

São Carlos, 27 e 28 de março de 2014

XVI ONPCE

Oficina Nacional de Problemas de Corte e Empacotamento,
Planejamento e Programação da Produção e Correlatos

Programação e Livro de Resumos

Programação

Programação

Dia 27 de março, quinta-feira	
07:50 – 8:10	Entrega do material
08:00 – 08:30	Abertura
08:30 – 09:30	Palestra: <i>Heróis ou Estúpidos?</i> Prof. Geraldo Robson Mateus, UFMG.
09:30 – 09:50	Café
09:50– 10:00	Abertura - Sessão 1 (Coordenador: Prof. Silvio Araújo)
10:00 – 10:20	Resolução do problema conjunto de carregamento de contêineres em navios 3D e programação de guindastes portuários via representação por regras e algoritmo genético. <i>Autores:</i> Anibal Tavares de Azevedo, Luiz Leduino de Salles Neto, Antônio Augusto Chaves, Antônio Carlos Moretti.
10:20–10:40	Programação de rotas de navios de petróleo cru da Petrobras no litoral brasileiro. <i>Autores:</i> Bruno Jensen Virginio da Silva, Amélia Stanzani, Denise Sato Yamashita, Maria Gabriela Furtado, Vinicius Rodrigues
10:40–11:00	O Problema de Alocação de Berços: Aspectos Teóricos e Computacionais. <i>Autores:</i> Flávia Barbosa, Antônio Carlos Moretti, Luiz Leduino de Salles Neto.
11:00–11:20	Algoritmo de otimização por colônia de formigas na roteirização de veículos. <i>Autores:</i> Ricardo Pinto Ferreira, Arthur Ferreira, Renato José Sassi.
11:20–11:40	Mapa auto-organizável de Kohonen na programação de veículos. <i>Autores:</i> Ricardo Pinto Ferreira, Andréa Martiniano da Silva, Renato José Sassi.
11:40–12:00	Análise estatística da decisão para tratar o problema de roteamento e cobertura com máximo retorno às áreas sensíveis. <i>Autores:</i> Maria José Pinto Lamosa, Mônica Maria De Marchi, Carlos Alberto Lima Araújo Júnior, Rodrigo Arnaldo Scarpel.
12:00–12:20	Planejamento da rede de logística reversa pós-consumo de equipamentos de informática. <i>Autores:</i> Maritha Silva, Eli Angela Vitor Toso.

12:20–14:00	Almoço: Armazém de Maria, Rodovia Washington Luiz, km 230
14:00–15:00	Palestra: <i>Métodos Eficientes para Problemas de Roteamento de Veículos.</i> Prof. Luis Satoru Ochi, UFF.
15:00–15:40	Pôsteres e reuniões subgrupos
15:40–16:00	Café
16:00–16:10	Abertura - Sessão 2 (Coordenadora: Profa. Débora Ronconi)
16:10–16:30	Análise comparativa de heurísticas para minimização de adiantamentos e atrasos em <i>flow shop</i> com tempos <i>desetup</i> . <i>Autores:</i> Helio Yochihiro Fuchigami, John Lennon Damas Davi.
16:30–16:50	Um modelo de programação inteira mista para a programação da produção em <i>flowshop</i> híbrido com buffers limitados. <i>Autores:</i> Pedro Luis Miranda Lugo, Rodolfo Florence Teixeira Jr.
16:50–17:10	Proposta de framework com o foco na análise e representação de dados, para pesquisas de otimização baseados em heurísticas aplicadas a problemas de sistemas de produção. <i>Autores:</i> Ronaldo Castro de Oliveira, Roberto Fernandes Tavares Neto, Hercules Tadeu Asato Dantas.
17:10–17:30	Aplicação do algoritmo genético de chaves aleatórias viciadas ao problema de escalonamento de técnicos de campo. <i>Autores:</i> Ricardo de Brito Damm, Débora Pretti Ronconi.
17:30–17:50	Um estudo de problemas de otimização combinatória em hospitais. <i>Autores:</i> Tiago Tiburcio, Socorro Rangel.
17:50	Reunião pesquisadores do projeto temático.

Dia 28 de março, sexta-feira	
08:00–08:10	Abertura - Sessão 3 (Coordenadora: Profa. Franklina Toledo)
08:10–08:30	Um algoritmo memético no problema do Corte Unidimensional Inteiro. <i>Autores:</i> Angelo Aliano Filho, Antônio Carlos Moretti.
08:30–08:50	Planejamento na divisão de talhões para o plantio da cana-de-açúcar. <i>Autores:</i> Adriana Cristina Cherri, Andréa Vianna, Helenice Florentino, Rômulo Ramos.
08:50–09:10	Uma heurística para um problema de carregamento de caixas em um contêiner compartimentado. <i>Autores:</i> Rodolfo Ranck Junior, Horacio Hideki Yanasse, Reinaldo Morabito.
09:10–10:30	Palestra: <i>Multidimensional dual-feasible functions and fast lower bounds for the vector packing problem.</i> Prof. José Valério de Carvalho, Universidade do Minho.
10:30–11:20	Reuniões sub-grupos e café.
11:20–11:40	Problema de Localização-Roteamento-Empacotamento com Demanda Incerta. <i>Autores:</i> Thiago Alves de Queiroz, José Fernando Oliveira, Maria Aantónia Carravilla, Flávio Keidi Miyazawa.
11:40–12:00	Um modelo compactado para o problema de corte de peças irregulares. <i>Autores:</i> Marcos Okamura Rodrigues, Aline A. S. Leão, Franklina M. B. Toledo.
12:00–12:20	Um método heurístico para o problema de corte de peças irregulares. <i>Autores:</i> Luiz Henrique Cherri, Franklina Maria Bragion de Toledo; Maria Antónia Carravilla.
12:20–14:00	Almoço: Armazém de Maria. Rodovia Washington Luiz, km 230
14:00–15:00	Mesa redonda: Otimização: passado, presente e futuro. Mediador: Prof. Reinaldo Morabito, UFSCar. Participantes: Prof. Dr. Geraldo Robson Mateus, UFMG; Prof. Luis Satoru Ochi, UFF. Prof. José Valério de Carvalho, Universidade do Minho.

15:00–15:10	Abertura - Sessão 4 (Coordenador: Prof. Douglas Alem)
15:10–15:30	Restrições do tipo multi-commodity flow aplicadas ao problema integrado de dimensionamento e sequenciamento da produção. <i>Autores:</i> Michelli Maldonado, Socorro Rangel.
15:30–15:50	Planejamento e programação da produção na indústria de polpa moldada. <i>Autores:</i> Karim Pérez Martínez, Reinaldo Morabito, Eli Vitor Toso
15:50–16:10	Uma abordagem de solução híbrida baseada em algoritmo genético para o problema de planejamento e sequenciamento da produção em indústria de papel e celulose. <i>Autores:</i> Marcos Mansano Furlan, Bernardo Almada-Lobo, Maristela Oliveira Santos, Reinaldo Morabito.
16:10–16:30	Café
16:30–16:50	Uma formulação não-linear para a restrição de ciclos da serra no problema integrado de dimensionamento de lotes e corte de estoque. <i>Autores:</i> Gislaíne Mara Melega, Silvio Alexandre de Araujo e Socorro Rangel.
16:50–17:10	Problema integrado de produção e distribuição: uma abordagem heurística. <i>Autores:</i> Márcio Antônio Ferreira Belo Filho, Pedro Amorim, Bernardo Almada-Lobo.
17:10–17:30	Modelos híbridos para o Problema da Mochila Bidimensional com conflitos. <i>Autores:</i> Pedro Hokama, Thiago A. de Queiroz, Flavio Keidi Miyazawa.
17:30–17:50	Otimização estocástica na programação de bombas de redes de abastecimento urbano. <i>Autores:</i> Jonathan Justen de La Vega Martínez, Douglas Alem.
17:50	Encerramento

Palestras

Palestra 1: Heróis ou Estúpidos?

Prof. Geraldo Robson Mateus

Departamento de Ciência da Computação
UFMG, Instituto de Ciências Exatas

Resumo: Um grande desafio para os profissionais de Pesquisa Operacional é entender os problemas que se apresentam, solucionar e implantar no mundo real. Se apresentam desde contextos amplos e complexos até problemas específicos e de fácil solução. Como atuar no estado da arte na pesquisa e retornando resultados concretos para sociedade? É um ato de heroísmo ou de estupidez? Nesta linha, daremos uma visão iniciando pelo contexto amplo de Cadeia de Suprimentos a problemas específicos de roteamento de veículos. Muitos são os problemas de roteamento de tráfego e veículos, explorando diversos tipos de serviços, tecnologias e aplicações. Estão presentes nas redes de transporte, telecomunicações e energia. Para estes problemas são propostas diversas formulações, bem como algoritmos. Trata-se, portanto, de uma linha de pesquisa, e aplicações, específicas, que por si só são amplas e complexas, com grandes desafios tanto em modelagem com em desenvolvimento de novas técnicas. E o mundo real?

Palestra 2: Métodos Eficientes para Problemas de Roteamento de Veículos

Prof. Luis Satoru Ochi

Departamento de Ciência da Computação
UFF, Instituto de Computação

Resumo: Nesta palestra serão apresentadas estratégias que tem se mostrado serem as mais eficientes na resolução de diferentes variantes de Problemas de Roteamento de Veículos (The Vehicle Routing Problems - VRP). Neste contexto, mostramos que métodos híbridos tanto sequencias ou paralelos: conjugando diferentes metaheurísticas, ou reunindo conceitos de metaheurísticas com programação matemática (matheuristics) tem se mostrado muito promissores em VRP. Como ilustração, descreveremos um método híbrido de nossa autoria para uma variante do VRP cujos resultados se mostraram muito competitivos quando confrontados com os resultados da literatura. Maiores informações sobre este tema pode ser encontrado em: <http://www2.ic.uff.br/~satoru/>

Palestra 3: Multidimensional dual-feasible functions and fast lower bounds for the vector packing problem

Prof. José Valério de Carvalho

Departamento de Produção e Sistemas
Universidade do Minho, Campus de Gualtar

Artigo: <http://dx.doi.org/10.1016/j.ejor.2013.08.011>

Resumo: We address the 2-dimensional vector packing problem where an optimal layout for a set of items with two independent dimensions has to be found within the boundaries of a rectangle. Many practical applications in areas such as the telecommunications, transportation and production planning lead to this combinatorial problem. Here, we focus on the computation of fast lower bounds using original approaches based on the concept of dual-feasible functions (DFF). Until now, all the dual-feasible functions proposed in the literature were 1-dimensional functions. We extend the principles of dual-feasible functions to the m -dimensional case. Several difficulties arise when one wants to generalize the results obtained for 1-dimensional DFF. They are induced by the fact that the comparison between item sizes has to be done componentwise, and therefore, monotonicity and superadditivity take a different sense. We propose several new families of functions, and analyze the properties which are relevant to evaluate the quality of the lower bounds that these functions can generate. We describe different lower bounding procedures based on these functions, and we report on the complexity of each one. Our approaches were tested on benchmark instances of the literature. Our computational experiments show that high quality bounds can be obtained for this problem using the vector packing dual-feasible functions proposed in this paper. Furthermore, we tested our lower bounding procedures within a branch-and-bound algorithm to evaluate their potential impact on the convergence of this type of algorithms. Our computational results show that these functions can approximate very efficiently the best known lower bounds for this problem and improve significantly the convergence of branch-and-bound algorithms. The utility of multidimensional dual-feasible functions goes far beyond the computation of lower bounds for vector packing problems. Indeed, these functions can be used to obtain valid inequalities for general mixed integer programs, i.e., any integer programming problem with multidimensional knapsack constraints.

Trabalhos Completos

Sessão 1

Resolução do problema conjunto de carregamento de contêineres em navios 3D e programação de guindastes portuários via representação por regras e algoritmo genético.

Autores: Anibal Tavares de Azevedo, Luiz Leduino de Salles Neto, Antônio Augusto Chaves, Antônio Carlos Moretti.

Programação de rotas de navios de petróleo cru da Petrobras no litoral brasileiro.

Autores: Bruno Jensen Virginio da Silva, Amélia Stanzani, Denise Sato Yamashita, Maria Gabriela Furtado, Vinicius Rodrigues

O Problema de Alocação de Berços: Aspectos Teóricos e Computacionais.

Autores: Flávia Barbosa, Antônio Carlos Moretti, Luiz Leduino de Salles Neto.

Algoritmo de otimização por colônia de formigas na roteirização de veículos.

Autores: Ricardo Pinto Ferreira, Arthur Ferreira, Renato José Sassi.

Mapa auto-organizável de Kohonen na programação de veículos.

Autores: Ricardo Pinto Ferreira, Andréa Martiniano da Silva, Renato José Sassi.

Análise estatística da decisão para tratar o problema de roteamento e cobertura com máximo retorno às áreas sensíveis.

Autores: Maria José Pinto Lamosa, Mônica Maria De Marchi, Carlos Alberto Lima Araújo Júnior, Rodrigo Arnaldo Scarpel.

Planejamento da rede de logística reversa pós-consumo de equipamentos de informática.

Autores: Maritha Silva, Eli Angela Vitor Toso.

RESOLUÇÃO DO PROBLEMA CONJUNTO DE CARREGAMENTO DE CONTÊINERES EM NAVIOS 3D E PROGRAMAÇÃO DE GUINDASTES PORTUÁRIOS VIA REPRESENTAÇÃO POR REGRAS E ALGORITMO GENÉTICO

Anibal Tavares de Azevedo

UNICAMP

anibal.azevedo@fca.unicamp.br

Luiz Leduino de Salles Neto

UNIFESP

luiz.leduino@unifesp.br

Antônio Augusto Chaves

UNIFESP

antonio.chaves@unifesp.br

Antônio Carlos Moretti

UNIFESP

antonio.moretti@fca.unicamp.br

RESUMO

A eficiência de um terminal portuário depende um planejamento apropriado da movimentação de contêineres, também chamado de plano de estiva. A partir do plano de estiva é possível estimar o tempo, e assim também o custo, para realizar os processos de carregamento e descarregamento de contêineres de um navio. Para tanto, este artigo propõe uma abordagem para resolver o problema de planejamento de estiva 3D integrado ao problema de programação ou *scheduling* de guindastes portuários. Ambos os problemas são NP-Hard e demanda grande número de variáveis binárias para representar uma solução em uma formulação inteira. A alternativa empregada neste trabalho é a aplicação sucessiva da representação por regras para o problema de estiva e o programação de guindastes portuários com um algoritmo genético. Soluções para problemas de grande porte que demandam aproximadamente 40 milhões de variáveis binárias, só para o plano de estiva, são obtidas em menos de 15 minutos.

PALAVRAS-CHAVE: Plano de estiva, Programação de Guindastes Portuários, Representação por Regras, Algoritmo Genético.

1. Introdução

De acordo com (Guan et al. 2013; Steenken et al. 2004), as operações em um terminal portuário de contêineres podem ser divididas em cinco problemas principais:

- (1) Alocação de berços: a saída deste problema é a programação de qual navio será atendido em qual berço de modo que uma distância mínima de segurança seja respeitada entre dois navios, e dois navios não podem compartilhar o mesmo berço no mesmo período de tempo;
- (2) Plano de estiva: este problema consiste em determinar como organizar os

contêineres em um navio de modo a minimizar o número de movimentos de carregamento e descarregamento;

- (3) Designação de Guindastes no Cais: o tempo que o navio porta-contêiner leva para carregar ou descarregar depende da programação de guindastes portuários alocados para cada seção do navio. Os guindastes não podem ultrassapar uns aos outros por terem seus movimentos limitados a um trilho comum. Além disso, uma distância de segurança entre eles também deve ser observada;
- (4) Transporte do Cais: é necessário determinar que máquinas serão empregadas e quais serão suas trajetórias ligando o navio ao pátio do porto e vice-versa;
- (5) Transporte do Pátio: A organização de pilhas no pátio do porto depende de como realizar operações eficientes para empilhar e desempilhar os contêineres que são trazidos ou levados por caminhões e trens.

Este trabalho irá abordar a resolução conjunta dos problemas (2) e (3). Para a resolução do problema (2) é necessário definir um plano de estiva que é fortemente relacionado com a estrutura especial do navio porta-contêiner como dado na Figura 1.

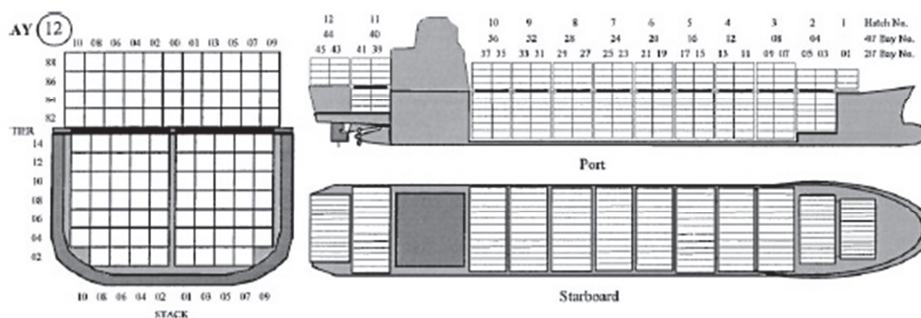


Fig. 1: Estrutura de um navio porta-contêiner (Source: Wilson and Roach 2000).

A partir do plano de estiva, é possível necessária programação de guindastes portuários tal como dado na Figura 2.

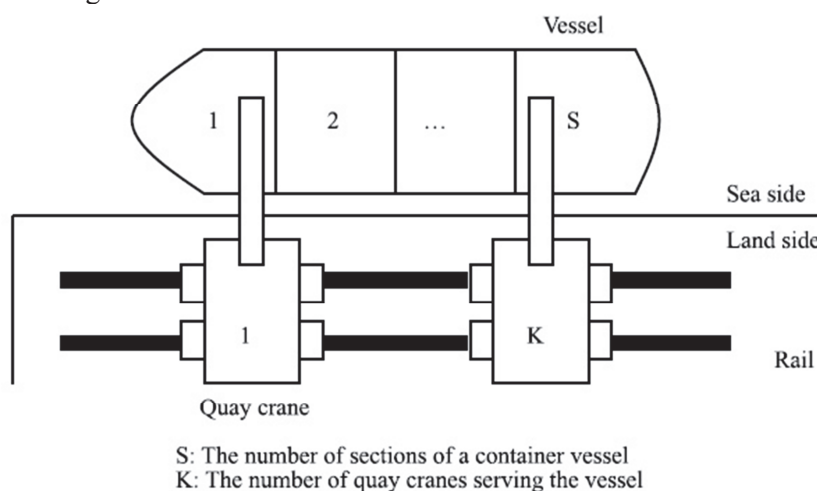


Fig. 2: Problema de programação de guindastes portuários (Source: Lee et al. 2008).

2. Descrição do problema

O mesmo modelo adotado para o plano de estiva em (Azevedo et al. 2012) será empregado neste artigo de modo que apenas a descrição do que as restrições representam será apresentada aqui. As seguintes considerações foram adotadas:

(a) O navio porta-contêiner possui um formato rectangular e pode ser representado por uma matriz com linhas ($r = 1, 2, \dots, R$), colunas ($c = 1, 2, \dots, C$) e baias ($d = 1, 2, \dots, D$) totalizando uma capacidade máxima de $R \times C \times D$ contêineres;

(b) Todos os contêineres possui o mesmo peso e tamanho;

(c) Todos os navios começam a ser carregados no porto 1 e sem nenhuma carga inicial;

(d) O navio visita os portos 2, 3, ..., N tal que o navio estará vazio no ultimo porto, pois é considerado que o mesmo realiza uma rota circular onde o porto N, na verdade, representa o porto 1;

(e) Em cada porto $i = 1, 2, \dots, N$, o navio deve ser carregado com contêineres cujo destino são portos $i+1, \dots, N$.

(f) O navio porta-contêineres pode sempre carregar todos os contêineres disponíveis em cada porto e isto não irá exceder sua capacidade máxima.

Para a programação dos guindastes portuários o modelo descrito em (Lee et al. 2008) foi adotado e as seguintes considerações foram consideradas sob a forma de restrições e função objetivo:

(a) Ao invés de se considerar o navio dividido em seções como em (Lee et al. 2008), foi considerada cada baia do navio. Dessa forma, é possível transformar a informação de movimentação de contêineres fornecida pelo plano de estiva sob a forma do total de esforço necessário por baia;

(b) Foi assumido que o esforço total equivale ao número de contêineres que devem ser movidos e que cada container demanda uma unidade de tempo para ser carregado ou descarregado;

(c) Cada baia no navio só pode ter um único guindastes em um dado período de tempo;

(d) Uma vez que um guindaste começou o serviço (carregamento ou descarregamento) em uma baia, o mesmo só irá parar quando não houver mais serviço para realizar;

(e) Quando um guindaste troca sua operação de uma baia para outra, ele leva um tempo constante e igual a três unidades de tempo para mover um contêiner;

(f) Uma distância mínima entre os guindastes deve ser observada. Isto significa que, às vezes, um guindaste não pode ser movido e não pode começar o seu serviço até que um segundo guindaste termine o seu trabalho e mova-se para uma nova posição que garanta a distância mínima entre os guindastes;

(g) Um guindaste não deve ultrapassar outro, pois eles compartilham o mesmo trilho;

(h) Todos os guindastes possuem a mesma e constant taxa de serviço de movimentação de uma unidade de trabalho por unidade de tempo.

