

UM ESTUDO DE DIFERENTES MÉTODOS DE BUSCA E A METAHEURÍSTICA VNS PARA OTIMIZAR A ESCALA DE MOTORISTAS DE ÔNIBUS URBANO

Alexandre Fortes da Silva Reis

Departamento de Engenharia de Produção
Universidade Federal de Ouro Preto

Gustavo Peixoto Silva

Departamento de Computação
Universidade Federal de Ouro Preto

RESUMO

Este trabalho explora diferentes métodos de busca associados à metaheurística *Variable Neighborhood Search-VNS* para resolver o Problema de Programação de Tripulações do Sistema de Transporte Público. Inicialmente a metaheurística VNS foi implementada utilizando o método clássico de busca *Variable Neighborhood Descent-VND*. Posteriormente foi utilizada a técnica *Very Large-scale Neighborhood Search-VLNS* como procedimento de busca. Esta técnica realiza uma busca em um espaço maior do que as buscas clássicas, permitindo a realocação de tarefas em uma série de diferentes tripulações. Consequentemente, suas soluções ótimas locais devem ser melhores do que aquelas obtidas pelo método clássico. As duas versões do VNS foram aplicadas a um conjunto de problemas de uma empresa que opera em Belo Horizonte, produzindo resultados melhores do que a solução adotada pela empresa. Os resultados mostram também que as soluções obtidas pelo método VNS-VLNS são mais econômicas e apresentam características mais apropriadas à operação do que aquelas obtidas pelo método clássico.

ABSTRACT

This paper explores different methods associated with Metaheuristic Variable Neighborhood Search-VNS to solve the Crew Scheduling Problem from Public Transportation Systems. Initially, the VNS was implemented using the classical Variable Neighborhood Descent-VND. Further, we used the Very Large-Scale Neighborhood Search-VLNS technique as local search procedure. This technique is able to search for better solutions in a larger space than the classic search, allowing tasks reallocation in a sequence of different crews. Consequently, local optimal solutions must be better than those obtained by classical search methods. Both versions of VNS have been applied to a set of problems from a company that operates in Belo Horizonte, producing better results than the solution adopted by the company. The results also show that the solutions obtained by the VNS-VLNS are more economical and present characteristics more appropriate to the operation than those obtained by the classical method.

1. INTRODUÇÃO

Atualmente as empresas de todos os setores da economia buscam usar ao máximo as inovações tecnológicas para melhorar o desempenho de seus processos. Mas as empresas do sistema de transporte público brasileiro fazem pouco uso de softwares e modelos de otimização para alocar os seus equipamentos e a mão de obra necessária para a sua operação, isto é, os veículos e as tripulações (motoristas e cobradores). Isso se deve porque o uso de modelos de otimização requer: o levantamento de dados precisos; o cumprimento das regras especificadas; a flexibilização da operação do sistema, entre outras práticas ainda pouco difundidas no setor. Por outro lado, em sistemas mais exigentes, como no sistema de transporte aéreo, o uso de modelos de otimização para montar as escalas de suas tripulações, é uma prática quase obrigatória. Além de possibilitar a geração de escalas viáveis, os modelos conseguem reduzir os custos com esta mão de obra, que no caso do transporte aéreo, são muito elevados. Dessa forma, o estudo e a implementação de métodos eficientes para a resolução do problema de programação de tripulações do sistema de transporte público, assim como a sua difusão no meio, são tarefas importantes não somente para reduzir os custos operacionais, mas também para disseminar a utilização de sistemas de apoio à decisão pela alta gerência das empresas que atuam neste setor.

O Problema da Programação das Tripulações (PPT) consiste em determinar o número mínimo de tripulações de forma a cobrir totalmente a programação de veículos realizada previamente.

A solução deste problema envolve o sequenciamento das atividades dos motoristas, gerando um conjunto de jornadas de trabalho. As jornadas devem satisfazer diversas restrições devido às leis trabalhistas, acordos sindicais e ainda as regras operacionais das empresas. Desta forma, o problema se torna *NP-hard*, para o qual não existe algoritmo polinomial que obtenha a solução ótima (Fischetti et al. 1987).

A abordagem clássica para tratar esse problema formula-o como um problema de programação linear inteira de recobrimento ou particionamento (*set covering* ou *set partitioning model*). Para resolver o problema é utilizada a estratégia de geração de colunas (Smith e Wren 1988, Desrochers e Soumis 1989, Fores et al. 1999, Barnhart et al 1998). Entretanto, modelos exatos são limitados quando aplicados a problemas de grande porte. Desta forma, torna-se fundamental a utilização de métodos heurísticos para resolver problemas práticos, os quais são normalmente de grande porte.

Um grupo precursor nesta área, denominado *Scheduling and Constraint Management Group* da Universidade de Leeds, realizou uma série de implementações heurísticas utilizando Algoritmos Genéticos (Kwan et al, 1999; Li e Kwan, 2003), Busca Tabu (Shen e Kwan, 2001), Colônia de Formigas (Forsyth e Wren, 1997) entre outras. Os modelos desenvolvidos por este grupo são largamente utilizados no Reino Unido tanto para realizar a programação das tripulações quanto da frota em operação (Wren, 2004).

