

UMA HEURÍSTICA BASEADA EM BUSCA EM VIZINHANÇA VARIÁVEL PARA O PROBLEMA DE AGRUPAMENTO DE ENTREGAS EM VEÍCULOS DE UMA FROTA HETEROGÊNEA.

Jorge von Atzingen dos Reis

Claudio Barbieri da Cunha

Programa de Pós-Graduação em Engenharia de Transportes
Escola Politécnica da Universidade de São Paulo

RESUMO

Este artigo trata do problema de distribuição física que envolve o agrupamento e a alocação de entregas a uma frota de veículos visando minimizar o frete total. Este problema surge em diferentes situações práticas em que os pontos a serem atendidos estão suficientemente próximos entre si de tal modo que as distâncias percorridas entre paradas consecutivas podem ser consideradas irrelevantes para o custo total da frota e, portanto, podem ser desconsideradas para o cálculo dos fretes pagos a terceiros que realizam esse serviço. O problema pode ser modelado como um problema de *bin-packing* com *bins* de tamanho variável (do inglês *Variable Sized Bin-Packing Problem* - VSBPP), uma generalização do tradicional problema de *bin-packing* no qual *bins* (veículos) de diferentes capacidades e custos estão disponíveis para a alocação de um conjunto de objetos (cargas) de modo que o custo total dos *bins* (veículos) utilizados seja mínimo. É proposta uma estratégia de solução baseada na meta-heurística VNS (*Variable Neighborhood Search*) para a resolução desse problema. O método proposto foi testado com problemas teste derivados de instâncias *benchmarking* da literatura. Os resultados obtidos indicam que a heurística permite obter soluções de alta qualidade em tempos de processamento bastante reduzidos.

ABSTRACT

In this paper we deal with the physical distribution problem which comprises grouping and assigning deliveries to a heterogeneous fleet of vehicles aiming to minimize the total freight cost. This problem arises in several practical situations in which the customers to be served are geographically close enough to each other such that the distances traveled between consecutive stops are irrelevant when compared to the overall vehicle costs and thus not considered when establishing the freight rate to be paid to third-party carriers. The problem can be modeled as Variable Sized Bin-Packing Problem (VSBPP), a generalization of the traditional bin-packing problem, where bins (vehicles) with different capacities and costs are available for the assignment of different objects (deliveries) such that the total cost of the used bins (vehicles) is minimized. We propose an approach to solve this problem, based on the meta-heuristic VNS (Variable Neighborhood Search). The algorithm was evaluated considering benchmark instances from the literature. The results show it can produce high quality results in very short times.

1. INTRODUÇÃO

No contexto logístico, a distribuição física de produtos geralmente envolve o roteamento de veículos, que consiste em determinar que entregas devem ser alocadas a cada veículo de uma frota disponível, e qual o roteiro (ou sequência de paradas) de cada veículo, de forma a minimizar o custo total do serviço, geralmente composto da soma ponderada dos custos fixos e dos custos proporcionais à distância percorrida pelos veículos e ao tempo de viagem. Entretanto, em várias situações práticas pode surgir um problema diferente, que envolve a distribuição de produtos para certo destino (ou destinos próximos entre si), quando se busca otimizar o agrupamento de entregas nos veículos de modo a reduzir a frota necessária e o custo com o frete pago aos transportadores.

Muitos sistemas de distribuição física, tais como a entrega de cimento ou aço para construção abrangem inúmeros clientes, localizados em municípios distintos e atendidos a partir de uma única base de origem dos veículos. Em muitos casos a distância de deslocamento entre base de origem, onde o veículo é carregado, até um município ou região onde são realizadas as entregas é muito superior às distâncias de percurso em rota (i.e, entre entregas consecutivas). Características da carga, tempos envolvidos (de entregas e de deslocamento do veículo) podem acarretar um número de entregas não elevado para cada veículo (geralmente cerca de até 10); conseqüentemente, normalmente é necessário despachar vários veículos a um

município ou a municípios vizinhos, próximos entre si, de forma a realizar as entregas para todos os clientes ali localizados (Miura, 2008).

Deve-se destacar que, na prática, não se admite que as entregas possam ser divididas (ou fracionadas) em mais de um veículo (estratégia conhecida como *split deliveries*), de modo a facilitar a sua acomodação e o melhor aproveitamento das capacidades dos veículos. Isso ocorre porque é indesejável para os clientes receber a mercadoria fracionada, em mais de uma entrega, uma vez que isso dificulta e até impede a sua conferência em termos dos itens e respectivas quantidades em conformidade com o pedido; já para os embarcadores, o fracionamento da carga implica na multiplicação de tarefas administrativas e operacionais, tais como a emissão de mais documentos e a maior dificuldade de conferência da carga sendo embarcada.

A forma peculiar de contratação do transporte no Brasil, e de remuneração dos transportadores, favorece esse agrupamento por município, uma vez que, usualmente, o valor do frete a ser pago ao transportador não é determinado com base na distância total da rota (apurada ou estimada), como é prática usual em muitos países, principalmente os mais avançados, onde grande parte da literatura de roteamento de veículos se origina. Segundo a prática no mercado brasileiro, existe uma tabela pré-acordada entre o embarcador e o transportador que define um valor fixo diário por município e tamanho de veículo ao qual são alocadas entregas, independente do número de entregas e da sua ocupação; nesse caso, um custo adicional é incorrido apenas quando as entregas ocorrem em municípios distintos e mais distantes entre si (em geral, distâncias superiores a 10-20 km), uma vez que o veículo deve percorrer um trajeto rodoviário maior a fim de concluir as entregas. Além do custo envolvido, esse trajeto rodoviário entre entregas prejudica a produtividade do veículo, em razão do menor tempo disponível para as entregas (que geralmente se restringem ao horário comercial), decorrente do percurso rodoviário extra.

