



Otimização na alocação dinâmica de veículos no transporte rodoviário de cargas completas entre terminais

Optimization in dynamic allocation of vehicles on road transportation of full load between terminals

Rejane Arinos Vasco¹
Reinaldo Morabito¹

Resumo: Este artigo trata do problema da alocação dinâmica (multiperíodos) de veículos (PADV) no transporte rodoviário de cargas completas entre terminais. O PADV pertence à classe de problemas de alocação de recursos multiperíodos e consiste em definir “movimentos” de uma frota de veículos que realiza viagens entre terminais geograficamente dispersos que interagem entre si. Esses movimentos podem ser: veículos carregados com carga completa, vazios para reposicionamento ou mantidos em um terminal de um período para outro como provisão para o atendimento de demandas futuras. A ênfase é dada na caracterização do problema em situações reais, na modelagem matemática do problema e na sua solução utilizando um método exato e métodos aproximados, incluindo uma heurística GRASP. Resultados baseados em um estudo de caso de uma transportadora brasileira são apresentados e analisados mostrando que a abordagem pode ser efetiva para apoiar decisões na prática.

Palavras-chave: Alocação dinâmica de veículos. Otimização multiperíodo. Transporte rodoviário. Heurística GRASP.

Abstract: *This paper deals with the dynamic (multi-period) vehicle allocation problem (DVAP) in the road transportation of full loads between terminals. The DVAP belongs to a class of problems dealing with multi-period resource allocation and consists of defining the “movements” of a fleet of vehicles that transport goods between terminals with a wide geographical distribution and which interact among themselves. These movements may be of fully-laden vehicles, unladen for repositioning, or held at a terminal to meet future demands. Emphasis is given on the characterization of the problem in real situations, mathematical modeling of the problem, and the use of exact and heuristic methods to solve the problem, including the GRASP heuristic. Results based on a case study of a transportation company are presented and analyzed, showing that the approach can be effective in supporting practical decisions.*

Keywords: *Dynamic vehicle allocation. Multi-period optimization. Freight transportation. GRASP heuristic.*

1 Introdução

Este estudo se propõe a contribuir para apoiar decisões no nível operacional da gestão de frotas no transporte rodoviário de cargas. Em particular, na otimização do uso de veículos de média e grande capacidade no transporte de transferência de cargas entre terminais, tendo como fator crítico e determinante a maximização da utilização dos recursos disponíveis nas operações.

A alocação de veículos vazios devido às diferenças geográficas de demanda por serviços de transporte muitas vezes acumula veículos em uma região onde eles não são necessários, ou gera um déficit de veículos em regiões em que há demanda. Dessa forma, os veículos devem mover-se vazios, ou cargas adicionais de retorno devem ser encontradas, com o objetivo de

reposicioná-los onde são necessários para atenderem uma demanda conhecida ou prevista nos próximos períodos de planejamento em nível operacional.

A movimentação de veículos vazios não contribui diretamente para o lucro da empresa mas é essencial para a continuidade das operações e manutenção do nível de serviço. O balanço de veículos vazios, ou melhor, a distribuição de veículos vazios para balancear a oferta e a demanda em períodos futuros, é um componente central do planejamento e controle das operações de empresas brasileiras de transporte rodoviário de cargas, constituindo-se inclusive num dos maiores desafios para o gerenciamento da frota.

O problema de alocação dinâmica de veículos (PADV) surge em uma variedade de situações que

¹ Departamento de Engenharia de Produção, Universidade Federal de São Carlos – UFSCar, Rod. Washington Luiz, Km 235, CEP 13565-905, São Carlos, SP, Brasil, e-mail: morabito@ufscar.br

envolvem o gerenciamento de uma frota ao longo de um dado período de tempo, possuindo aplicações no transporte rodoviário de cargas (completas e fracionadas), no transporte ferroviário e no transporte marítimo, com o reposicionamento de contêineres vazios. Para o transporte rodoviário de cargas, um dos primeiros problemas estudados na literatura foi o PADV para o gerenciamento de uma grande frota de caminhões.

O trabalho de Powell, Sheffi e Thiriez (1984) foi um dos pioneiros na literatura que estudou o PADV para o transporte rodoviário de cargas. Foi apresentado um modelo dinâmico não linear, levando em conta as incertezas nas previsões de demanda. Em seguida, Powell (1986) refinou e estendeu o modelo anterior e incorporou incertezas na quantidade de veículos mantidos em estoque, também acompanhando tanto o movimento de veículos cheios como vazios.

Em estudos subsequentes, Powell (1987, 1988) apresentou modelos alternativos para serem usados em ambientes operacionais, na determinação de como gerenciar uma frota de veículos, antecipando as consequências das decisões tomadas. Ele também revisou o PADV no contexto de transporte de cargas completas por caminhões, com atenção especial ao despacho e reposicionamento de caminhões em antecipação a demandas futuras previstas, e revisou quatro metodologias: (a) Redes de transporte determinísticas; (b) Redes não lineares estocásticas; (c) Processos de programação estocástica; e (d) Processos de decisão markovianos. Ele comentou que os modelos determinísticos, apesar de serem mais simples, são viáveis na prática. Uma aplicação prática em uma transportadora de carga completa nos Estados Unidos, tratando não somente o PADV mas também o problema de alocação de motoristas e incorporando incertezas na demanda futura por transportes é reportada por Powell (1988).

Dejax e Crainic (1987) estudaram e propuseram uma taxonomia para o problema de fluxo de veículos vazios no transporte de cargas, fazendo ainda uma ampla revisão da literatura existente sobre o tema. Frantzeskakis e Powell (1990) desenvolveram uma nova heurística para solução do PADV com demandas incertas e Braklow et al. (1992) apresentaram um modelo interativo de otimização em larga escala, chamado pelos autores de SYSNET, com o objetivo de definir o roteamento da carga parcelada em uma rede de transporte e também o reposicionamento de veículos vazios, a fim de responder ao desbalanceamento da demanda.

Powell, Jaillet e Odoni (1995) aprofundaram e estenderam o estudo do problema de alocação dinâmica de veículos e suas diversas variações para os modais rodoviário, ferroviário, aéreo e também para o caso de reposicionamento de contêineres. Os autores apresentam ainda uma formulação básica

do problema, expressa como uma rede dinâmica para um único tipo de veículo e, também, para múltiplos tipos de veículos. Estudos subsequentes, Powell (1996) e Powell e Carvalho (1998a, b), apresentaram um modelo híbrido para combinar as melhores características dos modelos apresentados anteriormente, inclusive um teste experimental com uso de um simulador denominado pelos autores de MIDAS (*Micro Dispatch and Simulation*). O problema foi representado em uma rede denominada *Logistics Queueing Network* (LQN).

Hall (1999) desenvolveu métricas/dimensões para o desbalanceamento espacial e temporal nos fluxos de carga e estudou a aplicação dessas métricas em uma rede de transporte rodoviário. Hall e Zhong (2002) investigaram métodos de controle descentralizados para o gerenciamento de equipamentos no transporte de cargas parceladas. O desempenho foi medido no que diz respeito a: (i) custo de se mover equipamentos vazios, (ii) custo de propriedade dos equipamentos e (iii) pedidos atrasados devido a falta de equipamentos. A proposta incluiu duas etapas: resolver um programa linear para determinar onde obter veículos vazios e reprogramar os pedidos para determinar quando mover um equipamento vazio. Outra discussão do PADV aparece em Ghiani, Laporte e Musmanno (2003).

Essa breve revisão do PADV mostra que foram desenvolvidos alguns estudos principalmente para o modal rodoviário, sendo que a maioria dos autores incorporou nos modelos matemáticos a incerteza na demanda de cargas a serem transportadas. De forma geral, o problema a ser resolvido integra o nível de decisão operacional, tendo ainda alguns trabalhos apresentado resultados de aplicações práticas em empresas transportadoras de cargas nos Estados Unidos. Foram ainda propostos modelos matemáticos não lineares como, por exemplo, modelos em que a função objetivo é côncava. Com relação aos métodos de solução empregados, certos trabalhos aplicaram técnicas de otimização estocástica, simulação e o desenvolvimento de heurísticas. Não foi encontrado nenhum estudo aplicando essas técnicas para resolver esses problemas no transporte rodoviário de cargas no Brasil.