Os dois modelos descritos anteriormente podem ser acoplados ao se observar que o número de movimentos por baia pode ser transformado no esforço total por baia como dado na Fig. 3.

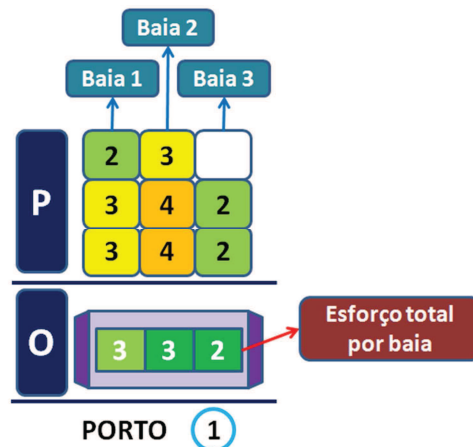


Fig. 3: Acoplando o plano de estiva (P) com a operação de guindastes (O).

3. Métodos e abordagens

Ao invés de empregar um modelo com variáveis binárias, será empregada a resolução dos dois problemas através da representação por regras. As regras para carregamento e descarregamento de navios descritas em (Azevedo et al. 2012) foram combinadas com quatro regras de operações de guindastes. A Fig. 4 ilustra o passo-a-passo do funcionamento de uma das regras de operação de dois guindastes portuários.

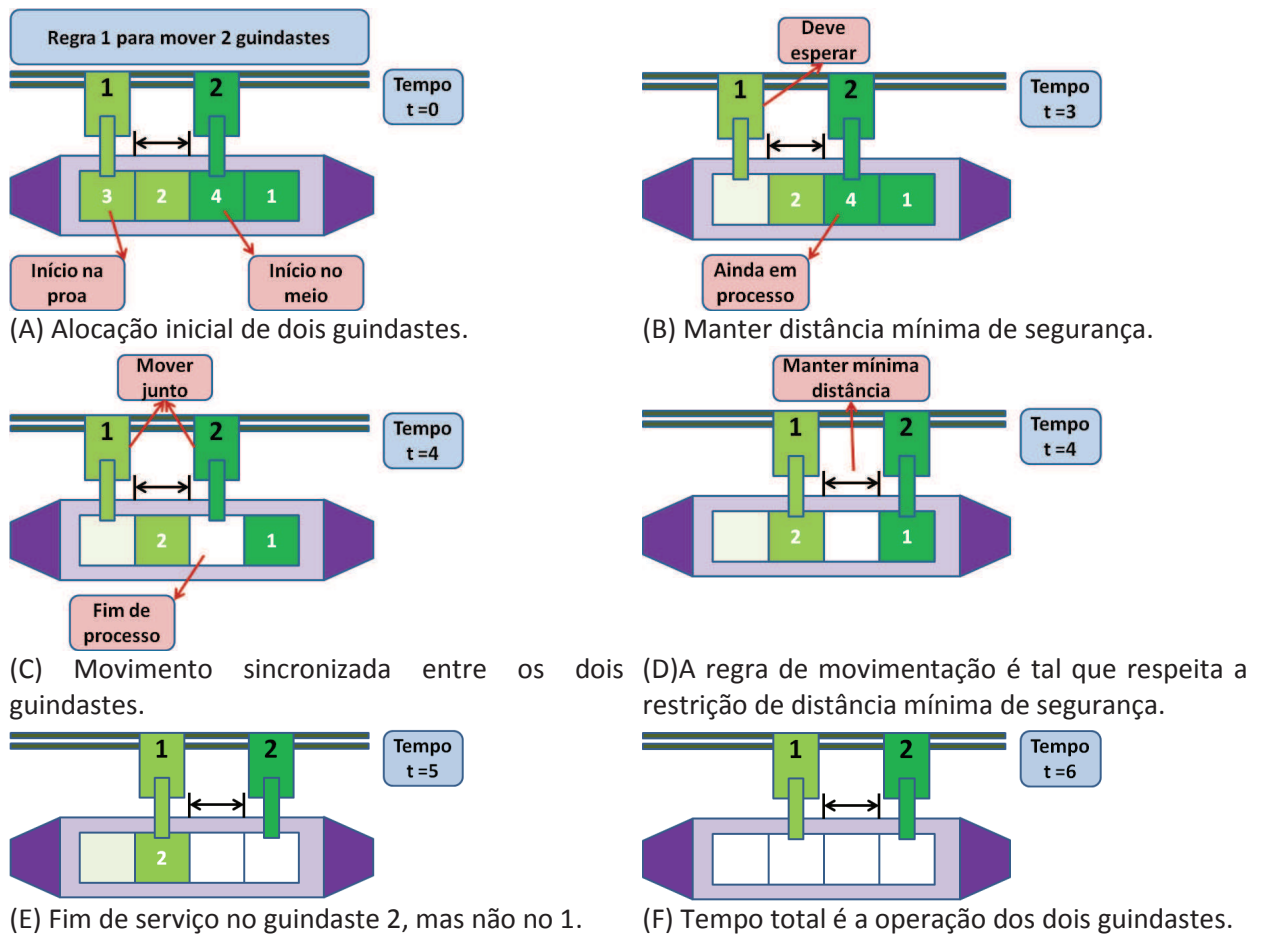


Fig. 4: Descrição detalhada do funcionamento da regra 1 de operação de dois guindastes portuários.

4. Resultados

Foram obtidos resultados para instância com até cerca de 44 milhões de restrições e 42 milhões de variáveis binárias utilizando um programa criado em Matlab 7.0, e uma máquina com um processador 1.66 GHz Core Duo Intel Processor, memória RAM memory de 2 GB e sistema operacional Windows Vista Operational System com Service Pack 2. Duas funções objetivos foram testadas: uma que busca minimizar o tempo total de operação (Fig. 5(A) e (B)) e outra que busca minimizar a instabilidade do arranjo dos contêineres no navio (Fig. 5(C) e (D)). Parte das melhores soluções obtidas são mostradas na Fig. 5.

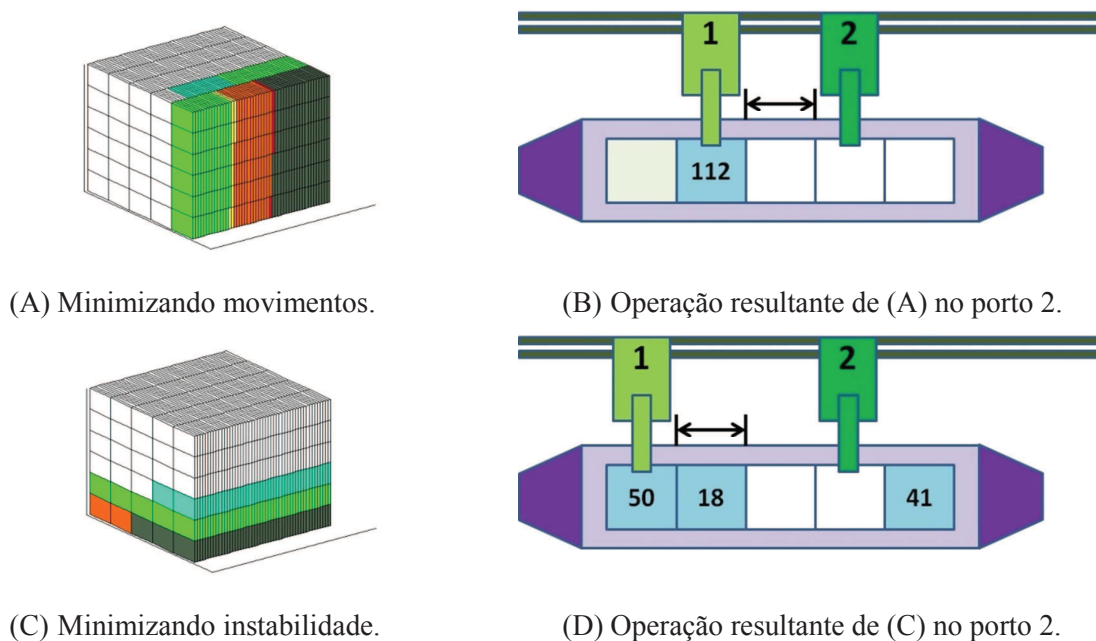


Fig. 5: Comparação de duas soluções em termos plano de estiva e operação de guindastes.

5. Conclusões e perspectivas futuras

Neste trabalho a representação por regras foi estendida para resolver o problema conjunto de plano de estiva e programação de guindastes em um terminal portuário de contêineres. A metodologia proposta conseguiu obter soluções para problemas com cerca de 44 milhões de restrições e 42 milhões de restrições em menos de 15 minutos. Uma perspectiva futura é ampliar a utilização da representação por regras de modo a incluir outras operações portuárias tais como a movimentação de contêineres no pátio do porto.

Referências

- Azevedo, A., Ribeiro, C., Sena, G., Chaves, A., Salles, L., Moretti, A. (2012). Solving the 3D Container Ship Loading Planning Problem by Representation by Rules and Beam Search, in ICORES'12, pp. 132-141.
- Guan, Y., Yang, K-H., Zhou, Z. (2013). The Crane Scheduling: models and solution approaches", Annals of Operations Research 203, p. 119-139.
- Lee, D., Wang, D., Miao, L. (2008). Quay crane scheduling with non-interference constraints in port container terminals, Transportation Research E 44, pp. 124-135.
- Steenken, D., Voss, S., Stahlbock, R. (2004). Container terminal operation and operations research - a classification and literature review, OR Spectrum 26 (1), pp. 3-49.
- Wilson, I., Roach, P. (2000). Container stowage planning: a methodology for generating computerised solutions, Journal of the Operational Research Society 51, pp. 1248-1255.

PROGRAMAÇÃO DE ROTAS DE NAVIOS DE PETRÓLEO CRU DA PETROBRAS NO LITORAL BRASILEIRO

Amélia Stanzani

Universidade Federal de São Carlos
mel.stanzani@hotmail.com

Bruno Jensen Virginio da Silva

Universidade Federal de São Carlos
brunojen@gmail.com

Denise Sato Yamashita

Universidade Federal de São Carlos
dsyamashita@gmail.com

Maria Gabriela Furtado

Universidade Federal de São Carlos
gabisfurtado@gmail.com

Vinícius P. Rodrigues

Universidade Federal de São Carlos
viniciuspr88@gmail.com

RESUMO

Este estudo aborda o problema de roteamento de navios petroleiros da Petrobras que transportam petróleo entre as plataformas de produção no oceano e terminais de entrega no litoral brasileiro. O petróleo deve ser escoado ao menor custo logístico possível por uma frota heterogênea, respeitando restrições específicas do problema. Neste trabalho, um modelo matemático inteiro misto de roteamento de veículos com coleta e entrega e janelas de tempo é proposto e implementado em GAMS/CPLEX. Os resultados dos experimentos computacionais mostram que o modelo encontra soluções de qualidade para casos reais pequenos.

PALAVRAS-CHAVE: Petróleo, Roteamento de navios, Petrobras

1. Introdução

Este trabalho é parte de um projeto entre a UFSCar (Universidade Federal de São Carlos e a Petrobras, que tem por objetivo desenvolver métodos e algoritmos para resolver um problema de logística na produção de petróleo no litoral brasileiro.

No Brasil, grande parte do petróleo está no oceano e é produzido em plataformas *offshores*. Navios petroleiros devem coletar esse petróleo e transportá-lo até os terminais localizados na costa brasileira, onde serão redistribuídos ou processados.

As decisões de quais navios devem fazer quais coletas e entregas de petróleo não são triviais e envolvem grandes quantidades de recursos e custos. Com a descoberta de petróleo na região conhecida como pré-sal, essas operações serão mais importantes, complexas e custosas para a Petrobras. Não foi encontrado na literatura trabalhos que se assemelhassem as operações com as mesmas características que a brasileira, nem mesmo em regiões do Mar do Norte na Noruega ou do Golfo do México nos Estados Unidos.

Portanto, é uma operação em que a utilização de métodos e algoritmos customizados

tem grande potencial de melhorar as decisões que são tomadas pelos operadores. Nesse estudo é apresentado o problema real e alguns resultados encontrados para experimentos com dados reais fornecidos pela empresa.

Outros trabalhos em situações de transporte marítimo semelhantes mas com características diferentes podem ser encontradas em Christiansen (1999) e Christiansen et al. (2011) para transporte de amônia e cimento na Europa e Seung (2005) para transporte de derivados de petróleo na Ásia. Christiansen et. al (2004) apresenta uma revisão da literatura bastante completa.

Na seção 2 é feita uma descrição do problema real. Na seção 3 são discutidos os resultados dos experimentos e na seção 4 são apresentadas as perspectivas futuras de desenvolvimento.

2. Descrição do problema

O problema estudado no presente trabalho trata de uma situação real de transporte de petróleo cru extraído do fundo do oceano no Brasil por plataformas e que deve ser transportado para terminais no litoral brasileiro. Nos terminais, esse petróleo pode ser transbordado para outro navio, no caso em que será exportado, ou descarregado no terminal que irá transportá-lo para uma refinaria. O escopo deste trabalho é restrito ao transporte marítimo das plataformas para os terminais. A exportação e o transporte para as refinarias não estão no escopo.

A frota de navios é heterogênea em várias características como capacidade total, velocidade, custo de navegação, calado, comprimento, entre outras. Esta frota é formada por navios próprios da Petrobras e por navios de terceiros que possuem contrato de transporte com a Petrobras.

O horizonte de planejamento é rolante, com duração máxima de 1 mês. Portanto, o planejamento é do tipo operacional. Dentro do horizonte de planejamento, a frota é constante, então, não há decisões de mix de frota.

As plataformas estão espalhadas pelo oceano e são também heterogêneas, produzindo diferentes tipos de petróleo, com taxas de produção, capacidade de estocagem e características de atracação específicas. A quantidade de terminais é menor do que a quantidade de plataformas e todos os terminais estão localizados na costa brasileira. Estes terminais também diferem entre si, demandando conjuntos específicos de produtos, em quantidades diferentes e com várias limitações de tipos de navio que podem receber.

Para atracar em uma plataforma ou terminal, os navios devem respeitar janelas de tempo que são previamente conhecidas em função do limite de capacidade de estoque das plataformas, níveis dos estoques iniciais e taxa média de produção e os planos de produção das refinarias servidas pelos terminais.

As quantidades de petróleo a serem coletadas em cada plataforma e entregues em cada terminal são geradas por um planejamento em nível hierárquico maior e não podem ser modificadas. Portanto, as quantidades e as respectivas origens e destinos dos lotes de produção são previamente conhecidos.

O objetivo do problema consiste em programar os navios e as coletas e entregas de produtos, minimizando os custos de atracação e de deslocamento dos navios.

O problema foi representado por um modelo matemático inteiro misto, baseado na formulação clássica de Ropke e Cordeau (2009) para o problema de coleta e entrega com janelas de tempo. A partir do modelo original de coleta e entrega com janelas de tempo, foram incorporadas outras restrições específicas para o problema da Petrobras como, por exemplo, calado flexível, impossibilidade de atracação e posicionamento dinâmico.

3. Resultados

Foram feitos vários experimentos computacionais para diversos conjuntos de dados reais. Os conjuntos de dados compreendem um horizonte de tempo de 1 mês, e o conjunto completo tem cerca de 120 coletas. Este conjunto completo foi dividido em períodos de planejamento menores, de 16, 22 e 44 pares de coletas e entregas.

Os experimentos foram feitos no GAMS 24.1.1, com CPLEX 12.5.0.1 em uma *workstation* com 2 processadores Intel Xeon E5-2680 2.70GHz e 192 GB de RAM. Nas configurações do CPLEX, foi ativada a opção para utilizar todos os núcleos ao mesmo tempo para processamento.

Os resultados encontrados e os respectivos tempos computacionais são apresentados na tabela 1 a seguir:

Tabela 1: Resultados dos experimentos

Tamanho do problema	Valor da Função Objetivo	GAP (%)	CPU (s)
16	2.158,76	0	00:00:22
22	2.686,57	0	01:28:58
44	-	-	05:00:00

Para os experimentos com 22 pares de coletas e entregas, foi possível encontrar e provar a otimalidade da solução. Nos experimentos com 44 pares coletas e entregas, não foi encontrada nenhuma solução factível dentro do tempo disponibilizado.

No contexto prático, o tempo razoável para aguardar soluções é de até 10 minutos. Pelos resultados dos experimentos, problemas com 16 pares de coletas e entregas atendem esse requisito facilmente, entretanto, o problema com 22 coletas encontrou uma solução de 2.742,8 e GAP de 23,5%.

Essas soluções foram apresentadas aos especialistas da Petrobras e a modelagem foi considerada uma representação satisfatória do contexto prático, capturando todas as restrições e características relevantes do problema. Não foi possível comparar as soluções dos modelos com as praticadas na realidade pois estas não foram fornecidas pela empresa.

4. Conclusões e perspectivas futuras

O modelo proposto para representar o problema real de logística marítima de petróleo, segundo avaliação de especialistas da empresa, consegue representar o problema real de forma razoável. Os resultados dos experimentos mostraram que para horizontes de planejamento que considerem 16 ou 22 pares de coletas e entregas, é possível utilizar o modelo proposto para sugerir decisões, embora nem sempre ótimas.

Para problemas com 44 pares de coletas e entregas ou mais, novas melhorias no modelo devem ser desenvolvidas e testadas. Métodos aproximados como heurísticas e metaheurísticas podem ser utilizados como perspectivas de bons resultados, principalmente para os problemas grandes. Métodos exatos também serão explorados e ao que indica a literatura, pode-se esperar que os resultados sejam satisfatórios para exemplares de tamanho razoável.

Outras perspectivas são explorar formulações diferentes do problema real, inspiradas na literatura de programação da produção e formulações que utilizam outros paradigmas, como tamanho e quantidade de lotes como decisões do modelo.

Referências

- Christiansen, M., Fagerholt, K., Nygreen, B., Ronen, D.** (2004). *Handbook in OR & MS*, Vol. 14, Cap. 4, Elsevier.
- Christiansen, M.** (1999). Decomposition of a Combined Inventory and Time Constrained Ship Routing Problem. *Transportation Science*, 33(1), 3-16.
- Christiansen, M., Fagerholt, K., Flatberg, T., Haugen, O., Kloster, O., Lund, E.** (2011). Maritime inventory routing with multiple products: A case study from the cement industry. *European Journal of Operational Research*, 208, 86-94.
- Ropke, S., Cordeau, J.F.** (2009). Branch and cut and price for the pickup and delivery problem with time windows. *Transportation Science*, 43 (3), 267-286.
- Seung-June, H.** (2005). *Inventory Constrained Maritime Routing and Scheduling for Multi Commodity Liquid Bulk*. Tese de Doutorado. Georgia Institute of Technology, 133.

O Problema de Alocação de Berços: Aspectos Teóricos e Computacionais

Flávia Barbosa

UNICAMP

flaflabarbosa0@gmail.com

Antônio Carlos Moretti

UNICAMP

moretti@ime.unicamp.br

Luiz Leduíno de Salles Neto

UNIFESP

luiz.leduino@unifesp.br

RESUMO

Este trabalho irá estudar as questões acerca do problema de alocação de berços. Nesse contexto, formas mais eficazes para a operação de descarga do carvão feita pela companhia Vale serão propostas com o intuito de reduzir os custos.

PALAVRAS-CHAVE: alocação, corte, esquematização

1. Introdução

O comércio internacional é profundamente dependente do transporte marítimo e por isso, os portos tem sido forçados a investir em infra-estrutura e logística. Nesse contexto, o presente trabalho aborda o Problema de Alocação de Berços: como alocar navios a berços em um dado horizonte de planejamento de modo a minimizar os custos operacionais. No Brasil a companhia Vale é responsável pela exportação da matéria-prima minério de ferro utilizado na fabricação do aço. Assim, será proposto métodos que otimizem suas operações portuárias, mais especificamente para o Terminal de Praia Mole no Porto de Tubarão. Para tanto, dois modelos matemáticos e duas heurísticas foram implementadas. Os modelos, adaptados de casos existentes na literatura, são executados com o GLPK e o CPLEX e os resultados obtidos são comparados para que a melhor opção seja encontrada.

2. Descrição do problema

O Problema de Alocação de Berços (Berth Allocation Problem - BAP) consiste em determinar os horários e as posições de atracação de navios nos berços do cais. Primeiramente vamos trabalhar com uma interpretação de problema de corte.

No caso que será estudado aqui, o eixo horizontal representa o horizonte de planejamento e o eixo vertical representa o tamanho do cais. Analogamente, o navio será representado como um retângulo cujo comprimento é o tempo de processamento do navio e cuja altura é tamanho do navio. Assim, nosso problema será determinar as coordenadas do canto inferior esquerdo do retângulo no diagrama.

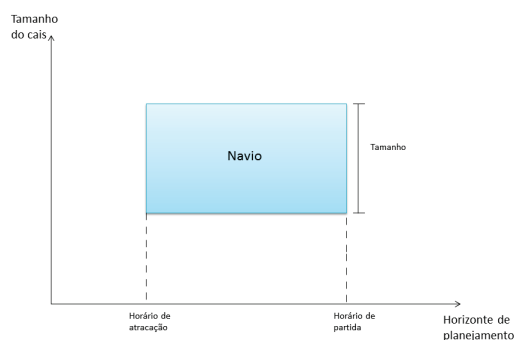


Figura 1: Interpretação do BAP como um problema de corte.