Embora o Problema de Programação da Tripulação tenha sido largamente estudado e aplicado nos países mais desenvolvidos, suas técnicas de resolução são pouco difundidas e raramente aplicadas à realidade brasileira. Isso se deve, em parte, pela falta de dados e organização necessária, por parte das empresas do setor, como também pela carência de modelos que representem a realidade operacional brasileira.

Dentre os estudos voltados para a resolução do PPT na realidade brasileira, podemos destacar aqueles desenvolvidos por um dos autores deste trabalho, que utilizam as metaheurísticas *Simulated Annealing*, Busca Tabu, GRASP e VNS (Silva et al. 2002; Soares et al. 2006, Souza et al. 2004). Estas implementações foram testadas com dados de empresas que atuam no sistema brasileiro de transporte público e os resultados mostram que existem grandes possibilidades de redução dos custos em relação às soluções adotadas pelas empresas. Entretanto novas regras para o problema e modernas técnicas de busca surgiram nos últimos anos, e que podem ser utilizadas na resolução do problema. Desta forma, este trabalho explora a utilização de uma técnica recente de busca local baseada na representação em grafos e no uso de algoritmos de fluxo em redes para realizar buscas mais complexas do que aquelas inerentes aos procedimentos clássicos de busca local.

A técnica de busca denominada *Very Large-scale Neighborhood Search* - VLNS (Ahuja, 2000) foi empregada para resolver o PPT por Silva e Cunha (2010). O modelo desenvolvido utiliza a técnica VLNS como o procedimento de busca da metaheurística GRASP. Neste trabalho foi observado que o resultado de uma busca do tipo VLNS é fortemente dependente da solução inicial. Sendo assim, foi proposto adotar heurísticas construtivas que realizam perturbações periódicas na solução corrente através de movimentos diversos. Nesta classe destaca-se a heurística *Variable Neighborhood Search* (VNS), que consiste em explorar o espaço de soluções através de trocas sistemáticas de estruturas de vizinhança.

Assim, este trabalho tem como objetivo resolver o Problema de Programação da Tripulação utilizando a metaheurística VNS, combinada com a técnica de busca VLNS. Para verificar a

eficiência da combinação proposta, a metaheurística VNS também foi implementada em sua versão clássica, que usa o método *Variable Neighborhood Descent* (VND) tendo como procedimento de busca local subjacente o método da Primeira Solução de Melhora (*First Improvement Method*). As duas versões da metaheurística foram testadas com dados de uma empresa nacional que opera em um sistema de transporte público, e os diferentes resultados foram comparados permitindo uma avaliação quantitativa e qualitativa das soluções.

2. O PROBLEMA DE PROGRAMAÇÃO DAS TRIPULAÇÕES

Programar a realização de algum evento é uma atividade que se torna gradualmente mais difícil com o aumento da quantidade de itens a serem programados e também com o aumento das restrições do problema. Não diferente, a programação das tripulações do transporte urbano torna-se um problema cada vez mais difícil de resolver, devido às restrições trabalhistas e operacionais envolvidas. Assim, uma programação eficiente equivale à diminuição dos custos para a empresa, ao aumento da produtividade e à satisfação dos funcionários e dos usuários do sistema.

A escala da tripulação é criada a partir de uma programação previamente definida para os veículos. Sendo assim, a programação dos veículos, as regras operacionais e a legislação trabalhista são os dados de entrada do problema. A maior parcela dos seus custos é composta pela remuneração das tripulações, dessa forma, definir uma programação de tripulantes com custo mínimo leva a uma grande economia para a empresa.

A programação de uma tripulação é formada por um conjunto de tarefas e denominada de *jornada*, ou jornada diária da tripulação. Uma *tarefa* é um conjunto de viagens de um mesmo veículo que deve, necessariamente, ser realizado por uma mesma tripulação. O conjunto de todas as jornadas constitui numa escala para as tripulações, também dita programação das tripulações. As jornadas são divididas em dois tipos: Pegada Simples (PS) ou Dupla Pegada (DP). Na PS as tarefas são realizadas de uma única vez e os intervalos de tempo entre as tarefas são inferiores a duas horas. Caso ocorra um intervalo maior do que duas horas, a jornada é classificada como DP. Este tipo de jornada está associado aos picos de demanda por viagens existentes nos dias úteis e o intervalo maior do que duas horas não é remunerado.

Para agrupar tarefas e formar uma jornada, inúmeras restrições operacionais e normas trabalhistas devem ser levadas em conta. Neste caso foram consideradas as seguintes restrições: *i*) as jornadas têm uma remuneração por 6 horas e 40 minutos de trabalho; *ii*) as jornadas do tipo PS devem ter uma pausa de pelo menos 20 minutos para descanso e refeição; *iii*) as jornadas não podem conter mais do que duas horas-extras de trabalho; *iv*) o intervalo de tempo entre o final de uma jornada e o seu início no dia seguinte deve ser de pelo menos 11 horas; *v*) a troca de tripulações, durante a operação, só pode ocorrer em pontos e horas pré-determinados.

O modelo heurístico implementado neste trabalho procura minimizar os custos fixos e variáveis da escala, satisfazendo todas as restrições mencionadas acima. Os custos fixos são representados pelo número de jornadas e os custos variáveis são computados em função do total de horas extras encontrado na escala.

