

OTIMIZAÇÃO NO SISTEMA DE TRANSPORTE PÚBLICO

Gleudson Fonseca Soares

Universidade Federal de Ouro Preto
Campus Universitário – Morro do Cruzeiro, Ouro Preto – MG, CEP: 35400-000
gleudson_fonsecasoares@yahoo.com.br

Gustavo Peixoto Silva

Universidade Federal de Ouro Preto
Campus Universitário – Morro do Cruzeiro, Ouro Preto – MG, CEP: 35400-000
gustavo@iceb.ufop.br

Euler Horta Marinho

Universidade Federal de Ouro Preto
Campus Universitário – Morro do Cruzeiro, Ouro Preto – MG, CEP: 35400-000
eulerhm@yahoo.com.br

Emiliana Lopes Simões

Universidade Federal de Ouro Preto
Campus Universitário – Morro do Cruzeiro, Ouro Preto – MG, CEP: 35400-000
mililopessimes@yahoo.com.br

Marcone Jamilson Freitas Souza

Universidade Federal de Ouro Preto
Campus Universitário – Morro do Cruzeiro, Ouro Preto – MG, CEP: 35400-000
marcone@iceb.ufop.br

Resumo

Este trabalho apresenta os métodos de resolução e os resultados obtidos em um estudo de caso no qual é realizada a otimização dos veículos e das suas tripulações para um conjunto de empresas que atuam numa mesma região dentro de um sistema de transporte público. Para resolver tais problemas foi implementada a metaheurística *Iterated Local Search (ILT)* na otimização dos veículos e utilizada uma implementação da Busca Tabu para o PPT. Os procedimentos foram aplicados aos dados de todas as empresas que atuam na Bacia do Barreiro, uma das regionais metropolitanas de Belo Horizonte, MG. A análise dos resultados mostra possibilidades de redução de custos com a utilização de métodos de otimização e a necessidade de abordar os problemas de forma integrada. Este trabalho consiste na primeira etapa de um projeto maior que tem como objetivo resolver de forma integrada a programação dos veículos e tripulações no sistema de transporte público.

Palavras-Chaves: Programação de veículos e tripulações; Transporte público; Metaheurísticas.

Abstract

This paper presents the methodologies and results obtained in a study case developed for a set of companies operating in a same area into a transit mass system. In order to solve such problems, it was implemented the *Iterated Local Search (ILS)* metaheuristic for the vehicle scheduling problem and it was adopted the Tabu Search to solve the crew scheduling problem. The algorithms were applied to the data concerning to all companies operating on a same metropolitan area from Belo Horizonte city, in Minas Gerais state. The results show that it is possible to reduce costs and also points out that both problems should be solved in an integrated approach. This work presents relative results from a more width project, with has

as goal to solve the vehicle and the crew scheduling from transportation system in an integrated way.

Keywords: Vehicle and crew scheduling; Public transport; Metaheuristics.

1. INTRODUÇÃO

A otimização da alocação de veículos e tripulações dos sistemas de transportes públicos torna-se cada vez mais indispensável nas grandes cidades brasileiras. A crescente demanda por estes serviços acompanhada pela constante preocupação com a qualidade do serviço, com questões ambientais nos centros urbanos e influenciados por fatores econômicos, justificam qualquer iniciativa de pesquisas nesta área. As entidades que fazem parte do sistema, sejam as empresas operadoras, assim como a sua gestora, estão sempre empenhadas em melhorar seus processos produtivos mantendo a qualidade dos serviços oferecidos e diminuindo os seus custos.

O planejamento de um Sistema Municipal de Transporte Público é bastante complexo e, conseqüentemente, é dividido em alguns subproblemas. Os dados de saída referentes a resolução de um subproblema são os dados de entrada do subproblema seguinte. As principais etapas deste processo são: o planejamento das rotas, a definição da tabela de horários, a programação dos veículos, a programação das tripulações e finalmente o rodízio das tripulações.

Neste trabalho são apresentados os resultados relativos à resolução do Problema de Programação dos Veículos (PPV) e o Problema de Programação das Tripulações (PPT) de um conjunto de empresas que atuam na região do Barreiro em Belo Horizonte. Os gastos das empresas são fortemente influenciados pela forma como estas programações são geradas. A maior parte destes custos é decorrente da remuneração da tripulação e da manutenção da frota empenhada ([1]). Assim, o estudo deste caso visa testar o comportamento e os resultados obtidos pelos modelos de resolução do PPV e do PPT, desenvolvidos pelos autores em trabalhos anteriores, quando aplicados a uma massa de dados de grande porte.

O Problema de Programação de Veículos (PPV) consiste em definir a frota em operação com o menor custo possível. Já o Problema de Programação de Tripulações (PPT) está voltado para a alocação mais econômica dos motoristas e cobradores que conduzirão a frota empenhada na operação. Estes problemas, da forma como são abordados neste caso, pertencem à classe dos problemas NP-difíceis. Nestes casos, são utilizados métodos heurísticos para a resolução dos mesmos.

O sistema inglês BOOST ([2]) contém um dos mais importantes métodos heurísticos para a resolução do problema de programação de veículos e tem servido como referência na área. Neste método, as viagens são ordenadas de acordo com seus horários de partida e as n primeiras viagens são atribuídas aos n veículos da garagem. O número n de veículos pode ser fornecido pelo usuário ou calculado automaticamente pelo modelo. Após verificar cada uma das viagens, o sistema procura a próxima viagem que pode ser executada pelo mesmo veículo e atribui esta viagem ao veículo. Se nenhuma viagem posterior puder ser executada pelo veículo, ele retorna à garagem. É possível que a programação inicial tenha algumas ligações infactíveis de sobreposição de viagens executadas por um mesmo veículo. Neste caso, a programação é refinada através de um processo iterativo, em que cada par de ligações são quebrados e religados no sentido oposto. O sistema retorna a melhor solução encontrada considerando o número de veículos escolhido anteriormente. Caso este número seja muito baixo, são apontadas as viagens cujos horários devam ser alterados, tal que a programação com o número de veículos especificado seja factível. Na próxima iteração, o número de veículos é aumentado de uma unidade ou de acordo com a opção do usuário e o sistema volta a aplicar o método de 2-ótimo produzindo uma nova solução. Este processo continua até que todas as ligações infactíveis sejam eliminadas.