3. Métodos e abordagens

3.1. Modelo de corte para o caso da Vale

- N : número de retângulos a serem alocados;
- m : número total de folhas de estoque disponíveis;
- M : número grande arbitrário;
- (p_i) : parâmetros indicando tempo de processamento do navio i ;
- x_i : variável contínua indicando o horário do início do processamento do navio i .
- y_i , tal que $1 \leq y_i \leq 2$: variável inteira que indica em qual berço o navio i foi alocado
- e_i : horário de chegada do navio i

$$a_{ik} = \begin{cases} 1 & \text{se o retângulo } i \text{ está à esquerda do retângulo } k. \\ 0 & \text{caso contrário.} \end{cases}$$

$$b_{ik} = \begin{cases} 1 & \text{se o retângulo } i \text{ está à direita do retângulo } k. \\ 0 & \text{caso contrário.} \end{cases}$$

$$c_{ik} = \begin{cases} 1 & \text{se o retângulo } i \text{ está atrás do retângulo } k. \\ 0 & \text{caso contrário.} \end{cases}$$

$$d_{ik} = \begin{cases} 1 & \text{se o retângulo } i \text{ está na frente do retângulo } k. \\ 0 & \text{caso contrário.} \end{cases}$$

$$\min \quad \sum_i (e_i - x_i) \quad (1)$$

$$\text{s.a} \quad x_i + p_i \leq x_k + (1 - a_{ik})M \quad \forall i, k, i < k \quad (2)$$

$$x_k + p_k \leq x_i + (1 - b_{ik})M \quad \forall i, k, i < k \quad (3)$$

$$y_i + 1 \leq y_k + (1 - c_{ik})M \quad \forall i, k, i < k \quad (4)$$

$$y_k + 1 \leq y_i + (1 - d_{ik})M \quad \forall i, k, i < k \quad (5)$$

$$a_{ik} + b_{ik} + c_{ik} + d_{ik} \geq 1 \quad \forall i, k, i < k \quad (6)$$

$$x_i \geq e_i \quad \forall i \quad (7)$$

$$1 \leq y_i \leq 2 \quad (8)$$

$$l_{xi}, h_{yi}, a_{ik}, b_{ik}, c_{ik}, d_{ik} \in 0, 1 \quad \forall i, k \quad (9)$$

$$x_i \geq 0 \quad \forall i \quad (10)$$

$$y_i \geq 0 \text{ e inteiro } \forall i. \quad (11)$$

3.2. Modelo de Esquematização de Navios a Berços

$$z_{ij} = \begin{cases} 1 & \text{se o navio } i \text{ está no berço } j. \\ 0 & \text{caso contrário.} \end{cases}$$

$$a_{ik} = \begin{cases} 1 & \text{se o retângulo } i \text{ está à esquerda do retângulo } k. \\ 0 & \text{caso contrário.} \end{cases}$$

x_i = posição do início do atendimento do navio no horizonte de planejamento.

$$\min \quad \sum_i (e_i - x_i) \quad (12)$$

$$\text{s.a} \quad \sum_j z_{ij} = 1 \quad \forall i \quad (13)$$

$$z_{ij} + z_{kj} - a_{ik} - a_{ki} \leq 1 \quad \forall i \neq k, j \quad (14)$$

$$x_i + p_i - (1 - a_{ik})M \leq x_k \quad \forall i, k \quad (15)$$

$$z_{ij} + z_{kh} + a_{ik} + a_{ki} \leq 2 \quad \forall i \neq k, j \neq h \quad (16)$$

$$x_i \geq e_i \quad \forall i. \quad (17)$$

3.3. Heurística Aloca-Berço

Os parâmetros usados pela heurística são os mesmos. A primeira ação a ser feita é colocar os navios em ordem crescente de horário de partida ($w_i = e_i + p_i$). Então são criadas duas listas vazias, uma representando o Berço 1 e outra o Berço 2. Também deve ser criada um vetor L em que na posição j é armazenada a soma do tempo de atendimento do Berço j ($j = 1, 2$).

No Passo k do algoritmo (quando o k -ésimo navio deve ser alocado a um berço) verifica-se qual berço possui o menor tempo total de atendimento. Se $L(1) < L(2)$, então o Navio k é alocado no Berço 1 e $L(1)$ é atualizado para $L(1) + p(k)$. Caso contrário, o Navio k é alocado no Berço 2 e $L(2)$ é atualizado para $L(2) + p(k)$. Esse procedimento é feito até que todos os navios sejam atribuídos a um berço.

3.4. Heurística *First Come, First Serve* (FCFS)

O terminal utiliza o princípio *First Come, First Serve* para a atribuição de seus navios nos dois berços: o primeiro navio a chegar é o primeiro a ser alocado ao berço que estiver livre. É interessante implementar esse algoritmo para que seja possível comparar os resultados dos modelos e da heurística com a estratégia de alocação que tem sido usado pela Vale.

Para tanto, basta fazer uma simples modificação no algoritmo da heurística apresentada anteriormente: os navios devem ser colocados em ordem crescente de horário de chegada.

4. Resultados

Problemas de diferentes tamanhos serão estudados e para cada um desses tamanhos foram gerados 10 cenários. Os horários de chegada dos navios foram gerados com base em uma distribuição uniforme entre $[0, 25]$ e os tempos de processamento foram gerados por uma distribuição binomial com 16 tentativas e $p = 0.5$.

Os valores encontrados para a função-objetivo (soma dos tempos de espera) com GLPK e com CPLEX serão comparados para ambos modelos, o valor da função objetivo do problema de corte e do problema de esquematização quando utilizamos o CPLEX e o melhor resultado dos modelos com as heurísticas. O critério de parada para a execução dos problemas nos solvers foi o tempo, limitado em 1 hora.

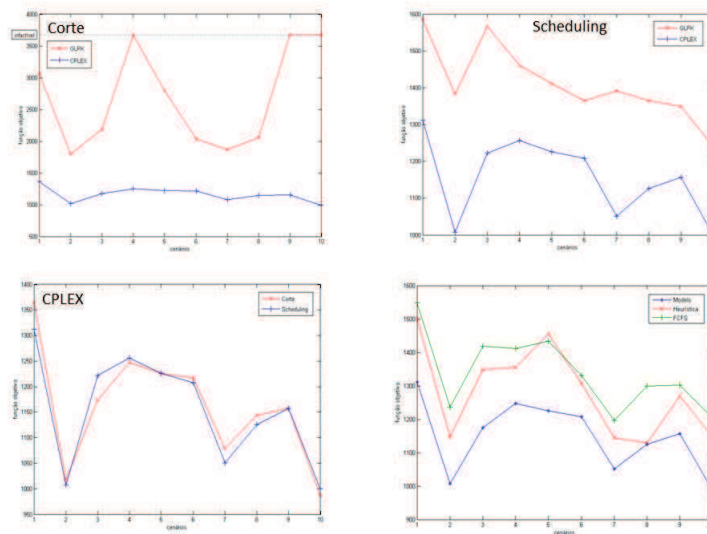


Figura 2: Comparações para 30 navios.

5. Conclusões

O objeto de estudo do presente trabalho foi o Problema de Alocação de Berços: qual a melhor maneira de alocar os navios que chegam ao terminal de modo a minimizar o tempo de espera desses. Após um abrangente estudo sobre as variações que estão disponíveis na literatura, dois modelos matemáticos foram adaptados e uma heurística foi construída. A otimalidade só foi garantida para os cenários com 10 navios. Pelos gráficos e tabelas apresentados no Capítulo , o desempenho do CPLEX foi notavelmente melhor para ambos modelos, apresentado o melhor resultado de função-objetivo (mesmo sem garantia de otimalidade). Em alguns cenários, o GLPK

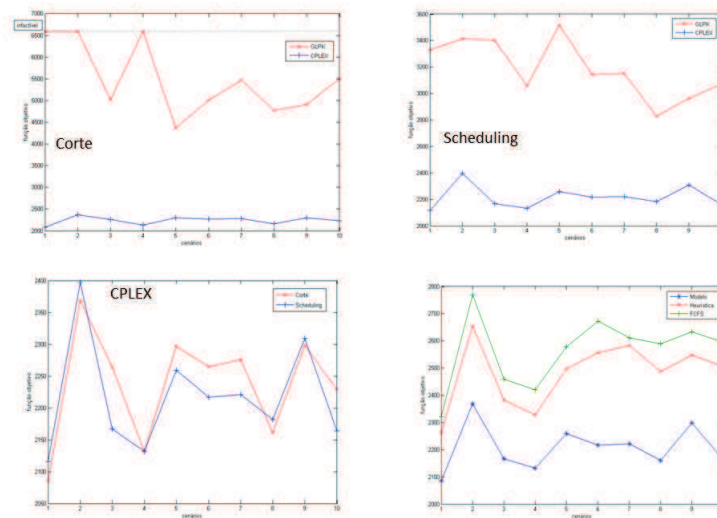


Figura 3: Comparações para 40 navios.

não conseguiu sequer obter uma solução factível dentro do tempo estipulado. Quando os modelos foram comparados, a diferença apresentada foi pequena, mas em pouco mais da metade dos casos o modelo de esquematização retornou um valor melhor para a função-objetivo minimizada.

Depois do resultado do modelo, está o resultado obtido pela heurística. Também é possível ver que o *FCFS* em quase todas as vezes foi o método que teve o pior resultado. De modo geral, pode-se concluir que uma maneira mais eficaz de fazer o planeamento das operações de atracação no terminal foi obtida.

ALGORITMO DE OTIMIZAÇÃO POR COLÔNIA DE FORMIGAS NA ROTEIRIZAÇÃO DE VEÍCULOS

Ricardo Pinto Ferreira

Universidade Nove de Julho

kasparov@uninove.br

Arthur Ferreira

Escola Técnica Estadual - ETEC

arthur.ferreira16@etec.sp.gov.br

Renato José Sassi

Universidade Nove de Julho

sassi@uninove.br

RESUMO

As formigas quando em busca de alimento exploram aleatoriamente o ambiente em torno de seu ninho de uma forma aparentemente desordenada. Ao percorrerem esse trajeto as formigas liberam uma substância química odorífera chamada feromônio que as formigas podem depositar e cheirar durante o trajeto. Ao mesmo tempo, as formigas são influenciadas pela presença de feromônios no ambiente e tem uma tendência probabilística a seguir na direção em que a concentração de feromônios é mais forte, pois deve-se escolher um caminho entre todos os outros existentes a partir da posição atual. Esse comportamento das formigas é um exemplo de otimização o que motiva a aplicação na roteirização de veículos. O objetivo deste artigo é apresentar um experimento computacional aplicando o algoritmo de otimização por colônia de formigas na roteirização de veículos.

PALAVARAS-CHAVE: Roteirização de veículos, Coleta e distribuição, Algoritmo de otimização por colônia de formigas.

1. Introdução

Na última década, o Brasil sofreu profundas transformações, caminhando a largos passos de uma economia fechada e ineficiente para uma economia de mercado (Lumare Júnior, 2007). Ao transporte, coleta e distribuição restaram acompanhar essas tendências. Em razão disso, o serviço de coleta e distribuição tornou-se altamente complexo e competitivo.

Neste artigo foi aplicado para a roteirização de veículos, o algoritmo de otimização por colônia de formigas. O resultado aponta que a roteirização de veículos com o algoritmo de otimização por colônia de formigas é capaz de gerar bons roteiros com tempo de processamento reduzido.

O artigo está organizado após essa seção introdutória da seguinte forma: na seção 2 é apresentada a descrição do problema, na seção 3 é apresentado os métodos e abordagens, na seção 4 é apresentado os resultados computacionais, na seção 5 o artigo é encerrado com as conclusões e perspectivas futuras.

2. Descrição do problema

O problema de roteirização de veículos consiste em definir itinerários, fazendo com que minimizem o custo total de transporte e atendimento. Com isso, visam que os trajetos iniciem e terminem no depósito ou base dos veículos, assegurando que cada ponto seja visitado exatamente uma vez e a demanda, em qualquer rota, não exceda a capacidade do veículo que a atende (Bodin et al., 1983; Laporte et al., 2000).

3. Métodos e abordagens

O primeiro algoritmo de colônia de formigas *Ant Colony Optimization* (ACO) foi inspirado na observação de colônias de formigas que são capazes de encontrar o caminho mais curto entre seus ninhos e as fontes de alimento (Dorigo e Stützle, 2004).

As formigas quando em busca de alimento exploram aleatoriamente o ambiente em torno de seu ninho de uma forma aparentemente desordenada. Ao percorrerem esse trajeto as formigas liberam feromônio e são influenciadas pela presença de feromônios no ambiente e tem uma tendência probabilística a seguir na direção em que a concentração de feromônios é mais forte. A Figura 1 ilustra de forma simplificada essa probabilidade na sequência (a, b, c, d, e).

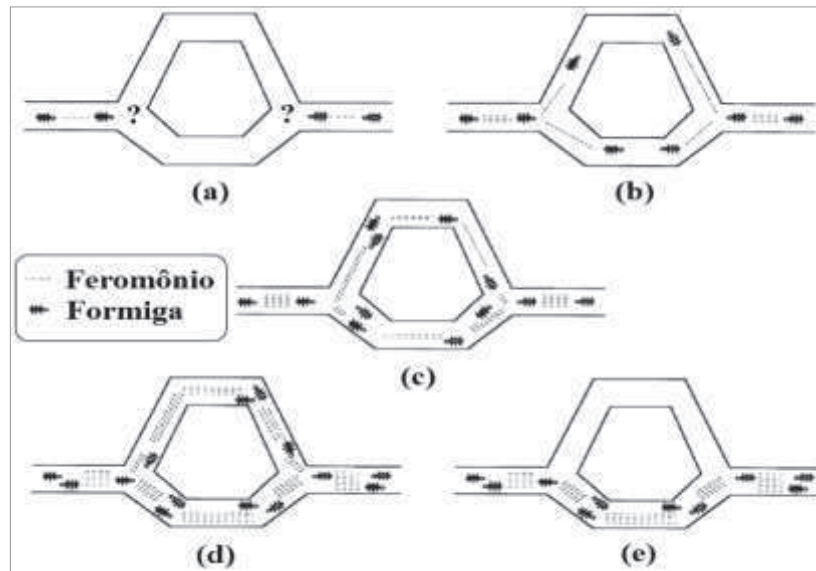


Figura 1 – Trajeto das formigas adaptado de (Dorigo e Gambardella, 1997).

A escolha do arco baseia-se na quantidade de feromônios acumulado no caminho das formigas precedentes e na distância existente em cada arco. Para esses valores é atribuído um peso diferente através dos parâmetros do algoritmo: *alfa* e *beta*, respectivamente (Santos e Leal, 2007).

Esses parâmetros servem para determinar a influência do feromônio e da informação heurística.

A atratividade de cada arco pode ser descrito conforme a Equação 1:

$$a_{ij}(t) = \frac{[\tau_{ij}(t)]^\alpha [n_{ij}]^\beta}{\sum_{l \in N_i} [\tau_{il}(t)]^\alpha [n_{il}]^\beta} \quad (1)$$

A partir da equação 1 é calculada a atratividade $a_{ij}(t)$ para uma formiga que está em um nó i qualquer, e para todo nó j pertencente a N_i , em que N_i é o conjunto dos nós adjacentes ao nó i . A variável τ_{ij} representa a quantidade de feromônio acumulada no arco ij , e a variável η_{ij} é igual ao inverso do comprimento do arco ij . O parâmetro t , indica que os valores estão variando a cada iteração.

Após o cálculo de todos os valores dos arcos incidentes a_i , calcula-se a probabilidade de uma formiga k escolher percorrer o arco ij . A Equação 2 descreve essa probabilidade:

$$p_{ij}^k(t) = \frac{a_{ij}(t)}{\sum_{l \in N_i^k} a_{il}(t)} \quad (2)$$

Na equação 2, N_i^k é o subconjunto de N_i que apresenta apenas os nós que ainda não foram visitados pela formiga k .

A formiga k seleciona um arco entre todas as possíveis alternativas através do cálculo da probabilidade acumulada a partir das probabilidades de cada arco p_{ij} e sorteia-se um número aleatório no intervalo entre 0 e 1. O arco escolhido será aquele em que o número sorteado possuir valor maior ou igual à $p_{ij}-1$ e menor que p_{ij} . Uma vez que todas as formigas k percorreram os arcos, o feromônio é atualizado em todos os arcos de acordo com Dorigo e Gambardella (1997). Esta atualização consiste na evaporação dos feromônios e na deposição de novos feromônios.

Arcos mais curtos com feromônio elevado têm maior probabilidade de seleção (Haupt e Haupt, 2004). A Figura 2 ilustra a seleção baseada nesse critério.

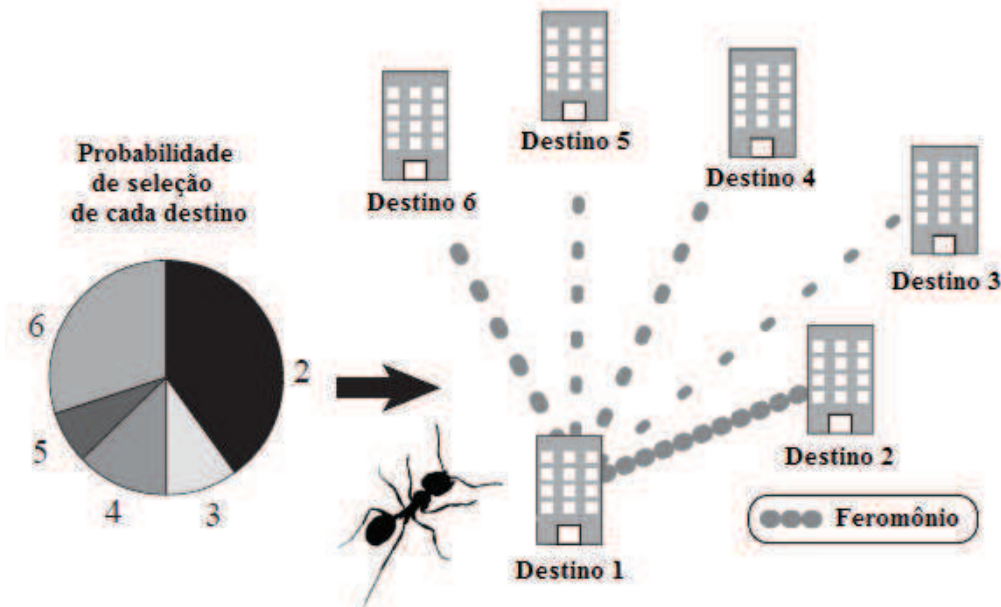


Figura 2 – Exemplo da probabilidade de seleção. Fonte: Adaptado de Yoshikawa e Otani (2010).

A Equação 3 descreve a atualização dos feromônios:

$$\tau_{ij}(t) \leftarrow (1 - \rho)\tau_{ij}(t) + \sum_{k=1}^m \Delta\tau_{ij}^k(t) \quad (3)$$

Na equação 3, o parâmetro ρ representa o nível de evaporação dos feromônios, que atua

como um coeficiente de decaimento. O somatório representa todas as contribuições individuais de cada formiga k que tenha passado pelo arco ij na atual iteração e m é o número de formigas (Silva Júnior e Leal, 2009).

O $\Delta\tau_{ij}^k$ é a quantidade de feromônios que a formiga k deposita no arco ij , e é definida como o inverso do comprimento total do percurso L_k que a formiga k percorreu (Dorigo et al., 1996). Quanto mais formigas passarem ao longo de um arco, mais feromônios serão depositados no arco, e o tamanho destas deposições é proporcional à qualidade das soluções encontradas.

Os experimentos computacionais foram realizados no *software* Matlab usando o problema proposto por Solomon (1987).

Os parâmetros utilizados para aplicação do algoritmo de otimização por colônia de formigas foi: $\alpha = 1$, $\beta = 5$, $\rho = 0,65$. O número máximo de iterações = 200 com $k = 20$ formigas cada uma. Número de pontos atendidos = 100.

4. Resultados computacionais

A Figura 3 ilustra o roteiro gerado pelo algoritmo de otimização por colônia de formigas. O tempo de processamento foi de 57 segundos.