3. MÉTODO DE RESOLUÇÃO DO PROBLEMA

Para resolver o problema, foram implementadas duas versões da metaheurística VNS, proposto por Mladenović e Hansen (1997): o VNS Clássico, que utiliza como busca local a técnica *Variable Neighborhood Descent* – VND, e o VNS-VLNS, que utiliza a técnica *Very*

Large-scale Neighborhood Search como procedimento de busca local. As duas versões da metaheurística foram testadas resolvendo problemas de grande porte da realidade brasileira.

3.1. A Metaheurística VNS

A metaheurística VNS parte de uma solução inicial factível e de um conjunto de diferentes estruturas de vizinhança para realizar buscas locais. A partir da solução inicial, o procedimento gera um vizinho qualquer segundo uma estrutura de vizinhança e realiza uma busca local a partir deste vizinho. Se a busca local gerar uma solução melhor do que a solução corrente, esta é atualizada e o processo se repete a partir da primeira estrutura de vizinhança. Caso contrário, um novo vizinho é gerado considerando a próxima estrutura de vizinhança e a busca local é aplicada neste vizinho. Este processo se repete até que a condição de parada seja satisfeita. A melhor solução encontra neste processo é retornada pelo procedimento. O Algoritmo 1 sintetiza a metaheurística VNS.

Procedimento VNS(solução s)

Início:

1. Seja um conjunto de k_{max} estruturas de vizinhanças;
2. Encontre uma solução inicial s e determine uma condição de parada;
3. Enquanto a condição de parada não for satisfeita faça:
4. $k \leftarrow 1$;
5. Enquanto $k \leq k_{max}$ faça:
6. Gere um vizinho qualquer s' de s na k -ésima vizinhança;
7. Encontre o melhor vizinho s'' de s' na vizinhança k ;
8. Se $f(s'')$ for melhor do que $f(s)$
9. então $s \leftarrow s''$ e $k \leftarrow 1$;
10. senão $k \leftarrow k + 1$;
11. Fim_enquanto;
12. Fim_enquanto
13. Retorna a solução s ;

Fim

Algoritmo 1. Pseudo-código da Metaheurística VNS.

3.2. Método de Busca *Variable Neighborhood Descent* - VND

A heurística VNS clássica se caracteriza por tem o VND como o método de refinamento subordinado, utilizado para encontrar o melhor vizinho s'' de s' na linha 7 do Algoritmo 1. A busca aceita somente soluções melhores que a corrente, retornando à primeira estrutura de vizinhança se ocorrer uma melhora, senão vai para a próxima vizinhança. Uma síntese do procedimento é apresentada no pseudocódigo do Algoritmo 2.

Procedimento VND(solução s , vizinhança k)

Início

1. Seja s uma solução corrente dada;
2. Seja k uma estrutura de vizinhança dada;
3. $r \leftarrow 1$;
4. Enquanto $r \leq k$ faça:
5. Encontre o melhor vizinho s' de s na vizinhança r ;
6. Se $f(s')$ for melhor do que $f(s)$
7. então $s \leftarrow s'$ e $k \leftarrow 1$;
8. senão $r \leftarrow r + 1$;
9. Fim_enquanto
10. Retorna a solução s ;

Fim

Algoritmo 2. Pseudo-código do Procedimento VND.

De acordo com o pseudocódigo do VNS, a busca VND é realizada a partir de um vizinho da solução corrente e uma vizinhança k , fornecida ao procedimento como limite de variação das diferentes estruturas. Para a resolução do PPT, este parâmetro define o número máximo de tarefas consecutivas que serão removidas de uma jornada e introduzidas em outra. Neste trabalho foi adotado k variando de uma a três tarefas.

3.3. Estrutura de Vizinhança

Uma vez definida a quantidade máxima de tarefas movimentadas, é necessário estabelecer os diferentes tipos de movimento que caracterizam cada uma das vizinhanças de uma solução. Foram adotados dois tipos de movimentos: realocação e troca de tarefas entre duas jornadas, sem haver sobreposição. Estes movimentos são realizados para encontrar o melhor vizinho de uma solução corrente. Exemplificando, para um dado $k \leq 3$, considere duas jornadas i e j , obtidas aleatoriamente. Então são sorteadas k tarefas consecutivas da jornada i a serem introduzidas na jornada j . Logo, pode ocorrer uma das seguintes situações: 1) As k tarefas de i podem ser introduzidas em j , sem a necessidade de remover qualquer tarefa de j . Neste caso o movimento é aceito e será avaliado. 2) A introdução das k tarefas em j exige a retirada de uma ou mais tarefas desta jornada. Neste caso, se as tarefas removidas de j puderem ser reinseridas na jornada i , sem haver qualquer sobreposição com as tarefas que permaneceram em i , então o movimento é aceito, caso contrário ele é descartado. Em ambos os casos, as modificações são aceitas se e somente se as jornadas resultantes respeitarem todas as restrições do problema.

Para encontrar o melhor vizinho da solução corrente, dentro do VND, foi utilizado o método clássico de descida com Primeiro Movimento de Melhora Randômico (Mladenović e Hansen 1997), que além de evitar a exploração exaustiva da vizinhança, analisa as jornadas em uma ordem aleatória renovada a cada iteração. Isso evita que as primeiras jornadas da escala apresentem qualidade superior em detrimento da qualidade das últimas jornadas.