Baita et al. [3] abordam um caso prático do PPV comparando o desempenho do modelo tradicional de designação com duas novas heurísticas propostas para o problema. O

modelo de designação é resolvido com o algoritmo Húngaro descrito por Carpaneto et al. [4]. Neste modelo, a função de custo considera as viagens mortas entre os terminais, mudanças de linhas e o tempo de espera nos terminais. A primeira heurística proposta é baseada na técnica de Inteligência Artificial ou Programação em Lógica. Uma vez que o PPV pode ser visto como um problema de otimização multiobjetivo, o algoritmo tenta conectar as viagens mais próximas que pertencem à mesma linha e ao mesmo terminal, satisfazendo os tempos mínimos e máximos de terminal. Se a primeira estratégia não cobrir todas as viagens, então a restrição de pertinência à mesma linha é retirada, e são conectadas viagens que partem do mesmo terminal. Se for necessário, são conectadas viagens que partem de diferentes terminais, iniciando a busca pelos terminais mais próximos.

No algoritmo genético proposto em Baita et al. [3], uma programação é representada por um gene, sendo que cada alelo representa um veículo, e sua posição representa a viagem atribuída ao veículo. Para manter a factibilidade da solução após uma operação de cruzamento, aplica-se um mecanismo específico de reparação. A escolha dos indivíduos a serem reproduzidos na próxima geração é feita utilizando dois procedimentos não convencionais, além do método tradicional, segundo o qual a probabilidade de um indivíduo ser selecionado é proporcional ao valor da função objetivo.

Dada a complexidade do PPT, os sistemas heurísticos foram os primeiros a serem utilizados e com o surgimento das metaheurísticas, tais como Algoritmos Genéticos ([5]), Busca Tabu ([6]) e *Simulated Annealing* ([7]) abriu-se um novo horizonte na resolução deste problema. Embora tais métodos também não garantam a obtenção do ótimo global, eles contêm mecanismos capazes de escapar de ótimos locais. Na literatura destacam-se os trabalhos de Clement e Wren [8], Wren e Wren [9] e Kwan et al. [10] que utilizam Algoritmos Genéticos, enquanto Shen e Kwan [11] utilizam Busca Tabu.

Lourenço et al. [12] utilizam as metaheurísticas Busca Tabu e Algoritmo Genético para selecionar soluções não dominadas, segundo o conceito multiobjetivo. As colunas de tais soluções, geradas ao longo do processo, são armazenadas para constituir o domínio sobre o qual é aplicado um método exato para o problema de particionamento, quando se tratar de instâncias de pequeno porte. Para instâncias mais elevadas, é aplicado um método GRASP ([13]), desenvolvido pelos autores para esse fim.

No âmbito nacional, pode-se destacar uma série de trabalhos que abordam casos práticos por meio de heurísticas e metaheurísticas. Yunes [14] explora técnicas de programação por restrições e a resolução independente dos problemas de programação dos veículos e das tripulações. Silva et al. [15] utilizam as metaheurísticas *Simulated Annealing* para resolver o PPT. No trabalho de Souza et al. [16] três métodos são comparados a partir do uso de três estruturas diferentes de vizinhança: *Simulated Annealing*, Busca Tabu e o Método de Pesquisa em Vizinhança Variável. Mostrou-se que este último método foi o mais eficiente, tanto em qualidade de solução final, quanto em rapidez em apresentar uma solução de boa qualidade. Marinho [17] explora a utilização da Busca Tabu com relaxação adaptativa para resolver o PPT. Mauri [18] utiliza algoritmos genéticos com diferentes procedimentos de cruzamento e melhoria da solução.

No presente trabalho, foi desenvolvida uma metaheurística do tipo *Iterated Local Search - ILS* ([19]) para resolver o PPV, a qual incorpora restrições que não podem ser incluídas nos modelos exatos para o problema. Já para a resolução do PPT foi utilizado o algoritmo desenvolvido por Marinho [17].

2. PROGRAMAÇÃO DOS VEÍCULOS

O Problema de Programação de Veículos (PPV) em um Sistema de Transporte Público consiste em determinar o número mínimo de veículos necessários para realizar um dado conjunto de viagens e definir a seqüência de viagens a serem executadas por cada veículo da frota mínima, tal que o custo da operação seja minimizado. Os dados de entrada para o problema é um conjunto de viagens contidas na tabela de horários definida pela

BHTRANS. As viagens são caracterizadas pelos seguintes atributos: *Ponto de Partida*, *Horário de Partida*, *Ponto de Chegada*, *Horário de Chegada*, *Tempo de Embarque*, *Tempo de Desembarque* e *Linha* a que pertence. Além do conjunto de viagens, devem ser fornecidas também informações relativas às distâncias entre os vários pontos iniciais e finais de cada linha e a garagem. De posse dessas informações, é possível determinar uma solução para o problema.

A resolução do PPV tem como saída um *Bloco de Viagens* para cada veículo da frota mínima. Um bloco de viagens contém todas as viagens a serem realizadas por um veículo desde o momento que ele parte da garagem até o momento que ele retorna a garagem no final de sua jornada de trabalho. Além dos atributos que caracterizam cada viagem, os blocos apresentam dados de qualidade da solução, que são: *Tempo Total de Viagem Vazia* e *Tempo Total de Terminal*. As viagens vazias se referem ao deslocamento dos veículos entre dois terminais ou entre uma garagem e os terminais, estando o veículo fora de operação. O tempo de terminal indica quanto tempo o veículo permanece fora de operação parado no terminal entre duas viagens consecutivas.