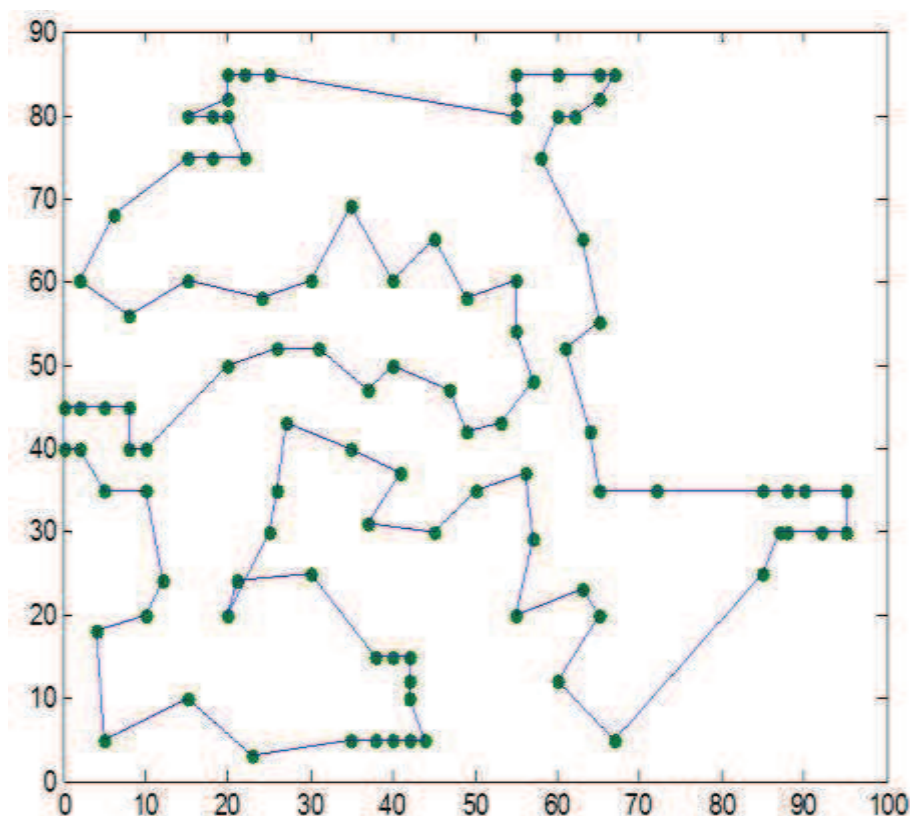


Figura 3 – Roteiro gerado pelo algoritmo de otimização por colônia de formigas.

Observa-se que o roteiro gerado pelo algoritmo de otimização por colônia de formigas não apresenta sobreposição ou cruzamentos de arcos o que indica inicialmente um roteiro adequado para o problema proposto.

5. Conclusões e perspectivas futuras

O algoritmo de otimização por colônia de formigas foi capaz de gerar um bom roteiro com tempo de processamento reduzido o que é animador para a resolução de problemas de roteirização em que normalmente o tempo de processamento é bastante expressivo. Assim, conclui-se que a aplicação do algoritmo proposto no artigo para a roteirização de veículos pode auxiliar na construção de roteiros de coleta e distribuição de maneira produtiva e otimizada.

Como perspectivas futuras pretende-se: realizar novos experimentos utilizando outros parâmetros para o algoritmo de otimização por colônia de formigas; realizar novos experimentos com maior número de pontos de coleta e distribuição; utilizar algoritmos híbridos como por exemplo: algoritmos genéticos e colônia de formigas. Pretende-se ainda integrar a proposta apresentada nesse artigo a um sistema de roteirização de veículos por hierarquia como proposto por Ferreira (2011) utilizando no nível operacional o algoritmo de otimização por colônia de formigas.

Referências

- Bodin, L. D.; Golden, B.; Assad, A.; Ball, E.M.** (1983). *Routing and scheduling of vehicles and crews: The state of the art. Computers and Operations Research*, 10(2), 289–315.
- Dorigo, M.; Gambardella, L. M.** (1997). *Ant Colony System: A Cooperative Learning Approach to the Traveling Salesman Problem. IEEE Transactions on Evolutionary Computation*, 1(1), 53–66.
- Dorigo, M.; Maniezzo, V.; Colomi, A.** (1996). *The Ant System: Optimization by a colony of cooperating agents. IEEE Transactions on Systems, Man, and Cybernetics–Part B*, 26(1), 1-26.
- Dorigo, M.; Stützle, T.** (2004). *Ant Colony Optimization. Bradford Book*, 2004.
- Ferreira, R. P.** (2011) *Combinação de Técnicas da Inteligência Artificial para Previsão do Comportamento do Tráfego Veicular Urbano na Cidade de São Paulo*. 2011. Dissertação (Mestrado) – Universidade Nove de Julho, Engenharia de Produção, São Paulo. 107 p.
- Haupt, R. L.; Haupt, S. E.** (2004). *Practical genetic algorithms, Hoboken, New Jersey. Wiley & Sons, Inc.*
- Laporte, G.; Gendreau, M.; Potvin, J.Y.; Semet, E. F.** (2000). *Classical and modern heuristics for the vehicle routing problem, International Transactions in Operational Research*, 7(4/5), 285–300.
- Lumare Júnior, G.** (2007). *Valor econômico do cliente: uma teoria das encomendas*. São Paulo: Prentice Hall.
- Santos, R. L.; Leal, J. E.** (2007). *Solução de um problema de roteirização com janelas de tempo através de um algoritmo de múltiplas colônias de formigas. Transportes (Rio de Janeiro)*, v. XV, 2, 5-16.
- Silva Júnior, O. S.; Leal, J. E.** (2009). *Roteirização dinâmica de veículos com janelas de tempo usando de um algoritmo de colônias de formigas*. In: XXIII ANPET - Congresso de Pesquisa e Ensino em Transportes. Vitória. XXIII ANPET.
- Solomon, M. M.** (1987). *Algorithms for the vehicle routing and scheduling problems with time windows constraints. Operations Research*, 35(2), 254–265.
- Yoshikawa, M.; Otani, K.** (2010). *Ant Colony Optimization Routing Algorithm with Tabu Search. Proceedings of the Internacional MultiConference of Engineers and Computer Scientists*, v. III. IMECS 2010. Hong Kong.

MAPA AUTO-ORGANIZÁVEL DE KOHONEN NA PROGRAMAÇÃO DE VEÍCULOS

Ricardo Pinto Ferreira

Universidade Nove de Julho
kasparov@uninove.br

Andréa Martiniano da Silva

Universidade Nove de Julho
andrea.martiniano@gmail.com

Renato José Sassi

Universidade Nove de Julho
sassi@uninove.br

RESUMO

A expansão rápida do comércio eletrônico no país deu um novo impulso ao setor de transportes. Este novo cenário provocou mudanças no mercado de coleta e distribuição, tornando essa atividade complexa e competitiva afetando diretamente a eficiência deste serviço. Neste artigo foi aplicada uma rede neural artificial do tipo Mapa Auto-Organizável de Kohonen ou do inglês *Self-Organizing Map* (SOM) para contribuir com a programação de veículos em uma empresa de logística. O objetivo deste artigo é apresentar um experimento computacional capaz de contribuir na programação de veículos a partir de agrupamentos gerados pela rede SOM. Os resultados iniciais apontam que a aplicação da rede SOM na programação de veículos pode contribuir para o aumento da eficácia na coleta e distribuição física.

PALAVRAS-CHAVE: Programação de veículos, Coleta e distribuição, Redes neurais artificiais, Mapa Auto-Organizável.

1. Introdução

O crescimento do comércio eletrônico (*e-commerce*) e das transações *business-to-consumer* (B2C) e *business-to-business* (B2B) tem provocado um aumento das dimensões, volume e do valor das encomendas transportadas, o problema de programação de veículos ganha maior importância, tendo em vista a necessidade de melhorar a qualidade na distribuição física.

A programação de veículos envolve: quantidade de veículos, capacidade de cada veículo (cubagem em m³), pontos de paradas para coleta ou distribuição, tempo necessário para cada parada, alocação da tripulação, jornada de trabalho, etc. Em razão disso, princípios operacionais, que resultem em boas soluções, podem ser muito úteis, tendo em vista que normalmente o tempo para a programação de veículos é curto.

A rede SOM é uma rede neural artificial de aprendizagem competitiva organizada em duas camadas, a primeira camada representa o vetor dos dados de entrada, a segunda corresponde a uma grade de neurônios, geralmente bidimensional, totalmente conectada aos componentes do vetor de entrada. Cada neurônio possui um vetor de código associado (Kohonen, 2001). Neste artigo foi aplicada uma rede SOM para contribuir na programação de veículos.

O artigo está organizado após essa seção introdutória da seguinte forma: na seção 2 é apresentada a descrição do problema, na seção 3 é apresentado os métodos e abordagens, na seção 4 é apresentado os resultados computacionais, na seção 5 o artigo é encerrado com as conclusões e perspectivas futuras.

2. Descrição do problema

No Problema de Programação de Veículos, as condicionantes temporais devem ser consideradas, ou seja, as restrições adicionais relacionadas aos horários em que as atividades serão executadas devem ser consideradas no tratamento do problema, entre outras restrições como tipo de carga, forma de carregamento, janela de atendimento, restrições de circulação, restrições da dimensão do veículo, os dois últimos bastante comuns nas grandes cidades.

Segundo Chopra e Meindl (2003), as decisões operacionais mais importantes relacionadas ao transporte dizem respeito às rotas e aos cronogramas (programação) de coletas e distribuição.

O Problema de Programação de Veículos é uma extensão do problema de roteirização de veículos, em que restrições realistas são incluídas, tais como janela de atendimento, precedência de tarefas e alocação da tripulação, etc.

Quanto às restrições, deve-se completar a rota com os recursos disponíveis, atendendo a todos os clientes, tempo e capacidade do veículo; obedecer a limites de jornada de trabalho dos motoristas e ajudantes; obedecer à velocidade máxima permitida; obedecer a restrição do tamanho de veículo permitido em vias públicas; obedecer os horários de carga e descarga, etc. A solução para este problema de coleta e distribuição está em encontrar soluções que respeitem todas as restrições, minimizando o custo total de transporte e satisfazendo a demanda dos clientes (Bodin et al., 1983; Laporte et al., 2000).

3. Métodos e abordagens

A rede SOM desenvolvida por Teuvo Kohonen no início da década de 80 é o modelo de rede neural artificial competitiva e não supervisionada em maior uso, e tem sido aplicada em diversos problemas, incluindo agrupamento e visualização de dados multidimensionais (Kohonen, 2001).

O algoritmo de aprendizado da rede SOM é composto por três fases: competição, cooperação e adaptação:

a) **Competição.** Para cada padrão de entrada, os neurônios da grade calculam seus respectivos valores de uma função discriminante. Esta função discriminante fornece a base para a competição entre os neurônios. O neurônio com o maior valor da função discriminante é declarado vencedor da competição (Haykin, 2001).

O princípio para o processo de aprendizado competitivo é a concorrência entre os neurônios, com o objetivo de determinar o vencedor da competição, pois o processo é não supervisionado, ou seja, não tem a saída desejada (Silva et al., 2010).

O prêmio para o neurônio vencedor da competição é o ajuste dos seus pesos, proporcionalmente aos valores do padrão de entrada apresentado, visando melhorar o seu estado para a próxima competição, se todas as conexões laterais deste neurônio vencedor forem nulas, implica que somente os seus pesos serão ajustados e assume-se a estratégia do vencedor leva tudo (*winner-takes-all*) (Silva et al., 2010; Braga et al., 2011).

b) **Cooperação.** O neurônio vencedor determina a localização espacial de uma vizinhança topológica de neurônios excitados, fornecendo assim a base para a cooperação entre os neurônios vizinhos.

c) **Adaptação Sináptica.** Este último mecanismo permite que os neurônios excitados aumentem seus valores individuais da função discriminante em relação ao padrão de entrada através de ajustes adequados aplicados a seus pesos sinápticos. Os ajustes feitos são tais que a

resposta do neurônio vencedor à aplicação subsequente de um padrão de entrada similar é melhorada (Haykin, 2001).

Para cada dado apresentado à rede haverá uma competição entre todos os neurônios pelo direito de representá-lo. Vence a competição o neurônio que tiver o vetor de pesos mais próximo do vetor de dados. A Figura 1 mostra a arquitetura da rede SOM.

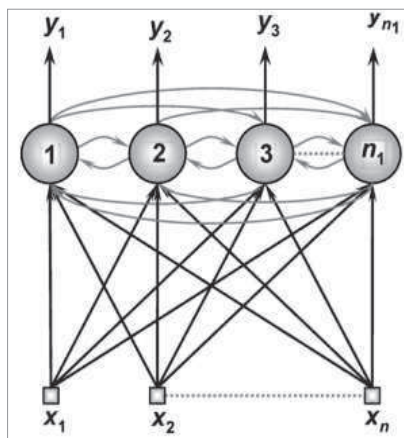


Figura 1 – Arquitetura da rede SOM (adaptado de Silva et al., 2010).

As conexões laterais entre os neurônios apresentadas na Figura 1 revelam o papel que um neurônio pode influenciar na resposta de saída produzida por outro neurônio.

Embora a rede SOM seja utilizada para diversas aplicações em diferentes áreas do conhecimento: Descoberta de Conhecimento em Bases de Dados (Sassi, 2006), Identificação do Perfil de empregados Absenteístas e Presenteístas (Martiniano, 2012) os trabalhos mais bem difundidos tratam sobre problemas que envolvem classificação de padrões e agrupamento de dados ou clusterização (Silva et al., 2010).

A plataforma de *hardware* utilizada nos experimentos foi um computador com processador Intel® Core™ 2 Duo CPU T6600 de 2,20 GHz com 4,00 GB de memória RAM DDR3, 320 GB de disco rígido e sistema operacional Windows 7 com 64 bits.

Com relação ao *software* utilizado para gerar os Mapas Auto-Organizáveis, foi utilizado o *software* Viscovery® SOMine versão 5.2.2.

A base de dados utilizada na rede SOM é composta por 9 atributos e 162 registros e foi coletada através de registros de coleta e distribuição em uma empresa de logística.

Os atributos foram: quantidade de tripulantes, forma de acondicionamento/transbordo da carga (manual ou mecanizada/paletizada), dificuldade de atendimento (tempo de espera, identificação do entregador etc.), distância do depósito em quilômetros, média de carga mensal por ponto, média de carga diária por ponto, média de volumes transportados por dia, ocupação média do veículo (%) e tipo de veículo utilizado.

Os parâmetros utilizados na estrutura da rede SOM foram os seguintes: número de neurônios igual a 64 (8 x 8), vizinhança topológica hexagonal e função de vizinhança topológica gaussiana.

Os parâmetros utilizados na fase de treinamento da rede SOM foram os seguintes: número de épocas igual a 1000, taxa de aprendizagem igual a 0,5 na fase inicial de treinamento e na fase de convergência igual a 0,05 (Kaski e Kohonen, 1997).

4. Resultados computacionais

Foram utilizados três tipos de veículos: (1=C1) com capacidade de até 0,7 tonelada, (2=C2) com capacidade até 1,5 tonelada e (3=C3) com capacidade até 3 toneladas, no experimento foi utilizado um veículo de cada tipo para atender todos os pontos.

A Figura 2 mostra os *clusters* do atributo (tipo de veículo) gerado pela rede SOM.

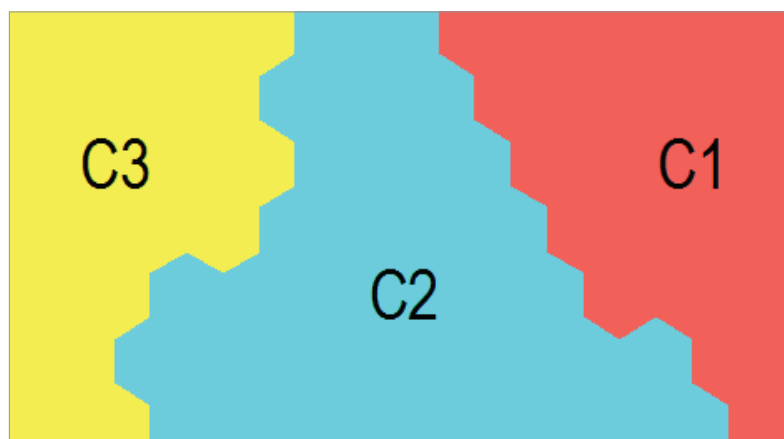


Figura 2 – *Clusters* gerados pela rede SOM segundo o atributo tipo de veículo.

A Figura 3 mostra os agrupamentos (*clusters*) gerados pela rede SOM com toda a base de dados, observam-se as características de cada veículo utilizado no atendimento aos 18 pontos de distribuição. Cada mapa representa da esquerda para a direita e de cima para baixo os atributos: quantidade de tripulantes, forma de acondicionamento da carga/transbordo da carga (manual ou mecanizada/paletizada), dificuldade de atendimento (tempo de espera, identificação do entregador etc.), distância do depósito em quilômetros, média de carga mensal por ponto, média de carga diária por ponto, média de volumes transportados por dia, ocupação média do veículo (%) e tipo de veículo utilizado.

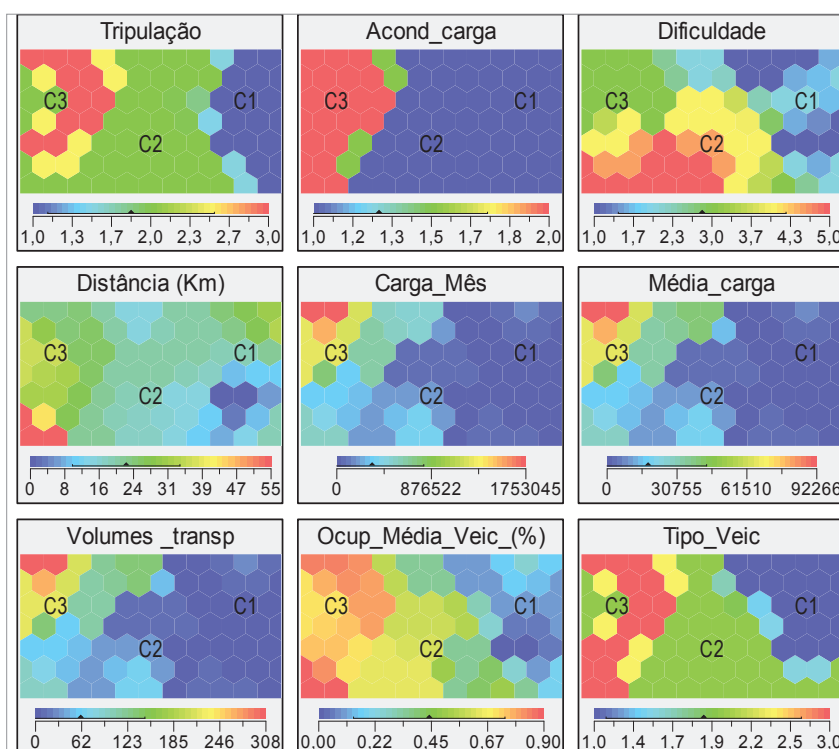


Figura 3 – *Clusters* gerados pela rede SOM.

Observa-se que os *clusters* gerados pela rede SOM apresentam uma forma visual intuitiva para contribuir na tomada de decisão quanto à programação de veículos.

5. Conclusões e perspectivas futuras

Os resultados iniciais demonstraram que a rede SOM agrupou os registros com base na similaridade dos atributos que devem ser levados em consideração na programação de veículos permitindo uma análise visual intuitiva dos *clusters* para identificar o veículo que melhor atende determinado grupo de clientes. Assim, conclui-se que a rede SOM pode contribuir de forma a apoiar a tomada de decisão para compor a programação de veículos.

Como perspectivas futuras pretende-se coletar novos atributos em empresas distintas para realizar novos experimentos com a finalidade de validar o modelo inicial apresentado.

Pretende-se ainda melhorar o ajuste da rede SOM para que as bordas dos mapas (veja Figura 3) não deixem dúvidas na leitura dos contornos dos *clusters*.

Espera-se também criar novas formas de codificar os atributos para que a leitura dos mapas seja mais simples, com o objetivo de facilitar a leitura pelo tomador de decisão responsável pela programação de veículos na empresa de logística.

Referências

- Bodin, L. D.; Golden, B.; Assad, A.; Ball, E.M.** (1983). *Routing and scheduling of vehicles and crews: The state of the art. Computers and Operations Research*, 10(2), 289–315.
- Braga, A. P.; Carvalho, A .C. P. L. F.; Ludemir, T. B.** (2011). *Redes Neurais Artificiais: Teoria e Aplicações*. 2ª Ed. Rio de Janeiro: LTC.
- Chopra, S.; Meindl, P.** (2003). *Gerenciamento da Cadeia de Suprimentos*. Tradução Claudia Freire; revisão técnica Paulo Roberto Leite. – São Paulo: Prentice Hall.
- Haykin, S.** (2001). *Redes Neurais – Princípios e Práticas*. 2ª edição. Porto Alegre. Bookman.
- Kaski, S.; Kohonen, T.** (1997). *Winner-Takes-All Networks. Triennial Report 1994 – 1996, Neural Networks Research Centre & Laboratory of Computer and Information Science, Helsinki University of Technology*, Finland, 72-75.
- Kohonen, T.** (2001). *Self-Organizing Maps. 3ª Edition*. New York. Springer.
- Laporte, G.; Gendreau, M.; Potvin, J.Y.; Semet, E. F.** (2000). *Classical and modern heuristics for the vehicle routing problem, International Transactions in Operational Research*, 7(4/5), 285–300.
- Martiniano, A.** (2012). *Aplicação de Técnica da Mineração de Dados na Identificação do Perfil de empregados Absenteístas e Presenteístas em uma Empresa de Courier da Cidade de São Paulo*. 2012. Dissertação (Mestrado) – Universidade Nove de Julho, Engenharia de Produção, São Paulo. São Paulo. 120 p.
- Sassi, R. J.** (2006). *Uma Arquitetura Híbrida para Descoberta de Conhecimento em Bases de Dados: Teoria dos Rough Sets e Redes Neurais Artificiais Mapas Auto-Organizáveis*. 2006. Tese (Doutorado) – Escola Politécnica da Universidade de São Paulo, Departamento de Engenharia Elétrica, São Paulo. 169 p.
- Silva, I. N.; Spatti, D. H.; Flauzino, R. A.** (2010). *Redes Neurais Artificiais para Engenharia e Ciências Aplicadas*. SP: Artliber.