No presente artigo, o PADV é estudado com foco no transporte rodoviário de cargas completas entre terminais. Os movimentos podem ser: veículos carregados com carga completa, veículos vazios para reposicionamento, ou veículos mantidos em um terminal de um período para outro como provisão para o atendimento de demandas futuras. A ênfase é dada na caracterização do problema em situações reais de transportadoras brasileiras, com base em estudos de caso realizados por Vasco (2012).

O objetivo é propor um modelo de programação matemática que represente satisfatoriamente o PADV dessas transportadoras e estudar métodos de solução

para resolvê-lo em situações práticas. Experimentos computacionais são apresentados, mostrando que a abordagem pode ser útil no apoio a decisões na prática.

Na próxima seção é apresentado o modelo de programação linear inteira utilizado. Na seção 3 são apresentadas uma heurística construtiva gulosa, um procedimento de melhoria utilizando busca local e uma aplicação da heurística *Greedy Randomized Adaptive Search Procedure* (GRASP) (FEO; RESENDE, 1989, 1995) para a solução do PADV. Já na seção 4 são apresentados os resultados obtidos com os experimentos computacionais realizados para a solução de um conjunto de problemas de tamanho e complexidade similares aos encontrados na prática, utilizando as heurísticas propostas. As conclusões e perspectivas para trabalhos futuros são apresentadas na seção 5.

2 Modelagem matemática do PADV

Dentre os modelos revisados na literatura para representar o PADV, utilizou-se o modelo apresentado em Ghiani, Laporte e Musmanno (2003) como ponto de partida para a modelagem do problema em estudo. Esse modelo baseia-se em formulações anteriores de Powell, Sheffi e Thiriez (1984) e Powell (1986). No modelo proposto foram considerados aspectos importantes do problema prático do transporte rodoviário de cargas encontrados em empresas brasileiras como, por exemplo, a situação em que certos grupos de veículos não podem realizar determinadas rotas.

Neste trabalho optou-se pela modelagem determinística para resolver o problema em função de algumas razões: (i) maior facilidade para compreensão, desenvolvimento e implementação da abordagem de otimização; (ii) os modelos determinísticos são computacionalmente mais fáceis de serem resolvidos do que os modelos estocásticos (BIRGE; LOUVEAUX, 1997); (iii) esses modelos não necessitam de muitos dados históricos como, por exemplo, nos modelos estocásticos, para se caracterizar distribuições de probabilidade das variáveis aleatórias envolvidas; e (iv) esses modelos são flexíveis para análises do tipo *what-if*, assim como permitem fáceis e importantes análises de sensibilidade e de cenários alternativos.

A seguir são descritos os conjuntos, parâmetros e variáveis utilizados para a modelagem do PADV, com foco na solução do problema prático encontrado em empresas de transporte rodoviário de cargas no Brasil.

2.1 Conjuntos

Os conjuntos usados no modelo são:

- Conjunto N : Composto por todos os terminais da rede de transporte em estudo, entendendo-se como terminal o ponto onde se originam ou para o qual se destinam cargas. Esses terminais

podem ainda ter função mista, exercendo também a atividade de consolidação e de transbordo de cargas.

- Conjunto E : Formado pela subdivisão de toda a frota disponível em grupos para o transporte de cargas, foi proposto neste trabalho distinguir os veículos, seja pela forma de sua remuneração, pelas rotas praticadas ou, ainda, pelo tratamento individual de cada veículo, placa a placa, sendo nesse caso $|E|$ igual à quantidade de veículos existentes na frota. A frota foi considerada como “homogênea” no que diz respeito à capacidade transportada de cargas completas entre terminais e desempenho, resultando em velocidades médias equivalentes nos deslocamentos entre terminais.
- Conjunto T : Formado por períodos discretizados e definido a partir do horizonte de planejamento usualmente utilizado em situações práticas, em que são realizados planos semanais, muitas vezes de maneira informal. Para se aproximar da realidade, tanto do tempo de viagem entre terminais como também do período para demanda, cada dia pode ser subdividido, por exemplo, em seis períodos de quatro horas (ou menos, dependendo da frequência das decisões envolvidas), totalizando 36 períodos. Uma quantidade menor de períodos, apesar de tornar o problema menor e mais simples de ser resolvido, poderia gerar planos que não poderiam ser utilizados na prática devido à maior imprecisão no tempo de viagem entre terminais e também no período da demanda.

2.2 Parâmetros

Os parâmetros do modelo são:

- τ_{ij} : Tempo de viagem entre terminais, medido em quantidade de períodos, entre os terminais i e j , $i \in N, j \in N$. Em termos práticos, esse parâmetro representa o tempo médio de deslocamento e independe do tipo de veículo utilizado, uma vez que a frota foi considerada homogênea.
- d_{ijt} : Demanda por serviços de transporte (em número de veículos cheios) entre os terminais i e j no período t , $i \in N, j \in N, t \in T$. Uma demanda consiste em uma carga com peso (volume ou frete) suficiente para que se justifique a liberação de um veículo para viagem do terminal de origem ao terminal de destino.
- p_{ij}^v : Lucro (receita - custos operacionais diretos) obtido ao se realizar a rota de i para j utilizando-se um veículo do tipo v , $i \in N, j \in N, v \in E$.

- c_{ij}^v : Custo de deslocamento vazio de veículos do tipo v de i para j , $i \in N, j \in N, v \in E$. Para este trabalho, para o caso de frota própria, esse custo está diretamente ligado ao custo operacional, composto por custos fixos e variáveis. Para o caso da frota própria, o custo por quilômetro é calculado pela somatória dos custos fixos dividido pela distância percorrida no período. Já o custo variável é obtido diretamente por quilômetro. Para se calcular o valor de c_{ij}^v no caso de frota própria, multiplica-se a distância em quilômetros de i para j pelo custo operacional por quilômetro. Para veículos terceirizados, há uma negociação prévia do valor a ser pago por quilômetro percorrido, que é multiplicado pela distância no percurso de i para j .
- m_{it}^v : Quantidade de veículos do tipo (ou grupo) v que entram no sistema no terminal i no período t , $i \in N, t \in T, v \in E$.
- A_{ij}^v : Parâmetro binário que assume valor igual a 1 caso o veículo do tipo v possa ser utilizado para realizar o percurso de i para j , ou 0, caso contrário, $i \in N, j \in N, v \in E$.

Esse parâmetro é utilizado para representar situações práticas em que certos veículos não realizam determinadas rotas por motivos diversos como, por exemplo: (i) rotas que não incluam passagem (na ida ou na volta) no município de domicílio da família do motorista, (ii) falta de locais específicos para manutenção do veículo, (iii) inexistência de pontos para troca de motoristas, (iv) transportadores autônomos de cargas que nem sempre conhecem todos os percursos, (v) oferta ou escassez de carga de retorno, (vi) condições físicas da rodovia versus condições de manutenção dos veículos, (vii) risco do percurso (ou rota) versus o valor limite (em unidades monetárias) permitido pela seguradora ao condutor, dentre outros.

2.3 Variáveis

As variáveis de decisão do modelo são:

- X_{ijt}^v : Fluxo (número) de veículos cheios do tipo v usados para satisfazer a demanda d_{ijt} , $i \in N, j \in N, t \in T, v \in E$. Vale ressaltar que $X_{ijt}^v > 0$ somente se $A_{ij}^v = 1$, caso contrário essa variável não existirá na construção do modelo matemático, ou será simplesmente pré-fixada com valor nulo.
- Y_{ijt}^v : Fluxo (número) de veículos vazios do tipo v de i para j no período t , $i \in N, j \in N, t \in T, v \in E$. Vale ressaltar que se $i = j$, o movimento é dito de “estoque”, e o veículo v é mantido no terminal de origem i de um período a outro. Da

mesma forma que em $X_{ijt}^v, Y_{ijt}^v > 0$ somente se $A_{ij}^v = 1$, caso contrário essa variável não existirá na construção do modelo matemático, ou será simplesmente pré-fixada com valor nulo.