Em muitos problemas pode-se assumir produto único ou homogêneo, como é o caso, por exemplo, do cimento (Miura, 2008), permitindo desconsiderar a questão do arranjo espacial das cargas nos veículos. Assim, esse problema de distribuição pode ser modelado como um problema de *bin-packing*.

Problemas de *bin-packing* constituem um grupo de problemas desafiadores de otimização combinatória que pertencem a uma classe mais geral denominada “problemas da mochila” (do inglês *knapsack problems*), e que vêm sendo intensamente estudados há várias décadas, atraindo teóricos e práticos. O interesse dos teóricos deve-se, principalmente, à estrutura simples do problema, o que permite, além da exploração de diversas propriedades do mesmo, a sua resolução como parte de problemas mais complexos. Já do ponto de vista prático, estes problemas aparecem em diversas aplicações reais, incluindo, entre outros, carregamento de veículos (Eilon e Christofides, 1971; Gehring *et al.*, 1990) e corte e empacotamento (Dyckhoff, 1990; Marques e Arenales, 2007; Hoto *et al.*, 2007).

O problema de *bin-packing* (BPP) pode ser descrito da seguinte forma: dados $j = 1, \dots, n$ objetos ou itens com seus respectivos pesos w_j e $i = 1, \dots, n$ bins (ou mochilas) idênticas de capacidade finita c , determinar a alocação (ou designação) dos n itens aos bins (sendo $w_j \leq c, \forall j$), de tal modo que o número de bins utilizados seja mínimo, e respeitando-se as restrições de capacidade em cada um dos bins.

Assim, no contexto do BPP aplicado à distribuição física, busca-se determinar, para cada

município ou região composta por municípios para o qual o frete é o mesmo, quantos veículos idênticos serão necessários (correspondendo aos *bins*), e a alocação dos pontos de entrega (que correspondem aos objetos) aos mesmos, de modo que a frota total utilizada seja mínima.

No caso mais geral, a frota de distribuição pode ser composta por veículos de diferentes tamanhos e capacidades (em geral, de dois a cinco tipos), aos quais estão associados custos diferentes. Nesse caso, busca-se não mais minimizar o número total de veículos, mas sim o custo total dos veículos utilizados; normalmente o frete dos veículos maiores, embora maior, costuma ser menor em termos unitários (por unidade de capacidade) do que o dos veículos menores, uma vez que o custo fixo diário do veículo não é diretamente proporcional à sua capacidade de carga (por exemplo, o custo do motorista é o mesmo, independente do tamanho do veículo).

Esse problema mais geral com frota heterogênea dá origem ao chamado problema de *bin-packing* com *bins* heterogêneos, ou de tamanho variável (do inglês “*variable sized bin packing problem*”, ou simplesmente VSBPP), que é objeto do presente trabalho, tendo em vista sua aplicação prática no contexto logístico. Ao contrário do problema de *bin packing* tradicional, em que os *bins* são idênticos, este é um problema mais complexo e ainda não muito explorado na literatura.

Assim, motivados pela resolução de problemas reais de distribuição física de produtos, é proposta, neste trabalho, uma heurística baseada em Busca em Vizinhança Variável (VNS) para o problema de *bin-packing* com *bins* heterogêneos. A heurística proposta é avaliada considerando novas instâncias derivadas de instâncias *benchmarking* na literatura. Busca-se encontrar um método de solução eficiente, que produza boas soluções em tempos de processamento reduzidos, a fim de permitir a sua aplicação na programação diária da distribuição de produtos aos pontos de entrega. Tal programação compreende a definição, para cada município ou região de destino, de quantos veículos de qual tipo serão necessários, e a alocação dos pontos de entrega aos mesmos, de modo que o custo total da frota utilizada seja mínimo. Tendo em vista a sua aplicação à programação diária de entregas, e ao prazo reduzido para essa atividade, o tempo de processamento, assim como a qualidade das soluções, são aspectos importantes para a avaliação da heurística proposta.

Este artigo está organizado da seguinte forma: na próxima seção é apresentada a formulação matemática do problema. Em seguida, na seção 3 é descrita a heurística proposta baseada em Busca em Vizinhança Variável (*Variable Neighborhood Search* - VNS). Na seção 4 são apresentados os resultados dos experimentos computacionais realizados, incluindo uma comparação dos resultados com os obtidos através de um pacote de otimização linear. Por fim, as conclusões e recomendações são apresentadas na seção 5.