ANÁLISE ESTATÍSTICA DA DECISÃO PARA TRATAR O PROBLEMA DE ROTEAMENTO E COBERTURA COM MÁXIMO RETORNO ÀS ÁREAS SENSÍVEIS

Maria José Pinto, Mônica Maria De Marchi
Instituto de Estudos Avançados (IEAv)
E-mails: maju@ieav.cta.br, monica@ieav.cta.br

Carlos Alberto Lima Araújo Júnior, Rodrigo Arnaldo Scarpel
Instituto Tecnológico de Aeronáutica (ITA)
E-mails: carlosalbertolajr@gmail.com, rodrigo@ita.br

RESUMO

O problema de roteamento e cobertura com máximo retorno às áreas sensíveis busca determinar rotas eficientes para cobrir um conjunto de áreas em uma região onde algumas destas áreas, denominadas sensíveis, precisam ser revisitadas o mais breve possível. Em trabalhos anteriores, o problema foi denominado CRPMRCA (do inglês, *Covering-Routing Problem with Maximum Returns to Critical Areas*) e considerado multiobjetivo devido ao *trade-off* entre a minimização do custo e a maximização dos retornos às áreas sensíveis. Neste trabalho, estamos propondo uma nova abordagem utilizando conceitos de Análise Estatística da Decisão, onde o CRPMRCA será tratado como um problema de decisão onde, dadas diferentes configurações de rotas possíveis e seus respectivos retornos às áreas sensíveis, a decisão consistirá em definir a rota com maior retorno possível em utilidade. Como aplicação, o problema tem sido considerado em cenários de gerenciamento de crises, busca e salvamento e vigilância do espaço aéreo.

PALAVRAS-CHAVE: Roteamento, Cobertura, Análise Estatística da Decisão, Áreas Sensíveis.

1. Introdução

O processo de tomada de decisão em cenários complexos envolvendo incertezas e uma grande variedade de informações e de entidades responsáveis pela tomada de decisão é extremamente desafiador. Sistemas de gerenciamento de crises (enchentes, terremotos, movimentos sociais, etc), de busca e salvamento e de vigilância do espaço aéreo são exemplos deste tipo de cenário.

Neste contexto, pretende-se apoiar a tomada de decisão aplicando conceitos da Pesquisa Operacional para tratar um problema de roteamento específico onde o objetivo é determinar uma rota eficiente para cobrir (monitorar) um conjunto de áreas (ou pontos) em uma determinada região onde algumas destas áreas, denominadas sensíveis, precisam ser revisitadas o mais breve possível. Neste caso, a rota gerada poderia ser utilizada para monitoramento da região e geração de informações.

Neste trabalho, fazemos uso da teoria de Análise Estatística da Decisão para resolver o problema onde, para considerar o risco inerente a este tipo de cenário, incorporamos conceitos da Teoria de Utilidade (Bekman e Neto, 1980).

Na Seção 2, apresentamos uma descrição mais detalhada do problema e das abordagens que temos utilizado para tratá-lo. Nas Seções 3 e 4, a nova abordagem será apresentada, juntamente com os resultados obtidos até o momento; e, na Seção 5, serão feitas algumas considerações finais.

2. Descrição do problema

Para garantir que as áreas sensíveis permaneçam descobertas o mínimo de tempo possível é interessante buscar a maximização dos retornos a estas áreas. Por outro lado, se as áreas sensíveis forem revisitadas muitas vezes, a rota pode resultar em um custo muito alto. Assim, o problema pode ser visto como multiobjetivo pois existe um *trade-off* entre a maximização de retornos às áreas sensíveis e a minimização dos custos.

Em Lamosa *et al.* (2011), propomos uma formulação matemática multiobjetivo para o CRPMRCA e, em Lamosa *et al.* (2012), apresentamos a primeira tentativa de resolver o problema utilizando uma metodologia multiobjetivo, baseada na formulação matemática proposta. Dentre os métodos clássicos citados em Deb (2001), foi utilizado o “método ϵ -restrito”, que consiste em reformular o problema considerando somente um dos objetivos e incluindo os demais como restrições do problema limitados a um valor específico, o qual é atualizado iterativamente de forma a gerar os valores da curva de *trade-off*. Foram consideradas duas abordagens, sendo que a primeira consistiu em maximizar os retornos, limitados a restrições de custo e, a segunda, consistiu em minimizar os custos, com restrições em relação a quantidade de retornos às áreas sensíveis, sendo que a segunda abordagem apresentou melhores resultados.

Em Lamosa *et al.* (2013), foram apresentadas outras duas tentativas para tratar o CRPMRCA, utilizando ainda a formulação matemática proposta. A primeira tentativa consistiu em reduzir o custo dos arcos que passam por uma área sensível de forma a permitir um número maior de retornos à estas áreas. Esta redução foi feita utilizando uma porcentagem fixa dos valores originais. Em seguida, a formulação matemática foi utilizada e, no final, o custo original restaurado para mostrar o aumento resultante no custo final da rota. A segunda tentativa consistiu em manter no grafo somente os nós referentes às áreas sensíveis e utilizar como custo dos arcos o valor do caminho mínimo entre estas áreas. As áreas não sensíveis que, eventualmente, não tenham sido consideradas na geração dos caminhos mínimos são incluídas no grafo. Em seguida, a formulação foi utilizada considerando os nós do novo grafo. Em ambas as tentativas, a formulação matemática considerou somente o objetivo de minimização do custo e o retorno às áreas sensíveis foi permitido. A primeira abordagem obteve resultados satisfatórios ao contrário da segunda abordagem.

3. Método e abordagens

Nos trabalhos anteriores, buscava-se rotas otimizadas em relação ao custo e ao número de retornos. Entretanto, considerando os cenários de aplicação e da forma como o problema foi definido, o CRPMRCA pode ser abordado como um problema de decisão, onde a questão de interesse será, dadas inúmeras configurações de rotas possíveis e os respectivos retornos às áreas sensíveis, definir a solução com maior valor de utilidade possível. Ou seja, além do custo e do número de retornos, serão considerados fatores como risco ou a probabilidade de se encontrar o que se está buscando ao realizar uma determinada rota com um determinado número de retornos em uma área (por exemplo, a probabilidade de encontrar um sobrevivente em um cenário de busca e salvamento).

Para auxiliar neste tipo de decisão, estamos propondo uma nova abordagem utilizando conceitos da teoria de Análise Estatística da Decisão e, para considerar o risco inerente a este tipo de cenário, incorporamos conceitos da Teoria de Utilidade (Bekman e Neto, 1980).

Para chegar a cada configuração possível, contemplando variabilidade em relação ao número de retornos às áreas sensíveis, será incluído um nó virtual correspondente a cada retorno. Os custos das arestas que ligam cada nó virtual aos demais nós da rede serão exatamente iguais aos custos relativos ao nó parental. Será considerado, ainda, um custo virtualmente infinito entre os nós virtuais e seu nó parental, a fim de evitar ciclos. Desta forma, o problema passa a ser visto como um PRV (Problema de Roteamento de Veículos). Assim, para resolução do problema pode ser utilizada qualquer metodologia da literatura para resolver o PRV. Neste trabalho, utilizamos a metaheurística *vrp_rtr* da biblioteca VRPH (Groer, 2014). Como resultado obtemos os valores de

custo, as rotas e o número de entidades que realizam o roteamento.

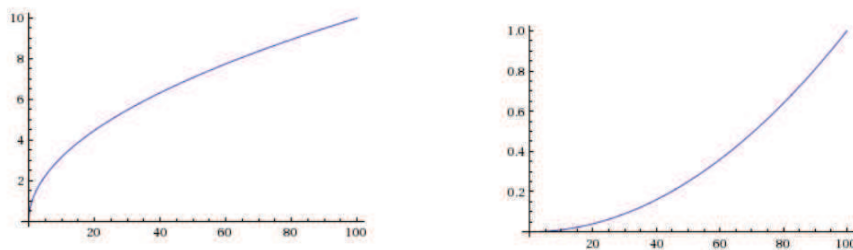
Para avaliar e comparar as configurações obtidas foi utilizada uma função de utilidade baseada no custo, na expectativa de retorno em relação a cada nó visitado e no risco. Essa função foi utilizada considerando que os cenários em estudos envolvem eventos com incertezas associadas. Desta forma, a metodologia proposta permite definir quais configurações maximizam o retorno em utilidade e, conseqüentemente, otimizam a alocação de recursos.

4. Resultados

Os testes computacionais foram realizados utilizando algumas instâncias obtidas da TSPLIB (Reinelt, 1991) para o PRV, onde são fornecidos os pontos (áreas a serem cobertas), o custo dos arcos entre eles, o número e a capacidade das entidades que realizarão o roteamento. Para utilizar estas instâncias, selecionamos aleatoriamente algumas das áreas disponíveis para serem consideradas sensíveis. Os casos de teste utilizados foram:

- n7, r1-5 : caso de teste TSPLIB n7, considerando o nó 4 como área sensível e de 1 a 5 retornos.
- n13, r1-5 : caso de teste TSPLIB n13, considerando o nó 10 como área sensível e de 1 a 5 retornos.
- n31, r1-2 r1-2 : caso de teste TSPLIB n31, considerando os nós 24 e 30 como áreas sensíveis, e uma combinação de 1 a 2 retornos para cada nó.

Para uma análise comparativa dos resultados, foram utilizadas duas configurações de exposição ao risco, representadas pelas curvas de utilidade das Figuras 1a e 1b. A Figura 1a mostra o perfil de um decisor avesso ao risco e a Figura 1b mostra o perfil de um decisor afim ao risco. Conforme será apresentado adiante, a prescrição do comportamento face ao risco é de extrema relevância a fim de definir a escolha da configuração que melhor atende as condições específicas de cada problema em questão.



(a) \sqrt{x} (b)
Figura 1: (a) Curva de utilidade 1: \sqrt{x} (b)
(b) Curva de utilidade 2: $x^2/10000$

As Tabelas 1 a 4 apresentam os resultados obtidos utilizando uma planilha Excel para realizar as simulações. Nas tabelas, o parâmetro **P** corresponde à probabilidade de se encontrar, em um retorno aleatório, o que se está buscando na operação de monitoramento a ser realizada dentro do cenário em questão. Por exemplo, vítimas de um acidente (cenário de gerenciamento de crise ou de busca e salvamento), atividades ilegais em zona de fronteira (monitoramento do espaço aéreo), áreas com perigo de deslizamento (gerenciamento de crise), etc. O parâmetro **M** corresponde ao montante de recursos iniciais disponíveis para a operação, que pode ser definido em unidades monetárias ou em unidades de recursos disponíveis como, por exemplo, tempo disponível, litros de combustível disponíveis, etc. O parâmetro **Prêmio** corresponde ao montante

equivalente em recursos obtidos por se encontrar o que se está pretendendo naquela operação de monitoramento. Para cada configuração, dada a probabilidade a priori, foi calculada a probabilidade de se encontrar o objeto de busca dado o correspondente número de retornos. Para um melhor detalhamento sobre o cálculo de probabilidades, sugere-se consultar Bekman e Neto (1980).

As colunas **R**, **C** e **qE** correspondem ao número de retornos, ao custo e à quantidade de entidades necessárias para realizar o roteamento da configuração correspondente. As colunas **F1** e **F2** são os resultados correspondentes à função de utilidade 1 e 2, respectivamente.

A Tabela 1 apresenta o caso de teste n7 considerando uma baixa quantidade inicial de recurso **M**. Observa-se que o decisor com comportamento de aversão ao risco tenderá a uma escolha conservadora, adotando uma rota com baixo número de retornos (destacado em negrito na coluna **F1**). Já o decisor com alta afinidade ao risco tenderá a ter uma maior utilidade em arriscar a ganhar o prêmio esperado, adotando uma configuração com alto número de retornos (destacado em negrito na coluna **F2**).

Tabela 1: Caso de teste n7 com baixa quantidade inicial de recurso.

P	M	Prêmio	R	C	qE	F1	F2
0,01	400	2000	0	111	2	17,30843495	13,5081
			1	128	3	17,11276677	17,52352
			2	164	3	16,3104683	20,2537744
			3	186	3	15,90639046	23,71417754
			4	210	4	15,40203321	26,93873625
			5	250	4	14,24418369	29,16913128

Na Tabela 2 observamos que uma alteração nos parâmetros de entrada alteram a solução final. Neste caso, reduzimos o valor do prêmio **M** e obtivemos o mesmo comportamento para ambos os decisores. Este resultado é esperado pois, mesmo para uma postura de afinidade ao risco, não convém correr riscos para um prêmio não atrativo.

Tabela 2: Caso de teste n7 com prêmio menor.

P	M	Prêmio	R	C	qE	F1	F2
0,01	400	2000	0	111	2	17,11089144	8,8911
			1	128	3	16,71714257	8,43718
			2	164	3	15,71178401	7,0130686
			3	186	3	15,10521345	6,407945136
			4	210	4	14,39587928	5,766437804
			5	250	4	13,02269915	4,590794024

Foram obtidos comportamentos semelhantes considerando os casos n13 e n31 de acordo com a variação de parâmetros imposta, ou seja, o modelo comportou-se conforme o esperado em relação à exposição ao risco, custo, quantidade de recursos disponíveis e prêmio.

5. Conclusões e perspectivas futuras

Neste trabalho, utilizamos Análise Estatística da Decisão como uma forma de tratar o CRPMRCA. Os resultados obtidos até o momento mostraram-se satisfatórios, pois a solução

sugerida pela metodologia mostrou um comportamento esperado em relação ao perfil de exposição ao risco de cada decisor, representado pelas diferentes curvas de utilidade. Ainda, a configuração que teve a maior utilidade esperada não necessariamente foi a configuração que teve o menor custo ou o maior valor monetário esperado.

Entretanto, apesar da metodologia ter mostrado coerência nos resultados considerando os parâmetros estabelecidos, sugere-se uma calibração estatística dos parâmetros da função utilidade para que estes sejam ainda mais representativos em relação à prescrição do comportamento dos decisores face ao risco para uma aplicação real. Esta necessidade advém do fato das funções de utilidade tomadas até o momento serem funções matemáticas padronizadas, tendo por objetivo apresentar o comportamento desejado para cada caso que se pretendeu demonstrar.

Como continuidade do trabalho, encontra-se em desenvolvimento uma metodologia multiobjetivo para tratar o problema baseada no GRASP (Goldberg e Luna, 2000). Além disso, um novo modelo está sendo desenvolvido para incluir limitações e características ao problema que até o momento não estavam sendo consideradas como, por exemplo, o tempo associado ou a autonomia/capacidade da entidade que realizará o roteamento.

Referências

- Bekman, O. R.; Costa Neto, P. L.** (1980). *Análise Estatística da Decisão*. São Paulo: Editora Edgard Blucher Ltda.
- Deb, K.** (2001). *Multi-objective Optimization using Evolutionary Algorithms*. John Wiley & Sons.
- Goldberg, M. C., Luna, H. P. L.** (2000). *Otimização Combinatória e Programação Linear: Modelos e Algoritmos*. Rio de Janeiro: Editora Campus.
- Groer, C.** VRPH. Disponível em <<https://projects.coin-or.org/VRPH/wiki>>. Acesso em 10/02/2014.
- Lamosa, M. J. P., Lamosa, D. M., De Marchi, M. M.** (2011). Heurística modificada para um problema de roteamento e cobertura com maximização de retornos às áreas sensíveis – resultados computacionais. *XLIII Simpósio Brasileiro de Pesquisa Operacional (SBPO)*.
- Lamosa, M. J. P., De Marchi, M. M. e Lamosa, D. M.** (2012). O problema de roteamento e cobertura num contexto de gerenciamento de crise. *XLIV Simpósio Brasileiro de Pesquisa Operacional (SBPO)*.
- Lamosa, M. J. P., De Marchi, M. M. e Lamosa, D. M.** (2013). Resolução do problema multiobjetivo de roteamento e cobertura com retorno às áreas sensíveis. *XLV Simpósio Brasileiro de Pesquisa Operacional (SBPO)*.
- Reinelt, G.,** (1991). TSPLIB: A traveling salesman problem library, *ORSA Journal on Computing*, 3, 376-384.

PLANEJAMENTO DA REDE DE LOGÍSTICA REVERSA PÓS-CONSUMO DE EQUIPAMENTOS DE INFORMÁTICA

Maritha Gomes Silva de Oliveira
Universidade Federal de São Carlos
marithasilva@yahoo.com.br

Eli Angela Vitor Tosos
Universidade Federal de São Carlos
eli@ufscar.br

RESUMO

O objetivo deste projeto é o estudo da configuração da cadeia de logística reversa pós-consumo de equipamentos de informática. Uma motivação para o trabalho são os desafios logísticos inerentes ao descarte de resíduos sólidos urbanos, que envolvem comprometimento entre sociedade, governo e empresas privadas. No caso de equipamentos eletroeletrônicos, além dos aspectos legais previstos no Plano Nacional de Resíduos Sólidos (PNRS), as oportunidades de recuperação de valor para os produtos são relevantes e podem aumentar sua vida útil. Particularmente no caso de equipamentos de informática, uma forma de recuperação é através dos Centros de Recondicionamento de Computadores (CRCs), que são locais onde equipamentos de informática são recondicionados e disponibilizados para utilização por comunidades carentes. Os CRCs fazem parte do projeto 'Computadores para Inclusão' do Governo Federal (MINISTÉRIO DAS COMUNICAÇÕES, 2012). Uma das dificuldades para o planejamento neste projeto é a configuração da rede logística, que consiste: (i) na definição de pontos de descarte dos computadores; (ii) na localização dos centros de recondicionamento, considerando os locais existentes e possibilidades de expansão; e (iii) na localização de Telecentros, que são locais para acesso digital para comunidade carente. Neste contexto, a abordagem deste projeto é a proposição de modelo matemático para configuração da rede logística do projeto 'Computadores para Inclusão' contribuindo para coleta e destinação apropriada de equipamentos de informática pós-consumo em todo território brasileiro.

PALAVRAS-CHAVE: Logística Reversa de Eletroeletrônicos, Pesquisa Operacional, Modelagem Matemática.

1.Introdução

O CLM (*Council of Logistics Management*, 2010), define logística como um processo para atender as exigências dos clientes. Através do planejamento, da implementação e do controle eficiente e eficaz, a um custo justo, do fluxo e da armazenagem, das matérias primas até o produto acabado, assim como as informações inerentes aos processos que ocorrem nesse caminho. A logística é composta por dois tipos de canais de distribuição: o direto e o reverso. O canal de distribuição direto realiza o fluxo de produtos e informações no sentido do produtor ao cliente por diversos meios, como distribuidores, lojas de atacado e varejo, etc. O canal de distribuição reverso é o fluxo oposto, a partir do consumidor final até o ponto de origem (*Council Of Logistics Management*, 2010).

A prática de reutilização de produtos e materiais não é um fenômeno recente. Há algum tempo já era realizada a reciclagem de metais, papéis e garrafas de bebidas. Esses eram os produtos cuja reciclagem trazia grandes benefícios econômicos. A crescente preocupação ambiental, as ideias sobre redução e os esforços para ciclos de vida mais econômicos ganharam maior dimensão no fim do século XX (FLEISCHMANN et al., 1997). Há também uma crescente preocupação acadêmica com este problema que é demonstrada pela quantidade de trabalhos recentes sobre temas como: Gerenciamento sustentável de redes de suprimento (*Sustainable Supply Network Management*); Gerenciamento ambiental de cadeias de suprimento (*Supply Chain Environmental Management*); Logística Verde (*Green Logistics*); Cadeias de Suprimento Verde (*Green Supply Chain*); e, Gerenciamento de Cadeias de Suprimento Verde (*Green Supply Chain Management*) (DEKKER; BLOEMHOF; MALLIDIS, 2012; LAMBERT; RIOPEL; ABDUL-KADER, 2011; SARKIS; ZHU; LAI, 2011; SBIHI; EGGLESE, 2009; SRIVASTAVA, 2007). Particularmente os trabalhos de (DEKKER; BLOEMHOF; MALLIDIS, 2012) e (SBIHI; EGGLESE, 2009) apontam direções e oportunidades de pesquisas acadêmicas sobre aplicações de Pesquisa Operacional para Logística Verde. Segundo estes mesmos autores, dentre as oportunidades de pesquisa relevantes destacam-se os problemas de configuração de rede logística.

Com o rápido desenvolvimento tecnológico houve um aumento da frequência das trocas de equipamentos eletrônicos, como computadores, celulares, impressoras, televisores, etc. O resultado deste novo padrão de consumo é a geração de um tipo de resíduo: o resíduo eletroeletrônico, também conhecido como lixo *high tech*, e-lixo ou Resíduos de Equipamentos Eletroeletrônicos (REEE).