2.4 Modelo matemático

O modelo a seguir é uma modificação do modelo usualmente utilizado para representar o PADV (GHIANI; LAPORTE; MUSMANNO, 2003). No modelo incorporou-se a possibilidade de utilização de mais de um tipo (ou grupo) de veículos e também foi feita a inclusão da restrição de que certos tipos (ou grupos) de veículos não podem realizar determinadas rotas. Essas modificações no modelo original tiveram como objetivo representar melhor a realidade prática das empresas transportadoras de carga no Brasil, pois nem todos os veículos podem ser tratados da mesma forma (em um único grupo) como, por exemplo, (i) próprios, (ii) terceiros (agregados), (iii) transportador autônomo de carga, entre outros.

O PADV com múltiplos tipos de veículo (conjunto E) pode ser modelado como um problema linear de fluxo *multicommodity* (multifluxo) de mínimo custo, transformando a restrição de atendimento da demanda em uma restrição de mochila (*bundle constraint*) (POWELL; JAILLET; ODONI, 1995). Dessa maneira, diferente do modelo usualmente utilizado para representar o PADV, ele é de difícil solução exata (NP-difícil) (GHIANI; LAPORTE; MUSMANNO, 2003). Considerando os conjuntos, parâmetros e variáveis já definidos, o modelo matemático determinístico para representar o PADV pode ser definido como o seguinte modelo de programação linear inteira:

$$\max Z_1 = \sum_{i \in N} \sum_{j \in N} \sum_{t \in T} \sum_{v \in E} (p_{ij}^v X_{ijt}^v - c_{ij}^v Y_{ijt}^v) \quad (1)$$

Sujeito a

$$\sum_{j \in N} (X_{ijt}^v + Y_{ijt}^v) - \sum_{\substack{k \in N \\ k \neq i \\ t > \tau_{ik}}} (X_{kit}^v + Y_{kit}^v) = m_{it}^v + Y_{it}^v(t-1), \quad (2)$$

$$\forall i \in N, \forall t \in T, \forall v \in E$$

$$\sum_{v \in E} X_{ijt}^v \leq d_{ijt} \quad \forall i, j \in N, \forall t \in T \quad (3)$$

$$X_{ijt}^v = 0, e Y_{ijt}^v = 0, se A_{ij}^v = 0, \forall i, j \in N, \forall t \in T, \forall v \in E \quad (4)$$

$$X_{ijt}^v \geq 0, Y_{ijt}^v \geq 0, \text{inteiros}, \forall i, j \in N, \forall t \in T, \forall v \in E \quad (5)$$

A função objetivo (1) visa maximizar a diferença entre a receita obtida com o transporte de cargas (movimentos cheios) e o custo de se deslocar veículos vazios para reposicionamento da frota. As restrições

(2) impõem a conservação de fluxo no início de cada período, enquanto as restrições (3) asseguram que o número de veículos cheios seja menor ou igual à demanda. Como consequência, temos que as diferenças $\sum_{v \in E} X_{ijt}^v - d_{ijt}^v, \forall i, j \in N, \forall t \in T$, representam as cargas que não puderam ser atendidas. As restrições (3) tornam o modelo de multifluxos (*multicommodity*), em cada arco (i, j) em t , e, portanto, um modelo de programação linear inteira (restrições (5)). Caso o problema fosse de monofluxo, envolvendo apenas uma *multicommodity*, poderia ser simplesmente modelado e resolvido por programação linear (GHIANI; LAPORTE; MUSMANNO, 2003).

Esse modelo matemático pode ser útil para apoiar decisões de alocação de veículos vazios, selecionando quais demandas deverão ser atendidas, levando-se em conta os diferentes (grupos de) veículos e a contribuição ao lucro da empresa. Por meio da identificação prévia das demandas que não poderão ser atendidas via modelo, a empresa transportadora poderia se antecipar e contratar veículos de terceiros em certos terminais e períodos para que todas as cargas sejam transportadas, sem necessariamente gerar atrasos.

A restrição (4) visa garantir que veículos de um dado tipo transitem, cheios ou vazios, somente em um determinado conjunto de percursos. Convém salientar que essas restrições podem ser substituídas por simples procedimentos de fixação de variáveis, disponíveis nos pacotes de otimização, ou ainda no momento da geração do modelo, eliminando-se essas variáveis do problema.

Na prática de algumas empresas transportadoras brasileiras é comum utilizar-se a placa do veículo ou o número de frota no momento do planejamento do atendimento das demandas e definição dos movimentos cheios e vazios a serem realizados. Como pretende-se que o modelo proposto possa ser devidamente validado em ambientes reais de operação, a classificação de veículos em grupos não se mostrou totalmente adequada, pois o tomador de decisão espera que a saída do modelo defina os movimentos a serem realizados por cada veículo individualmente e não em grupos. Assim, cada veículo passa a ser considerado como um grupo diferente. Para isso, fixou-se $m_{it}^v = \{0,1\}$, $\forall i \in N, \forall t \in T, \forall v \in V$, representando se o veículo v entra no sistema no terminal i no período t , satisfazendo a Equação 6:

$$\sum_{i \in N} \sum_{t \in T} \sum_{v \in E} m_{it}^v = |E| \quad (6)$$

Ao se tratar individualmente cada veículo, espera-se que a complexidade para solução do modelo matemático aumente consideravelmente, em função do aumento do número de variáveis e restrições. Convém observar que o modelo permite fáceis adaptações e é razoavelmente flexível para incorporar outros

aspectos da operação. Por exemplo, nos casos em que parte da demanda d_{ijt}^v só pode ser atendida por um grupo de veículos v (como, por exemplo, em *green logistics*), basta separar essa demanda como um parâmetro específico d_{ijt}^v e incluir uma restrição adicional ao modelo, do tipo: $X_{ijt}^v \leq d_{ijt}^v$. Esse pode ser o caso de alguns clientes que exigem veículos adequados, que atendam as especificações dos seus programas de sustentabilidade.

3 Heurísticas para solução do PADV

Nesta seção são apresentadas heurísticas para solução do PADV para casos em que os veículos pertencem a muitos grupos. Por exemplo, o caso em que cada veículo necessita ser tratado individualmente, permitindo distinguir os movimentos realizados pela frota em problemas de tamanho e complexidade similares aos encontrados na prática, a resolução de modelos via linguagem de modelagem e métodos exatos de *softwares* de otimização apresentou-se limitada em função dos requisitos de tempo e memória computacional, conforme mostrado na próxima seção.

Foi desenvolvida uma heurística “gulosa” (*greedy*) para geração de uma solução inicial para o problema, seguida de um procedimento de busca local para a melhoria da solução inicial gerada. Com o objetivo de adicionar variabilidade no processo de obtenção de soluções e também tentar escapar de ótimos locais, foi também desenvolvido um procedimento aplicando-se GRASP para a solução do problema, que utiliza as heurísticas construtiva e de busca local.

3.1 Uma heurística “gulosa” para geração de soluções iniciais para o PADV

Heurísticas gulosas para a obtenção de soluções iniciais viáveis para problemas de difícil otimização são muito utilizadas, pois permitem rapidamente, e muitas vezes de forma intuitiva e com conhecimento específico do problema, gerar ótimos locais, servindo de ponto de partida para outras heurísticas mais sofisticadas, que visam ir além de ótimos locais obtidos pelo procedimento “guloso”. Para a obtenção de uma solução inicial viável para o PADV, optou-se pelo desenvolvimento de uma heurística construtiva “gulosa” que levasse em conta as demandas a serem atendidas, nos diferentes terminais e em diferentes períodos, priorizando as demandas mais lucrativas (com maior p_{ij}^v), verificando-se qual (ou quais) veículos podiam ser deslocados vazios, caso necessário, para o atendimento da demanda. Essa heurística foi utilizada como ponto de partida para um procedimento de melhoria, ou ainda para outras heurísticas que utilizam busca local ou utilizam uma população de soluções, como o GRASP.

No pseudocódigo apresentado na Figura 1 é descrita a heurística gulosa proposta para a geração

de soluções iniciais viáveis para o PADV. A ideia básica da heurística é verificar para cada carga não atendida se existe algum veículo que pode “chegar a tempo” para atender a demanda. Entende-se por “chegar a tempo” o tempo de viagem a partir da posição atual do veículo até a origem da carga. É necessário também analisar se esse veículo pode realizar o percurso vazio da sua posição atual até a origem da carga e também da origem da carga até seu destino.