2. O PROBLEMA DE *BIN-PACKING* COM FROTA HETROGÊNEA

O Problema de *Bin-Packing* com frota heterogênea (VSBPP) é *NP-hard*, uma vez que se reduz ao BPP no caso particular em que todos os *bins* são idênticos, que é *NP-hard* (Garey e Johnson, 1979). Para o BPP, diversas heurísticas têm sido propostas para a sua resolução. As mais populares são as que os objetos são ordenados em ordem não-crescente de peso (w_j) e aplicadas regras para a alocação dos objetos aos *bins*, tais como *first-fit*, *best-fit* ou *next-fit* (Coffman *et al.*, 1997). Uma boa revisão dos principais trabalhos na literatura para o BPP é apresentada por Scholl (1997), que também propõe um algoritmo exato para a sua resolução. Carvalho (1999) propôs um algoritmo exato baseado em geração de colunas e *branch-and-bound*. Gupta e Ho (1999) propõem uma heurística construtiva baseada na melhor folga do

bin, denominada *minimum bin slack* (MBS), com resultados superiores. Fleszar e Hindi (2002) propõem alterações na heurística MBS proposta por Gupta e Ho (1999), possibilitando encontrar resultados melhores com tempos computacionais menores; adicionalmente é também proposto um algoritmo baseado em VNS (*Variable Neighborhood Search*). Motivados pela sua aplicação em problemas de distribuição física, Cunha *et al.* (2008) propuseram três heurísticas para a solução desse problema: duas baseadas em GRASP e uma baseada em Busca em Vizinhaça Variável (VNS). Experimentos computacionais realizados usando problemas *benchmarking* evidenciaram que as três heurísticas, que são simples e facilmente implementáveis, fornecem resultados de boa qualidade em tempos de processamento reduzidos, possibilitando a sua aplicação a problemas práticos em logística.

Já no caso do VSBPP, a literatura é bem mais restrita, e só mais recentemente esta generalização do problema clássico passou a atrair a atenção dos pesquisadores. Segundo Zhang (1997) as heurísticas clássicas para o BPP não fornecem bons resultados. O trabalho de Friesen e Langston (1986) é um dos pioneiros sobre o VSBPP definindo *bounds* e apontando a direção de pesquisas futuras com a utilização de heurísticas. Burkard e Zhang (1997) e Kang e Park (2003) propõem algoritmos aplicados ao problema de VSBPP e determinam o desempenho no pior caso dos algoritmos para casos específicos. Correia *et al.* (2006) utilizam uma reformulação do modelo de otimização discreta para sua solução. Mais recentemente, Haouari e Serairi (2009) analisaram o desempenho empírico de seis heurísticas baseadas em otimização em conjuntos de dados randomicamente gerados, evidenciando o sucesso da heurística baseada em cobertura de conjunto (*set covering*).

2.1. Formulação Matemática para o VSBPP

O problema de *bin-packing* com *bins* heterogêneos (VSBPP) pode ser descrito da seguinte forma: dados $j = 1, \dots, n$ objetos ou itens com seus respectivos pesos w_j (representando as entregas a serem realizadas) e $i = 1, \dots, m$ *bins* (ou mochilas, denotando os veículos a serem utilizados), para os quais são conhecidos seus respectivos custos f_i e suas capacidades b_i , determinar a alocação (ou designação) dos n itens (entregas) aos m *bins* (veículos), de tal modo a minimizar o custo total dos *bins* utilizados, e respeitado-se as restrições de capacidade nos *bins*. Assume-se, sem perda de generalidade, que os objetos j estejam ordenados em ordem não crescente de peso ($w_1 \leq w_2 \leq \dots \leq w_n$) e que os *bins* i estejam ordenados em ordem não decrescente de tamanho ($b_1 \leq b_2 \leq \dots \leq b_m$); adicionalmente, assume-se ainda que $w_n \leq b_m$ e que o número de bins m seja suficientemente elevado a fim de assegurar a viabilidade do problema. Assim, a formulação matemática do VSBPP pode ser escrita da seguinte forma:

$$\text{Min} \sum_{i=1}^m f_i y_i \quad (1)$$

$$\sum_{j=1}^n w_j x_{ij} \leq b_i y_i \quad \forall i=1, \dots, m \quad (2)$$

$$\sum_{i=1}^m x_{ij} = 1 \quad \forall j=1, \dots, n \quad (3)$$

$$x_{ij} \in \{0,1\} \quad \forall i=1, \dots, m \quad \forall j=1, \dots, n \quad (4)$$

$$y_i \in \{0,1\} \quad \forall i=1, \dots, m \quad (5)$$

Onde: x_{ij} é uma variável binária que assume o valor 1 quando o objeto j é alocado ao bin i e recebe o valor zero caso contrário; y_i é uma variável binária que recebe o valor 1 quando o bin i é utilizado na solução e assume o valor zero caso contrário.

A função objetivo (1) busca minimizar o custo total dos veículos (ou *bins*) utilizados. A restrição (2) assegura que a capacidade de cada veículo utilizado não seja violada. A restrição (3) garante que cada entrega (ou objeto) j seja alocado a exatamente um veículo. As restrições (4) e (5) asseguram a integralidade das variáveis de decisão.

3. META-HEURÍSTICA VNS APLICADA AO VSBPP

O Método de Busca em Vizinhança Variável (VNS) é um método de busca local que consiste em explorar o espaço de soluções através de trocas sistemáticas de estruturas de vizinhança. Contrariamente a outras meta-heurísticas baseadas em métodos de busca local, o método VNS não segue uma trajetória, mas sim explora vizinhanças gradativamente mais “distantes” da solução corrente e focaliza a busca em torno de uma nova solução se, e somente se, um movimento de melhora é realizado.

