, do valor da melhor solução encontrada em cada abordagem comparada com o custo da melhor solução obtida na abordagem seqüencial. O “Desvio” corresponde ao desvio médio de cada abordagem em relação ao valor da melhor solução encontrada nos testes, isto é:

$$desvio = \frac{FOMedio - MelhorFO}{MelhorFO} \quad (5.1)$$

Por fim, são apresentadas na Tabela 5.4 as médias para o desvio e a melhoria, considerando todas as instâncias. É importante frisar que os problemas sendo tratados são de minimização, ou seja, o objetivo é reduzir os custos das soluções.

Tabela 5.4: Resultados considerando os custos das soluções obtidas pelas abordagens independente, seqüencial, iterativa e integrada.

Intância	Avaliação	Abordagens			
		Independente	Seqüencial	Iterativa	Integrada
SEG-G01	Melhor FO	188184	206401	198690	199071
	FO Médio	188212	207752	200878	199486
	Desvio (%)	0.02	0.65	1.10	0.21
	Melhoria (%)	8.83	0.00	3.74	3.55

Intância	Avaliação	Abordagens			
		Independente	Seqüencial	Iterativa	Integrada
SEX-G01	Melhor FO	186960	205476	197726	198022
	FO Médio	194633	206860	199879	201015
	Desvio (%)	4.10	0.67	1.09	1.51
	Melhoria (%)	9.01	0.00	3.77	3.63
SAB-G01	Melhor FO	104029	114821	113601	109976
	FO Médio	104181	118892	113991	111540
	Desvio (%)	0.15	3.55	0.34	1.42
	Melhoria (%)	9.40	0.00	1.06	4.22
DOM-G01	Melhor FO	52785	56533	56355	56207
	FO Médio	53622	57822	57119	56368
	Desvio (%)	1.59	2.28	1.35	0.29
	Melhoria (%)	6.63	0.00	0.31	0.58
SEG-G02	Melhor FO	64143	69561	69103	69328
	FO Médio	64245	70335	69207	69782
	Desvio (%)	0.16	1.11	0.15	0.65
	Melhoria (%)	7.79	0.00	0.66	0.33
SEX-G02	Melhor FO	64338	70188	69454	68597
	FO Médio	64999	70562	69663	69510
	Desvio (%)	1.03	0.53	0.30	1.33
	Melhoria (%)	8.33	0.00	1.05	2.27
SAB-G02	Melhor FO	39122	41818	41023	39863
	FO Médio	39216	42668	41123	39920
	Desvio (%)	0.24	2.03	0.24	0.14
	Melhoria (%)	6.45	0.00	1.90	4.68
DOM-G02	Melhor FO	27823	30291	28507	29685
	FO Médio	27863	30857	29321	29833
	Desvio (%)	0.14	1.87	2.85	0.50
	Melhoria (%)	8.15	0.00	5.89	2.00
SEG-G03	Melhor FO	234780	267525	260576	252235
	FO Médio	237778	272265	266441	252888
	Desvio (%)	1.28	1.77	2.25	0.26
	Melhoria (%)	12.24	0.00	2.60	5.72

Intância	Avaliação	Abordagens			
		Independente	Seqüencial	Iterativa	Integrada
SEX-G03	Melhor FO	240457	267965	267924	257405
	FO Médio	241628	272355	271511	257756
	Desvio (%)	0.49	1.64	1.34	0.14
	Melhoria (%)	10.27	0.00	0.02	3.94
SAB-G03	Melhor FO	151853	173211	169502	164866
	FO Médio	152896	177362	171261	165222
	Desvio (%)	0.69	2.40	1.04	0.22
	Melhoria (%)	12.33	0.00	2.14	4.82
DOM-G03	Melhor FO	116330	131449	128258	124673
	FO Médio	117416	133678	130658	126119
	Desvio (%)	0.93	1.70	1.87	1.16
	Melhoria (%)	11.50	0.00	2.43	5.15
SEG-G04	Melhor FO	163948	189628	188030	175311
	FO Médio	165729	193688	193302	175976
	Desvio (%)	1.09	2.14	2.80	0.38
	Melhoria (%)	13.54	0.00	0.84	7.55
SEX-G04	Melhor FO	167986	188238	193405	177275
	FO Médio	168576	193709	194888	177492
	Desvio (%)	0.35	2.91	0.77	0.12
	Melhoria (%)	10.76	0.00	-2.74	5.82
SAB-G04	Melhor FO	116263	133575	130764	126656
	FO Médio	117397	138047	131423	128722
	Desvio (%)	0.98	3.35	0.50	1.63
	Melhoria (%)	12.96	0.00	2.10	5.18
DOM-G04	Melhor FO	92377	104791	104560	98676
	FO Médio	92990	108095	105246	98885
	Desvio (%)	0.66	3.15	0.66	0.21
	Melhoria (%)	11.85	0.00	0.22	5.84
SEG-G05	Melhor FO	346752	400873	404482	377157
	FO Médio	347752	403410	416529	378377
	Desvio (%)	0.29	0.63	2.98	0.32
	Melhoria (%)	13.50	0.00	-0.90	5.92