São armazenados em uma estrutura de dados específica todos os movimentos de cada veículo, estejam eles cheios ou vazios, com seus respectivos percursos, períodos, custos e lucros. Todas as cargas são ordenadas cronologicamente e, se houver mais de uma carga no mesmo período, são ainda ordenadas em ordem decrescente de lucro p_{ij}^v , a fim de favorecer cargas que estejam no mesmo período, porém com a possibilidade de serem mais favoráveis do ponto de vista do valor final da função objetivo. Resumidamente, essa ordenação teve como objetivo realizar o transporte de cargas mais próximas ao início do horizonte de planejamento para que os veículos, após esse transporte, estejam liberados o mais brevemente possível para novas designações, tentando aumentar assim a sua produtividade.

Seguindo o procedimento guloso para geração de soluções iniciais para o PADV, cada carga (mais

próxima ao início do horizonte de planejamento e com maior lucro p_{ij}^v) é analisada individualmente e todos os veículos que podem chegar a tempo (caso existam) para atender a demanda são candidatos a serem deslocados vazios para o atendimento. Além do tempo de viagem com o deslocamento vazio, é analisando ainda se o veículo v pode realizar o percurso desde a sua posição atual k até a origem i da carga e também o deslocamento da origem i da carga até o seu destino j , ou melhor, $A_{ki}^v = A_{ij}^v = 1$.

O veículo escolhido será o que possuir o menor custo de deslocamento vazio c_{ij}^v . Após escolhido o veículo, a demanda é atendida e são atualizados os dados de posicionamento do veículo escolhido, deslocando-o para o terminal de destino da carga, no seu respectivo período final $t + \tau_{ij}$, onde t é o período de início da demanda e τ_{ij} o tempo de viagem desde a origem i da carga até o seu destino j . Esse procedimento é realizado até que todas as cargas não atendidas tenham sido avaliadas e uma solução inicial viável para o PADV tenha sido construída.

3.2 Procedimento de melhoria utilizando busca local

Foi desenvolvido um procedimento de melhoria para, a partir da solução inicial da heurística construtiva, investigar quais dentre as cargas não

Procedimento Heurística Gulosa

Ordenar cargas de forma cronológica e em seguida decrescente de lucro p_{ij}^v ;

Enquanto houver carga a ser atendida **faça**

encontrou \leftarrow falso;

custo $\leftarrow \infty$;

para cada veículo v da frota **faça**

se veículo v pode chegar a tempo **e** pode realizar o percurso **então**

se $c_{ij}^v < \text{custo}$ **então**

encontrou \leftarrow verdadeiro;

custo $\leftarrow c_{ij}^v$;

fim se;

fim se;

fim para cada;

se encontrou **então**

realiza o movimento vazio;

desloca o veículo v para a origem i da carga;

realiza o movimento carregado;

desloca o veículo v para o destino j da carga;

grava os movimentos vazios e cheios;

atualiza a quantidade de cargas a serem atendidas;

atualiza a posição e período do veículo v ;

fim se;

fim enquanto;

Fim Heurística Gulosa;

Figura 1. Pseudocódigo para a heurística construtiva gulosa.

atendidas no procedimento guloso seriam mais interessantes de serem atendidas considerando-se o lucro p_{ij}^v .

O procedimento de melhoria descrito no pseudocódigo apresentado da Figura 2 consiste basicamente em verificar, para cada carga não atendida, possíveis veículos que podem atender essa demanda mas que foram deslocados para outra carga na estratégia gulosa.

É mantido um histórico detalhado de todos os movimentos realizados por cada veículo v , armazenando-se todas as suas paradas em um dado

terminal k e período t e, ainda, se o deslocamento realizado para se chegar de k em t foi vazio ou carregado. Com base nesse histórico, o procedimento de melhoria consiste em iterativamente analisar cada posicionamento do veículo em um dado terminal e em um dado período e verificar se ele pode ser deslocado desse espaço-tempo para atender outra carga não atendida. Da mesma forma que na heurística gulosa, é analisado se o veículo pode realizar o percurso da sua posição k para a origem i da carga e também se pode realizar o percurso desde a origem i até o destino j da carga, ou seja, $A_{ki}^v = A_{ij}^v = 1$. O veículo

Procedimento de Melhoria

Criar solução inicial gulosa conforme Figura 1;

Ordenar as cargas não atendidas em ordem decrescente de lucro p_{ij}^v ;

para cada carga não atendida **faça**

encontrou \leftarrow falso ;

valor $\leftarrow \infty$;

para cada veículo v da frota **faça**

para cada movimento realizado pelo veículo v **faça**

para $t \leftarrow T$ **faça**

se veículo v em t no terminal k pode chegar a tempo e $A_{ki}^v = 1$ **então**

seja f a contribuição do veículo v na função objetivo (FO) a partir de k e t ;

calcular a contribuição na FO caso o veículo v atenda a demanda;

$f' \leftarrow p_{ij}^v - c_{ki}^v$;

se $f' > f$ **então**

se $f' >$ valor **então**

encontrou \leftarrow verdadeiro;

valor $\leftarrow f'$;

fim se;

fim se;

fim se;

fim para;

fim para cada;

fim para cada;

se encontrou **então**

apaga movimentos do veículo v a partir de k e t ;

realiza o movimento vazio;

desloca o veículo v para a origem i da carga;

realiza o movimento carregado;

desloca o veículo v para o destino j da carga;

grava os movimentos vazios e cheios;

atualiza a quantidade de carga a ser atendida;

atualiza a posição e período do veículo;

atualiza valor da função objetivo;

fim se;

fim para cada;

Fim procedimento de melhoria;

Figura 2. Pseudocódigo do procedimento de melhoria para as soluções.

escolhido para alteração da solução será aquele que tiver uma maior contribuição no valor corrente da função objetivo.

3.3 Utilizando o GRASP para solução do PADV

Utilizando-se a heurística construtiva gulosa e o procedimento de melhoria, investigou-se se a combinação desses dois procedimentos, juntamente com um mecanismo aleatório para adicionar variabilidade à heurística gulosa, seria capaz de obter soluções de boa qualidade na solução do PADV. Assim, foi desenvolvida uma aplicação do GRASP para a solução do PADV. O GRASP é um procedimento multi-início (*multi-start*) utilizado para a solução de problemas de otimização combinatória. É considerada uma heurística semigulosa que adiciona variabilidade à heurística gulosa (FEO; RESENDE, 1989, 1995), e consiste em um processo iterativo, sendo que cada iteração consiste basicamente em duas fases: (i) uma fase construtiva e (ii) uma fase de busca local. Na fase construtiva cria-se uma solução inicial viável para o problema, enquanto que a vizinhança da solução é explorada até que um ótimo local seja encontrado pela busca local.

Na criação da solução inicial, elementos candidatos são todos os que podem ser incorporados na solução parcial em construção sem violar nenhuma restrição do problema. A definição do próximo elemento a ser incorporado na solução é determinada pela avaliação de todos os elementos candidatos de acordo com uma função de avaliação gulosa. Essa função gulosa geralmente representa o aumento incremental na função de custo devido à incorporação desse elemento na solução em construção. A avaliação do elemento por essa função leva à criação de uma lista restrita de candidatos, ou *Restricted Candidate List* (RCL), formada pelos melhores elementos, isto é, todos aqueles cuja incorporação na solução parcial resulte em um menor custo incremental.

O elemento a ser incorporado na solução parcial é aleatoriamente selecionado dentre os contidos na lista restrita, sendo esse o aspecto probabilístico da heurística. Uma vez que o elemento é selecionado e incorporado na solução parcial, a lista de candidatos é atualizada e os custos incrementais são reavaliados.

O GRASP foi aplicado com sucesso em diversos problemas e aplicações. Em Festa e Resende (2002) são apresentados 230 trabalhos, desde materiais introdutórios e tutoriais sobre GRASP até aplicações em diversos domínios e problemas como, por exemplo, na teoria de grafos, problemas encontrados na operação de transportes, roteamento de veículos, localização de instalações, biologia, sistemas elétricos, problemas de corte e empacotamento, problemas de designação

quadrática, telecomunicações, implementações paralelas, entre outros.