3.4. Método de Busca VLNS

Um aspecto crítico em algoritmos de busca em vizinhança diz respeito à escolha da estrutura da vizinhança, isto é, na maneira como ela é definida. Essa escolha define, em grande parte, se a estratégia de busca permitirá obter soluções de boa qualidade ou não. Em geral, quanto maior a vizinhança, melhor deverá ser a qualidade das soluções ótimas locais. Porém, vizinhanças de grande porte requerem um tempo elevado de pesquisa. Por essa razão, uma vizinhança maior não implica numa heurística melhor, exceto se a vizinhança for explorada de maneira eficiente.

Tais algoritmos são denominados *Very Large-scale Neighborhood Search Methods*, aplicáveis a problemas de particionamento (Ahuja et al, 2000). Esses algoritmos permitem explorar vizinhanças muito grandes, mantendo o tempo de busca em níveis bem reduzidos. Uma forma de alcançar tal eficiência é com a utilização de modelos de fluxo em rede para enumerar, de forma implícita, uma vizinhança, com a finalidade de encontrar soluções melhores.

Os métodos clássicos de busca em vizinhança se baseiam em realocação e trocas aos pares, de elementos entre os dois subconjuntos aos quais pertencem. Uma *troca cíclica* pode ser definida por uma sequência de elementos $i_1-i_2-i_3-...-i_r-i_1$ pertencentes a diferentes subconjuntos do particionamento. Considerando $S[i_k]$ o subconjunto ao qual pertence o elemento i_k , então a troca cíclica $i_1-i_2-i_3-...-i_r-i_1$ representa a alteração onde o elemento i_1 é movido de $S[i_1]$ para $S[i_2]$, o elemento i_2 de $S[i_2]$ para $S[i_3]$, e assim por diante. Finalmente, o elemento i_r é movido de $S[i_r]$ para $S[i_1]$. Uma *troca em caminho* (*path exchange*) é definida

por uma sequência de nós $i_1-i_2-i_3-...-i_r$ e difere da troca cíclica pelo fato de que o último elemento i_r não é movido de $S[i_r]$ para $S[i_1]$.

Note que a vizinhança de troca cíclica e de troca em caminho contempla a troca aos pares e ainda explora uma infinidade de outras soluções não alcançáveis pela troca aos pares. Portanto, é de se esperar que soluções ótimas locais obtidas por meio de múltiplas trocas sejam, em média, superiores às aquelas obtidas pela troca aos pares. Entretanto, uma vez que o tamanho da vizinhança em trocas múltiplas cresce exponencialmente com o tamanho do problema, torna-se necessário um método eficiente para encontrar um vizinho de menor custo na vizinhança. Este problema pode ser contornado utilizando o conceito de *grafo de melhoria* e algoritmos de fluxo em redes para explorar, de forma eficiente, uma dada vizinhança.

Um *grafo de melhoria* (*improvement graph*) para uma vizinhança com múltiplas trocas é definido para uma solução viável S do problema, sendo representado por $G(S)$. Seja $S[a_j]$ a jornada que contém a tarefa a_j . O grafo $G(S)$ é um grafo direcionado com n nós, onde cada nó i corresponde a uma tarefa $a_i \in S$. Um arco direcionado (i, j) em $G(S)$ significa que a tarefa a_i deixa a sua jornada atual e é movida para a jornada que contém a tarefa a_j , isto é, a jornada $S[a_j]$. Simultaneamente, a tarefa a_j deixa $S[a_j]$. Para se construir $G(S)$ são considerados todos os pares de tarefas a_i e a_j em S . O arco (i, j) é adicionado a $G(S)$ se: *i*) as tarefas a_i e a_j pertencerem a diferentes jornadas; *ii*) a jornada $\{a_i\} \cup S[a_j] \setminus \{a_j\}$ for viável. O custo c_{ij} no arco (i, j) é definido como $c(\{a_i\} \cup S[a_j] \setminus \{a_j\}) - c(S[a_j])$.

Denomina-se um *ciclo direcionado* W no grafo de melhoria $G(S)$ se as tarefas em S correspondentes aos nós em W pertencem a diferentes jornadas. Define-se um *ciclo válido* como um ciclo direcionado de custo negativo em $G(S)$. Assim, um ciclo válido em $G(S)$ corresponde a uma troca cíclica que leva a uma melhoria no valor da função objetivo do problema. Esta é uma forma eficiente de realizar buscas por soluções que melhoram o valor objetivo na vizinhança de S . Portanto, é necessário encontrar ciclos válidos no grafo de melhoramentos $G(S)$. Nesse trabalho foi utilizado o algoritmo do rotulamento modificado com a disciplina de fila *primeiro que entra – primeiro que sai* para identificar um ciclo válido (Ahuja et al., 1993).