A resolução do PPV considera um conjunto de restrições intrínsecas ao problema e que podem variar de acordo com regras e políticas operacionais definidas pela empresa. As restrições a serem atendidas são dispostas em dois grupos: *Essenciais* e *Não Essenciais*. As restrições essenciais são aquelas que devem ser satisfeitas obrigatoriamente para que a solução seja factível. As restrições não essenciais são aquelas que levam à melhoria na qualidade da solução.

As *Restrições Essenciais* levadas em conta neste estudo são:

- a) Cada veículo pode operar no máximo 23h30min durante o dia;
- b) Um veículo não pode executar duas viagens ao mesmo tempo;
- c) Antes de iniciar e após terminar uma dada viagem, o veículo deve cumprir os respectivos tempos de embarque e desembarque relativo à viagem;
- d) Um veículo não pode permanecer por mais de duas horas no terminal para cumprir a próxima viagem. Caso isso ocorra o veículo deverá se recolher para a área de estocagem;

As *Restrições Não Essenciais* são:

- a) O número total de veículos utilizados na programação deve ser minimizado;
- b) A ociosidade de cada veículo deve ser reduzido ao máximo;
- c) O tempo total de viagens mortas de cada veículo deve ser minimizado;
- d) O total de pegadas duplas não pode exceder um dado valor pré-estabelecido;
- e) O total de trocas de linhas realizadas durante a jornada de trabalho de cada veículo deve ser reduzido ao máximo.

3. PROGRAMAÇÃO DAS TRIPULAÇÕES

O Problema de Programação de Tripulações (PPT) em um Sistema de Transporte Público consiste na definição da jornada de trabalho de cada *Tripulação* pertencente ao quadro de funcionários de uma dada empresa. Uma tripulação é constituída por um motorista e um cobrador, responsáveis pela operação de um veículo. As tarefas são formadas a partir da solução do PPV, ou seja, dos blocos dos veículos, agrupando conjuntos de viagens de um mesmo veículo. Este agrupamento é feito de maneira que cada tarefa fica compreendida entre duas *Oportunidades de Trocas (OT)*. Uma OT é um intervalo de tempo suficiente, em um ponto apropriado, que possibilita a troca de tripulações de um veículo em operação. Durante a realização de uma tarefa, não é possível que haja troca de tripulação.

Assim, a jornada diária de trabalho de uma tripulação é constituída por um conjunto de tarefas. As jornadas são classificadas em dois tipos: a *Pegada Dupla* que é caracterizada por ter um intervalo superior a 2 horas entre duas tarefas consecutivas. Este tipo de jornada atua nos horários de pico e o intervalo livre entre as duas pegadas não é contabilizado na remuneração da tripulação. Na *Pegada Simples*, os intervalos entre tarefas são inferiores a 2

horas e são contabilizados na remuneração da tripulação. A duração normal de pegada dupla é de 06h40min de trabalho. No caso de pegada simples, a duração é de 07h00min contendo um intervalo de 20 minutos para descanso e alimentação.

Durante a alocação das tarefas para a formação das jornadas, uma série de restrições operacionais e trabalhistas deve ser satisfeitas e ao mesmo tempo o custo envolvido na execução da escala diária deve ser o menor possível. Assim como na programação dos veículos, as restrições se dividem em dois grupos. As *Restrições Essenciais* atendidas neste trabalho são as seguintes:

- a) Uma tripulação não pode executar duas tarefas ao mesmo tempo;
- b) Uma tripulação não pode realizar duas tarefas consecutivas onde o ponto final da primeira esteja em um terminal diferente do ponto inicial da segunda e o intervalo entre elas seja menor do que duas horas;
- c) Uma tripulação só pode realizar a troca de veículos durante uma pegada simples caso haja um tempo entre as tarefas maior ou igual ao tempo mínimo necessário para a troca das tripulações;
- d) Uma tripulação pode exercer, no máximo, duas horas diárias de trabalho além da duração normal de sua jornada, ou seja, pode ter, no máximo, duas horas extras por dia;
- e) Toda jornada do tipo pegada simples deve ter uma folga mínima de 20 minutos, podendo esta ser fracionada em dois intervalos maiores ou iguais a 10 minutos corridos;
- f) O intervalo mínimo entre o final de uma jornada e o seu início no dia seguinte deverá ser de 11 horas;
- g) O total de tripulações com jornadas do tipo pegada dupla não pode exceder um certo número pré-fixado;

As *Restrições Não Essenciais* são:

- a) O número de trocas de pontos permitidas, contidas nas jornadas deve ser minimizado;
- b) O total de trocas de veículos permitidas realizadas durante as jornadas de trabalho deve ser reduzido ao máximo;
- c) O total de jornadas de trabalhos deve ser minimizado;
- d) O total de horas extras deve ser minimizado;
- e) A ociosidade em cada jornada deve ser reduzida tanto quanto possível. Esta é calculada pela diferença entre o tempo efetivamente trabalhado pelas tripulações (o somatório da duração de cada tarefa) e a duração da sua jornada de trabalho.

As restrições Essenciais e Não Essenciais descritas anteriormente têm como objetivo cumprir a Legislação e a Convenção Coletiva de Trabalho firmado entre o Sindicato das Empresas de Transporte de Passageiros de Belo Horizonte – SETRABH e o Sindicato dos Trabalhadores em Transportes Rodoviários de Belo Horizonte – STTRBH.