O GRASP implementado para a solução do PADV é descrito no pseudocódigo apresentado na Figura 3. Inicialmente, todas as cargas são ordenadas em ordem decrescente de lucro e enquanto a quantidade máxima de iterações não for alcançada, são geradas soluções gulosas aleatórias a partir desse vetor ordenado e o procedimento de melhoria descrito na Figura 2 é executado a cada solução gulosa gerada. Na criação da solução gulosa aleatória, cada carga ainda não inserida na solução em construção é avaliada individualmente e a sua contribuição na função objetivo, caso seja atendida, é calculada, entendendo-se como contribuição a diferença entre o lucro obtido no transporte e o custo de deslocamento de veículos vazios. Cada carga não atendida é uma candidata a ser inserida na solução em construção.

Após a avaliação da contribuição de cada carga na função objetivo, é criada uma lista restrita de candidatos (RCL), contendo todas as cargas candidatas que possuem contribuição f_c maior ou igual a um valor de corte, ou *cutoff*. O valor de *cutoff* é calculado a partir dos valores máximo e mínimo de contribuição de cada carga candidata e definido como: $cutoff = f_{max} - \alpha(f_{max} - f_{min})$, onde f_{max} e f_{min} representam os valores máximo e mínimo, respectivamente, de contribuição dos candidatos disponíveis. O parâmetro α é um dado de entrada e assume um valor aleatório entre 0 e 1 ($\alpha = UNIF[0,1]$) e fixo para todas as iterações. O valor α possui a função de balancear entre uma escolha puramente gulosa, quando $\alpha = 0$, e uma escolha puramente aleatória, quando $\alpha = 1$.

Foi utilizada a meta-heurística de busca em vizinhança variável, ou *Variable Neighborhood Search* (VNS), proposta por Mladenovic e Hansen (1997), para a realização da busca local após a solução inicial criada pelo GRASP. A busca em vizinhança variável é um método de busca local que explora o espaço de soluções através de trocas sistemáticas de estruturas de vizinhança. Diferentemente de outras meta-heurísticas baseadas em métodos de busca local, o método VNS não segue uma trajetória mas explora vizinhanças gradativamente mais “distantes” da solução corrente, focalizando a busca em torno de uma nova solução somente se um movimento de melhora é realizado.

No VNS parte-se de uma solução inicial s_0 e, a cada iteração, seleciona-se aleatoriamente um vizinho s' dentro da vizinhança $N^{(k)}(s)$ da solução corrente s . Esse vizinho é então submetido a um procedimento de busca local, gerando uma nova solução s'' . Se a solução ótima local s'' for melhor que a solução s corrente, a busca continua a partir de s'' , recomeçando da primeira estrutura de vizinhança $N^{(1)}(s)$. Caso contrário, continua-se a busca a partir da próxima estrutura de vizinhança $N^{(k+1)}(s)$. Com a

Procedimento GRASP

Ordenar as cargas em ordem decrescente de lucro p_{ij}^v ;

Armazenar o índice das cargas ordenadas em um vetor V ;

Enquanto a quantidade máxima de iterações não for alcançada **faça**

Enquanto a solução não estiver completa **faça**

Para cada carga c ainda não inserida na solução **faça**

Verificar a contribuição na função objetivo (FO) da carga $c - f_c$;

Inserir em um vetor a carga candidata c e sua contribuição na FO f_c ;

Atualizar máximos (f_{max}) e mínimos (f_{min}) das contribuições dos candidatos;

Se $f_c > f_{max}$ **então** $f_{max} \leftarrow f_c$;

Se $f_c < f_{min}$ **então** $f_{min} \leftarrow f_c$;

Fim para cada

$cutoff \leftarrow f_{max} - \alpha \times (f_{max} - f_{min})$;

Para cada carga candidata c **faça**

Se $f_c \geq cutoff$ **então**

Inserir candidato na lista restrita RCL;

Fim se

Fim para cada

Sortear aleatoriamente um candidato da lista restrita RCL e inserir na solução;

Fim enquanto

Executar o procedimento *Reduced VNS* (RVNS) conforme pseudocódigo da Figura 4;

Executar o *path-relinking* de intensificação conforme pseudocódigo da Figura 5;

Fim enquanto

Executar o *path-relinking* de pós-otimização entre todos os pares de soluções elite.

Fim procedimento GRASP;

Figura 3. Pseudocódigo da heurística GRASP para solução do PADV.

finalidade de manter o tempo de processamento em níveis aceitáveis, optou-se por utilizar uma variação do VNS chamada *Reduced VNS* (RVNS) (HANSEN; MLADENOVIC; PÉREZ, 2010) que elimina a etapa de busca local na solução s' obtida.

No caso do PADV, duas estruturas de vizinhança são utilizadas ($k = 2$), sendo: (i) Troca do tipo “2-opt”, em que são sorteadas aleatoriamente duas posições distintas do vetor de cargas a serem atendidas para trocarem de posição; e (ii) Troca do tipo “3-opt”, de forma similar à troca “2-opt”, porém com o sorteio aleatório de três posições do vetor de cargas para trocarem de posição.

Foi ainda incorporada uma estratégia de intensificação conhecida por *path-relinking*, originalmente proposta por Glover (1996), com o objetivo de explorar a trajetória ao se conectarem soluções elite obtidas ao longo da execução do GRASP. O *path-relinking* é executado entre um par de soluções: x_s (solução inicial ou *start*) e x_t (solução alvo ou *target*). O procedimento inicia computando-se a diferença simétrica $\Delta(x_s, x_t)$ entre as duas soluções, isto é, o conjunto de movimentos necessários para se alcançar x_t a partir de x_s . Um caminho de soluções é

gerado ligando-se x_s e x_t . A melhor solução x^* nesse caminho é retornada pelo procedimento. Mais detalhes sobre o GRASP com *path-relinking* são apresentados, por exemplo, em Resende e Ribeiro (2005).

O *path-relinking* é executado a cada iteração, denominando-se a solução obtida após a execução do RVNS de x_s e definindo-se uma solução x_t aleatoriamente escolhida dentro de um conjunto de soluções elite que é mantido ao longo da busca. Após a execução de todas as iterações do GRASP, o *path-relinking* é também executado entre todos os pares de soluções contidas no conjunto de soluções elite, como uma etapa de pós-otimização.

Em suma, cada iteração do GRASP consiste de três fases: (i) fase de construção, em que uma solução gulosa aleatória viável é criada, (ii) fase de busca local, em que utiliza-se o RVNS para a obtenção de um ótimo local, e (iii) fase de execução do *path-relinking*, como pós-otimização.

4 Experimentos e resultados computacionais

Nesta seção são apresentados os experimentos computacionais realizados utilizando-se a heurística

Procedimento RVNS

Enquanto o tempo máximo não for alcançado **faça**

$k \leftarrow 1$

Enquanto $k = 1$ **faça**

Sortear aleatoriamente dois pontos distintos do vetor de demandas V ;
 Execute movimentos de troca do tipo “2-opt” com as posições sorteadas;
 Gerar solução gulosa conforme pseudocódigo da Figura 1;
 Executar procedimento de melhoria conforme pseudocódigo da Figura 2;

Se houve melhora na FO **então**

Atualize a melhor solução obtida até o momento;

Senão $k \leftarrow k + 1$;

Fim enquanto

Enquanto $k = 2$ **faça**

Sortear aleatoriamente três pontos distintos do vetor de demandas V ;
 Execute movimentos de troca do tipo “3-opt” com as posições sorteadas;
 Gerar solução gulosa conforme pseudocódigo da Figura 1;
 Executar procedimento de melhoria conforme pseudocódigo da Figura 2;

Se houve melhora na FO **então**

Atualize a melhor solução obtida até o momento;

$k \leftarrow 1$;

Fim enquanto

Fim enquanto {tempo limite do RVNS}

Fim procedimento RVNS

Figura 4. Pseudocódigo da metaheurística *Reduced VNS* (RVNS).