Procedimento VLNS(solução s)

Início

1. Seja s uma solução corrente dada;
2. Construa o grafo de melhoria $G(s)$ referente a s ;
3. Enquanto $G(s)$ tiver ciclos negativos faça:
 4. Identifique um ciclo negativo em $G(s)$;
 5. Melhore a solução s devido às trocas do ciclo negativo;
 6. Atualize o grafo de melhoria $G(s)$;
7. Fim_enquanto
8. Retorna a solução s ;

Fim

Algoritmo 3. Pseudo-código do Procedimento VLNS.

A idéia dos algoritmos do tipo VLNS consiste em construir um grafo $G(S)$ para uma dada solução S , e encontrar um ciclo direcionado em $G(S)$ que forneça um vizinho melhor do que S . Após efetuar a troca cíclica, inerente ao ciclo válido, o grafo é atualizado e é procurado um novo ciclo válido. A busca termina quando o grafo de melhoria não apresentar qualquer ciclo válido. O pseudocódigo apresentado no Algoritmo 3 sintetiza a método.

3.5 Função de Avaliação

O custo associado a uma solução do PPT é computado por meio da combinação linear do custo fixo e dos custos variáveis, que são: *i*) o total de horas extras e *ii*) número de jornadas do tipo dupla pegada. A expressão final para o custo de uma solução do PPT é:

$$C_{ppt} = \sum_{i=1}^{tot_trip} Custo_Fixo + w_1 \times hora_extra_i + w_2 \times dupla_peg_i \quad (1)$$

onde *Custo_Fixo* representa a remuneração fixa de uma tripulação, w_1 é o peso por minutos de hora extra e $hora_extra_i$ o total de horas extras da jornada *i*, expresso em minutos. Finalmente, w_2 é o peso por dupla pegada e $dupla_peg_i$ é igual a 1 se a tripulação *i* faz uma dupla pegada em sua jornada e 0 caso contrário.

4. RESULTADOS OBTIDOS

Os algoritmos foram utilizados para resolver um conjunto de sete problemas referentes a uma semana de operação de uma empresa que atua no sistema de transporte público de Belo Horizonte. As duas heurísticas, VNS-Clássico e VNS-VLNS, foram implementadas na linguagem C/C++ e os testes foram realizados em um computador pessoal com processador Core2 Duo e 3 GB de memória RAM. As heurísticas foram executadas por uma hora e foram realizadas 10 execuções para cada problema da empresa.

Os resultados apresentados abaixo foram divididos em duas subseções em função dos pesos atribuídos ao total de duplas pegadas contido na solução. Este tipo de análise é fundamental para o problema, pois as empresas trabalham com o número de duplas pegadas variando entre 10 e 20 % do total de duplas. Por outro lado, os dois métodos de busca implementados neste trabalho produziram soluções com características significativamente diferentes em relação ao número de duplas pegadas.

4.1 Soluções com menor peso para as duplas pegadas

A Tabela 1 contém as características das soluções adotadas pela empresa. Nas tabelas a seguir, a linha “HE” se refere o total de horas extras, “Jornadas” ao total de jornadas, “DP” o total de jornadas do tipo dupla pegada contidas na solução e a “FO” se refere ao valor total da solução. O primeiro conjunto de testes foi realizado considerando os seguintes pesos: 10.000 para o *Custo_Fixo*, 4 para w_1 , tendo as horas extras expressas em minutos. Ao peso w_2 , referente às duplas pegadas, foi atribuído o valor 600. Estes pesos foram obtidos empiricamente para se produzir soluções com características desejáveis. Eles foram aplicados às soluções da empresa para efeito de comparação com as soluções produzidas pelas heurísticas.

Tabela 1. Dados da escala de mão de obra operada pela empresa.

	Segunda	Terça	Quarta	Quinta	Sexta	Sábado	Domingo
HE	86:41	85:55	106:05	120:14	108:11	54:35	26:57
Jornadas	134	130	149	162	155	124	68
DP	6	3	5	4	1	0	0
FO	1.364.404	1.322.420	1.518.460	1.651.256	1.576.564	1.253.100	686.468

4.1.1 Resultados considerando no máximo uma troca de veículo por jornada

Para os pesos mencionados anteriormente, foram realizados testes considerando que as jornadas pudessem realizar no máximo uma troca de veículo durante a operação. A heurística VNS-Clássico produziu os resultados apresentados na Tabela 2 e na Tabela 3 estão os

resultados do VNS-VLNS. As tabelas a seguir apresentam um resumo dos resultados obtidos nos quais os dados HE, Jornadas e DP se referem à solução com menor FO (FO Melhor). Além disso, são apresentadas as médias de todas as soluções obtidas, a FO Média, assim como o desvio médio dado por $(FO \text{ Média} - FO \text{ Melhor})/FO \text{ Média}$. Quanto menor for esta porcentagem, mais robusto é o método. Ou seja, a diferença entre as diversas soluções encontradas, que contam com um fator de aleatoriedade, não é significativa e a heurística tem a capacidade de produzir soluções muito parecidas.

Tabela 2. Características das soluções obtidas pelo VNS-Clássico.

	Segunda	Terça	Quarta	Quinta	Sexta	Sábado	Domingo
HE	96:58	99:46	122:37	132:06	125:56	90:43	53:51
Jornadas	117	111	133	141	134	101	52
DP	50	59	71	93	81	43	17
FO Melhor	1.223.272	1.169.344	1.402.028	1.497.504	1.418.824	1.057.572	543.124
FO Média	1.233.332	1.188.202	1.410.726	1.529.976	1.441.757	1.064.657	556.974
Desvio	0,82%	1,59%	0,62%	2,12%	1,59%	0,67%	2,49%

Tabela 3. Características das soluções obtidas pelo VNS + VLNS.