Intância	Avaliação	Abordagens			
		Independente	Seqüencial	Iterativa	Integrada
SEX-G05	Melhor FO	353728	400175	416129	381173
	FO Médio	354856	408269	420943	381818
	Desvio (%)	0.32	2.02	1.16	0.17
	Melhoria (%)	11.61	0.00	-3.99	4.75
SAB-G05	Melhor FO	223336	256646	255575	241439
	FO Médio	223880	260492	261490	241985
	Desvio (%)	0.24	1.50	2.31	0.23
	Melhoria (%)	12.98	0.00	0.42	5.93
DOM-G05	Melhor FO	137572	158808	157053	149947
	FO Médio	138775	162009	157967	150879
	Desvio (%)	0.87	2.02	0.58	0.62
	Melhoria (%)	13.37	0.00	1.11	5.58
SEG-G06	Melhor FO	280346	313502	312840	303197
	FO Médio	281366	321166	319985	303786
	Desvio (%)	0.36	2.44	2.28	0.19
	Melhoria (%)	10.58	0.00	0.21	3.29
SEX-G06	Melhor FO	295477	334657	339962	322667
	FO Médio	299173	338795	344935	323311
	Desvio (%)	1.25	1.24	1.46	0.20
	Melhoria (%)	11.71	0.00	-1.59	3.58
SAB-G06	Melhor FO	165118	193037	188598	179639
	FO Médio	166562	195165	192456	179997
	Desvio (%)	0.87	1.10	2.05	0.20
	Melhoria (%)	14.46	0.00	2.30	6.94
DOM-G06	Melhor FO	129604	150935	145810	140249
	FO Médio	130094	153483	146001	141200
	Desvio (%)	0.38	1.69	0.13	0.68
	Melhoria (%)	14.13	0.00	3.40	7.08
SEG-G07	Melhor FO	117124	131954	128405	126023
	FO Médio	118978	134641	133693	126240
	Desvio (%)	1.58	2.04	4.12	0.17
	Melhoria (%)	11.24	0.00	2.69	4.49

Intância	Avaliação	Abordagens			
		Independente	Seqüencial	Iterativa	Integrada
SEX-G07	Melhor FO	124389	136460	132810	129068
	FO Médio	125668	138069	135559	129832
	Desvio (%)	1.03	1.18	2.07	0.59
	Melhoria (%)	8.85	0.00	2.67	5.42
SAB-G07	Melhor FO	65074	71026	69286	68426
	FO Médio	65238	72410	70224	68581
	Desvio (%)	0.25	1.95	1.35	0.23
	Melhoria (%)	8.38	0.00	2.45	3.66
DOM-G07	Melhor FO	30875	33450	32084	32014
	FO Médio	31123	34064	33147	32137
	Desvio (%)	0.80	1.84	3.31	0.38
	Melhoria (%)	7.70	0.00	4.08	4.29
SEG-G08	Melhor FO	137326	148608	145909	144207
	FO Médio	138364	150788	150006	144723
	Desvio (%)	0.76	1.47	2.81	0.36
	Melhoria (%)	7.59	0.00	1.82	2.96
SEX-G08	Melhor FO	137141	153945	149802	152010
	FO Médio	141077	155694	152879	153518
	Desvio (%)	2.87	1.14	2.05	0.99
	Melhoria (%)	10.92	0.00	2.69	1.26
SAB-G08	Melhor FO	77555	87444	85303	85896
	FO Médio	77705	90141	86307	86313
	Desvio (%)	0.19	3.08	1.18	0.49
	Melhoria (%)	11.31	0.00	2.45	1.77
DOM-G08	Melhor FO	64171	73763	71370	72581
	FO Médio	65009	75332	71820	72669
	Desvio (%)	1.31	2.13	0.63	0.12
	Melhoria (%)	13.00	0.00	3.24	1.60
Média	Desvio (%)	0.85	1.85	1.54	0.50
	Melhoria (%)	10.67	0.00	1.53	4.18