Ao mesmo tempo em que representam um enorme risco para o meio ambiente, os REEE tem um potencial de reaproveitamento bastante lucrativo. Estudos indicam que até 2020 o mercado global de recuperação de lixo eletroeletrônico pode lucrar cerca de 21 bilhões de dólares. Para evitar o risco de contaminação pelos resíduos gerados, alguns países exportam grande parte de seu lixo eletroeletrônico para reciclagem, enquanto outros reprocessam estes produtos, atingindo taxas de mais de 80% de reciclagem. No Brasil, relatórios da Associação Brasileira da Indústria Elétrica e Eletrônica (ABIEE, 2012) mostram que os resíduos compostos por aparelhos de telefones fixos e celulares, televisores e computadores atingem o patamar de 200 mil toneladas ao ano, o que significa mais de 1 kg por habitante do país. No ritmo de crescimento atual, as estimativas são de que até 2030 serão geradas 6,6 milhões de toneladas deste tipo de lixo no Brasil..

2.Descrição do problema

O objetivo deste trabalho é o estudo da configuração da cadeia de logística reversa pós-consumo de equipamentos de informática, no estado do Rio Grande do Sul. Uma motivação para este estudo são os desafios logísticos inerentes ao descarte de resíduos sólidos urbanos, que envolvem comprometimento entre sociedade, governo e empresas privadas. No caso de equipamentos eletroeletrônicos, além dos aspectos legais previstos no Plano Nacional de Resíduos Sólidos (PNRS), as oportunidades de recuperação de valor para os produtos são

relevantes e podem aumentar sua vida útil. Particularmente no caso de equipamentos de informática, uma forma de recuperação é através dos Centros de Recondicionamento de Computadores (CRCs), que são locais onde equipamentos de informática são recondicionados e posteriormente disponibilizados para a utilização em Telecentros comunitários, escolas públicas, bibliotecas e outros projetos com impacto social nas comunidades nas quais estão inseridos. Os CRCs fazem parte do projeto 'Computadores para Inclusão' do Governo Federal. Uma das dificuldades para planejamento neste projeto é a configuração da rede logística, que consiste: (i) na definição de pontos de descarte dos computadores; (ii) na localização dos centros de recondicionamento (considerando os locais existentes e possibilidades de expansão); e (iii) na localização de Telecentros, que são locais para acesso digital para comunidade carente.

No Rio Grande do Sul temos cinco Telecentros em funcionamento, dois em Porto Alegre, um em Santa Maria, um em Caxias do Sul e um em São Leopoldo. No estado há apenas um CRC instalado, na capital Porto Alegre. Os Telecentros recebem material eletrônico da comunidade na qual estão inseridos. Em cada local onde existe um Telecentro há a possibilidade de instalarmos um CRC. Os CRC's abertos recebem material de Telecentros e de Órgãos Governamentais, cada CRC pode receber material de quantos Telecentros forem necessários para a factibilidade do modelo desde que não exceda sua capacidade máxima de recebimento de material.

Após o processamento do material pelos CRC é feito um processo de recondicionamento de material, o que é passível de aproveitamento formam Computadores que são direcionados a Telecentros, de acordo com a necessidade, e o excesso é encaminhado para escolas públicas e bibliotecas.

3.Métodos e abordagens

Foi desenvolvido um Modelo Multiperíodo com Taxas de Retorno direcionadas, onde o material que a comunidade descarta nos Telecentros é encaminhado para os CRC e processados. Os computadores reformulados são encaminhados aos Telecentros que são monitorados a fim de que exista uma quantidade mínima de computadores para que os mesmos mantenham seu funcionamento ao longo do horizonte de planejamento, e também não se exceda uma quantidade máxima de computadores que eles possuem instalações para comportar. O material não passível de reciclagem é encaminhado para empresas recicladoras. Os computadores reformulados que não forem encaminhados para os Telecentros são enviados para instituições de acesso digital gratuito.

A função objetivo pode ser dividida em três partes. A primeira visa minimizar os custos de abertura dos CRC's da rede, a segunda minimiza o funcionamento da rede, minimizando a multiplicação entre as distâncias percorridas pelo material na rede e o fator que representa o custo por quilômetro percorrido de material. Por último temos as penalizações do modelo que garante que o máximo possível de material será processado e impede que o material depois do processo de recondicionamento se acumule para a destinação à órgãos públicos ao invés de retornar a Telecentros.

4.Resultados

O CRC 1-Porto Alegre já está em funcionamento, o CRC 3-Caxias do Sul foi selecionado para a abertura. Note que, durante todo o horizonte de planejamento esses CRC's não utilizam com toda a sua capacidade, a porcentagem de ocupação de cada um deles durante os períodos está representado na Tabela 1.

Tabela 1. Porcentagem de Utilização

		Períodos											
CRC		1	2	3	4	5	6	7	8	9	10	11	12
1-Porto Alegre		65	65	66	66	66	66	66	66	66	66	77	77
		%	%	%	%	%	%	%	%	%	%	%	%
3-Caxias do Sul		42	42	44	44	44	44	44	44	44	44	33	33
		%	%	%	%	%	%	%	%	%	%	%	%

Analisando a tabela percebemos que o CRC que já estava em funcionamento, 1-Porto Alegre, não trabalha com a sua plena capacidade, mesmo assim é aberta uma nova instalação, 3-Caxias do Sul, isso ocorre porque os custos de transporte são muito altos, sendo mais vantajoso a abertura de uma nova unidade do que o transporte de material.

Esse material processado é encaminhado para Telecentros ou para instituições públicas. Os Telecentros necessitam de uma faixa entre 83% e 100% de sua capacidade para que possam operar, ou seja, uma faixa entre 25 e 30 máquinas. Nos primeiros dois períodos do horizonte de planejamento ele estão com a capacidade mínima, a partir do período três é que passam a receber o material retornável.

Na Tabela 2 temos a porcentagem de ocupação de computadores em cada Telecentro ao longo dos períodos.

Tabela 2. Porcentagem de Ocupação de Computadores nos Telecentros

		Períodos											
Telecentros		1	2	3	4	5	6	7	8	9	10	11	12
1-Porto Alegre		83%	83%	83%	100%	90%	100%	90%	83%	100%	90%	83%	100%
2-Porto Alegre		83%	83%	83%	100%	90%	100%	90%	83%	100%	90%	83%	100%
3-Caxias do Sul		83%	83%	83%	93%	83%	100%	90%	100%	93%	83%	100%	100%
4-Santa Maria		83%	83%	83%	93%	83%	100%	90%	100%	90%	83%	83%	83%
5-São Leopoldo		83%	83%	83%	100%	90%	100%	90%	83%	100%	90%	93%	83%

Os computadores reformulados que não foram encaminhados para Telecentros, cerca de 200 máquinas por mês, serão encaminhados para instituições públicas de acesso digital.

O CRC 1-Porto Alegre, recebe material de três Telecentros, 1 e 2-Porto Alegre durante todo o horizonte de planejamento e do Telecentro 5-São Leopoldo nos dois últimos períodos. O CRC 3-Caxias do Sul, recebe material de três Telecentros, os de 3-Caxias do Sul e 4-Santa Maria, mandam seu material total durante todo o horizonte de planejamento, já o 5-São Leopoldo envia material nos dez primeiros períodos.

Ocorre também o fluxo de computadores prontos. Há a necessidade de retornar computadores para os Telecentros, mantendo assim a quantidade mínima necessária em cada um deles para seu funcionamento. Na

Tabela 3 temos qual CRC fornece material para cada Telecentro no horizonte de planejamento.

Tabela 3 .Fluxo de Computadores Reformulados CRC Telecentros

		Períodos							
		3	5	7	8	9	10	11	12
1-Porto Alegre	1-Porto Alegre	8	6	1	8		1	8	124
	2-Porto Alegre	8	6	1	8		1	8	
	5-São Leopoldo	8	6	1	8		4		
3-Caxias do Sul	3-Caxias do Sul	6	8	6	1		8	3	84
	4-Santa Maria	6	8	6		1	3	3	

5. Conclusões e perspectivas futuras

O Modelo matemático implementado no estado do Rio Grande do Sul, será implementado novamente com os dados do programas Computadores para a Inclusão no Brasil todo.

Referências

ABINEE. Associação Brasileira da Indústria Elétrica e Eletrônica. Disponível em: <http://www.abinee.org.br/> Acessado em: 10 de fevereiro de 2013.

Council Of Logistics Managemen. Disponível em: http://cscmp.org/sites/default/files/user_uploads/resources/downloads/glossary-2013.pdf. Acessado em: 10 de fevereiro de 2013.

DEKKER, R.; BLOEMHOF, J.; MALLIDIS, I (2012). Operations Research for green logistics – An overview of aspects, issues, contributions and challenges. *European Journal of Operational Research*, v. 219, n. 3, p. 671–679.

E-Waste. Disponível em: <<http://svtc.org/our-work/e-waste/>>. Acessado em: 10 de fevereiro de 2013.

FLEISCHMANN, M. et al. (1997). Quantitative models for reverse logistics: A review. *European Journal of Operational Research*, v. 103, n. 1, p. 1–17.

LAMBERT, S.; RIOPEL, D.; ABDUL-KADER, W. (2011). A reverse logistics decisions conceptual framework. *Computers & Industrial Engineering*, v. 61, n. 3, p. 561–581.

SARKIS, J.; ZHU, Q.; LAI, K. (2011). An organizational theoretic review of green supply chain management literature. *International Journal of Production Economics*, v. 130, n. 1, p. 1–15.

SBIHI, A.; EGGLESE, R. W. (2009). Combinatorial optimization and Green Logistics. *Annals of Operations Research*, v. 175, n. 1, p. 159–175.

SRIVASTAVA, S. K. (2007). Green supply-chain management: A state-of-the-art literature review. *International Journal of Management Reviews*, v. 9, n. 1, p. 53–80.

Sessão 2

Análise comparativa de heurísticas para minimização de adiantamentos e atrasos em *flow shop* com tempos de *setup*.

Autores: Helio Yochihiro Fuchigami, John Lennon Damas Davi.

Um modelo de programação inteira mista para a programação da produção em *flowshop* híbrido com buffers limitados.

Autores: Pedro Luis Miranda Lugo, Rodolfo Florence Teixeira Jr.

Proposta de framework com o foco na análise e representação de dados, para pesquisas de otimização baseados em heurísticas aplicadas a problemas de sistemas de produção.

Autores: Ronaldo Castro de Oliveira, Roberto Fernandes Tavares Neto, Hercules Tadeu Asato Dantas.

Aplicação do algoritmo genético de chaves aleatórias viciadas ao problema de escalonamento de técnicos de campo.

Autores: Ricardo de Brito Damm, Débora Pretti Ronconi.

Um estudo de problemas de otimização combinatória em hospitais.

Autores: Tiago Tiburcio, Socorro Rangel.

ANÁLISE COMPARATIVA DE HEURÍSTICAS PARA MINIMIZAÇÃO DE ADIANTAMENTOS E ATRASOS EM *FLOW SHOP* COM TEMPOS DE *SETUP*

John Lennon Damas David
UFG/Campus Catalão
johnlennon_13@yahoo.com.br

Hélio Yochihiro Fuchigami
UFG/Campus Catalão
helio@catalao.ufg.br

RESUMO

Neste trabalho foram propostas seis heurísticas construtivas para programação de *flow shop* com tempos de *setup* independentes da sequência e minimização da soma dos adiantamentos e atrasos das tarefas. O algoritmo com o procedimento mais simples, que apenas sequencia pela regra de prioridade EDD (*Earliest Due Date*) e aplica uma etapa de melhoria, forneceu os melhores resultados, com mais de 45% de sucesso.

PALAVARAS-CHAVE: Programação da produção, *Flow shop*, Heurísticas, Atrasos, Adiantamentos.

1. Introdução

Este estudo aborda o problema de programação da produção em sistemas *flow shop* com tempos de *setup* explícitos, ou seja, separados dos tempos de processamento das tarefas, e independentes da sequência de execução. A medida de desempenho considerada neste problema é a minimização da soma dos adiantamentos e atrasos das tarefas. Esta é uma abordagem da filosofia *Just-in-Time*, que considera que o estoque gerado pela conclusão antecipada da tarefa é um desperdício, por ocupar espaço e representar capital parado, e que os atrasos geram punições e/ou perda de oportunidade.

Os problemas de programação da produção são alvo de incontáveis trabalhos de Pesquisa Operacional, desde a publicação de Johnson (1954), que aborda *flow shop* com duas e três máquinas. Inúmeras pesquisas têm sido realizadas na busca de métodos exatos e heurísticos para programação de *flow shop*.

Conforme Linn e Zhang (1999) e Quadt e Kuhn (2007), a maioria dos autores trata o problema de *flow shop* com medidas de desempenho relacionadas à passagem das tarefas pelo sistema, tais como minimização do *makespan* (duração total da programação) ou do tempo médio de fluxo. Recentemente, os trabalhos passaram a abordar este ambiente com medidas relacionadas ao atraso e/ou adiantamento, que devido a sua complexidade, geralmente são abordados por meio de métodos heurísticos.

Este objetivo está sendo cada vez mais alvo dos temas de estudos de problemas de programação da produção, principalmente em sistemas de produção *Just-in-Time*, que visam reduzir, entre outras medidas, os adiantamentos e atrasos.

2. Descrição do problema

O problema em estudo consiste no ambiente de *flow shop* com m máquinas e n tarefas com diferentes datas de entrega (d_j) e tempos de *setup* em cada máquina k independentes da sequência (s_{jk}). O objetivo é a minimização da soma dos adiantamentos (E_j) e atrasos (T_j) das tarefas. Na conhecida notação de três campos, o problema é representado por $Fm|s_{jk}, d_j|\sum(E_j+T_j)$. O objetivo é reduzir o intervalo de tempo entre o término da execução de cada tarefa e o seu respectivo prazo de entrega.

No *flow shop*, as tarefas devem ter um fluxo de processamento unidirecional e passar por todas as máquinas na mesma sequência. Cada tarefa tem m operações e cada operação op_{jk} de cada tarefa j é realizada na máquina k ($k = 1, \dots, m$). Os tempos de processamento de cada tarefa nas diferentes máquinas (p_{jk}) são conhecidos previamente e considerados fixos.

Quando, em cada máquina, a ordem de processamento em das tarefas for a mesma, tem-se o ambiente de produção *flow shop permutacional*, onde o número de programações possíveis para n tarefas é $n!$ No caso no *flow shop* não permutacional, o número de programações é $(n!)^m$.

No problema tratado foram consideradas as seguintes premissas: cada máquina está disponível continuamente, não havendo interrupções, e pode processar apenas uma tarefa de cada vez; cada tarefa pode ser processada por apenas uma máquina de cada vez; os tempos de processamento das tarefas nas diversas máquinas são determinados e fixos; as tarefas têm diferentes datas de entrega (prazos); os tempos de preparação (*setup*) das operações nas diversas máquinas são separados dos tempos de processamento, independentes da sequência e podem ser antecipados (iniciar antes do término do processamento da tarefa na máquina anterior); as operações nas diversas máquinas, uma vez iniciadas não devem ser interrompidas; as tarefas devem ser processadas em todas as máquinas; as datas de entrega de cada tarefa são diferentes de zero e conhecidas previamente; a execução de uma tarefa em uma máquina só pode começar após o término da sua operação no estágio anterior e desde que a máquina já esteja preparada; o problema é estático, ou seja, as datas de liberação são iguais e consideradas iguais a zero.

3. Métodos de solução

Neste estudo, foram propostos cinco heurísticas e também um método de solução aleatório (Algoritmo 6), para servir como base de comparação para os demais métodos propostos, assim como foi feito por Moursli (1999), conforme apresentado a seguir.

ALGORITMO 1

Passo 1 - Ordene as tarefas de acordo com a regra EDD (*Earliest Due Date*).

Passo 2 - Programe sequencialmente as tarefas nas máquinas e calcule os adiantamentos e atrasos.

Passo 3 (Melhoria)

Passo 3a) Se a última tarefa da última máquina estiver adiantada, atrase-a fazendo sua data de término coincidir com a sua data de entrega, ou seja, $C_j = d_j$.

Passo 3b) A partir da penúltima tarefa até a primeira, se houver adiantamento e tempo ocioso, atrase cada tarefa, reduzindo ao mínimo possível esse adiantamento, em relação ao seu prazo ou o intervalo ocioso.

ALGORITMO 2

Passo 1 - Calcule para cada tarefa j , o valor do índice I_j :

$$I_j = d_j - \left(\max\{s_{j1} + p_{j1}, s_{j2}\} + \sum_{k=2}^m p_{jk} \right)$$

Passo 2 - Ordene as tarefas de acordo com os índices I_j de forma crescente.

Passo 3 - Programe sequencialmente as tarefas nas máquinas e calcule os adiantamentos e atrasos.

Passo 4 (Melhoria)

Passo 4a) Se a última tarefa da última máquina estiver adiantada, atrase-a fazendo sua data de término coincidir com a sua data de entrega, ou seja, $C_j = d_j$.

Passo 4b) A partir da penúltima tarefa até a primeira, se houver adiantamento e tempo ocioso, atrase cada tarefa, reduzindo ao mínimo possível esse adiantamento, em relação ao seu prazo ou o intervalo ocioso.

O índice I_j objetiva priorizar as tarefas que contenham os menores valores de folga, ou seja, que apresentem a menor diferença entre a sua data de entrega e a sua carga total. A carga é calculada considerando a possibilidade de antecipação do *setup* na segunda máquina, ou seja, o maior valor entre o *setup* mais o processamento da tarefa na primeira máquina ($s_{j1} + p_{j1}$) e o *setup* na segunda máquina (s_{j2}), somando-se os tempos de processamento das máquinas 2 a n , sob a hipótese otimista de todos os *setups* nestas máquinas serem antecipados.

ALGORITMO 3

Passo 1 - Ordene as tarefas de acordo com a regra EDD.

Passo 2 - Programe sequencialmente as tarefas nas máquinas e calcule os adiantamentos e atrasos.

Passo 3 - Selecione a tarefa com o maior adiantamento e insira-a em todas as posições possíveis da sequência, mantendo a programação com o melhor valor da função objetivo.

Passo 4 - Se uma nova programação foi encontrada, repita o *Passo 3*. Caso contrário, *Pare*.

ALGORITMO 4

Passo 1 - Ordene as tarefas de acordo com a regra EDD.

Passo 2 - Programe sequencialmente as tarefas nas máquinas e calcule os adiantamentos e atrasos.

Passo 3 (Inserção com melhoria)

Passo 3a) Selecione a tarefa com o maior adiantamento e insira-a em todas as posições possíveis da sequência.

Passo 3b) Para cada posição verificada, se a última tarefa da última máquina estiver adiantada, atrase-a fazendo sua data de término coincidir com a sua data de entrega, ou seja, $C_j = d_j$.

Passo 3c) A partir da penúltima tarefa até a primeira, se houver adiantamento e tempo ocioso, atrase cada tarefa, reduzindo ao mínimo possível esse adiantamento em relação ao seu prazo ou ao intervalo ocioso.

Passo 3d) Mantenha a programação com o melhor valor da função objetivo.

Passo 4 - Se uma nova programação foi encontrada, repita o *Passo 3*. Caso contrário, *Pare*.

ALGORITMO 5

Passo 1 - Ordene as tarefas de acordo com a regra EDD.

Passo 2 - Programe sequencialmente as tarefas nas máquinas e calcule os adiantamentos e atrasos.

Passo 3 - Selecione a tarefa com o maior adiantamento e insira-a em todas as posições possíveis da sequência, mantendo a sequência com o melhor valor da função objetivo.

Passo 4 - Se uma nova programação foi encontrada, repita o *Passo 3*. Caso contrário, vá

para o *Passo 5*.

Passo 5 (Melhoria)

Passo 5a) Se a última tarefa da última máquina estiver adiantada, atrase-a fazendo sua data de término coincidir com a sua data de entrega, ou seja, $C_j = d_j$.

Passo 5b) A partir da penúltima tarefa até a primeira, se houver adiantamento e tempo ocioso, atrase cada tarefa, reduzindo ao mínimo possível esse adiantamento, em relação ao seu prazo ou o intervalo ocioso.

ALGORITMO 6

Passo 1 - Associe um número aleatório a cada tarefa e ordene-as de forma crescente.

Passo 2 - Programe sequencialmente as tarefas nas máquinas.

4. Resultados

Na experimentação computacional foram considerados problemas com $n = 6, 8, 10, 20, 50$ e 100 tarefas, $m = 2, 3, 5, 10$ e 20 máquinas, tempos de processamento no intervalo $U[1,99]$ e tempos de *setup* nos intervalos $U[1,49]$, $U[1,124]$ e $U[50,99]$. As datas de entrega foram geradas conforme Ronconi e Birgin (2012), no intervalo $[P(1-T-R/2), P(1-T+R/2)]$, onde T e R são dois parâmetros denominados fator de atraso e faixa de dispersão, respectivamente, e P é um limitante inferior para o *makespan*, em que neste trabalho foi utilizada uma adaptação para *setup* explícito daquele proposto por Taillard (1993). Os valores de T foram $0,2$ e $0,4$ e os de R $0,6$ e $1,2$. Para cada classe dos parâmetros descritos foram gerados 100 problemas, totalizando 36.000 problemas resolvidos.