4. PROBLEMA DA BACIA DO BARREIRO

O conjunto de linhas de transporte urbano que atendem a regional do Barreiro, localizado na capital mineira, é denominado de Bacia do Barreiro e conta com 8 empresas responsáveis pela realização de todas as viagens desta região. Juntamente, elas prestam serviços para 40 linhas de ônibus sendo 14 *Linhas Trocais* e 26 *Linhas Alimentadoras*. Linhas troncais são linhas de transporte público com modos de maior velocidade e capacidade que têm como o objetivo de realizar o deslocamento de passageiros entre duas ou mais regiões da cidade. Essas linhas operam em corredores de ônibus onde há alta demanda. Linhas alimentadoras coletam e distribuem passageiros pelas regiões vizinhas às linhas troncais. Os usuários podem ser coletados em uma determinada região e desembarcados em um terminal

da linha troncal ou podem ser recolhidos no terminal da linha troncal e distribuídos em regiões atendidas por esta linha. Portanto, as linhas alimentadoras têm a função primordial de captação e distribuição da demanda e juntamente com as troncais, complementam o transporte dos passageiros.

Na bacia estão presentes 2 terminais, o Barreiro com 12 plataformas de embarque e o Diamante com 10 plataformas de embarque. O terminal Barreiro (B) possui 22 pontos de início/final de linha e o terminal Diamante (D) conta com 13. Além disso, a bacia possui mais 3 pontos localizados fora dos terminais. São eles: PC1145 (PC1), PC3051 (PC2) e PC3054 (PC3).

Para cada empresa foram utilizados 3 conjuntos diferentes de dados de entrada referentes a dias úteis, sábados e domingos operados em maio de 2005. A Tabela 1 apresenta os dados referentes às empresas que atuaram na bacia na referida data.

Tabela 1 – Dados da situação da bacia do Barreiro

Empresa		E1	E2	E3	E4	E5	E6	E7	E8
Terminais		B e D	PC1	B, D e PC3	B e D	B e D	B e D	B e PC2	B
Dia útil	Total de linhas	2	1	4	7	13	7	5	1
	Total de viagens	260	98	505	468	1.038	639	293	206
	Tempo Total em viagem	348:10	141:20	516:56	387:05	773:58	622:30	253:34	296:41
Sábado	Total de linhas	2	1	4	7	13	6	4	1
	Total de viagens	172	69	372	359	769	441	191	130
	Tempo Total em viagem	217:16	91:13	379:55	282:50	541:49	410:20	147:36	182:50
Domingo	Total de linhas	2	1	4	7	13	6	3	1
	Total de viagens	90	52	286	298	538	332	115	108
	Tempo Total em viagem	124:27	64:33	281:17	221:04	336:21	316:15	60:13	147:14

Cada empresa é responsável por um conjunto de linhas cujo controle diário ocorre nos terminais das respectivas linhas. De acordo com a Tabela 1, a empresa E1 opera linhas com pontos de controle no Barreiro e no Diamante. A empresa E2, por sua vez, é responsável por uma linha cujo ponto de controle é no PC1. Para cada empresa foi listado na Tabela 1, o número de linhas de ônibus atendido pela mesma, o total de viagens a serem cumpridas nos respectivos dias e o tempo total em viagem, o qual é calculado a partir dos horários iniciais e finais de cada viagem. É bom ressaltar que o tempo total em viagem é menor do que o tempo total da frota em operação, pois o primeiro não inclui os tempos de deslocamento ocioso e nem os tempos de terminal, os quais devem ser minimizados.

5. O MÉTODO DE RESOLUÇÃO DO PPV

Para a resolução do PPV relativo à Bacia do Barreiro foi implementada uma versão do método *Iterated Local Search* proposto por Glover e Kochenberger [19], cujo pseudocódigo é apresentado na Figura 1. ILS é um método aplicado em conjunto com um procedimento de busca local com a finalidade de melhorar as soluções obtidas por este através da realização de perturbações. Uma perturbação consiste em modificações realizadas sobre uma solução por meio da aplicação de um ou mais tipos de movimentos.

Para iniciar a sua exploração o ILS precisa primeiramente de uma boa solução para o problema, para tanto, é gerada uma solução inicial que é melhorada por meio de um procedimento de busca local. A geração da solução inicial é feita através de um procedimento construtivo guloso. Esse procedimento consiste na alocação, a cada passo, de uma nova viagem de responsabilidade da empresa no bloco de algum veículo pertencente à frota. Durante a alocação, a combinação das viagens de cada veículo deve ser a que forneça o menor custo de operação possível. Esse custo é determinado pela aplicação de uma função de avaliação $f(.)$. Em seguida, são utilizadas duas variantes do Método de Descida: Descida com Realocação e Descida com Troca, que se diferenciam quanto ao tipo de movimento utilizado para realizar a busca.

Obtida uma solução inicial, é iniciada a fase de pesquisa sobre a sua vizinhança. Obtem-se uma solução vizinha s' da solução corrente s através de uma perturbação utilizando os movimentos de realocação e troca. Para refinar a solução perturbada, o procedimento de busca local utilizado é o Método Randômico de Descida (MRD), usado em duas versões distintas: uma que realiza apenas movimentos de troca (*DescidaRandomicaComTroca*), e outra que executa somente movimentos de realocação (*DescidaRandomicaComRealocacao*). A solução refinada é um ótimo local da solução perturbada. Este ótimo é então avaliado e caso seja satisfeito o critério de aceitação, ele será considerado a nova solução corrente e recomeça-se o processo de pesquisa pela vizinhança dessa nova solução. Caso o critério de aceitação não seja satisfeito, é aumentado o nível de perturbação, ou seja, torna-se a perturbação mais intensa e o processo de pesquisa é retomado. Este procedimento é repetido até que um critério de parada seja satisfeito.

Há doze níveis de perturbação, que se diferenciam tanto pelos tipos de movimentos realizados, quanto pelo número de vezes que estes são aplicados. Para cada nível k ímpar, aplica-se $\lfloor k/2 \rfloor + 1$ movimentos de realocação. Para os níveis k pares, aplicam-se $k/2$ movimentos de troca. Maiores detalhes desta implementação podem ser obtidos em Simões e Souza [20].