Com base nos resultados da Tabela 5.4, verifica-se que como era de se esperar as melhores soluções foram obtidas pela abordagem independente, porém, conforme alertado anteriormente, as programações dos veículos e das tripulações encontradas são incompatíveis e, portanto, impraticáveis. A abordagem independente está sendo usada

apenas para indicar o quão promissora é a resolução integrada do PPV e PPT, uma vez que considera isoladamente os dois problemas, quebrando dessa forma as limitações que a resolução do PPV impõe ao PPT e vice-versa. Considerando as demais abordagens, a que obteve melhor resultado foi a integrada, e em seguida a iterativa, evidenciando assim, a relevância da integração. O pior desempenho foi do método de resolução seqüencial. Dentre as 32 instâncias consideradas, em 25 delas o método integrado encontrou os melhores resultados e para as demais 7 instâncias (SEG-G01, SEX-G01, SEG-G02, DOM-G02, SEX-G08, SAB-G08, DOM-G08) o método de resolução iterativa teve melhor desempenho, alcançando os menores custos. Destaca-se que para as instâncias SEX-G04, SEG-G05, SEX-G05 e SEX-G05 a abordagem iterativa não conseguiu melhorar os resultados gerados de forma seqüencial, por outro lado, a abordagem integrada foi capaz de aprimorar o custo de todas as soluções obtidas seqüencialmente.

Ainda na Tabela 5.4, analisando os valores de desvio constata-se a robustez dos métodos desenvolvidos e, dentre os quais, o procedimento de resolução integrada se mostrou o mais estável.

Na Tabela 5.5 as melhores soluções obtidas pelas diferentes abordagens são detalhadas. Nessa tabela, a “Melhoria” representa o aprimoramento, em percentual, do valor da melhor solução encontrada em cada abordagem comparada com o custo da melhor solução obtida na abordagem seqüencial (similar à Tabela 5.4) e, em seguida, estão listados os atributos das programações, são eles: o número de veículos e tripulações utilizados, o tempo total de viagens mortas dos veículos e o tempo total de horas extras das tripulações. Por fim, nessa tabela, também são apresentados os valores totais dos atributos das programações e, os tempos médios de viagens mortas por veículo e de horas extras por tripulação, considerando todas as instâncias. Salienta-se que todos os tempos são dados em horas e minutos (formato hh:mm).

Tabela 5.5: Detalhamento das melhores soluções obtidas pelas abordagens independente, seqüencial, iterativa e integrada.

Intância	Atributos	Abordagens			
		Independente	Seqüencial	Iterativa	Integrada
SEG-G01	Melhoria (%)	8.83	0.00	3.74	3.55
	Veículos	40	40	40	40
	Tripulações	65	74	70	70
	Viagens mortas	29:44	30:12	31:52	36:04
	Horas extras	00:00	01:03	02:06	02:49