Os resultados foram analisados por meio da porcentagem de sucesso (percentual de vezes em que o método forneceu a melhor solução), desvio relativo e tempo médio de computação (medido em milissegundos). A geração dos problemas e a implementação dos métodos foram feitas em linguagem Delphi em uma máquina com as seguintes configurações: processador Intel Core i5 com $2,67$ GHz de frequência, memória RAM de 14 Gb e sistema operacional Windows.

As Tabelas 4.1 e 4.2 apresentam os resultados gerais dos métodos, ordenados de acordo com o desempenho.

TABELA 4.1 – Porcentagem de sucesso dos métodos de solução

Método	Algoritmo 1	Algoritmo 4	Algoritmo 5	Algoritmo 2	Algoritmo 3	Algoritmo 6
% Sucesso	45,73	42,56	40,40	17,43	15,79	2,16

TABELA 4.2 – Desvio relativo médio dos métodos de solução

Método	Algoritmo 1	Algoritmo 4	Algoritmo 2	Algoritmo 5	Algoritmo 3	Algoritmo 6
% DRM	18,87	29,68	30,51	52,11	184,66	601,22

Os Algoritmos 1, 4 e 5 obtiveram os melhores resultados, com média acima de 45% , 42% e 40% de sucesso, respectivamente. Além disso, os Algoritmos 1 e 4 obtiveram os menores desvios relativos médios. É interessante notar que o Algoritmo 1 é o que possui o procedimento mais simples e obteve o melhor desempenho. Desconsiderando os empates, o percentual de sucesso conjunto destes três melhores algoritmos foi de $73,2\%$, o que é bastante significativo.

Pode ser verificado na Tabela 4.1 que a soma do percentual de sucesso perfaz $164,07\%$, indicando que houve um grande número de empates no melhor resultado dos métodos.

Conforme esperado, o Algoritmo 3 teve desempenho inferior aos Algoritmos 4 e 5, que

possuem a etapa de melhoria, comprovante que esse passo trouxe um efeito positivo na resolução do problema.

A ordem de superioridade dos métodos foi quase a mesma considerando a porcentagem de sucesso e o desvio relativo médio. A única exceção foi com a inversão dos Algoritmos 5 e 2, como pode ser observado nas Tabelas 4.1 e 4.2.

Os resultados do Algoritmo aleatório 6 evidenciam a viabilidade de aplicação dos cinco primeiros algoritmos, pelo seu pior desempenho, com pouco mais de 2% de sucesso e desvio relativo de mais de 600%, extremamente distante dos demais. Para efeito de comparação, dentre os cinco melhores, o Algoritmo 3 foi o que obteve o pior resultado (cerca de 15% de sucesso e 183% de desvio relativo).

Houve variabilidade nos resultados dos métodos em relação aos cenários configurados pelo fator de atraso e pela faixa de dispersão, assim como nos diferentes portes de problema (número de tarefas e de máquinas). Já os intervalos de *setup* considerado não influenciaram significativamente.

O custo computacional mostrou-se desprezível, pois o tempo máximo de CPU consumido por um problema foi de 63 ms.

5. Conclusões e perspectivas futuras

Os resultados da experimentação computacional elucidaram a viabilidade de aplicação das heurísticas apresentadas na solução do problema tratado. Uma vez que o tempo computacional não constitui fator restritivo, uma proposta viável seria a criação de uma heurística híbrida, que verifica os resultados dos três melhores métodos – Algoritmos 1, 4 e 5, e forneça a melhor solução encontrada, atingindo assim mais de 73% de sucesso.

Para futuros trabalhos, sugere-se a consideração de novas restrições, como tarefas com prazos de entrega comuns ou então janelas de entrega, e também a inclusão de outras medidas na função objetivo, como tempo médio de fluxo ou duração total da programação. Além disso, as soluções de problemas de pequeno porte poderiam ser comparados com a fornecida por algum método de solução exata.

Referências

- Johnson, S. M.** (1954). Optimal two and three-stage production schedules with setup times included. *Naval Research Logistics Quarterly*, 1, 61-68.
- Linn, R.; Zhang, W.** (1999). Hybrid flow shop scheduling: a survey. *Computers and Industrial Engineering*, 37, 57-61.
- Moursli, O.** (1999). *Scheduling the hybrid flowshop: branch and bound algorithms*. Tese (Doutorado) – Faculte des Sciences Economiques, Sociales et Politiques, Université Catholique de Louvain, Louvain. Bélgica.
- Quadt, D.; Kuhn, H.** (2007). A taxonomy of flexible flow line scheduling procedures. *European Journal of Operational Research*, 178, 686-698.
- Ronconi, D. P.; Birgin, E. G.** (2012). Mixed-integer programming models for flow shop scheduling problems minimizing the total earliness and tardiness. In: *Just-in-Time Systems*, Y.A. Ríos-Solís and R.Z. Ríos-Mercado (Eds.), Springer Series on Optimization and Its Applications, P.M. Pardalos and Ding-Zhu Du (Series eds.).
- Taillard, E.** (1993). Benchmarcks for basic scheduling problems. *European Journal of Operational Research*, 64, 278-285.

UM MODELO DE PROGRAMAÇÃO INTEIRA MISTA PARA A PROGRAMAÇÃO DA PRODUÇÃO EM *FLOWSHOP* HÍBRIDO COM *BUFFERS* LIMITADOS

Pedro Luis Miranda Lugo

Universidade Federal de São Carlos – Departamento de Engenharia de Produção
pmiranda@ufscar.br

Rodolfo Florence Teixeira Jr

Universidade Federal de São Carlos – *Campus* Sorocaba – Departamento de Engenharia de
Produção
rodolfo.florence@ufscar.br

RESUMO

Esta pesquisa aborda o problema de programação da produção em sistemas *Flowshop* híbrido. Algumas restrições comumente encontradas em sistemas de produção reais, como máquinas paralelas não relacionadas, *buffers* limitados, tempos de preparação dependentes da sequência (antecipatórios e não antecipatórios), elegibilidade de máquinas, tempos de transporte e tempos de liberação das máquinas, são consideradas. O critério de otimização é o *makespan*, cuja minimização está diretamente relacionada com a utilização eficiente dos recursos de produção. Um modelo de programação inteira mista é proposto e resolvido através do *solver* comercial CPLEX. Os resultados da avaliação computacional indicam que o modelo é viável somente para resolver instâncias de até nove tarefas e cinco estações.

PALAVRAS-CHAVE: *Flowshop* híbrido, programação da produção, *buffers* limitados.

1. Introdução

Em um *Flowshop* um conjunto de tarefas deve ser processado através de múltiplas estações na mesma ordem, desde a primeira até a última estação, e cada estação tem unicamente uma máquina. Porém, como destacado por Ribas, Leisten e Framiñan (2010), em algumas indústrias a necessidade de incrementar ou equilibrar a capacidade das estações tem levado à duplicação de máquinas. Em outras, a crescente demanda de produtos customizados tem desencadeado a necessidade de comprar máquinas adicionais para algumas estações com o objetivo de dedicá-las à produção deste produtos. Em qualquer caso, a aquisição das novas máquinas não implica a substituição ou eliminação do equipamento existente, o que leva a coexistência de várias máquinas, possivelmente diferentes, em várias estações do processo de produção.

Do ponto de vista teórico, esta nova configuração de produção, conhecida como *Flowshop* Híbrido (HFS), pode ser considerada como a combinação de dois problemas particulares de programação da produção: o problema de máquinas paralelas e o problema do *Flowshop*. No problema de máquinas paralelas a decisão-chave é a alocação de tarefas a máquinas, enquanto no *Flowshop* a decisão-chave é o sequenciamento das tarefas através do chão de fábrica. Portanto, no HFS as decisões-chave são designar e programar as tarefas às máquinas em cada estação, isto é, determinar a ordem na qual as tarefas serão processadas nas diferentes máquinas de cada estação, com o objetivo de minimizar um ou vários critérios determinados.

O estudo deste problema é de grande relevância prática, dado que este tipo de configuração produtiva pode ser encontrado em diversos setores de manufatura, como na indústria petroquímica (Deal, Yang, & Hallquist, 1994), indústria automotiva (Agnētis et al., 1997), indústria eletrônica (Liu & Chang, 2000), indústria têxtil (Grabowski & Pempera, 2000),

manufatura de etiquetas autoadesivas (Lin & Liao, 2003), indústria de revestimentos cerâmicos (Ruiz, Şerifoğlu, & Urlings, 2008), entre outras.

2. Descrição do problema

Formalmente, um conjunto N de tarefas, $N = \{1, \dots, n\}$, deve ser processado em um conjunto M de estações, $M = \{1, \dots, m\}$. Em cada estação i , $i \in M$, há um conjunto $M_i = \{1, \dots, m_i\}$ de processadores paralelos não relacionados. Os processadores podem ser máquinas ou *buffers*, sendo os últimos considerados máquinas especiais com tempos de processamento e preparação iguais a zero. E_{ij} denota o conjunto de processadores capazes de processar a tarefa j , $j \in N$, na estação i . O parâmetro rm_{il} indica o tempo de liberação do processador l , $l \in M_i$, na estação i . Cada tarefa j deve ser processada por exatamente um dos processadores paralelos em cada estação i . O tempo de processamento da tarefa j no processador l da estação i é denotado como p_{ij} . O parâmetro S_{iljk} denota o tempo de preparação entre as tarefas j e k , $k \in N$, no processador l da estação i . Adicionalmente, o parâmetro binário A_{iljk} indica se a preparação entre as tarefas j e k no processador l da estação i é antecipatória ou não antecipatória. Operações de transporte entre estações devem ser feitas antes de iniciar o processamento de qualquer tarefa, assim t_{iq} indica o tempo de transporte entre as estações i e q , $q \in M$. O critério de otimização é o C_{\max} , tal que $C_{\max} = \max_{j \in N} \{C_j\}$ e C_j indica o tempo de finalização da tarefa j na última estação.

3. Modelo Matemático

Nesta seção apresenta-se um modelo de programação inteira mista, baseado no modelo de Ruiz et al. (2008), para o HFS abordado nesta pesquisa. O modelo é testado em um *benchmark* de 1.728 instâncias e os resultados são utilizados para determinar a viabilidade e desempenho do modelo proposto como ferramenta de solução.

Índices e Conjuntos:

j, k, h = tarefas;

i, q = estações;

l = processadores;

N = conjunto de tarefas;

M = conjunto de estações;

M_i = conjunto de processadores paralelos na estação i ;

E_{ij} = conjunto de processadores elegíveis para a tarefa j na estação i ;

G_{il} = conjunto de tarefas que podem ser processadas no processador l da estação i ;

Parâmetros:

r_{il} = data de liberação do processador l da estação i ;

p_{ij} = tempo de processamento da tarefa j no processador l da estação i ;

S_{iljk} = tempo de setup entre as tarefas j e k no processador l da estação i ;

t_{iq} = tempo de transporte entre as estações i e q ;

$$A_{ijk} = \begin{cases} 1, & \text{se o setup entre as tarefas } j \text{ e } k \text{ no processador } l \text{ da estação } i \text{ é antecipatório} \\ 0, & \text{caso contrário} \end{cases}$$

$FS(LS)$ = Primeira (última) estação do sistema.

Variáveis de Decisão:

$$X_{ijk} = \begin{cases} 1, & \text{se a tarefa } j \text{ precede a tarefa } k \text{ no processador } l \text{ da estação } i \\ 0, & \text{caso contrário} \end{cases}$$

C_{ij} = tempo de finalização da tarefa j na estação i ;

d_{ij} = tempo de saída da tarefa j da estação i ;

C_{\max} = *Makespan*.

$$\min C_{\max} \quad (3.1)$$

$$\sum_{\substack{j \in \{N,0\} \\ j \neq k}} \sum_{l \in E_{ij} \cap E_{ik}} X_{ijk} = 1, \quad k \in N, i \in M \quad (3.2)$$

$$\sum_{\substack{j \in N \\ j \neq k}} \sum_{l \in E_{ij} \cap E_{ik}} X_{ilk} \leq 1, \quad k \in N, i \in M \quad (3.3)$$

$$\sum_{\substack{h \in \{G_{il},0\} \\ h \neq j, h \neq k}} X_{ilhj} \geq X_{ijk}, \quad j, k \in N, j \neq k, i \in M, l \in E_{ij} \cap E_{ik} \quad (3.4)$$

$$\sum_{l \in E_{ij} \cap E_{ik}} (X_{ijk} + X_{ilk}) \leq 1, \quad j \in N, k = j+1, \dots, n, i \in M \quad (3.5)$$

$$\sum_{k \in G_{il}} X_{i0k} \leq 1, \quad i \in M, l \in M_i \quad (3.6)$$

$$C_{i0} = 0, \quad i \in M \quad (3.7)$$

$$C_{ik} + B(1 - X_{ijk}) \geq rm_{il} + (1 - A_{ijk})S_{ijk} + p_{ilk}, \quad k \in N, i \in M, l \in E_{ik}, \\ j \in \{G_{il}, 0\}, j \neq k \quad (3.8)$$

$$C_{ik} + B(1 - X_{ijk}) \geq d_{ij} + S_{ijk} + p_{ilk}, \quad k \in N, i \in M, l \in E_{ik}, \\ j \in \{G_{il}, 0\}, j \neq k \quad (3.9)$$

$$C_{ik} + B(1 - X_{ijk}) \geq C_{i-1,k} + t_{i-1,i} + (1 - A_{ijk})S_{ijk} + p_{ilk}, \quad k \in N, i \in \{M \setminus FS\}, \\ l \in E_{ik}, j \in \{G_{il}, 0\}, j \neq k \quad (3.10)$$

$$d_{i-1,k} = C_{ik} - \sum_{\substack{j \in \{N,0\} \\ j \neq k}} \sum_{l \in E_{ij} \cap E_{ik}} X_{ijk} \cdot p_{ilk} - \sum_{\substack{j \in \{N,0\} \\ j \neq k}} \sum_{l \in E_{ij} \cap E_{ik}} X_{ijk} (1 - A_{ijk})S_{ijk} - t_{i-1,i}, \quad (3.11)$$

$$k \in N, i \in \{M \setminus FS\}$$

$$d_{ik} \geq C_{ik}, \quad k \in N, i \in \{M \setminus LS\} \quad (3.12)$$

$$d_{ik} = C_{ik}, \quad k \in N, i = LS \quad (3.13)$$

$$C_{\max} \geq C_{ik}, \quad k \in N, i = LS \quad (3.14)$$

$$X_{ijk} \in \{0, 1\}, \quad j \in \{N, 0\}, k \in N, j \neq k, i \in M, l \in E_{ij} \cap E_{ik} \quad (3.15)$$

$$C_{ij}, d_{ij} \geq 0, \quad j \in N, i \in M \quad (3.16)$$

A função objetivo (3.1) visa à minimização do *makespan*. As restrições (3.2) asseguram que cada tarefa k deve ser precedida por uma e somente uma tarefa j em apenas um processador l em cada estação i . As restrições (3.3) indicam que cada tarefa k deve ter, no máximo, um sucessor j . O conjunto de restrições (3.4) garante que se a tarefa j é predecessora de alguma tarefa k em algum processador l da estação i , então a tarefa j deve ter um predecessor h no mesmo processador l . O restrições (3.5) evitam a ocorrência de precedência cruzada. As restrições (3.6) indicam que a tarefa fictícia zero pode ser predecessora de, no máximo, uma

tarefa k em cada processador l em cada estação i . O conjunto de restrições (3.7) garante que a tarefa fictícia zero seja finalizada no tempo zero em cada estação $i \in M$. As restrições (3.8) indicam que se a tarefa k é designada ao processador l da estação i , então seu processamento não pode começar até que o processador l seja liberado pela programação de produção anterior. O conjunto de restrições (3.9) indica que o processamento da tarefa k no processador l da estação i não pode começar até que a tarefa predecessora j tenha saído da estação i e a correspondente preparação do processador l tenha sido feita. As restrições (3.10) garantem que cada tarefa k seja processada subsequentemente em cada estação i . As restrições (3.11) calculam o tempo de saída da tarefa k da estação i . As restrições (3.12) indicam que o tempo de saída da tarefa k da estação i não pode ser menor que seu tempo de finalização nesta estação. O conjunto de restrições (3.13) garante que a tarefa k sai do sistema assim que finalizar seu processamento na última estação. As restrições (3.14) define o *makespan* e, finalmente, as restrições (3.15) e (3.16) definem o domínio das variáveis de decisão.

4. Avaliação Computacional

O modelo de programação inteira mista da seção 3 foi implementado no *GAMS* e um conjunto de 1.728 instâncias foi resolvido utilizando o *CPLEX 12* em um *notebook* com processador Intel Core i5 2.67 GHz com 4 Gigabytes de memória RAM. Devido à quantidade de instâncias, um tempo limite de execução de 3.600 segundo foi imposto para cada uma delas.

Na Tabela 1 são apresentados os resultados consolidados do modelo. Cada célula representa a média para cada tamanho de instância. Assim, são apresentadas a percentagem de instâncias cuja solução ótima foi encontrada (%Ótimo); a percentagem de instâncias para as quais uma solução inteira factível foi encontrada (%IntSol) e a percentagem de instâncias para as quais nenhuma solução inteira factível foi encontrada (%NãoSol).

n		m		
		3	5	7
5	%Ótimo	100,00	100,00	100,00
	%IntSol	0,00	0,00	0,00
	%NãoSol	0,00	0,00	0,00
7	%Ótimo	100,00	97,92	86,81
	%IntSol	0,00	2,08	9,72
	%NãoSol	0,00	0,00	3,47
9	%Ótimo	94,44	52,08	22,92
	%IntSol	5,56	46,53	70,83
	%NãoSol	0,00	1,39	6,25
11	%Ótimo	40,97	13,19	1,39
	%IntSol	59,03	81,94	79,17
	%NãoSol	0,00	4,86	19,44

Tabela 1: Resultados consolidados do modelo de programação inteira mista

O *gap* médio observado é relativamente alto, entre 22,25% e 56,67%, e tende a ser maior com o aumento do número de tarefas e estações. Este comportamento era esperado porque, como mostrado pela Tabela 1, quanto maior número de tarefas e estações, maior a dificuldade teve o *CPLEX* para resolver as instâncias.

A Figura 1 ilustra o comportamento do tempo médio utilizado pelo *solver* em função do número de tarefas e estações. Note que, para instâncias além de nove tarefas e cinco estações, atingir uma solução ótima é inviável do ponto de vista prático, dado a quantidade de tempo requerida.

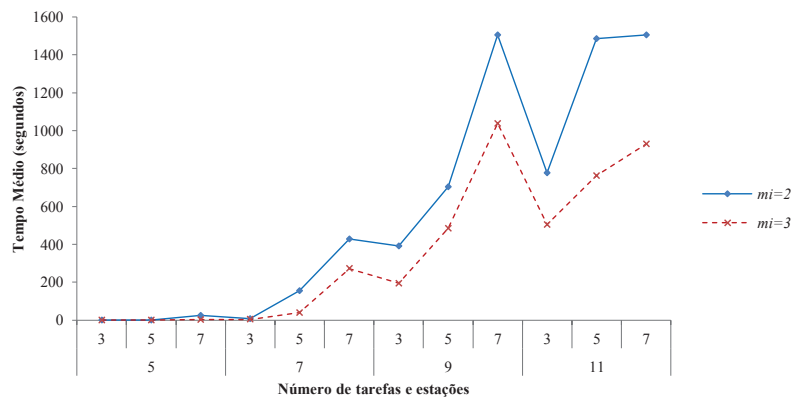


Figura 1: Tempo Médio utilizado pelo solver para encontrar uma solução ótima

5. Conclusões e perspectivas futuras

Nesta pesquisa um modelo de programação inteira mista foi apresentado para resolver o problema de programação da produção em HFS com máquinas paralelas não relacionadas, tempos de preparação dependentes da sequência (antecipatórios e não antecipatórios), tempos de transporte entre estações, *buffers* limitados e elegibilidade de máquinas, visando minimizar o *makespan*. O modelo foi testado em um conjunto de 1.728 instâncias com tempo limite de 3.600 segundos e os resultados encontrados indicam que o modelo é computacionalmente viável para instância de até nove tarefas e cinco estações. Em termos gerais, o CPLEX encontrou ao menos uma solução ótima para 1.166 instâncias. Para 511 instâncias, encontrou ao menos uma solução inteira factível e, finalmente, para as 51 instâncias restantes nenhuma solução inteira factível foi encontrada depois de 3.600 segundos.

Como pesquisa futura, pretende-se implementar e utilizar métodos meta-heurísticos para resolver o problema estudado. Estes métodos precisam maior tempo computacional que outros métodos mais simples, como as heurísticas, mas em contrapartida espera-se encontrar soluções de melhor qualidade para instâncias de qualquer tamanho.