As soluções vizinhas ou vizinhança de uma solução são todas as modificações que podem ser feitas na solução corrente, ou solução atual, de tal forma a gerar uma nova solução distinta. Uma estrutura de vizinhança é uma forma pré-determinada de se percorrer o espaço de soluções vizinhas à solução corrente. O pseudocódigo desta meta-heurística é apresentado na Figura 1. Maiores detalhes desse algoritmo podem ser encontrados em Mladenovic & Hansen (1997).

<p>Início VNS</p> <ol style="list-style-type: none"> 1. Seja s_0 uma solução inicial e r o número de estruturas de vizinhança; 2. $s \leftarrow s_0$; {Solução corrente} 3. <u>Enquanto</u> (Critério de parada não satisfeito) <u>faça</u> 4. $k \leftarrow 1$; {Tipo de estrutura de vizinhança} 5. <u>enquanto</u> ($k \leq r$) <u>faça</u> 6. Gere um vizinho qualquer $s' \in N^{(k)}(s)$; 7. $s'' \leftarrow \text{BuscaLocal}(s')$; 8. <u>Se</u> $f(s'') < f(s)$ 9. <u>então</u> $s \leftarrow s''$; 10. $k \leftarrow 1$; 11. <u>senão</u> $k \leftarrow k + 1$; 12. <u>Fim-se</u>; 13. <u>Fim-enquanto</u>; 14. <u>Fim-enquanto</u>; 15. Retorne s; <p>Fim VNS</p>

Figura 1: Pseudocódigo com a estrutura geral de funcionamento do VNS.

No VNS, parte-se de uma solução inicial qualquer e, a cada iteração, seleciona-se aleatoriamente um vizinho s' dentro da vizinhança $N^{(k)}(s)$ obtida através da estrutura de vizinhança de ordem k ($1 \leq k \leq r$) a partir da solução s corrente. Esse vizinho s' é então submetido a um procedimento de busca local. Se a nova solução obtida s'' , resultante dessa busca local no entorno de s' , for melhor que a solução corrente s (i.e. $f(s'') < f(s)$), a busca prossegue a partir da nova solução encontrada s'' , reiniciando a partir da primeira estrutura de vizinhança $N^{(1)}(s)$, que corresponde a $k=1$. Caso contrário, a busca prossegue a partir da próxima estrutura de vizinhança $N^{(k+1)}(s)$. Este procedimento é encerrado quando uma

condição de parada for atingida (por exemplo, tempo de processamento, número máximo de iterações ou número máximo de iterações consecutivas sem melhorias). A solução s' é obtida de maneira aleatória no passo seis do pseudocódigo da Figura 1, a fim de evitar ciclos, situação que pode ocorrer caso alguma regra determinística seja utilizada.

3.1. Solução Inicial

Para a aplicação do VNS, é necessária a geração de uma solução inicial a qual consiste de uma solução s na qual todas as entregas são alocadas aos veículos da frota. A solução inicial para a heurística aplicada ao VSBPP é obtida através de uma heurística construtiva, cujo pseudocódigo é apresentado na Figura 2.

A heurística construtiva para a geração da solução inicial, primeiramente, ordena as cargas (objetos) em ordem decrescente de peso e os veículos (*bins*) em ordem decrescente de folga (capacidade de carga ociosa) utilizando o método de ordenação do algoritmo *QuickSort*. Maiores detalhes sobre o algoritmo *QuickSort* podem ser obtidos em Hoare (1962).

<p>Início SoluçãoInicial</p> <ol style="list-style-type: none">1. Ordene cargas/ entregas em ordem decrescente de peso;2. Ordene veículos ordem decrescente de folga de capacidade;3. <u>Para</u> $i = 1$ <u>até</u> tot_entregas <u>faça</u>4. <u>Se</u> $\text{folgaVeic}[j] > 0$ <u>então</u>5. $\text{Veic}[j] \leftarrow \text{entrega}[i];$6. Atualiza $\text{custo}[j];$7. <u>Senão</u>8. $j \leftarrow j + 1;$9. <u>Fim-Para</u>; <p>Fim SoluçãoInicial</p>
--

Figura 2: Pseudocódigo da heurística construtiva para obtenção da solução inicial.

Em seguida, cada uma das cargas é inserida, uma a uma e em ordem, no primeiro veículo que possuir espaço suficiente para recebê-la. Esta heurística construtiva gera somente soluções viáveis, isto é, garante que todos os objetos serão alocados a exatamente um único veículo e que nenhum veículo possuirá a sua capacidade violada.

A frota de veículos é considerada suficientemente grande para alocar todos os objetos a um único tipo de veículo caso seja necessário. O número de veículos de cada tamanho é grande o suficiente para transportar todos os objetos do problema.