Intância	Atributos	Abordagens			
		Independente	Seqüencial	Iterativa	Integrada
SEX-G01	Melhoria (%)	9.01	0.00	3.77	3.63
	Veículos	39	39	39	39
	Tripulações	65	74	70	70
	Viagens mortas	30:56	30:20	33:54	37:08
	Horas extras	00:38	03:42	03:54	04:28
SAB-G01	Melhoria (%)	9.40	0.00	1.06	4.22
	Veículos	23	23	23	23
	Tripulações	35	40	40	37
	Viagens mortas	13:48	13:48	14:16	16:36
	Horas extras	04:17	08:41	01:45	14:10
DOM-G01	Melhoria (%)	6.63	0.00	0.31	0.58
	Veículos	10	10	10	10
	Tripulações	18	20	20	20
	Viagens mortas	06:00	06:00	06:56	07:10
	Horas extras	12:55	11:31	10:13	09:19
SEG-G02	Melhoria (%)	7.79	0.00	0.66	0.33
	Veículos	13	13	13	13
	Tripulações	22	25	25	25
	Viagens mortas	09:58	09:58	10:44	10:44
	Horas extras	06:45	03:31	00:43	01:58
SEX-G02	Melhoria (%)	8.33	0.00	1.05	2.27
	Veículos	13	13	13	13
	Tripulações	22	25	25	24
	Viagens mortas	09:58	09:58	10:44	10:44
	Horas extras	07:50	07:00	02:40	09:01
SAB-G02	Melhoria (%)	6.45	0.00	1.90	4.68
	Veículos	7	7	7	7
	Tripulações	14	15	15	14
	Viagens mortas	05:22	05:22	05:22	05:22
	Horas extras	05:10	09:02	04:37	09:17
DOM-G02	Melhoria (%)	8.15	0.00	5.89	2.00
	Veículos	5	5	5	5
	Tripulações	10	11	10	11
	Viagens mortas	03:50	03:50	03:50	03:50
	Horas extras	03:01	05:37	06:49	02:15

Intância	Atributos	Abordagens			
		Independente	Seqüencial	Iterativa	Integrada
SEG-G03	Melhoria (%)	12.24	0.00	2.60	5.72
	Veículos	47	47	47	48
	Tripulações	83	100	96	90
	Viagens mortas	16:35	16:16	19:17	27:58
	Horas extras	08:25	01:33	06:23	15:59
SEX-G03	Melhoria (%)	10.27	0.00	0.02	3.94
	Veículos	47	47	47	48
	Tripulações	86	100	100	93
	Viagens mortas	15:33	15:33	17:40	24:47
	Horas extras	06:58	04:14	03:18	12:26
SAB-G03	Melhoria (%)	12.33	0.00	2.14	4.82
	Veículos	27	27	27	27
	Tripulações	52	65	64	60
	Viagens mortas	08:42	08:08	09:42	18:28
	Horas extras	51:27	25:51	15:50	31:36
DOM-G03	Melhoria (%)	11.50	0.00	2.43	5.15
	Veículos	21	21	21	21
	Tripulações	41	50	48	45
	Viagens mortas	06:28	06:24	07:14	13:05
	Horas extras	24:04	08:05	12:18	23:46
SEG-G04	Melhoria (%)	13.54	0.00	0.84	7.55
	Veículos	31	31	31	31
	Tripulações	57	72	71	62
	Viagens mortas	12:25	12:07	14:10	20:59
	Horas extras	30:31	06:37	08:10	35:14
SEX-G04	Melhoria (%)	10.76	0.00	-2.74	5.82
	Veículos	31	31	31	31
	Tripulações	59	70	74	62
	Viagens mortas	12:30	12:37	14:39	21:25
	Horas extras	30:42	20:57	04:32	46:00
SAB-G04	Melhoria (%)	12.96	0.00	2.10	5.18
	Veículos	20	20	20	20
	Tripulações	41	51	50	46
	Viagens mortas	07:39	07:33	09:20	16:45
	Horas extras	31:08	16:14	11:08	30:17