Procedimento *Path_Relinking_Intensificação*

Dado: solução x_s incluindo o vetor de demandas V ;

Verificar a quantidade de soluções existentes no conjunto de soluções elite L ;

Se $|L| < 2$ **então**

Inserir a solução x_s no conjunto de soluções elite;

Senão

Sortear aleatoriamente uma solução alvo x_t entre as soluções elite;

Assegurar que $x_s \neq x_t$;

Calcular a diferença Δ entre x_s e x_t ;

Enquanto $\Delta > 0$ **faça**

Eliminar uma diferença entre os vetores de demanda de x_s e x_t ;

Gerar solução gulosa conforme pseudocódigo da Figura 1;

Executar procedimento de melhoria conforme pseudocódigo da Figura 2;

Atualizar melhor solução obtida até o momento;

Fim enquanto

Fim procedimento *Path_Relinking_Intensificação*

Figura 5. Pseudocódigo do procedimento *path-relinking* de intensificação.

gulosa e o procedimento de melhoria descritos na seção 3 e também o GRASP proposto para a solução do PADV. O desempenho das heurísticas foi comparado com a solução ótima obtida pelo método *branch-and-cut* do *software* de otimização CPLEX (INTERNATIONAL..., 2008), versão 11.1.1 para a plataforma de 64 bits. Foram gerados aleatoriamente

30 problemas de tamanho e complexidade comparáveis aos encontrados na prática de uma transportadora brasileira possuindo: 53 terminais; horizonte de tempo de programação de 36 períodos, tendo cada período 4 horas; demanda de 300 cargas a serem transportadas entre os terminais; e 130 veículos, especificamente caminhões articulados com capacidade

de carga de aproximadamente 25.000 kg, compostos por um veículo trator (cavalo mecânico 4x2), e um semirreboque (carreta) do tipo baú com três eixos.

Os problemas foram gerados variando-se a quantidade de grupos E de 1 até 17, que foi o tamanho limite de problema possível de ser resolvido utilizando-se o CPLEX. Os parâmetros $m_{i,v}$ e A_{ij}^v variaram convenientemente, de acordo com a quantidade de grupos. Foram ainda gerados outros 30 problemas possuindo 130 grupos, exatamente a quantidade de veículos do problema real em que cada veículo pertence a um grupo (i.e., placa a placa), sendo também resolvidos utilizando-se as heurísticas desenvolvidas. Portanto temos 30 problemas testes e 18 grupos (i.e., $|E| = 1, \dots, 17$ e $|E| = 130$), resultando em 540 problemas teste distintos.

Para o GRASP, cada problema foi resolvido em 11 rodadas independentes, utilizando em cada rodada um valor diferente de semente para geração de números aleatórios. Assim, no total, o GRASP resolveu 5.940 problemas e as heurísticas gulosa e de busca local foram executadas uma vez para cada problema. Os parâmetros utilizados para o GRASP foram: quantidade máxima de iterações igual a dez, $\alpha = 0,1$ e o tempo de execução do RVNS igual a 20 segundos. O valor tamanho máximo de soluções contidas no conjunto de soluções elite foi fixado como igual à quantidade de iterações do GRASP, ou seja, igual a dez. Todos os parâmetros foram definidos experimentalmente e mantidos fixos para todos os problemas e rodadas independentes executadas.

Como para os problemas contendo 130 grupos a solução ótima não é conhecida, isto é, não foi possível a obtenção da solução ótima utilizando o CPLEX, nem a obtenção de limitantes inferiores ou superiores para essa solução via CPLEX, dado que não houve memória suficiente para compilar esses problemas no AMPL/CPLEX, foi então utilizada a melhor solução obtida dentre as heurísticas como referência para comparação. As heurísticas foram implementadas na linguagem de programação C++ e todos os processamentos foram realizados em um *laptop* equipado com sistema operacional Linux, CPU Intel(R) Core (TM) 2 DUO T6600 2,2GHz e 4 GB de memória RAM. Somente um processador foi utilizado para os experimentos e o executável foi gerado utilizando-se o compilador Oracle Solaris Studio versão 12.3.

Por limitações de espaço, são aqui apresentadas somente as soluções obtidas para os problemas contendo 17 e 130 grupos. Em Vasco (2012) são apresentadas as soluções completas para todos os grupos e problemas gerados e resolvidos. A heurística gulosa e busca local, juntamente com a melhor solução obtida pelo GRASP, são apresentadas na Tabela 1. Em cada tabela, temos as colunas:

- *Problema*: Nome do problema no formato “ $pxx - vy$ ”, onde x representa o número do problema e y , a quantidade de grupos;
- *Sol. Ótima*: Solução ótima obtida utilizando-se o CPLEX, para os casos em que $|E| = 17$, ou a melhor solução para o caso em que $|E| = 130$;
- *CPUt(s)*: Tempo de execução, em segundos, necessário para se obter a solução reportada;
- *FO*: Valor obtido para a função objetivo;
- *GAP*: Desvio percentual da solução ótima (ou melhor solução obtida) do problema, calculado como $GAP = \frac{Sol.Ótima - FO}{Sol.Ótima} \times 100$.

Na Tabela 1 são apresentados os resultados dos 30 problemas gerados aleatoriamente para o caso em que a frota composta por 130 veículos foi distribuída em 17 grupos. Para esse caso, por exemplo, o desvio médio da solução ótima da heurística construtiva gulosa foi de 10,61%, com um tempo médio de 0,02 segundos para se obter cada uma das soluções. No caso da busca local, para o mesmo conjunto de dados, o desvio médio da solução ótima foi de 9,21%, com um tempo médio de 0,05 segundos por problema. E para o GRASP, o desvio médio foi reduzido para 2,01% com um tempo médio de 39,08 segundos. Para o mesmo conjunto de dados, o CPLEX necessitou em média de 318,95 segundos para resolver otimamente cada problema. O desvio padrão da função objetivo e do tempo de processamento para as 11 rodadas independentes executadas foram de, em média, 84,83 (σ_n) e 0,80 (σ_t), respectivamente.

Já para os problemas contendo 130 grupos, ou seja, o caso em que cada veículo é tratado placa a placa, a solução ótima não é conhecida, pois, como já relatado, não foi possível a obtenção da solução ótima nem de um bom limitante superior utilizando-se o CPLEX. Assim, para esses problemas, na Tabela 1, a solução utilizada como referência foi a melhor solução obtida pelo GRASP. Nesse caso, o desvio médio da melhor solução para a heurística gulosa foi de 8,65%, com um tempo médio de 0,25 segundos para cada problema, e a busca local ficou em média 7,21% distante da melhor solução obtida, com um tempo médio de processamento de 0,72 segundos para cada problema. O tempo médio gasto pelo GRASP foi de 32,68 segundos.

É possível verificar, ainda, que, além de ter obtido ótimos locais para o problema, o GRASP pode ser considerado adequado para a solução do PADV, o que pode ser evidenciado pela dispersão da média dos valores obtidos com a função objetivo para as 11 rodadas independentes. O desvio padrão dos tempos de processamento foi praticamente inexistente e, ainda, a amplitude total, medida pela diferença entre o melhor e o pior valor, foi considerada relativamente pequena para cada problema resolvido.

Tabela 1. Resultados obtidos com as heurísticas gulosa, busca local e GRASP.