Algoritmo ILS

1: Entrada: $Tmax$,	{ Tempo máximo admissível de processamento }
2: $CiclosMax$,	{ Número máximo de ciclos permitidos }
3: $NumNiveis$;	{ Número máximo de níveis de perturbação }
4: $T \leftarrow 0$;	{ Tempo de processamento corrente do algoritmo }
5: $ciclos \leftarrow 0$;	{ Número de ciclos realizados }
6: $nivel \leftarrow 1$;	{ Nível de perturbação corrente }
7: $s \leftarrow GerarSolucaoInicial()$;	
8: $s \leftarrow DescidaComRealocacao(s)$;	
9: $s \leftarrow DescidaComTroca(s)$;	
10: $s^* \leftarrow s$;	{ Melhor solução até então }
11: enquanto (($ciclos < CiclosMax$) e ($T < Tmax$)) faça	
12: $s' \leftarrow Perturbacao(s^*, nivel)$;	
13: $s' \leftarrow DescidaRandomicaComRealocacao(s')$;	
14: $s' \leftarrow DescidaRandomicaComTroca(s')$;	
15: se ($f(s') < f(s^*)$) então	

```

16:    $s^* \leftarrow s'$ ;
17:    $nivel \leftarrow 1$ ;
18:    $ciclos \leftarrow 0$ ;
19:   fim-se
20:   se ( $f(s') = f(s^*)$ ) então
21:      $s^* \leftarrow s'$ ;
22:      $nivel \leftarrow nivel + 1$ ;
23:     se ( $nivel > NumNiveis$ ) então
24:        $ciclos \leftarrow ciclos + 1$ ;
25:        $nivel \leftarrow 1$ ;
26:     fim-se
27:   fim-se
28:   se ( $f(s') > f(s^*)$ ) então
29:      $nivel \leftarrow nivel + 1$ ;
30:   fim-se
31:   se ( $nivel > NumNiveis$ ) então
32:      $ciclos \leftarrow ciclos + 1$ ;
33:      $nivel \leftarrow 1$ ;
34:   fim-se
35: fim-enquanto
36: retorne  $s^*$ ;
Fim ILS;

```

Figura 1 – Algoritmo Iterated Local Search aplicado ao PPV

6. O MÉTODO DE RESOLUÇÃO DO PPT

Para a resolução do PPT, utilizou-se o método BTAR-FI-IC proposto por Marinho [17]. Este método, cujo pseudocódigo é apresentado na Figura 2, é baseado no método de Busca Tabu acrescido de um mecanismo de Relaxação Adaptativa.

Partindo de uma solução inicial, a cada interação, o método BTAR-FI-IC explora um subconjunto V da vizinhança $N(s)$ da solução corrente s . Neste método, o melhor vizinho é procurado dentro de um percentual reduzido da vizinhança, no caso igual a 50%. A vizinhança é composta pelo conjunto $N^R(s)$, baseada na realocação de tarefas e pelo conjunto $N^T(s)$ decorrente dos movimentos de troca de tarefas. A busca nessa vizinhança é interrompida quando um movimento que melhore a solução corrente for encontrado. A idéia de reduzir o espaço de busca se deve ao fato da interação do método de Busca Tabu ser custosa computacionalmente para o PPT. Essa interrupção da busca, conhecida como primeiro de melhora (ou *First Improvement - FI*), é uma idéia aplicada em métodos heurísticos e visa reduzir o esforço computacional para explorar a vizinhança.

A escolha do melhor vizinho a cada interação é utilizada para tentar escapar de mínimos locais. Porém, este critério de escolha pode fazer com que o algoritmo entre em ciclo, ou seja, que retorne a uma solução já visitada anteriormente. Para evitar a ciclagem, existe um mecanismo chamada memória de curto prazo ou lista tabu. O objetivo dessa lista é tentar evitar movimentos que levem a regiões já visitadas, o que usualmente é alcançado pela proibição dos últimos movimentos realizados. Esses movimentos são mantidos nesta lista por um certo número de iterações. Durante estas iterações, os movimentos armazenados ficam proibidos de serem realizados. Mas essa estratégia pode ser muito restritiva e desconsiderar soluções de alta qualidade. Para que isto não ocorra, movimentos com *status* tabu podem ser aceitos se a nova solução produzida satisfizer ao critério de aspiração por objetivo, isto é, se a solução, ainda que tabu, for melhor que a melhor solução gerada até então. De forma a reduzir ainda mais a probabilidade de ciclagem, o método BTAR-FI-IC utiliza uma lista tabu de tamanho dinâmico.

O mecanismo de Relaxação Adaptativa utilizado pelo método BTAR-FI-IC é baseado naquele descrito por Schaerf [21], onde os pesos para cada fonte de inviabilidade são ajustados dinamicamente, como proposto originalmente por Gendreau et al. [22].

Para cada inviabilidade i o peso β_i é multiplicado por um fator σ_i que varia de acordo

com o seguinte esquema:

- 1) No início da busca $\sigma_i \leftarrow 1$.
- 2) A cada k movimentos:
 - Se todas as k soluções visitadas são factíveis em relação à restrição i então $\sigma_i \leftarrow \sigma_i / \gamma$.
 - Se todas as k soluções visitadas são infactíveis em relação à restrição i então $\sigma_i \leftarrow \sigma_i \times \gamma$.
 - Se algumas soluções são factíveis e outras são infactíveis, considerando a restrição i , então σ_i permanece inalterado.

O parâmetro γ é randomicamente selecionado, a cada k movimentos, no intervalo $[1,8; 2,2]$ conforme proposto por Schaerf [21]. Cada valor de σ_i é limitado por duas constantes σ_{min} e σ_{max} , de forma a evitar que a relaxação adaptativa incremente/decremente indefinidamente os pesos para restrições que são sempre insatisfeitas/satisfeitas.