Intância	Atributos	Abordagens			
		Independente	Seqüencial	Iterativa	Integrada
DOM-G04	Melhoria (%)	11.85	0.00	0.22	5.84
	Veículos	16	16	16	16
	Tripulações	32	40	40	35
	Viagens mortas	06:18	06:23	06:47	11:56
	Horas extras	30:13	10:16	08:51	30:00
SEG-G05	Melhoria (%)	13.50	0.00	-0.90	5.92
	Veículos	70	70	70	70
	Tripulações	122	150	151	135
	Viagens mortas	30:05	29:49	34:17	44:33
	Horas extras	12:29	02:08	09:35	32:08
SEX-G05	Melhoria (%)	11.61	0.00	-3.99	4.75
	Veículos	70	70	70	70
	Tripulações	125	148	157	137
	Viagens mortas	30:42	30:21	34:30	44:01
	Horas extras	17:42	20:18	07:33	32:24
SAB-G05	Melhoria (%)	12.98	0.00	0.42	5.93
	Veículos	42	42	42	42
	Tripulações	75	96	96	85
	Viagens mortas	17:01	16:56	18:05	25:57
	Horas extras	72:45	24:30	18:10	59:14
DOM-G05	Melhoria (%)	13.37	0.00	1.11	5.58
	Veículos	24	24	24	24
	Tripulações	46	60	60	53
	Viagens mortas	09:53	10:21	12:03	20:45
	Horas extras	61:53	24:09	13:50	49:14
SEG-G06	Melhoria (%)	10.58	0.00	0.21	3.29
	Veículos	55	55	55	55
	Tripulações	99	116	115	109
	Viagens mortas	36:26	35:55	41:25	47:15
	Horas extras	14:30	09:59	15:35	26:44
SEX-G06	Melhoria (%)	11.71	0.00	-1.59	3.58
	Veículos	60	60	60	60
	Tripulações	103	123	125	115
	Viagens mortas	39:32	38:49	46:55	51:09
	Horas extras	13:55	09:36	14:09	27:46

Intância	Atributos	Abordagens			
		Independente	Seqüencial	Iterativa	Integrada
SAB-G06	Melhoria (%)	14.46	0.00	2.30	6.94
	Veículos	30	30	30	30
	Tripulações	55	73	71	63
	Viagens mortas	18:17	18:00	20:56	26:41
	Horas extras	65:07	20:19	16:54	54:06
DOM-G06	Melhoria (%)	14.13	0.00	3.40	7.08
	Veículos	23	23	23	23
	Tripulações	44	58	54	50
	Viagens mortas	13:51	13:39	17:40	22:01
	Horas extras	46:21	09:22	24:00	36:06
SEG-G07	Melhoria (%)	11.24	0.00	2.69	4.49
	Veículos	24	24	24	24
	Tripulações	41	48	46	45
	Viagens mortas	11:12	11:11	13:24	15:50
	Horas extras	03:24	08:01	09:47	06:51
SEX-G07	Melhoria (%)	8.85	0.00	2.67	5.42
	Veículos	25	25	25	25
	Tripulações	44	50	48	45
	Viagens mortas	11:08	10:40	13:30	16:32
	Horas extras	02:37	03:10	04:10	15:42
SAB-G07	Melhoria (%)	8.38	0.00	2.45	3.66
	Veículos	12	12	12	12
	Tripulações	23	26	25	24
	Viagens mortas	04:42	04:48	05:50	09:20
	Horas extras	10:24	10:06	11:12	16:22
DOM-G07	Melhoria (%)	7.70	0.00	4.08	4.29
	Veículos	5	5	5	5
	Tripulações	11	13	12	12
	Viagens mortas	01:10	01:10	01:53	01:34
	Horas extras	09:45	01:50	05:07	04:50
SEG-G08	Melhoria (%)	7.59	0.00	1.82	2.96
	Veículos	26	26	26	26
	Tripulações	49	53	52	51
	Viagens mortas	43:20	43:20	43:20	45:00
	Horas extras	00:22	18:36	14:43	15:49

Intância	Atributos	Abordagens			
		Independente	Seqüencial	Iterativa	Integrada
SEX-G08	Melhoria (%)	10.92	0.00	2.69	1.26
	Veículos	28	28	28	28
	Tripulações	47	55	52	54
	Viagens mortas	46:40	46:40	46:40	50:00
	Horas extras	04:47	09:15	19:34	08:30
SAB-G08	Melhoria (%)	11.31	0.00	2.45	1.77
	Veículos	14	14	14	14
	Tripulações	27	32	31	32
	Viagens mortas	23:20	23:20	23:20	23:20
	Horas extras	13:25	12:48	12:01	04:12
DOM-G08	Melhoria (%)	13.00	0.00	3.24	1.60
	Veículos	13	13	13	13
	Tripulações	21	26	25	26
	Viagens mortas	21:40	21:40	21:40	21:40
	Horas extras	14:07	11:51	09:40	05:17
Total	Veículos	911	911	911	913
	Tripulações	1634	1959	1938	1800
	Viagens mortas	554:45	551:08	611:55	748:39
	Horas extras	617:37	339:52	309:17	673:50
Média	Viagens mortas por veículo	00:37	00:36	00:40	00:49
	Horas extras por tripulação	00:22	00:10	00:10	00:22