3.2. Estrutura de vizinhança

O método de busca em vizinhança variável necessita gerar uma solução vizinha, a partir da qual é realizada uma busca local. Esta solução vizinha é gerada a partir de uma estrutura de movimentos de vizinhança que visa buscar soluções cada vez mais “distantes” da solução corrente. A sequência proposta de movimentos para o VNS é a seguinte:

- Troca 1-1 \rightarrow Escolhe aleatoriamente uma entrega de um veículo e tenta trocar sistematicamente com outra entrega de outro veículo até encontrar um movimento viável;
- Troca 2-0A \rightarrow Escolhe aleatoriamente duas entregas de um mesmo veículo e tenta sistematicamente inserir ambas em outro veículo até encontrar um movimento viável;
- Troca 2-0B \rightarrow Escolhe aleatoriamente duas entregas de um mesmo veículo e tenta

sistematicamente inserir em outro veículo (sendo que ambas as entregas podem ser inseridos no mesmo veículo ou não) até encontrar um movimento viável;

- Troca 2-1 → Escolhe aleatoriamente duas entregas de um mesmo veículo v_1 e tenta sistematicamente inserir ambas em outro veículo v_2 , enquanto é removida uma entrega de v_2 e inserida no veículo v_1 , até encontrar um movimento viável;
- Perturbação A → Escolhe aleatoriamente uma entrega de um veículo v_1 e insere em um veículo v_2 ao mesmo tempo em que retira uma entrega de v_2 e insere no veículo v_3 ; repetir até encontrar um movimento viável;
- Perturbação B → Escolhe aleatoriamente duas entregas de um veículo v_1 e insere em um veículo v_2 , retira uma entrega de um veículo v_2 e insere no veículo v_3 ; repetir até encontrar um movimento viável.

3.3. Busca Local

Em seguida à geração de uma solução vizinha à solução corrente é realizada uma busca local. Os veículos são ordenados em ordem decrescente de folga (capacidade ociosa). A etapa seguinte é tentar realocar todas as entregas deste veículo para outros veículos e sempre buscando o veículo com a menor folga de capacidade para receber a carga sendo redistribuída. Este procedimento é repetido até que todas as cargas tenham sido realocadas ou não seja possível realocar mais nenhum objeto. A Figura 3 apresenta o procedimento de busca local:

<p>Início BuscaLocal</p> <ol style="list-style-type: none">1. Ordene veículos ordem decrescente de folga ();2. Ordene cargas ordem decrescente de peso ();3. <u>Para</u> $i = 1$ até tot_veículos <u>faça</u>4. <u>Para</u> $j = i$ até tot_veículos <u>faça</u>5. <u>Se</u> $\text{folgaVeic}[j] > \text{peso_objeto}[k][i]$ <u>então</u>6. $\text{Veic}[j] \leftarrow \text{objeto}[k][i]$;7. Atualiza $\text{custo}[j]$;8. Atualiza $\text{custo}[i]$;9. <u>Fim-Para</u>;10. <u>Fim-Para</u>; <p>Fim BuscaLocal</p>

Figura 3: Pseudocódigo do procedimento de busca local.

O procedimento de busca local apresentado na Figura 3 visa reduzir o número de veículos utilizados realocando os objetos dos veículos que possuem menor quantidade de carga atribuída para veículos que possuem uma maior quantidade de carga mas ainda não estão transportando carga suficiente para atingir o limite de capacidade.

3.4. Função de Avaliação

A abordagem baseada em VNS para o VSBPP proposta neste trabalho foi testada com dados de derivados de instâncias da literatura. As novas instâncias são derivadas das instâncias *benchmarking* do BPP disponíveis na *OR Library* (<http://people.brunel.ac.uk/~mastjjb/jeb/orlib/binpackinfo.html>), originalmente propostas por Falkenauer (1996) para o BPP, no qual os *bins* são idênticos, e que haviam sido utilizadas por Cunha *et al.* (2008). A diferença básica das novas instâncias foi a geração de mais dois tamanhos de *bins* (veículos) além do de capacidade igual a 150, considerada pelo autor. Considerando por base o tamanho de *bin* correspondente à capacidade igual a 150 conforme proposto por Falkenauer (1996), e atribuindo um custo de 100, foram criados mais dois tipos

de *bins* da seguinte forma: um tipo de *bin* com capacidade 25% inferior e custo 20% inferior e um tipo de *bin* com capacidade 25% superior e custo 20% superior. Dessa forma, o *bin* de maior tamanho é o que apresenta o menor custo por unidade de capacidade, tal como ocorre na prática onde veículos maiores são os de menor custo unitário.

Tabela 1: Custo e Capacidade dos veículos.

	Grande	Médio	Pequeno
Custo (\$)	120	100	80
Capacidade	187	150	112

A função de avaliação tem o objetivo de avaliar a qualidade de uma solução vizinha obtida através de um movimento realizado por uma das estruturas de vizinhança utilizada pela meta-heurística VNS. A função de avaliação proposta visa a avaliar a qualidade de uma solução para o VSBPP considerando o custo da frota alocada e pode ser calculada através da expressão (6).

$$FA = \sum_{i=1}^3 \text{Custo}_i \text{NumBinTipo}_i \quad (6)$$

Onde: Custo_i é o custo por utilizar um veículo do tipo i ; NumBinTipo_i é a quantidade de veículos (*bins*) do tipo i utilizados.

4. EXPERIMENTOS COMPUTACIONAIS

Os testes foram realizados em um total de três conjuntos de dados com problemas com o número de objetos variando entre 10 e 1.000. Para cada instância do problema, o programa desenvolvido em C++ foi rodado cinco vezes e as melhores soluções obtidas para cada instância estão apresentadas nas Tabelas 2, 3 e 4. O computador utilizado para os testes apresentados neste trabalho foi um Core2Quad 2.4GHz com 2GB de RAM, entretanto é possível rodar o programa desenvolvido (heurística baseada em VNS) em computadores mais modestos.