Algoritmo BTAR-FI-IC

```

1:  $s \leftarrow GerarSolucaoInicial()$ ;
2:  $s^* \leftarrow s$ ;                                     { Melhor solução até então }
3:  $ListaTabu \leftarrow \emptyset$ ;                       { Lista Tabu }
4:  $Iter \leftarrow 0$ ;                                  { Contador do número de iterações }
5: repita
6:    $Iter \leftarrow Iter + 1$ ;
7:   Selecione a vizinhança corrente  $N(s) \in \{ N^R(s), N^T(s) \}$ ;
8:   Defina um subconjunto  $V \subset N(s)$ ;
9:    $melhorMovimento \leftarrow movimentoRandomico(V)$ ;   { Melhor Movimento em  $V \subset N(s)$  }
10:   $melhorCustoDinamico \leftarrow \infty$ ;              { Melhor custo dinâmico }
11:  para todo ( movimento  $m \in V$  ) faça
12:    se (  $f(s \oplus m) < f(s^*)$  ) então
13:       $melhorMovimento \leftarrow m$ ;
14:      interromper;
15:    senão
16:      se (  $m \notin ListaTabu$  ) então
17:        se (  $f(s \oplus m) < f(s)$  ) então
18:           $melhorMovimento \leftarrow m$ ;
19:          interromper;
20:        senão
21:          se (  $custoPorPesosDinamicos(s \oplus m) < melhorCustoDinamico$  ) então
22:             $melhorMovimento \leftarrow m$ ;
23:             $melhorCustoDinamico \leftarrow custoPorPesosDinamicos(s \oplus m)$ ;
24:          fim-se
25:        fim-se
26:      fim-se
27:    fim-se
28:  fim-para
29:   $s \leftarrow s \oplus melhorMovimento$ ;
30:  se (  $f(s) < f(s^*)$  ) então
31:     $s^* \leftarrow s$ ;
32:  fim-se
33:   $AtualizarListaTabu()$ ;
34:  se  $IteracaoAtualizacao()$  então
35:     $AtualizarPesosDinamicos()$ ;
36:  fim-se
37: até que ( Critério de parada seja satisfeito )
38: retorne  $s^*$ ;
Fim BTAR-FI-IC;

```

Figura 2 – Algoritmo de Busca Tabu com Relaxação Adaptativa aplicado ao PPT

7. RESULTADOS COMPUTACIONAIS

Os algoritmos utilizados para solucionar o PPV e PPT aplicados ao problema da Bacia do Barreiro, conforme descrito anteriormente, foram desenvolvidos na linguagem C++, utilizando o ambiente de desenvolvimento Borland C++ Builder, versão 6.0. Os testes foram executados em um microcomputador com processador Intel Pentium IV, 2.4 GHz, com 512 MB de memória RAM, sob o sistema operacional Microsoft Windows XP Service Pack 2.

A seguir são apresentados os resultados obtidos tendo como entrada os dados reais das oito empresas que atuaram na bacia do Barreiro e considerando o tempo mínimo de troca de tripulações contidos na Tabela 2. Assim, uma viagem que inicia às 04h00min deve chegar à plataforma às 03h59min. Da mesma forma, para que haja a troca entre duas tripulações de um veículo, no intervalo entre as 05h00min e as 07h59min são necessários no mínimo 2 minutos de terminal entre as viagens deste veículo. Os tempos para o embarque e desembarque são considerados no início e no final das viagens respectivamente.

Tabela 2 – Tempo mínimo para embarque/desembarque e troca de tripulações

Faixa de horário	00:00-04:59	05:00-07:59	08:00-15:59	16:00-18:59	19:00-23:59
Tempo	1	2	1	2	1

Para que fosse possível obter uma análise consistente dos dados, cada instância foi executada 10 vezes, cada qual partindo de uma semente diferente de números aleatórios, com uma duração máxima de execução de 60 minutos.

7.1. PROBLEMA DE PROGRAMAÇÃO DE VEÍCULOS – PPV

Na Tabela 3 são apresentadas as soluções utilizadas por cada uma das empresas, denominada “Manual”, e a melhor solução obtida pelo método ILS descrito na seção 5. Essas programações são comparadas em termos da função de avaliação baseada na penalização das restrições não atendidas. Os pesos atribuídos às restrições essenciais são tais que forçam a eliminação de inviabilidades e a melhoria da qualidade das soluções. O limite para o número de veículos com pegadas duplas foi de 60% sobre o total de veículos utilizados na solução da empresa. Foram analisadas as soluções referentes a um dia útil, um sábado e um domingo para cada uma das oito empresas.

Tabela 3 – Programação não otimizada versus programação otimizada

Empresa	Dia útil		Sábado		Domingo	
	Manual	ILS	Manual	ILS	Manual	ILS
E1	3.450	2.506	1.921	1.491	1.027	840
E2	1.343	1.019	809	655	527	370
E3	5.429	4.100	3.874	2.733	2.394	1.900
E4	3.605	3.102	3.098	2.686	2.514	2.285
E5	8.163	7.106	5.476	4.373	4.132	3.586
E6	6.564	4.624	3.935	2.940	2.940	2.328
E7	2.155	1.813	1.347	1.248	902	791
E8	2.183	1.651	1.635	1.117	1.645	926

Com base nos resultados apresentados na Tabela 3, verifica-se que a heurística ILS supera todas as soluções propostas pelas empresas. As reduções em relação aos valores da função objetivo variam entre 13,95% para a empresa E4 e 29,56% para a empresa E6 no dia

útil, 7,35% para E7 e 31,68% para E8 no sábado e finalmente de 12,31% para a empresa E7 e 43,71% para a empresa E8 no domingo. Para uma análise mais detalhada são apresentadas, na Tabela 4, as características das soluções de três empresas dentre as oito que participaram do estudo. Essas soluções são comparadas em relação ao número de veículos utilizados (NV), o tempo de espera no terminal (TE), tempo de viagem morta (TVM) e o número de duplas pegadas (NDP).