	Segunda	Terça	Quarta	Quinta	Sexta	Sábado	Domingo
HE	50:39	56:28	77:30	70:42	70:33	57:21	34:21
Jornadas	120	117	135	149	141	107	55
DP	19	14	19	17	14	10	7
FO Melhor	1.223.556	1.191.952	1.380.000	1.517.168	1.435.332	1.089.764	562.444
FO Média	1.235.435	1.203.345	1.409.582	1.534.114	1.451.791	1.106.525	572.411
Desvio	0,96%	0,95%	2,10%	1,10%	1,13%	1,51%	1,74%

Tabela 4. Melhorias alcançadas no número de jornadas em relação à solução da empresa.

	Heurística	Segunda	Terça	Quarta	Quinta	Sexta	Sábado	Domingo
Jornadas	VNS-Clássico	12,69%	14,62%	10,74%	12,96%	13,55%	18,55%	23,53%
	VNS-VLNS	10,45%	10,00%	9,40%	8,02%	9,03%	13,71%	19,12%
HE	VNS-Clássico	-11,9%	-16,1%	-15,9%	-9,8%	-16,5%	-66,3%	-100,4%
	VNS-VLNS	41,6%	34,3%	26,9%	41,2%	34,8%	-5,1%	-27,5%
%DP	VNS-Clássico	42,7%	53,2%	53,4%	66,0%	60,4%	42,6%	32,7%
	VNS-VLNS	15,8%	12,0%	14,1%	11,4%	9,9%	9,3%	12,7%
FO	VNS-Clássico	10,3%	11,6%	7,7%	9,3%	10,0%	15,6%	20,9%
	VNS-VLNS	10,3%	9,9%	9,1%	8,1%	9,0%	13,0%	18,1%

As melhorias apresentadas na Tabela 4 mostram uma redução de até 13,55% e de 10,45% no número de jornadas obtidas pelo VNS-Clássico e pelo VNS-VLNS respectivamente para os dias úteis e de até 23,53% e 19,12% para o final de semana. As reduções de jornadas mais significativas foram obtidas pelo VNS-Clássico. Para tanto, houve um aumento no número de horas extras, que chega a 16,5% nos dias úteis e até 100,4% no final de semana. Por outro lado, o VNS-VLNS conta com uma redução de até 41,6% das horas extras nos dias úteis. As soluções obtidas pelo VNS-VLNS são de interesse prático pois estão dentro do limite de 20% do total de jornadas. O mesmo não ocorre com as soluções produzidas pelo VNS-Clássico.

Como a empresa admite até duas trocas de veículo por motorista ao longo de sua jornada diária de trabalho, foram realizados testes segundo esta condição, o que representa uma flexibilização na operação da mão de obra.

4.1.2 Resultados considerando no máximo duas trocas de veículo por jornada

Os resultados das Tabelas 5 e 6 consideram que as jornadas podem conter até duas trocas de veículo ao longo da operação. Esta é mais uma variação nas diferentes possibilidades operacionais admitidas pela empresa.

Tabela 5. Características das soluções obtidas pelo VNS Clássico.

	Segunda	Terça	Quarta	Quinta	Sexta	Sábado	Domingo
HE	96:58	94:51	113:45	125:23	105:51	88:02	43:35
Jornadas	117	111	132	144	136	101	52
DP	50	58	69	84	67	43	17
FO Melhor	1.223.272	1.167.564	1.388.144	1.520.492	1.425.068	1.057.572	543.124
FO Média	1.233.332	1.184.599	1.404.902	1.538.868	1.441.251	1.064.102	556.974
Desvio	0,82%	1,46%	1,21%	1,21%	1,14%	0,62%	2,55%

Tabela 6. Características das soluções obtidas pelo VNS-VLNS.

	Segunda	Terça	Quarta	Quinta	Sexta	Sábado	Domingo
HE	50:39	55:52	62:10	53:36	56:18	43:12	36:21
Jornadas	120	117	137	149	141	107	55
DP	19	10	15	17	17	10	7
FO Melhor	1.223.556	1.189.408	1.393.920	1.515.196	1.437.060	1.089.764	562.444
FO Média	1.235.435	1.203.465	1.411.193	1.527.566	1.448.626	1.106.525	572.557
Desvio	0,97%	1,18%	1,24%	0,82%	0,81%	1,54%	1,80%

Os resultados obtidos mostram que o aumento no número de trocas de veículos realizadas pelas tripulações levam, em alguns casos, a uma economia significativa no número de horas extras. Este é o caso dos dados da quinta e da sexta-feira observados no método VNS-VLNS que chegam a 24,4% e 20,4% de redução entre o primeiro e o segundo cenário operacional.