Problema	CPLEX		Heurística gulosa			Busca local			GRASP		
	Sol. Ótima	CPUt (s)	FO	CPUt (s)	Gap (%)	FO	CPUt (s)	Gap (%)	FO	CPUt (s)	Gap (%)
p1r-v17	17.079,40	417,43	14.220,80	0,02	16,74	14.653,80	0,04	14,20	16.536,20	38,99	3,18
p2r-v17	18.906,80	239,68	16.768,40	0,02	11,31	16.914,40	0,05	10,54	18.596,00	42,31	1,64
p3r-v17	15.789,00	433,53	13.708,40	0,02	13,18	14.342,80	0,04	9,16	15.271,80	36,33	3,28
p4r-v17	19.352,00	283,05	17.674,00	0,02	8,67	17.674,00	0,04	8,67	18.903,80	43,13	2,32
p5r-v17	17.567,20	274,45	15.329,80	0,02	12,74	15.522,40	0,04	11,64	17.231,80	34,69	1,91
p6r-v17	19.298,00	325,37	17.540,20	0,02	9,11	17.667,80	0,04	8,45	19.042,40	40,34	1,32
p7r-v17	18.878,20	262,55	17.620,60	0,02	6,66	17.620,60	0,03	6,66	18.558,80	39,91	1,69
p8r-v17	17.609,20	246,76	15.516,00	0,02	11,89	15.825,40	0,04	10,13	17.402,40	42,14	1,17
p9r-v17	15.015,00	214,61	13.391,60	0,02	10,81	13.672,20	0,05	8,94	14.804,60	42,03	1,40
p10r-v17	18.093,80	244,48	15.406,00	0,02	14,85	16.377,80	0,05	9,48	17.642,60	34,65	2,49
p11r-v17	16.387,00	192,65	14.093,60	0,02	14,00	14.432,40	0,06	11,93	15.896,00	34,94	3,00
p12r-v17	21.729,80	342,13	20.542,60	0,01	5,46	20.566,40	0,03	5,35	21.494,80	41,89	1,08
p13r-v17	18.939,20	387,85	17.366,80	0,02	8,30	17.366,80	0,04	8,30	18.519,80	39,03	2,21
p14r-v17	19.241,80	278,88	17.833,00	0,02	7,32	17.872,20	0,05	7,12	19.020,60	41,17	1,15
p15r-v17	19.796,20	298,37	17.825,20	0,01	9,96	18.588,40	0,07	6,10	19.431,20	40,83	1,84
p16r-v17	17.294,60	275,78	15.338,20	0,02	11,31	15.476,40	0,04	10,51	17.007,80	39,31	1,66
p17r-v17	16.719,80	323,83	14.681,00	0,02	12,19	15.166,00	0,05	9,29	16.274,00	35,80	2,67
p18r-v17	17.435,40	359,55	16.145,80	0,02	7,40	16.208,20	0,05	7,04	17.024,00	39,70	2,36
p19r-v17	17.777,80	222,18	15.267,20	0,02	14,12	15.453,00	0,06	13,08	17.341,40	34,16	2,45
p20r-v17	19.878,60	357,98	16.868,40	0,02	15,14	17.108,80	0,05	13,93	19.450,60	39,66	2,15
p21r-v17	16.694,80	403,03	15.290,20	0,02	8,41	15.290,20	0,03	8,41	16.234,00	42,10	2,76
p22r-v17	18.541,60	547,59	17.397,20	0,02	6,17	17.397,20	0,03	6,17	18.269,20	41,26	1,47
p23r-v17	21.119,20	353,53	19.345,60	0,02	8,40	19.469,40	0,05	7,81	20.866,20	42,50	1,20
p24r-v17	17.489,40	236,83	15.487,20	0,02	11,45	16.533,20	0,06	5,47	17.140,20	35,10	2,00
p25r-v17	16.819,40	268,81	14.757,60	0,02	12,26	14.795,80	0,04	12,03	16.214,60	40,03	3,60
p26r-v17	19.995,40	293,20	17.934,20	0,02	10,31	18.106,80	0,06	9,45	19.704,00	36,30	1,46
p27r-v17	19.557,60	387,43	17.765,00	0,02	9,17	17.789,20	0,05	9,04	19.359,20	39,66	1,01
p28r-v17	17.807,60	324,72	15.324,80	0,02	13,94	15.799,00	0,05	11,28	17.607,00	39,73	1,13
p29r-v17	18.473,60	495,97	16.923,80	0,02	8,39	16.992,20	0,05	8,02	17.989,00	37,78	2,62
p30r-v17	16.288,60	276,32	14.867,80	0,02	8,72	14.972,00	0,05	8,08	15.941,60	36,85	2,13
	Média	318,95	Média	0,02	10,61	Média	0,05	9,21	Média	39,08	2,01
p1r-v130			14.725,20	0,25	15,15	14.989,40	0,71	13,63	17.355,40	32,32	0,00
p2r-v130			16.556,40	0,27	12,37	16.755,80	0,83	11,31	18.892,80	37,08	0,00
p3r-v130			14.884,00	0,25	9,25	15.189,80	0,68	7,38	16.400,40	27,33	0,00
p4r-v130			17.043,20	0,23	11,62	17.048,20	0,68	11,60	19.284,60	37,74	0,00
p5r-v130			15.978,60	0,25	12,32	16.273,40	0,64	10,70	18.223,40	26,35	0,00
p6r-v130			18.377,40	0,24	6,39	18.398,40	0,64	6,28	19.631,80	34,91	0,00
p7r-v130			18.511,20	0,25	3,03	18.511,20	0,44	3,03	19.090,40	33,48	0,00
p8r-v130			16.050,00	0,28	11,23	16.450,20	0,68	9,02	18.080,80	38,63	0,00
p9r-v130			13.928,40	0,29	7,88	14.219,20	0,90	5,96	15.119,80	36,33	0,00
p10r-v130			14.957,80	0,24	16,56	16.587,20	0,81	7,47	17.925,40	26,47	0,00
p11r-v130			15.001,00	0,25	9,37	15.081,00	0,72	8,89	16.551,80	25,07	0,00
p12r-v130			20.596,20	0,23	4,58	20.596,20	0,36	4,58	21.584,00	37,52	0,00
p13r-v130			18.061,60	0,24	6,17	18.073,20	0,67	6,11	19.248,80	35,87	0,00
p14r-v130			17.597,00	0,30	8,68	17.630,40	0,78	8,51	19.269,60	36,85	0,00
p15r-v130			18.393,20	0,25	9,61	19.143,00	0,96	5,93	20.349,80	34,35	0,00
p16r-v130			15.964,00	0,25	6,28	16.102,20	0,62	5,47	17.033,20	33,32	0,00
p17r-v130			15.090,20	0,26	9,91	15.703,40	0,75	6,25	16.750,20	27,07	0,00
p18r-v130			16.610,40	0,24	5,91	16.610,40	0,93	5,91	17.654,20	35,44	0,00
p19r-v130			16.315,00	0,25	8,77	16.555,20	0,90	7,42	17.882,40	23,82	0,00
p20r-v130			17.764,60	0,26	9,24	18.243,00	0,70	6,80	19.574,00	32,37	0,00

Tabela 1. Continuação...

Problema	CPLEX		Heurística gulosa			Busca local			GRASP		
	Sol. Ótima	CPUt (s)	FO	CPUt (s)	Gap (%)	FO	CPUt (s)	Gap (%)	FO	CPUt (s)	Gap (%)
p21r-v130			15.634,00	0,22	5,56	15.634,00	0,48	5,56	16.554,40	34,09	0,00
p22r-v130			17.437,00	0,25	5,39	17.444,80	0,50	5,34	18.429,60	37,02	0,00
p23r-v130			19.439,60	0,31	6,49	19.496,20	0,84	6,22	20.788,60	39,48	0,00
p24r-v130			15.927,40	0,25	10,43	17.106,20	0,91	3,80	17.781,40	25,23	0,00
p25r-v130			15.027,80	0,28	10,37	15.350,40	0,68	8,45	16.766,40	37,55	0,00
p26r-v130			18.321,60	0,25	8,50	18.352,40	0,72	8,35	20.024,20	30,16	0,00
p27r-v130			18.161,80	0,25	5,47	18.300,60	0,66	4,75	19.213,00	31,75	0,00
p28r-v130			15.893,80	0,26	11,10	16.086,00	0,66	10,03	17.879,00	34,40	0,00
p29r-v130			17.843,20	0,26	4,67	17.843,20	0,78	4,67	18.718,20	30,33	0,00
p30r-v130			15.075,20	0,26	7,10	15.104,60	0,82	6,92	16.227,00	28,19	0,00
			Média	0,25	8,65	Média	0,72	7,21	Média	32,68	0,00

5 Conclusões e perspectivas

Neste artigo foi abordado o PADV no contexto do transporte rodoviário de cargas completas entre terminais. O problema foi modelado por programação linear inteira considerando casos em que os veículos pertencem a muitos grupos e, no caso especial em que cada veículo é tratado individualmente e se necessita acompanhar o movimento cheio e vazio, veículo a veículo. Essa situação é comum em ambientes práticos onde os operadores do setor de tráfego de uma empresa de transportes necessitam definir qual veículo realizará cada rota. Dada a dificuldade de se resolver problemas de tamanho e complexidades similares ao encontrado na prática das transportadoras brasileiras utilizando-se pacotes comerciais de otimização como o CPLEX, optou-se pelo desenvolvimento de heurísticas explorando conhecimento específico do problema e capazes de produzir relativamente boas soluções em tempos computacionais aceitáveis na prática: (i) heurística gulosa para a geração de soluções iniciais viáveis; (ii) busca local para melhoria da solução; e (iii) GRASP com VNS e *path-relinking*.