Tabela 4 - Detalhamento das programações de algumas empresas

Empresa	Atributos	Dia útil		Sábado		Domingo	
		Manual	ILS	Manual	ILS	Manual	ILS
E3	NV	53	52	38	29	28	22
	TE	100:19	44:44	89:42	42:35	69:35	52:13
	TVM	23:42	23:12	12:18	11:04	08:42	07:13
	NDP	26	22	3	5	1	1
E4	NV	34	33	27	24	21	19
	TE	123:24	70:24	94:06	60:03	84:54	59:12
	TVM	16:33	16:15	11:36	10:11	09:02	08:03
	NDP	12	8	6	2	5	1
E6	NV	63	62	45	32	35	26
	TE	162:54	79:47	108:58	72:53	95:50	63:01
	TVM	59:33	45:34	33:24	21:08	22:38	16:08
	NDP	40	13	13	2	6	0

Pode-se observar que o método de otimização conseguiu uma redução em quase todos os atributos destacados na programação dos veículos. Pode-se ressaltar as reduções no número de veículos, que corresponde ao principal componente na composição dos custos com a frota. Conforme observado nos dados de sábado da empresa E6, a redução no número de veículos foi de 28,9%. Na empresa E3, para a programação do sábado, houve uma redução de 23,7% e, novamente, na empresa E6, para os dados de domingo, houve uma redução de 25,7% no número de veículos.

7.2. PROBLEMA DE PROGRAMAÇÃO DE TRIPULAÇÕES – PPT

Os dados de entrada utilizados para a resolução do PPT foram a solução do PPV obtida pelo método ILS e as restrições trabalhistas descritas anteriormente. O número máximo de tripulações utilizado na construção da solução inicial do PPT foi de duas vezes e meia do total de blocos de veículos gerados para o PPV. O limite para o número de tripulações com pegadas duplas foi de 10% sobre o máximo de tripulações utilizados. Estes valores foram definidos para atender as regras adotadas pelas empresas. Para definir as tarefas das tripulações e classificar as trocas de veículos como permitidas ou proibidas, foram consideradas as variações de tempos mínimos descritas na Tabela 2.

As soluções obtidas pela metaheurística são direcionadas pelos pesos adotados na função de avaliação (penalização), a qual é uma composição das restrições essenciais e não essenciais para o problema. Este estudo de caso teve como objetivo diminuir o número de horas extras auferidos pelas tripulações. Assim, a condição de minimização do número de horas extras, embora seja uma restrição não essencial, foi tratada com o objetivo de atingir o valor nulo.

A Tabela 5 mostra as principais características das soluções das oito empresas que atuaram na Bacia do Barreiro. Os atributos considerados foram: número de tripulações utilizadas (NT) e total de tarefas executadas (NTA).

Tabela 5 - Programação das empresas que atuam na bacia do Barreiro

Empresa	Atributos	Dia útil	Sábado	Domingo
---------	-----------	----------	--------	---------

E1	NT	73	45	25
	NTA	208	153	84
E2	NT	27	19	12
	NTA	87	61	46
E3	NT	106	73	55
	NTA	416	322	250
E4	NT	81	60	48
	NTA	395	318	282
E5	NT	170	115	60
	NTA	839	656	491
E6	NT	130	80	64
	NTA	551	400	299
E7	NT	54	30	13
	NTA	253	187	111
E8	NT	58	35	28
	NTA	156	112	97

A Tabela 6 mostra detalhadamente as características das soluções das três empresas apresentadas na Tabela 4. Os atributos utilizados para avaliação foram: número de duplas pegadas (NDP), tempo total trabalhado pelas tripulações (TT), tempo total ocioso das tripulações (TO), horas extras das tripulações (TE) e total de trocas de linhas (TL), de pontos (TP) e de veículos (TV) efetuados pelas tripulações.

Tabela 6 - Detalhamento das programações de algumas empresas

Empresa	Atributos	Dia útil	Sábado	Domingo
E3	NDP	13	8	6
	TT	597:14	437:12	334:14
	TO	179:04	104:10	80:41
	TE	12:28	07:45	02:51
	TL	7	10	3
	TP	7	11	2
	TV	5	6	3
E4	NDP	9	6	5
	TT	473:50	354:10	295:32
	TO	143:50	112:49	94:41
	TE	06:37	05:08	02:59
	TL	17	17	18
	TP	16	18	19
	TV	2	3	8
E6	NDP	16	8	7
	TT	743:47	499:10	384:48
	TO	213:28	112:59	99:27

	TE	15:59	11:10	03:54
	TL	19	11	11
	TP	21	11	10
	TV	5	3	3

Pode-se observar que foi priorizada a redução no total de horas extras de cada tripulação procurando manter o número de tripulações utilizadas na solução inferior ao número máximo estabelecido inicialmente. Embora ainda não se tenha os dados das soluções adotadas pelas empresas que atuaram na bacia do Barreiro, comparando as soluções obtidas neste trabalho com aquelas praticadas e observadas *in loco*, pode-se assegurar que houve uma redução significativa na remuneração dispendida com horas extras. A diretriz adotada, de minimizar as horas extras, teve como consequência um aumento no número de horas ociosas e de tripulações, como era de se esperar.

8. CONCLUSÕES

Neste trabalho foi apresentado um estudo de caso que envolve a resolução do Problema de Programação dos Veículos e de suas respectivas Tripulações para um conjunto de empresas que atuam em uma mesma região da cidade de Belo Horizonte. Os métodos de resolução dos respectivos problemas estão baseados em metaheurísticas, uma vez que as restrições inerentes aos problemas dificultam a resolução por métodos de programação matemática, conhecidos como métodos exatos de otimização.