4.2 Soluções com maior peso para as duplas pegadas

A seguir são apresentados os resultados devido aos pesos escolhidos para produzir escalas com um número menor de duplas pegadas DP pela heurística VNS-Clássico. Nas tabelas a seguir foram considerando os seguintes pesos: : 10.000 para o *Custo_Fixo*, 4 para w_1 , tendo as horas extras expressas em minutos. Ao peso w_2 , referente às duplas pegadas, foi atribuído o valor 5.000. Estes pesos são obtidos empiricamente, com o objetivo de reduzir o número de duplas pegadas e manter o número de jornadas em um valor próximo dos níveis dos valores obtidos anteriormente.

4.2.1 Resultados considerando no máximo uma troca de veículo por jornada

Considerando a possibilidade de cada tripulação realizar no máximo uma troca de veículo ao longo da operação e os novos pesos, foram geradas escalas com as duas versões da heurística VNS. Os resultados são apresentados nas Tabelas 7 e 8.

Tabela 7. Características das soluções obtidas pelo VNS Clássico com novos pesos.

	Segunda	Terça	Quarta	Quinta	Sexta	Sábado	Domingo
HE	98:00	96:40	116:16	121:58	116:06	95:02	46:12
Jornadas	120	114	135	152	142	104	54
DP	29	31	30	27	23	24	9
FO Melhor	1.368.520	1.318.200	1.527.904	1.684.272	1.562.864	1.182.808	596.088
FO Média	1.381.195	1.330.934	1.540.384	1.696.964	1.575.031	1.198.170	609.063
Desvio	0,93%	0,97%	0,82%	0,75%	0,78%	1,30%	2,18%

Tabela 8. Características das soluções obtidas pelo VNS-VLNS com novos pesos.

	Segunda	Terça	Quarta	Quinta	Sexta	Sábado	Domingo
HE	65:07	75:44	67:24	77:41	87:55	52:18	27:35
Jornadas	120	114	140	148	139	109	57
DP	11	11	11	17	12	10	5
FO Melhor	1.270.628	1.213.176	1.471.176	1.583.644	1.471.100	1.152.552	601.296
FO Média	1.282.988	1.232.681	1.484.256	1.602.478	1.487.842	1.164.025	606.620
Desvio	0,97%	1,61%	0,89%	1,19%	1,14%	1,00%	0,88%

Com os novos pesos fica mais clara a superioridade das soluções produzidas pela heurística VNS-VLNS sobre o VNS-Clássico, não somente por meio da comparação da FO, como também pela comparação do número de jornadas, número de jornadas com duplas pegadas, mas principalmente pelo total de horas extras.

4.2.2 Resultados considerando no máximo duas trocas de veículo por jornada

Os resultados das Tabelas 9 e 10 consideram que as jornadas podem conter até duas trocas de veículo ao longo da operação.

Tabela 9. Características das soluções obtidas pelo VNS Clássico.

	Segunda	Terça	Quarta	Quinta	Sexta	Sábado	Domingo
HE	93:36	89:25	111:57	113:42	110:20	98:03	34:06
Jornadas	120	115	139	151	143	102	54
DP	29	28	23	30	24	26	11
FO Melhor	1.348.720	1.306.088	1.526.608	1.663.964	1.547.540	1.148.672	595.472
FO Média	1.370.574	1.316.318	1.535.872	1.683.994	1.562.788	1.178.420	612.465
Desvio	1,62%	0,78%	0,61%	1,20%	0,98%	2,59%	2,85%

Tabela 10. Características das soluções obtidas pelo VNS-VLNS.

	Segunda	Terça	Quarta	Quinta	Sexta	Sábado	Domingo
HE	51:34	58:25	58:19	58:33	66:05	49:39	27:12
Jornadas	121	118	139	150	139	110	58
DP	12	8	11	14	10	11	6
FO Melhor	1.264.724	1.220.112	1.449.952	1.580.160	1.461.288	1.141.916	597.496
FO Média	1.281.180	1.233.742	1.465.691	1.589.927	1.478.001	1.155.537	605.801
Desvio	1,30%	1,12%	1,09%	0,62%	1,14%	1,19%	1,39%

Novamente foi verificada uma redução no valor da FO Melhor quando se permite que uma tripulação faça até duas trocas de veículo ao longo da operação. Esta redução se deve principalmente à redução das horas extras e no número de jornadas do tipo dupla pegada.

5. CONCLUSÕES

Os resultados obtidos mostram que as duas versões da metaheurística VNS produziram resultados que melhoram a solução da empresa em praticamente todos os testes efetuados. Comparando os resultados do VNS, a versão VNS-VLNS produz soluções melhores, segundo a função objetivo (FO), mantendo o número de jornadas do tipo duplas pegadas (DP) dentro dos limites aceitáveis do ponto de vista operacional, que é de no máximo de 20% do total de jornadas e reduzindo ao máximo o total de horas extras. Por outro lado, em alguns casos o VNS clássico produz escalas com um número menor de jornadas, como nos problemas de quarta, sábado e domingo. Mas como estas soluções apresentam um número maior de horas extras e de duplas pegadas, seus custos acabam sendo superiores. Este tipo de análise e escolha da melhor solução depende das características mais valorizadas na prática.

O fato das soluções do VNS-VLNS produzirem um número menor de duplas pegadas mostra que o método de busca empregado não provoca modificações tão drásticas nas características da solução inicial, construídas com a mesma filosofia da empresa. Ainda assim, ela é capaz de reduzir os custos dados pelo número de jornadas e o total de horas extras.