A partir da Tabela 5.5 observa-se que os tempos totais de viagens mortas obtidos nas abordagens independente e seqüencial foram menores do que os encontrados de forma iterativa ou integrada. Este era um resultado previsível, uma vez que diferentemente do que ocorre na resolução seqüencial, no PPVT limita-se o aprimoramento da programação de veículos ao considerar os efeitos que esta melhoria irá provocar na programação de tripulações. Porém, ainda assim, o número de veículos utilizados manteve-se o mesmo em todas as abordagens, exceto para as instâncias SEG-G03 e SEX-G03 em que a resolução integrada implicou na utilização de um veículo a mais. As melhorias proporcionadas pelas abordagens iterativa e integrada estão concentradas nas soluções para o PPT. Observa-se que o ganho se dá, principalmente, em relação ao número de tripulações empregadas. Nesse sentido, é interessante relembrar que

uma tripulação envolve a mão-de-obra de dois funcionários, um motorista e um cobrador. Portanto, ao analisar na Tabela 5.5 o total de tripulações utilizadas, verifica-se que comparadas com a abordagem seqüencial, a abordagem iterativa proporciona uma redução de 21 tripulações, ou seja, 42 funcionários e a abordagem integrada evita a utilização de 159 tripulações que correspondem a 318 funcionários.

Destaca-se que o objetivo primordial dos testes realizados foi a redução do número de veículos e tripulantes empregados, conforme pode ser observado pelos custos dados aos atributos das programações. Secundariamente, visou-se reduzir as horas extras das tripulações e os deslocamentos fora de operação realizados pelos veículos.



# Capítulo 6

## Conclusões e Trabalhos Futuros

Esse trabalho abordou o Problema de Programação Integrada de Veículos e Tripulações (PPVT) considerando um cenário real envolvendo empresas do sistema de transporte público por ônibus da cidade de Belo Horizonte. Esse é um problema de grande importância, uma vez que através da sua resolução as empresas do setor podem reduzir custos operacionais, aprimorar processos produtivos e, conseqüentemente, oferecer um serviço de qualidade, eficiente e mais barato aos usuários.

A definição simultânea da rotina de operação da frota de veículos e das jornadas de trabalho das tripulações é uma metodologia ainda pouco explorada na literatura devido à sua complexidade, mas é considerada promissora no que diz respeito à qualidade das soluções geradas. Portanto, a principal contribuição desse trabalho foi propor uma nova estratégia para tratar o PPVT e verificar a sua efetividade na resolução de problemas reais.

Foram propostas heurísticas, baseadas na metaheurística Busca Local Iterada, para definição das programações de veículos e tripulações considerando quatro abordagens distintas:

1. Abordagem Seqüencial Tradicional: Primeiro é resolvido o PPV, cuja solução consiste em um conjunto de blocos de viagens, sendo cada bloco a jornada diária de trabalho de um veículo. Antes de ser iniciada a resolução do PPT, os blocos de viagens são simplificados em tarefas. Cada tarefa consiste em um conjunto de viagens entre as quais não é possível realizar a troca de tripulações e onde são contemplados os deslocamentos do veículo que não correspondem a viagens;
2. Abordagem Independente: PPV e PPT são resolvidos separadamente, em qualquer ordem e de forma a não se relacionarem. Nesta abordagem tanto a programação de veículos quanto a programação de tripulações são determinadas a partir das viagens a serem realizadas, não havendo a formação de tarefas. Assim,

dificilmente as soluções obtidas são compatíveis e factíveis na prática. A principal finalidade desta metodologia independente é a determinação de uma melhor programação para as tripulações desconsiderando os veículos. Desta forma, uma comparação entre as soluções obtidas nesta abordagem e na seqüencial tradicional permite identificar os possíveis benefícios de se resolver PPV e PPT de forma integrada;