A primeira coluna da Tabela 2 apresenta o número de objetos (cargas) a serem alocadas. A segunda coluna apresenta o limite inferior (*lower bound*) para cada uma das instâncias de teste. O limite inferior representa o custo mínimo para cada instância e é calculado considerando que todas as cargas sejam alocadas ao veículo com a melhor relação custo/benefício e este limite pode ser obtido pela expressão (7).

$$\text{Lower Bound} = \frac{\text{Custo}_{GDE} * \sum_{i=1}^n \text{Peso}_i}{\text{Capacidade}_{GDE}} \quad (7)$$

onde: Custo_{GDE} é o custo por utilizar um veículo do tipo grande; Capacidade_{GDE} é a capacidade dos veículos do tipo grande; Peso_i é o peso da carga i ; n é o total de cargas da instância do problema.

A coluna *Custo Sol* da Tabela 2 apresenta o custo da solução obtida para cada instância pelo CPLEX (o qual obteve a solução ótima somente para as instâncias da Tabela 2) e pelo VNS. A coluna *Tempo CPU* apresenta o tempo computacional, em segundos, para obter a solução apresentada. As colunas *Veic Grande*, *Veic Médio* e *Veic Pequeno* apresentam, respectivamente, o número de veículos do tipo grande, o número de veículos do tipo médio e o número de veículos do tipo pequeno utilizados na solução encontrada para cada instância.

Tabela 2: Resultados obtidos para instâncias de pequeno porte.

Número de Objetos	<i>Lower Bound</i>	Solução Ótima					Solução VNS				
		Custo Sol	Tempo CPU (seg)	Veic Grande	Veic Médio	Veic Pequeno	Custo Sol	Tempo CPU (seg)	Veic Grande	Veic Médio	Veic Pequeno
10	371	400	2,1	1	2	1	400	0,2	1	2	1
12	443	460	2,5	3	1	0	460	0,3	3	1	0
12	464	480	4,6	3	0	0	480	0,5	3	0	0
15	589	600	0,2	5	0	0	600	0,3	5	0	0
20	681	700	1,1	5	1	0	700	0,4	5	1	0
25	1.037	1.060	9,3	8	1	0	1.060	0,6	8	1	0
40	1.634	1.660	8,8	13	1	0	1.660	1,2	13	1	0
40	1.542	1.560	1,2	13	0	0	1.580	1,3	11	1	2
40	1.635	1.660	48,0	13	1	0	1.660	1,2	13	1	0
40	1.542	1.560	3,4	13	0	0	1.580	1,1	9	5	0
40	1.634	1.660	10,0	13	1	0	1.680	1,1	14	0	0
40	1.592	1.620	5,7	12	1	1	1.640	1,2	12	2	0
40	1.530	1.540	2.104,0	12	1	0	1.560	1,1	13	0	0
40	1.496	1.520	1.460,0	11	2	0	1.520	1,1	11	2	0
40	1.633	1.660	1.671,0	13	1	0	1.660	1,1	13	1	0
40	1.551	1.560	2.823,0	13	0	0	1.600	1,1	11	2	1
40	1.503	1.520	77,0	11	2	0	1.540	1,0	12	1	0
Total	20.877	21.220	8.154,9	162	15	2	21.380	14,8	155	25	4

Os resultados apresentados na Tabela 2 são relativos aos problemas de pequeno porte com o número de entregas a serem alocadas aos veículos variando entre 10 e 40. Este conjunto de problemas foi definido como pequeno porte devido a ser possível encontrar a solução ótima para cada um deles utilizando um pacote de otimização linear. A solução ótima apresentada na Tabela 2 foi obtida resolvendo o modelo matemático apresentado na seção 2.1 utilizando o pacote de otimização linear ILOG CPLEX 11. Os resultados destacados mostram as instâncias nas quais o método proposto baseado em VNS obteve a solução ótima o que ocorreu em 10 das 17 instâncias testadas. Nas instâncias em que a metodologia proposta não atingiu a solução ótima os resultados estiverem em média 1,46% acima (entre 1,2% e 2,6%). E o tempo computacional (apresentado na tabela em segundos) utilizado pela metodologia proposta é expressivamente inferior ao tempo utilizado para a obtenção da solução ótima.

Outra observação interessante é que a solução ótima esteve em média 2,1% acima do valor do limite inferior (*lower bound*) e a solução obtida pela metodologia proposta esteve em média 3,3% acima.

Os problemas apresentados na Tabela 3 possuem entre 30 e 500 objetos a serem alocados e foram considerados como instâncias de médio porte devido ao CPLEX não conseguir encontrar uma solução ótima em um tempo inferior a uma hora (3600 segundos). A solução CPLEX apresentada na Tabela 3 apresenta a melhor solução encontrada dentro de uma hora e a coluna *GAP* apresenta a porcentagem que esta solução está acima do limite inferior calculado pelo CPLEX.

Entre os 23 problemas resolvidos de médio porte em cinco deles a metodologia proposta encontrou uma solução pior que a obtida pelo CPLEX (em média 0,92% acima). Em 11 dos problemas a metodologia proposta encontrou uma solução melhor que a obtida pelo CPLEX (em média 1,81% abaixo). A metodologia proposta ficou em média 2,59% acima do valor do

limite inferior, ou seja, muito próximo do valor que a solução ótima esteve acima do limite inferior (2,31%) nos problemas de pequeno porte.

Tabela 3: Resultados obtidos para instâncias de médio porte.