Para avaliar o desempenho das heurísticas, foram gerados diversos problemas de tamanho similar ao encontrado na prática, considerando diferentes quantidades de grupos de veículos. O GRASP obteve um bom desempenho relativamente ao método *branch-and-cut* do CPLEX, sendo que para o conjunto de problemas contendo 17 grupos, o desvio médio da solução ótima foi de 2,01%, com um tempo médio de processamento de 39,08 segundos. Para o conjunto de problemas contendo 130 grupos, ou melhor, no limite em que cada veículo pertence a um grupo, sendo tratado individualmente placa a placa, o tempo médio de processamento foi de 32,68 segundos. Nesse caso não foi possível a comparação com o AMPL/CPLEX, pois a linguagem de modelagem com o *solver* não conseguiu compilar e resolver nenhum

problema com mais de 17 grupos, em função dos altos requisitos de memória computacional.

Pode-se também concluir que o GRASP proposto é adequado para a obtenção de ótimos locais para o PADV, servindo como alternativa para a obtenção de soluções adequadas e aplicáveis ao caso prático de empresas transportadoras de cargas parceladas.

Como perspectiva para trabalhos futuros pretende-se desenvolver novas heurísticas e meta-heurísticas para a solução do PADV como, por exemplo, o *simulated annealing*, a colônia de formigas (*Ant Colony Optimization*), dentre outras. Pretende-se ainda reportar em breve os resultados de um processo de validação prática do modelo e dos métodos de solução propostos em ambientes reais de operação em uma empresa de transporte rodoviário de cargas parceladas no Brasil.

Agradecimentos

Os autores agradecem aos revisores anônimos pelos úteis comentários e sugestões, e à Patrus Transportes pela importante colaboração e apoio a esta pesquisa. Este estudo também contou com apoio parcial do CNPq e da Fapesp.

Referências

- BIRGE, J. R.; LOUVEAUX, F. **Introduction to Stochastic Programming**. Springer, 1997. (Springer Series in Operations Research and Financial Engineering).
- BRACKLOW, J. W. et al. Interactive Optimization Improves Service and Performance for Yellow Freight System. **Interfaces**, v. 22, n. 1, p. 147-172, 1992. <http://dx.doi.org/10.1287/inte.22.1.147>
- DEJAX, P. J.; CRAINIC, T. G. A review of empty flows and fleet management models in freight transportation. **Transportation Science**, v. 21, n. 4, p. 227-247, 1987. <http://dx.doi.org/10.1287/trsc.21.4.227>
- FEO, T.; RESENDE, M. A probabilistic heuristic for a computationally difficult set covering problem.

- Operations Research Letters**, v. 8, p. 67-71, 1989. [http://dx.doi.org/10.1016/0167-6377\(89\)90002-3](http://dx.doi.org/10.1016/0167-6377(89)90002-3)
- FEO, T.; RESENDE, M. Greedy randomized adaptive search procedures. **Journal of Global Optimization**, v. 6, p. 109-133, 1995. <http://dx.doi.org/10.1007/BF01096763>
- FESTA, P.; RESENDE, M. G. C. GRASP: An annotated bibliography. In: RIBEIRO, C. C.; HANSEN, P. **Essays and Surveys in Metaheuristics**. Kluwer Academic Publishers, 2002. p. 325-367. (Operations Research/Computer Science Interfaces Series Volume 15). http://dx.doi.org/10.1007/978-1-4615-1507-4_15
- FRANTZESKAKIS, L. F.; POWELL, W. B. A successive linear approximation procedure for stochastic, dynamic vehicle allocation problems. **Transportation Science**, v. 24, n. 1, p. 40-57, 1990. <http://dx.doi.org/10.1287/trsc.24.1.40>
- GHIANI, G.; LAPORTE, G.; MUSMANNO, R. **Introduction to logistics systems planning and control**. John Wiley & Sons Ltd., 2003. (Wiley-Interscience series in systems and optimization).
- GLOVER, F. Tabu search and adaptative memory programming - Advances, applications and challenges. In: BARR, R. S.; HELGASON, R. V.; KENNINGTON, J. L. (Eds.). **Interfaces in Computer Science and Operations Research**. Kluwer, 1996. p. 1-75. PMID:8842811.
- HALL, R. W. Stochastic freight flow patterns: implications for fleet optimization. **Transportation Research Part A: Policy and Practice**, v. 33, n. 6, p. 449-465, 1999. [http://dx.doi.org/10.1016/S0965-8564\(98\)00063-9](http://dx.doi.org/10.1016/S0965-8564(98)00063-9)
- HALL, R. W.; ZHONG, H. Decentralized inventory control policies for equipment management in a many-to-many network. **Transportation Research Part A: Policy and Practice**, v. 36, n. 10, p. 849-865, 2002. [http://dx.doi.org/10.1016/S0965-8564\(01\)00025-8](http://dx.doi.org/10.1016/S0965-8564(01)00025-8)
- HANSEN, P.; MLADENOVIC, N.; PÉREZ, J. A. M. Variable neighborhood search: methods and applications. **Annals of Operations Research**, v. 175, n. 1, p. 367-407, 2010. <http://dx.doi.org/10.1007/s10479-009-0657-6>
- INTERNATIONAL BUSINESS MACHINES - IBM. **ILOG CPLEX User's Manual**. 2008. v. 11.1.1.
- MLADENOVIC, N.; HANSEN, P. Variable neighborhood search. **Computers & Operations Research**, v. 24, n. 11, p. 1097-1100, 1997. [http://dx.doi.org/10.1016/S0305-0548\(97\)00031-2](http://dx.doi.org/10.1016/S0305-0548(97)00031-2)
- POWELL, W. B. A stochastic model of the dynamic vehicle allocation problem. **Transportation Science**, v. 20, p. 117-129, 1986. <http://dx.doi.org/10.1287/trsc.20.2.117>
- POWELL, W. B. An operacional planning model for the dynamic vehicle allocation problem with uncertain demands. **Transportation Research Part B**, v. 21, n. 3, p. 217-232, 1987. [http://dx.doi.org/10.1016/0191-2615\(87\)90005-1](http://dx.doi.org/10.1016/0191-2615(87)90005-1)
- POWELL, W. B. **A comparative review of alternative algorithms for the dynamic vehicle allocation problem**. Vehicle Routing: The Methods and Studies, Elsevier Science Publisher B. V., 1988.
- POWELL, W. B. A stochastic formulation of the dynamic assignment problem, with an application to truckload motor carrier. **Transportation Science**, v. 30, n. 3, p. 195-219, 1996. <http://dx.doi.org/10.1287/trsc.30.3.195>
- POWELL, W. B.; CARVALHO, T. A. Dynamic control of logistics queueing networks for large-scale fleet management. **Transportation Science**, v. 32, n. 2, p. 90-109, 1998a. <http://dx.doi.org/10.1287/trsc.32.2.90>
- POWELL, W. B.; CARVALHO, T. A. Real-time optimization of containers and flatcars for intermodal operations. **Transportation Science**, v. 32, n. 2, p. 110-126, 1998b. <http://dx.doi.org/10.1287/trsc.32.2.110>
- POWELL, W. B.; JAILLET, P.; ODoni, A. R. Stochastic and dynamic networks and routing. In: BALL, M. O. et al. (Ed.). **Handbooks in Operations Research and Management Science**. Amsterdam: Elsevier Science, 1995. v. 8, p. 141-295.
- POWELL, W. B.; SHEFFI, Y.; THIRIEZ, S. The dynamic vehicle allocation problem with uncertain demands. In: INTERNATIONAL SYMPOSIUM ON TRANSPORTATION AND TRAFFIC THEORY, 9., 1984. **Proceedings...** p. 357-374.
- RESENDE, M. G. C.; RIBEIRO, C. C. GRASP with path-relinking: recent advances and applications. In: IBARAKI, T.; NONOBE, K.; YAGIURA, M. (Eds.). **Metaheuristics: progress as real problem solvers**. Springer US, 2005. v. 32, p. 29-63. http://dx.doi.org/10.1007/0-387-25383-1_2
- VASCO, R. A. **Otimização na alocação dinâmica de veículos no transporte rodoviário de cargas completas entre terminais**. 2012. Tese (Doutorado)-Universidade Federal de São Carlos, São Carlos, 2012.