Os resultados mostram que a utilização da metaheurística VNS na construção de escalas diárias de motoristas do sistema de transporte público pode levar uma economia considerável para as empresas do setor. Por outro lado, comparando os diferentes métodos de busca implementados, pode-se verificar que a técnica *Very Large-scale Neighborhood Search* – VLNS produziu soluções mais compatíveis com a prática adotada pela empresa e consequentemente mais econômicas, tendo em vista os parâmetros adotados.

Agradecimentos

Os autores agradecem ao CNPq, à FAPEMIG e à AUTUMN TI pelo apoio recebido durante o desenvolvimento deste trabalho.

REFERÊNCIAS BIBLIOGRÁFICAS

- Ahuja, R. K.; Magnanti, T. L. e J. B. Orlin (1993). *Network Flows: Theory, Algorithms, and Applications*. Prentice Hall, N. J.
- Ahuja, R. K., Orlin, J. B. e Sharma, D., (2000) Very large-scale neighborhood search, *International Transactions in Operational Research*, v.7, p. 301–317.
- Alves, J. M. C. B., Silva, G. P. e Souza, M. J. F., (2002) Resolução do Problema de Programação Diária da Tripulação de Ônibus Urbano via *Simulated Annealing*, *XVI Congresso de Pesquisa e Ensino em Transportes, Panorama Nacional de Pesquisa em Transportes*, v. 2, p. 95-104.
- Barnhart, C.; Johnson, E. L.; Nemhauser, G. L.; Savelsbergh, M. P. e P. H. Vance (1998) Branch-and-price: column generation for solving huge integer programs. *Operations Research*, v. 46, p. 316-329.
- Desrochers, M. e F. Soumis (1989) A Column Generation approach to the urban transit crew scheduling problem. *Transportation Science*, v. 23, p. 1-13.
- Fischetti, M.; Martello, S e P. Toth, (1987) The Fixed Job Schedule Problem with Spread-Time Constraint. *Operations Research* v. 35, p. 849-858.
- Forsyth, P. e A. Wren (1997) An Ant System for Bus Driver Scheduling, In: *7th International Workshop on Computer-Aided Scheduling of Public Transport*, Boston.
- Fores, S.; Proll, L. e A. Wren (1999) An Improved ILP System For Driver Scheduling. In: *Computer-Aided Transit Scheduling*, Wilson, N. H. M. (ed.), Springer, Berlin, p. 43-61.
- Kwan, A. S.; Kwan, R. S. K. e A. Wren (1999) Driver scheduling using genetic algorithms with embedded combinatorial traits. In: *Computer-Aided Transit Scheduling*, Wilson, N. H. M. (ed.), Springer, Berlin, p. 81-102.
- Li, J. e R. S. K. Kwan (2003) A fuzzy genetic algorithm for driver scheduling. *European Journal of Operations Research* v. 147, p. 334-344.
- Mladenović, N. e P. Hansen (1997) Variable Neighborhood Search, *Computers & Operations Research* v. 24, Issue 11, p. 1097-1100.
- Shen, Y. e R. S. K. Kwan (2001) Tabu search for driver scheduling. In: *Computer-Aided Scheduling of Public Transport*, Voss, S. e J. R. Daduna (eds.), Springer, Berlin, p. 121-135.

- Silva, G. P. e C. B. Cunha (2010) Uso da técnica de busca em vizinhança de grande porte para a programação da escala de motoristas de ônibus urbano. *Revista Transportes*, v. 18, p. 64-75.
- Silva, G. P., Souza, M. J. F. e J. M. C. B. Alves (2002) Resolução do Problema de Programação Diária da Tripulação de Ônibus Urbano via Simulated Annealing. In: *Panorama Nacional de Pesquisa em Transportes*, v. 2, p. 95-104.
- Smith, B. M. e A. Wren, (1988) A Bus Crew Scheduling System Using a Set Covering Formulation. *Transportation Research*, v. 22A, p.97-108.
- Soares, G. F.; Silva, G. P. e E. H. Marinho (2006) Alocação da mão de obra no Sistema de Transporte Público: Uma visão multiobjetivo, In: *Panorama Nacional de Pesquisa em Transportes*, p.693-704.
- Souza, M. J. F.; Carvalho, L. X. T.; Silva, G. P.; Rodrigues, M. M. S. 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.
- Wren, A. (2004) Scheduling Vehicles and Their Drivers – Forty Years’ Experience. In: *9th International Conference on Computer-Aided Scheduling of Public Transport*, p. 27-40.

Gustavo Peixoto Silva (gustavo@iceb.ufop.br)

Departamento de Computação, Instituto de Ciências Exatas e Biológicas, Universidade Federal de Ouro Preto
Campus Universitário do Morro do Cruzeiro, s/nº – 35.400-000 Ouro Preto, MG, Brasil

Allexandre Fortes da Silva Reis (allexandrefsr@gmail.com)

Departamento de Engenharia de Produção, Escola de Minas, Universidade Federal de Ouro Preto
Campus Universitário do Morro do Cruzeiro, s/nº – 35.400-000 Ouro Preto, MG, Brasil