Número de Objetos	<i>Lower Bound</i>	Solução CPLEX				Solução VNS					GAP (%)
		Custo Sol	Veic Grande	Veic Médio	Veic Pequeno	Custo Sol	Tempo CPU (seg)	Veic Grande	Veic Médio	Veic Pequeno	
30	1.215	1.240	7	4	0	1.260	0,8	8	3	0	1,32
35	1.362	1.400	11	0	1	1.400	0,9	10	2	0	2,24
40	1.444	1.500	11	1	1	1.480	1,2	9	4	0	3,26
45	1.966	2.000	15	2	0	2.000	1,5	15	2	0	1,28
50	2.002	2.040	17	0	0	2.060	1,4	14	3	1	1,60
55	1.998	2.040	17	0	0	2.060	1,5	13	5	0	1,82
60	2.356	2.400	20	0	0	2.420	1,8	16	5	0	1,58
70	2.783	2.860	23	1	0	2.840	2,2	22	2	0	2,33
120	4.921	5.000	41	0	1	5.000	6	41	0	1	1,26
120	4.930	5.040	42	0	0	5.040	8	37	6	0	1,91
120	4.644	4.720	38	0	2	4.720	5	36	4	0	1,36
120	4.687	4.760	39	0	1	4.760	6	38	2	0	1,29
120	4.687	4.760	39	0	1	4.760	5	38	2	0	1,29
250	9.507	9.680	80	0	1	9.620	17	78	1	2	1,53
250	9.737	9.880	81	0	2	9.860	18	79	3	1	1,38
250	9.659	9.820	81	1	0	9.780	17	79	3	0	1,17
250	9.572	9.680	80	0	1	9.700	18	77	3	2	0,86
250	9.731	9.880	81	0	2	9.860	18	79	3	1	1,25
500	18.968	19.780	159	7	0	19.160	73	154	6	1	4,43
500	19.282	20.060	164	3	1	19.460	96	158	5	0	3,62
500	19.339	20.260	164	5	1	19.540	79	159	3	2	3,72
500	16.567	20.400	165	6	0	19.720	85	161	4	0	2,69
500	19.691	20.540	167	5	0	19.900	95	160	7	0	3,88
Total	181.048	189.740	1.542	35	15	186.400	557,3	1.481	78	11	

Os resultados da Tabela 4 são relativos às instâncias de grande porte e possuem 1.000 cargas a serem alocados aos veículos. Para os problemas cujos resultados estão apresentados na Tabela 4, o computador utilizado nos testes (Core2Quad 2.4GHz, 2GB de RAM e 150 GB de HD) não conseguiu nem carregar o modelo matemático utilizado no CPLEX devido à falta de memória disponível no computador mesmo configurando o CPLEX para utilizar o espaço disponível no disco rígido para diminuir a necessidade de utilização da memória RAM do computador utilizado nos testes.

A heurística baseada no VNS proposta neste trabalho conseguiu resolver os problemas relativos às instâncias de grande porte com um tempo computacional, aproximadamente entre cinco e onze minutos (o tempo indicado na Tabela 4 está em segundos). Os resultados obtidos pela metodologia proposta estão 0,83% acima do limite inferior (*lower bound*), o que pode indicar que a metodologia proposta pode ter conseguido se aproximar da solução ótima. Cabe ressaltar que a heurística proposta consegue resolver estes problemas de grande porte mesmo em computadores mais modestos (entretanto o tempo de processamento será maior do que os apresentados na Tabela 4).

Tabela 4: Resultados obtidos para instâncias de grande porte.

Número de Objetos	Lower Bound	Solução VNS				
		FO	Tempo	Grande	Médio	Pequeno
1.000	38.249	38.560	372	315	6	2
1.000	38.905	39.260	328	319	9	1
1.000	39.380	39.720	320	322	10	1
1.000	39.444	39.760	320	325	6	2
1.000	38.087	38.420	635	312	9	1
1.000	38.256	38.600	403	312	10	2
1.000	37.844	38.140	408	309	9	2
1.000	38.704	39.000	319	322	2	2
1.000	38.250	38.540	409	315	5	3
1.000	38.105	38.400	348	313	6	3
1.000	38.337	38.640	350	314	8	2
1.000	38.450	38.740	306	319	3	2
1.000	37.655	38.000	540	307	10	2
1.000	37.947	38.260	579	310	9	2
1.000	37.814	38.160	308	310	8	2
1.000	38.574	38.900	580	313	11	3
1.000	38.691	39.000	306	319	4	4
1.000	38.765	39.140	323	321	3	4
1.000	38.227	38.520	503	311	8	5
1.000	38.336	38.660	316	314	5	6
Total	768.020	774.420	7.973	6.302	141	51

5. CONSIDERAÇÕES FINAIS

Este trabalho abordou um problema de distribuição física em que se busca a otimização do agrupamento de entregas e carregamento nos veículos de modo a reduzir a frota necessária e o custo com o frete pago aos transportadores, que é modelado como um problema de *bin-packing* com *bins* heterogêneos (VSBPP). Trata-se de um problema ainda pouco estudado na literatura, ao contrário do problema com *bins* idênticos.

A metodologia proposta baseia-se na meta-heurística VNS e para a sua avaliação foram consideradas instâncias de teste derivadas de instâncias *benchmarking* da literatura para o BPP, adaptadas para o VSBPP. Também foi feita uma comparação com os resultados exatos obtidos através da utilização de um pacote de otimização linear (ILOG CPLEX versão 11).