Os resultados para a programação dos veículos puderam ser comparados com aqueles adotados pelas empresas, uma vez que a BHTRANS, parceira neste estudo, disponibilizou tais informações. Assim, foi constatada uma grande margem de otimização, com a redução nos custos de todas as empresas. No quadro detalhado da seção 7.1 pode-se observar que houve uma redução em praticamente todos os componentes da função de avaliação.

Na resolução do problema de programação das tripulações foi priorizado minimizar o número de horas extras. Assim, embora não se conheça a prática das empresas, as soluções obtidas se mostram de excelente qualidade. Isto pode ser constatado para as empresas que tiveram seus dados detalhados na Tabela 6, sendo que o número de horas extras varia entre 3,1 e 8,4 minutos por tripulação.

Este trabalho explorou as metaheurísticas desenvolvidas em estágios anteriores para resolver uma série de problemas de operação de uma região do sistema de transporte público, no caso a região do Barreiro de Belo Horizonte. Esta é uma primeira etapa de um projeto maior que visa estudar a possibilidade de licitar as linhas de uma região para um conjunto de empresas, que operem de forma integrada. Desta forma, as linhas passam a pertencer a todas as empresas e uma determinada linha pode ser operada por mais de uma empresa ao mesmo tempo. Experimentos preliminares mostram que tal estratégia permite reduzir os custos do sistema.

9. REFERÊNCIAS BIBLIOGRÁFICAS

- [1] Bouzada, C. F. (2002) Análise das Despesas Administrativas no Custo do Transporte Coletivo por Ônibus no Município de Belo Horizonte. Dissertação de mestrado - Fundação João Pinheiro.
- [2] Kwan, R. K. e M. A. Rahin (1999) Object oriented bus vehicle scheduling - the BOOST system. *Computer-Aided Transit Scheduling*, Wilson, N. H. M. (ed.), Springer-Verlag, Berlin, p. 177-191.
- [3] Baita, F.; R. Pesenti; W. Ukovich e D. Favaretto (2000) A comparison of different solution approaches to the vehicle scheduling problem in a practical case. *Computers and Operations Research*, v. 27, p. 1249-1269.

- [4] Carpaneto, G.; S. Martello e P. Toth (1988) Algorithms and codes for the assignment problem. *Annals of Operations Research*, v. 13, p. 193-223.
- [5] Goldberg, D. E. (1989) Genetic Algorithms in Search. *Optimization and Machine Learning*, Addison-Wesley, Berkeley.
- [6] Glover, F. e M. Laguna (1997) *Tabu Search*, Kluwer Academic Publishers.
- [7] Kirkpatrick, S.; D. C. Gellat e M. P. Vecchi (1983) Optimization by Simulated Annealing. *Science*, v. 220, p. 671-680.
- [8] Clement, R. e A. Wren (1995) Greedy genetic algorithms, optimizing mutantis and bus driver scheduling. *Computer-Aided Transit Scheduling*, Daduna, J. R., I. Branco e J. M. P. Paixão (eds.), Springer-Verlag, Berlin, p. 213-235.
- [9] Wren, A. e D. O. Wren (1995) A genetic algorithm for public transport driver scheduling. *Computer and Operations Research*, 22, p. 101-110.
- [10] Kwan, R. S. K.; A. S. K. Kwan e A. Wren (2001) A genetic algorithm with fuzzy comprehensive evaluation for driver scheduling. *Evolutionary Computation*, 9, p. 445-460.
- [11] Shen, Y. e R. S. K. Kwan (2001) Tabu search for driver scheduling. Voss, S. e J. R. Daduna (eds.), *Computer-Aided Scheduling of Public Transport*, Springer-Verlag, p. 121-135.
- [12] Lourenço, H. R.; J. P. Paixão e R. Portugal (2001) Metaheuristics for the bus-driver scheduling problem. *Transportation Science*, 35(3), p. 331-343.
- [13] Feo, T. A. e M. G. C. Resende (1995) Greedy randomized adaptive search procedures. *Journal of Global Optimization*, 6, p. 109-133.
- [14] Yunes, T. H. (2000) Problemas de escalonamento no transporte coletivo: programação por restrições e outras técnicas. Dissertação de mestrado - Universidade de Campinas.
- [15] Silva, G. P.; M. J. F. Souza e J. M. C. B. Alves (2002) Resolução do Problema de Programação Diária da Tripulação de Ônibus Urbano via Simulated Annealing. *Anais do XVI Congresso de Pesquisa e Ensino em Transportes*, ANPET, v. 2, p. 95-104.
- [16] Souza, M. J. F.; L. X. T. Cardoso; G. P. Silva; M. M. S. Rodrigues e S. M. S. Mapa (2004) Metaheurísticas aplicadas ao Problema de Programação de Tripulações no Sistema de Transporte Público. *Tendências em Matemática Aplicada e Computacional*, v. 5.2, p. 357-368.
- [17] Marinho, E. H. (2005) Heurísticas Busca Tabu para o Problema de Programação de Tripulações de Ônibus Urbano. Dissertação de Mestrado - Universidade Federal Fluminense.
- [18] Mauri, G. R. (2005) Novas heurísticas para o problema de escalonamento de tripulações. Dissertação de mestrado - Instituto Nacional de Pesquisas Espaciais.
- [19] Glover, F. e G. Kochenberger (2003) *Handbook of Metaheuristics*. Kluwer Academic Publishers.
- [20] Simões e Souza (2006) Relatório Técnico - Universidade Federal de Ouro Preto.
- [21] Schaerf, A. (1996) Tabu search techniques for large high-school timetabling problems, Amsterdam.
- [22] Gendreau, M.; A. Hertz e G. Laporte (1994) A tabu search heuristic for the vehicle routing problem. *Management science*, Linthicum, v. 40, n. 10, p. 1276-1290.