3. Abordagem Seqüencial Iterativa: Como primeiro passo para a integração entre PPV e PPT foi desenvolvida uma estratégia iterativa de resolução destes problemas em que, ora a busca por melhoras se concentra especificamente no PPV e ora no PPT. Nesta abordagem a principal preocupação é manter as programações de veículos e tripulações compatíveis entre si;
4. Abordagem Integrada: PPV e PPT são resolvidos de forma simultânea. O desafio encontrado é, durante a resolução, encontrar uma maneira de manter as programações de veículos e tripulações sendo geradas compatíveis. Neste sentido, foi elaborada uma representação baseada na definição de uma nova estrutura denominada viagem expandida. Uma viagem expandida corresponde a uma viagem com atributos adicionais que definem como ela deve ser considerada no PPT, dada a sua disposição no PPV. Os novos atributos das viagens permitem contemplar as mesmas condições asseguradas ao se formar tarefas na abordagem seqüencial. Nesta estratégia tanto a programação de veículos quanto a programação de tripulações são determinadas a partir das viagens expandidas, criando grande flexibilização para obtenção da solução.

Testes foram realizados com instâncias reais relativas aos quatro dias da semana com quadros de horários das viagens distintos (segunda-feira, sexta-feira, sábado e domingo) de 8 empresas do sistema de transporte público por ônibus da cidade de Belo Horizonte. Dentre as 32 instâncias consideradas, em 25 delas o método integrado encontrou os melhores resultados e para as demais 7 instâncias o método de resolução iterativa teve melhor desempenho, alcançando os menores custos, evidenciando assim, a relevância da integração.

Como continuidade deste trabalho pode-se propor, primeiramente, realizar experimentos alterando a função de avaliação de forma a priorizar outras métricas. Lembrando que, nos testes apresentados, o objetivo primordial foi a redução do número de veículos e tripulantes empregados. Outro aprimoramento seria comparar as soluções obtidas pela heurística integrada proposta com soluções ótimas geradas por um método exato. Destaca-se que para as instâncias estudadas, provavelmente não seria possível

obter soluções ótimas em um tempo aceitável, portanto instâncias fictícias menores precisariam ser criadas.

Outra extensão deste trabalho pode considerar a integração entre os problemas de definir a tabela de horários das viagens, a programação dos veículos e a programação das tripulações. Assim, a tabela de horários seria definida de forma a estrategicamente reduzir o custo operacional das programações de veículos e tripulações, além de considerar a disponibilidade de infra-estrutura, os serviços requeridos pelos usuários e os aspectos de demanda.

A Programação Integrada de Veículos e Tripulações com Múltiplas Garagens também é uma proposta a ser considerada para a continuidade da pesquisa. Esse problema é resolvido considerando que a frota de veículos, disponível para realização das viagens, se encontra espalhada em diferentes garagens e os veículos não necessitam iniciar e terminar a sua rotina de operação diária em uma mesma garagem.



# Referências Bibliográficas

- ANTP (1999). Associação Nacional dos Transportes Públicos. Disponível na internet. URL: <http://www.antp.org.br>. Acessado em 17/10/2007.
- Baita, F.; Pesenti, R.; Ukovich, W. e Favaretto, D. (2000). A comparison of different solution approaches to the vehicle scheduling problem in a practical case. *Computers and Operations Research*, 27:1249–1269.
- Ball, M. O.; Bodin, L. D. e Dial, R. (1983). A matching based heuristic for scheduling mass transit crew and vehicles. *Transportation Science*, 1(17):4–31.
- Daduna, J. R. e Voss, S. (2001). Lecture notes in economics and mathematical systems. *Computer-Aided Scheduling of Public Transport*, 505. Berlim: Springer Publishers.
- Desrochers, M. e Soumis, F. (1989). A column generation approach to the urban transit crew scheduling problem. *Transportation Science*, 23:1–13.
- Elias, S. E. G. (1964). The use of digital computers in the economic scheduling for both man and machine in public transportation. Manhattan, Kansas.
- Freling, R. (1997). *Models and techniques for integrating vehicle and crew scheduling*. PhD thesis, Erasmus University Rotterdam, Amsterdam.
- Freling, R.; Boender, C. G. E. e Paixão, J. M. P. (1995). An integrated approach to vehicle and crew scheduling. *Technical Report 9503/A*. Econometric Institute, Erasmus University Rotterdam, Rotterdam.
- Freling, R.; Huisman, D. e Wagelmans, A. P. M. (2003). Models and algorithms for integration of vehicle and crew scheduling. *Journal of Scheduling*, 6(1):63–85.
- Freling, R.; Paixão, J. M. P. e Wagelmans, A. P. M. (2001). Models and algorithms for vehicle scheduling. *Transportation Science*, 35:165–180.
- Freling, R.; Wagelmans, A. P. M. e Paixão, J. M. P. (1999). An overview of models and techniques for integrating vehicle and crew scheduling. *Computer-Aided Transit Scheduling*, 471:441–460.