Os resultados indicam que a metodologia proposta permitiu obter a solução ótima na maioria dos testes, nos quais foi possível a obtenção da solução ótima da mesma. E nos demais testes a solução obtida sempre esteve próxima do limitante inferior (menos de 3% acima), sendo que as soluções ótimas encontradas encontram-se sempre acima dos respectivos limitantes (mais do que 2% em média), o que indica que a estratégia de solução proposta atinge resultados de excelente qualidade com tempos de processamento muito reduzidos. A heurística baseada em VNS implementada também se mostrou mais rápida que a solução ótima do modelo matemático e nos problemas de grande porte foi a única a conseguir obter uma solução, a qual esteve menos de 1% acima do limite inferior.

Inúmeras são as possíveis extensões para a continuidade desta pesquisa, tendo em vista o número reduzido de trabalhos na literatura, incluindo a investigação de novos métodos de solução e a aplicação e validação dos métodos propostos a problemas reais que envolvem o agrupamento de cargas no contexto da distribuição física de produtos.

Outra proposta para a continuidade deste trabalho é o estudo do problema de *bin-packing* duas dimensões (volume e peso dos objetos) e frota heterogênea. Este é um problema muito comum no cotidiano das empresas e ainda praticamente inexplorado na literatura científica.

Agradecimentos

Ao Conselho Nacional de Desenvolvimento Científico e Tecnológico – CNPq – Brasil pelo apoio para a realização da pesquisa que resultou no presente trabalho.

REFERÊNCIAS BIBLIOGRÁFICAS

- Burkard, R.E., Zhang, (1997), Bounded space on-line variable-sized bin packing. *Acta Cybernetica* v. 13, p. 63-76.
- Carvalho, J.M.V. (1999), Exact solutions of bin-packing problems using column generation and branch-and-bound, *Annals of Operations Research*, v. 86, p. 629-659.
- Coffman, E.G., Garey e M.R., Johnson, D.S. (1997) Approximation algorithm for bin-packing: a survey. In: *Hochbaum D (eds) Approximation algorithm for NP-hard problems*. PWS Publishing, Boston, p. 46-93.
- Correia, I., Gouveia, L. e Saldanha-da-Gama, F. (2008), Solving the variable size bin packing problem with discretized formulations. *Computers & Operations Research*, v. 35, p. 2103-2113.
- Cunha, C. B, et al. (2008) Heurísticas para o Problema de *Bin-Packing* no Contexto da Distribuição Física de Produtos. In: *XL SBPO - 40º Simpósio Brasileiro de Pesquisa Operacional*, 2008, João Pessoa. Anais p. 712-723.
- Dyckhoff, H. (1990), A typology of cutting and packing problems. *European Journal of Operational Research*, v. 44, p. 145-159.
- Eilon, S. e Chrisofides, N. (1971), The loading problem, *Management Science*, v. 17, p. 259-268.
- Falkenauer E. (1996) A Hybrid Grouping Genetic Algorithm for Bin Packing. *Journal of Heuristics* 2, p. 5-30.
- Fleszar, K. e Hindi, K.S. (2002), New Heuristics for one-dimensional bin-packing, *Computers and Operations Research*, v. 29, p. 821-839.
- Friesen, D.K. e Langston, M.A. (1986), Variable-sized bin packing. *SIAM J. Comput.* v. 15, p. 222-230.
- Garey, M.R. e Johnson, D.S. (1979) *Computers and intractability: a guide to the theory of NP-completeness*. W. H. Freeman, San Francisco
- Gupta, J.N.D. e Ho, J.C. (1999), A new heuristic algorithm for the one-dimensional bin-packing problem. *Production Planning & Control*, v. 10, p. 598-603.
- Haouari, M., Serairi, M. (2009), Heuristics, for the variable sized bin-packing problem, *Computers and Operations Research*, v. 36, p. 2877-2884.
- Hoare, C. A. R. (1962) Quicksort. *Computer Journal*, v. 5:1, p. 10-15.
- Hoto, R., Arenales, M. e Maculan, N. (2007), The one dimensional Compartmentalized Knapsack Problem: A case study. *European Journal of Operational Research* v. 183, p. 1183-1195.
- Kang, J., Park, S. (2003), Algorithms for the Variable Sized Bin-packing Problem, *European Journal of Operation Research*, v. 147, p. 365-372.
- Marques, F. P. e Arenales, M. N. (2007). The constrained compartmentalized knapsack problem. *Computers and Operations Research*, v. 34, p. 2109-2129.
- Miura, M. (2008) *Modelagem heurística no problema de distribuição de cargas fracionadas de cimento*. Dissertação de Mestrado. Escola Politécnica da USP, São Paulo.
- Mladenovic, N. e P. Hansen (1997) Variable Neighborhood Search. *Computers and Operations Research*, v. 24, p. 1097-1100.
- Scholl, A., Klein, R. e Jürgens, C. (1997), BISON: a fast hybrid procedure for exactly solving the one-dimensional bin-packing problem. *Computers & Operations Research*, v. 24, n. 7, p. 627-645.
- Zhang, G. (1997), A New Version of On-line Variable-sized Bin-packing, *Discrete Applied Mathematics*, v. 72, p. 193-197.