- Haase, K. e Friberg, C. (1999). An exact branch and cut algorithm for the vehicle and crew scheduling problem. *Computer-Aided Transit Scheduling*, pp. 63–80.
- Hoffman, K. L. e Padberg, M. (1993). Solving airline crew scheduling problems by branch-and-cut. *Management Science*, 39:657–682.
- Huisman, D.; Freling, R. e Wagelmans, A. P. M. (2005). Multiple-depot integrated vehicle and crew scheduling. *Transportation Science*, 39(4):491–502.
- Kirkman, F. (1968). Problems of innovation in the transport industry: a bus scheduling program. In *Proceedings of PTRC Public Transport Analysis Seminar, Planning and Transport Research and Computation Co. Ltd.*, volume 1, pp. 1–15.
- Laurent, B. e Hao, J. (2007). Simultaneous vehicle and driver scheduling: A case study in a limousine rental company. *Computers and Industrial Engineering*, 53(3):542–558.
- Lourenço, H. R.; Martin, O. C. e Stutzle, T. (2003). Iterated local search. In Glover, F. e Kochenberger, G., editores, *Handbook of Metaheuristics*. Springer.
- Marinho, E. H. (2005). Heurísticas busca tabu para o problema de programação de tripulações de ônibus urbano. Dissertação de mestrado. Universidade Federal Fluminense, Niterói - RJ.
- Mesquita, M. e Paias, A. (2008). Set partitioning/covering-based approaches for the integrated vehicle and crew scheduling problem. *Computers and Operations Research*, 35(5):1562–1575.
- Patrikalakis, I. e Xerocostas, D. (1992). A new decomposition scheme of the urban public transport scheduling problem. In *Computer-Aided Transit Scheduling: Proceedings of the Fifth International Workshop*, pp. 407–425.
- Ribeiro, C. C. e Soumis, F. (1994). A column generation approach to the multiple-depot vehicle scheduling problem. *Operations Research*, 42:41–52.
- Saha, J. L. (1970). An algorithm for bus scheduling problems. *Operational Research Quarterly*, 21:463–474.
- Sampaio, R. B.; Sampaio, Y. e Sampaio, L. M. B. (2006). Eficiência de sistemas de transporte público no nordeste com análise envoltória de dados (DEA). *Revista Econômica do Nordeste*, 37(2):261–275.

- Santos, A. G. e Mateus, G. R. (2009). General hybrid column generation algorithm for crew scheduling problems using genetic algorithm. In *2009 IEEE Congress on Evolutionary Computation*, pp. 1799–1806.
- Shen, Y. e Kwan, R. S. K. (2001). Tabu search for driver scheduling. *Computer-Aided Scheduling of Public Transport*, pp. 121–135.
- Silva, G. P. (2001). Uma metodologia baseada na técnica de geração de arcos para o problema de programação de veículos. Tese de doutorado. Escola Politécnica da USP, São Paulo - SP.
- Souza, M. J. F.; Silva, G. P. e Simões, E. M. L. (2007). Programação de ônibus urbano: Uma abordagem heurística. In *Transporte em transformação XI: Trabalhos vencedores do prêmio CNT de Produção Acadêmica 2006*, volume 1, pp. 39–57. Positiva.
- Tosini, E. e Vercellis, C. (1988). An interactive system for extra-urban vehicle and crew scheduling problems. In *Computer-Aided Transit Scheduling: Proceedings of the Fourth International Workshop*, pp. 41–53.
- Tsang, E. (1993). Foundations of constraint satisfaction. Academic press.
- Wren, A. (1972). Bus scheduling: an interactive computer method. *Transportation Planning and Technology*, 1:115–122.