Referências

- Agnetis, A., Pacifici, A., Rossi, F., Lucertini, M., Nicoletti, S., Nicolò, F., ... Pesaro, E. (1997). Scheduling of flexible flow lines in an automobile assembly plant. *European Journal of Operational Research*, 97(2), 348–362. doi:10.1016/S0377-2217(96)00203-2
- Deal, D. E., Yang, T., & Hallquist, S. (1994). Job scheduling in petrochemical production: Two-stage processing with finite intermediate storage. *Computers and Chemical Engineering*, 18(4), 333–344.
- Grabowski, J., & Pempera, J. (2000). Sequencing of jobs in some production system. *European Journal of Operational Research*, 125(3), 535–550. doi:10.1016/S0377-2217(99)00224-6
- Lin, H.-T., & Liao, C.-J. (2003). A case study in a two-stage hybrid flow shop with setup time and dedicated machines. *International Journal of Production Economics*, 86(2), 133–143. doi:10.1016/S0925-5273(03)00011-2
- Liu, C.-Y., & Chang, S.-C. (2000). Scheduling flexible flow shops with sequence-dependent setup effects. *IEEE Transactions on Robotics and Automation*, 16(4), 408–419. doi:10.1109/70.864235
- Ribas, I., Leisten, R., & Framiñan, J. M. (2010). Review and classification of hybrid flow shop scheduling problems from a production system and a solutions procedure perspective. *Computers & Operations Research*, 37(8), 1439–1454. doi:10.1016/j.cor.2009.11.001
- Ruiz, R., Şerifoğlu, F., & Urlings, T. (2008). Modeling realistic hybrid flexible flowshop scheduling problems. *Computers & Operations Research*, 35(4), 1151–1175. doi:10.1016/j.cor.2006.07.014

PROPOSTA DE FRAMEWORK COM O FOCO NA ANÁLISE E REPRESENTAÇÃO DE DADOS, PARA PESQUISAS DE OTIMIZAÇÃO BASEADOS EM HEURÍSTICAS APLICADAS A PROBLEMAS DE SISTEMAS DE PRODUÇÃO

Ronaldo Castro de Oliveira

Universidade Federal de Uberlândia – UFU
ronaldooliveira@facom.ufu.br

Roberto Fernandes Tavares Neto

Universidade Federal de São Carlos - UFSCar
tavares@dep.ufscar.br

Hercules Tadeu Asato Dantas

Universidade Federal de São Carlos - UFSCar
dantas.hercules@gmail.com

RESUMO

Uma das maiores dificuldades quando se avalia uma pesquisa que utiliza métodos de otimização na solução de problemas de sistemas de produção é a falta de padronizações aplicadas na construção da solução, o que envolve desde a especificação dos dados de entrada, os modelos e padrões de projeto utilizados na codificação, a especificação dos parâmetros de configuração e na análise e representação dos resultados de saída apresentados. Esta falta de padronização dificulta a comparação de pesquisas correlatas que tratam do mesmo assunto. Este artigo apresenta uma proposta de classificação dos resultados de saída destas pesquisas que usam otimizações baseadas em heurísticas para problemas de sistemas de produção. Esta classificação fornecerá uma ampla base para levantar e definir padrões de análise e representação dos dados de saída, de forma a facilitar a comparação e avaliação de pesquisas nesta área.

PALAVRAS-CHAVE: Framework, Otimização, Heurísticas, Problemas de sistemas de produção.

1.Introdução

Realizar pesquisas que tratam da solução de problemas de sistemas de produção em muitos casos é um grande desafio. O fluxo natural na busca das soluções para estes problemas costuma envolver a implementação de uma solução algorítmica e sua validação, seja através da comparação com resultados obtidos na literatura ou através da aplicação de outras técnicas. Um grande desafio deste processo é que os problemas de sistemas de produção são de altíssima complexidade e muitas vezes classificados como "NP-hard", o que na prática significa que uma solução algorítmica exata levaria um tempo computacional extremamente grande para conseguir chegar a uma solução ótima. Neste sentido, muitas pesquisas acabam usando heurísticas e meta-heurísticas para solucionar estes problemas. Uma heurística consegue chegar a valores aceitáveis, muitas vezes os próprios valores ótimos obtidos pelos métodos exatos, com a vantagem de usar um tempo computacional muito menor (Blum e Roli, 2003).

Mesmo com o crescimento do uso de heurísticas na solução dos problemas de sistemas de produção, cada pesquisador acaba construindo uma solução distinta, muitas vezes desenvolvida do zero. Na maioria dos vezes o desenvolvimento está relacionado com o conhecimento de cada pesquisador que muitas vezes não utiliza de padrões de desenvolvimento, sejam eles, padrões de definição dos dados de entrada, padrões de especificação da solução, padrões de configuração dos parâmetros de execução e padrões de análise e representação dos resultados. A Figura 1 apresenta a estrutura do framework proposto.

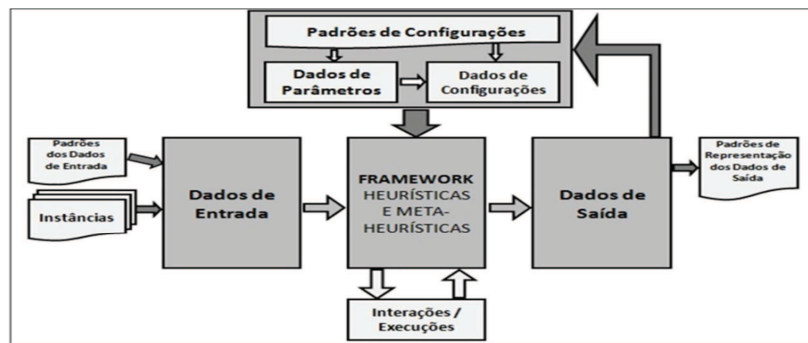


Figura 1 - Estrutura do framework proposto (elaborado pelos autores)

O maior problema é que todo este processo muitas vezes dificulta a comparação destas pesquisas. O objetivo deste trabalho é propor a especificação de um framework que forneça para os pesquisadores que trabalham nesta área, um conjunto de padrões que possam ser utilizados na especificação, construção, teste e armazenamento de informações das execuções realizadas, de forma a facilitar a comparação com outras pesquisas. Neste sentido tem como proposta uma classificação de pesquisas realizadas na área, de forma a especificar padrões de análise e representação de dados de saída destas pesquisas. Isto é necessário porque vários trabalhos publicados usam somente tabelas comparativas dos dados de saída, nem mesmo se preocupando em utilizar gráficos que poderia fornecer uma análise mais clara ao leitor. Mesmo aqueles trabalhos que utilizam gráficos, não seguem padrões especificado. As demais etapas apresentadas na Figura 1 serão desenvolvidas em trabalhos futuros.

Este artigo contempla cinco seções. A primeira seção é a Introdução (Seção 1) em que se contextualiza a pesquisa e delimita os objetivos. A seção 2 define os tipos de problemas relacionados com sistemas de produção e apresenta as principais heurísticas utilizadas. A seção 3 apresenta a classificação proposta no presente trabalho, e na seção 4 efetua um teste de conceito da classificação proposta com cinco artigos selecionados. A seção 5 apresenta as conclusões e perspectivas para trabalhos futuros. Por fim são apresentadas as referências utilizadas no presente trabalho.

2. Solução para problemas em sistemas de produção

Os aspectos do planejamento e controle da produção mudam ao longo do tempo. Slack et al. (1997), ilustra esta natureza de planejamento estabelecendo uma visão de curto, médio e de longo prazo. Segundo Arenales et al (2007) os problemas de sistemas e produção podem ser definidos como sendo problemas de produção e problemas de logística, e o relacionamento destes com o horizonte de planejamento de longo e curto prazo são apresentados na Tabela 1.

Tabela 1 - Problemas de sistemas de produção (Slack et al, 1997 e Arenales et al, 2007)

	Longo Prazo	Curto Prazo
Problemas de Produção	Dimensionamento de Lotes (<i>Lot Sizing</i>)	Programação de Tarefas (<i>Scheduling</i>)
Problemas de Logística	Localização de Facilidades (<i>Facility Location</i>)	Roteamento de Veículos (<i>Vehicle Routing</i>)

O dimensionamento do lote de produção especifica a quantidade que será produzida, de acordo com a capacidade de produção das instalações, de forma que a demanda dos clientes sejam atendidas onde a soma dos custos de produção e estocagem sejam minimizadas (Arenales et al., 2007). Já o problema de programação de tarefas consiste na alocação de recursos para tarefas, respeitando as restrições impostas e otimizando um ou mais critérios, dentre eles o *makespan* (instante de término da última tarefa). Os problemas de sequenciamento ou programação de tarefas em sistemas de produção são, tradicionalmente, classificados em função do fluxo das operações nas máquinas, conforme Nagano et al. (2004).

O problema de localização de facilidades está focado na localização de fábricas, armazéns e centros de distribuição, preocupando-se em estudar uma área específica a partir das

unidades de distribuição de produtos ou de prestação de serviços, com o objetivo de determinar a quantidade e a localização ideal destas. Por último o problema de roteamento de veículos envolvem projetos de rotas de entrega e/ou coleta de custo mínimo, por intermédio de uma frota de veículos, partindo de um ou mais depósitos para um número de clientes, sujeito a restrições adicionais que dependem de cada problema estudado. (Arenales et al., 2007).

Os problemas de um sistema de produção podem ser otimizados utilizando várias técnicas combinatórias diferentes. Para um grande conjunto desses problemas, é inviável o uso de algoritmos exatos de otimização combinatória que possam chegar a uma solução ótima usando um conjunto viável de recursos computacionais, principalmente pela característica não polinomial dos algoritmos. Devido a isso, tais algoritmos são apropriados somente para problemas de tamanho reduzido ou características especiais. Por esta razão normalmente utilizam-se algoritmos baseados em heurísticas para encontrar uma solução aceitável (não necessariamente a solução ótima). Estas soluções aproximadas tornam-se viáveis em função do tamanho do problema que se deseja resolver. Dentre as diferentes abordagens existem inúmeras meta-heurísticas foram desenvolvidas na busca por soluções, dentre elas: Simulated Annealing (SA), TABU Search, GRASP, Genetic Algorithmics (GA) e Ant Colony Optimization (ACO) (Blum e Roli, 2003).

3. Classificação proposta

A partir do levantamento teórico, chegou-se a classificação das análises e representações dos dados de saída. Esta classificação está dividida em quatro grandes caracterizações que são:

- Caracterização do processamento: define o tipo do problema de sistema de produção, o fluxo do processo da produção (Maccarthy, B. L.; Fernandes, F. C. F., 2000) e qual unidade de grandeza utilizada no trabalho;
- Caracterização da meta-heurística utilizada: define a meta-heurística utilizada, os dados de comparação de referência (benchmark) e o pré-processamento utilizado;
- Caracterização da representação dos dados: define como a análise de dados foi realizada e quais foram as representações gráficas utilizadas. Bruni (2007) e Anderson, Sweeney e Williams (2007) apresentam os principais conceitos estatísticos utilizados;
- Caracterização do tempo de execução: especifica como os autores do trabalho limitaram as execuções do(s) algoritmo(s) heurísticos na solução do problema.

Cada caracterização possui as suas subdivisões de classificação que são apresentadas de forma sumarizadas na Tabela 2, visualizando toda a classificação proposta.

4. Aplicação da classificação proposta e análise

Para testar a classificação proposta foram selecionados cinco artigos de problemas de sistemas de produção diferentes, usando heurísticas distintas (exemplo: para Arenales et al (2007) a classificação fica - T1/U\$/U#/H4/BE/PG/A3-A6/G2/E#). A Tabela 3 apresenta o teste da aplicação da classificação proposta com estes cinco artigos.

Pode-se observar alguns padrões entre os cinco artigos analisados. Quatro dos cinco artigos trabalham com unidade de grandeza numérica. Os artigos de programação de tarefas usam a unidade de grandeza tempo. Três dos cinco artigos usaram valores de comparação externo (benchmark), vindos de outros trabalhos, e os outros dois usaram valores de referência calculados de forma aleatória na distribuição normal. Em relação ao pré-processamento três dos cinco artigos trabalharam com valores absolutos e os outros dois com diferenças dos valores de referência. Ao estudar o conjunto de artigos pode-se verificar que os mesmos não fazem grandes análises de dados, limitando-se na maioria deles a médias simples ou no máximo o desvio médio dos valores de referência. A maioria dos artigos apresenta os resultados em gráficos de linha simples, e somente um deles apresentou um gráfico *boxplot*. Todos os artigos analisados executaram os seus experimentos limitados pelo número máximo de ciclos de execução.

Tabela 2 – Sumarização da Classificação Proposta (Elaborado pelos autores.)

Caracterização do Processamento		
Sigla	Descrição	Subcaracterística
T	Tipo de Problema	1 - Dimensionamento de Lotes de Produção (Lot Sizing Problem) 2 - Programação da Produção (Scheduling Problem) 3 - Localização de Facilidades (Location Facility Problem) 4 - Roteamento de Veículos (Vehicle Routing Problem)
F	Fluxo do Processo	1 - Estágio único com apenas uma máquina 2 - Estágio único com máquina idênticas em paralelo 3 - Estágio único com máquinas não idênticas em paralelo 4 - Processo multiestágios unidirecional (flow shop) 5 - Processo multiestágios unidirecional, que permite que estágios sejam pulados (overflow) 6 - Processo multiestágios unidirecional, com máquinas idênticas em paralelo 7 - Processo multiestágios unidirecional, com máquinas idênticas em paralelo, com overflow 8 - Processo multiestágios unidirecional, com máquinas idênticas em paralelo 9 - Processo multiestágios unidirecional, com máquinas não idênticas em paralelo, com overflow 10 - Processo multiestágios multidirecional, e.g., job shop 11 - Processo multiestágios multidirecional, com máquinas idênticas em paralelo 12 - Processo multiestágios multidirecional, com máquinas não idênticas em paralelo
U	Unidade de Grandeza	T - Tempo: 1 - Processamento (Makespan, Tempo de Finalização - $\sum C_i$, Temp. de Fluxo - $\sum F_i$ ou \bar{F}) 3 - Lateness ($L_i = C_i - D_i$; onde $-\infty < L_i < \infty$; C_i - data de finalização, D_i - data de entrega) 2 - Tradiness - Atraso ($T_i = \max \{L_i, 0\}$) 4 - Earliness - Adiantamento ($E_i = \max \{-L_i, 0\}$) \$ - Valores monetários # - Numérico M - Utilização de Multi-objetivo para análise S { U1, U2, ..., Un } - Soma ponderada, ex: S { T1, T2 }; onde U1, U2, ..., Un são elementos da unidade de grandeza P { U1, U2, ..., Un } - Pareto, ex: P { \$, T2 }; onde U1, U2, ..., Un são elementos da unidade de grandeza
Caracterização da Meta-heurística Utilizada		
H	Heurística	1 - SA - Simulated Annealing 2 - TABU 3 - GRASP 4 - AG - Algoritmo Genéticos 5 - Iterated Greedy 6 - ACO - Ant Colony Optimization 7 - BCO - Bees Colony Optimization 8 - VNS - Variable Neighborhood Search 9 - Estimation of distribution algorithm
B	Benchmark de Comparação dos Dados	E - Externo R - Real A - Aleatório: 1 - Distribuição Binomial 2 - Distribuição de Poisson 3 - Distribuição Normal 4 - Distribuição Uniforme
P	Pré-processamento	G - GAP (diferença entre os dados encontrados) A - Absoluto D - Distância Euclidiana
Caracterização da Representação dos Dados		
A	Análise dos dados	1 - Mínimo 2 - Máximo 3 - Média 4 - Mediana 5 - Moda 6 - Desvio Médio 7 - Variância 8 - Desvio Padrão 9 - ANOVA (Analysis of Variance) 10 - Planejamento de Experimentos 11 - N-vezes que atingiu a melhor solução conhecida 12 - Teste Paramétrico: a - Hipótese b - Bilateral c - Unilateral d - Igualdades e - Diferenças f - Análise Post-Hoc 13 - Teste Não Paramétrico: a - Qui-quadrático b - Sinais c - Wilcoxon d - Mann-Whitney e - Mediana f - Kruskal-Wallis g - Teste t 14 - % de Falha
G	Gráficos	1 - Barras ou Colunas 2 - Gráfico de Linha 3 - Diagrama Retangular 4 - Gráfico Circular ou de Setores 5 - Gráfico Polar 6 - Histograma 7 - Gráfico de Dispersão 8 - Boxplot
Caracterização da Representação dos Dados		
E	Execução	# - Limitado por número máximo de ciclos V - Limitado por valor específico T - Limitado por tempo

Tabela 3 – Teste da classificação proposta

N	Referência	Classificação
1	(Toledo et al., 2013)	T1/U\$-U#/H4/BE/PG/A3-A6/G2/E#
2	(Jin et al., 2012)	T4/U#/H2/BE/PA/A1-A6/E#
3	(Pasandideh e Niaki, 2010)	T3/U\$-U#/H4/BA3/PA/A1/G2/E#
4	(Tavares Neto; Godinho Filho, 2011)	T2/F4/UT1-U\$/H5/BA3/PG/A3/G8/E#
5	(Witkowski et al., 2010)	T2/F12/UT1-U#/H1-H3/BE/PA/A1-A2-A3/E#

5. Conclusões e perspectivas para trabalhos futuros

Este trabalho apresentou uma classificação dos dados de saída que faz parte de uma pesquisa maior que é a proposta de framework para pesquisas que usam otimizações baseada em heurísticas para solução de problemas de sistemas de produção. A proposta da classificação dos dados de saída tem como objetivo levantar e especificar padrões de análise e representação de resultados de pesquisas nesta área, fornecendo um passo a passo para categorização das publicações relacionadas. Um teste da classificação foi realizado com uma amostra heterogênea de artigos recentes, comprovando a viabilidade de uso, mostrando a facilidade de análise e comparação das características, e possibilitando a especificação de padrões a serem utilizados.

Como perspectivas de trabalhos futuros, espera-se a aplicação desta classificação em um conjunto mais amplo de trabalhos nesta área, que utilizam as principais heurísticas citadas, de forma que se possam estabelecer os padrões de análise e representação dos resultados. Espera-se também que estes padrões estabelecidos possam ser utilizados para construção de novos padrões de configuração, para fornecer uma base para a especificação do framework de execução de heurísticas na solução problemas de sistemas de produção.

Referências

- Anderson, D.R., Sweeney, D.J., Williams, T.A.** (2007). *Estatística Aplicada à Administração e Economia*. 2ª Edição, São Paulo: Cengage Learning.
- Arenales, M.; Armentano, V.; Morabito, R.** (2007). *Pesquisa operacional: para cursos de engenharia*. Campus.
- Blum, C.; Roli, A.** (2003). *Metaheuristics in Combinatorial Optimization: Overview and Conceptual Comparison*. ACM Computing Surveys, v. 35, n. 3, p. 268–308.
- Bruni, A.L.** (2007). *Estatística Aplicada à Gestão Empresarial*. 2ª Edição, São Paulo: Ed. Atlas.
- Jin, J.; Crainic, T. G.; Lokketangen, A.** (2012). *A parallel multi-neighborhood cooperative tabu search for capacitated vehicle routing problems*. European Journal of Operational Research, v. 222, n. 3, p. 441–451.
- Maccarthy, B. L.; Fernandes, F. C. F.** (2000). *A multi-dimensional classification of production systems for the design and selection of production planning and control systems*. Production Planning & Control, v. 11, n. 5, p. 481–496.
- Nagano, M. S.; Antonio, L.; Lorena, N.; Inpe, L. A. C.** (2004). *Programação da Produção Flow Shop Permutacional com Minimização do Tempo Médio de Fluxo*. XXXVI Simpósio Brasileiro de Pesquisa Operacional. São João Del-Rei.
- Pasandideh, S. H. R.; Niaki, S. T. A.** (2010). *Genetic application in a facility location problem with random demand within queuing framework*. Journal of Intelligent Manufacturing, v. 23, n. 3, p. 651–659.
- Slack, N., Chambers, S., Johnston, R.** (1997). *Administração da produção*. 2ª Edição, São Paulo: Atlas.
- Tavares Neto, R. F.; Godinho Filho, M.** (2011). *An ant colony optimization approach to a permutational flowshop scheduling problem with outsourcing allowed*. Computers & Operations Research, v. 38, n. 9, p. 1286–1293.
- Toledo, C. F. M.; Oliveira, R. R. R.; França, P. M.** (2013). *A hybrid multi-population genetic algorithm applied to solve the multi-level capacitated lot sizing problem with backlogging*. Computers & Operations Research, v. 40, n. 4, p. 910–919.
- Witkowski, T.; Antczak, P.; Aantczak, A.** (2010). *Solving the Flexible Open-Job Shop Scheduling Problem with GRASP and Simulated Annealing*. 2010 International Conference on Artificial Intelligence and Computational Intelligence, p. 437–442.

APLICAÇÃO DO ALGORITMO GENÉTICO DE CHAVES ALEATÓRIAS VICIADAS AO PROBLEMA DE ESCALONAMENTO DE TÉCNICOS DE CAMPO

Ricardo de Brito Damm

Departamento de Engenharia de Produção da Escola Politécnica, USP
rbdamm@usp.br

Débora Pretti Ronconi

Departamento de Engenharia de Produção da Escola Politécnica, USP
dronconi@usp.br

RESUMO

Um tema ainda pouco estudado na literatura é o problema de escalonamento de técnicos de campo, que consiste em programar um conjunto de tarefas que devem ser executadas por um grupo de técnicos. As tarefas estão distribuídas em uma região e têm diferentes prioridades e janelas de tempo. Os técnicos possuem habilidades distintas e diferentes horários de trabalho; cada tarefa é atendida por um único técnico e, como objetivo, procura-se principalmente maximizar o número ponderado de tarefas executadas em um dia, de acordo com as suas prioridades. O trabalho apresenta um modelo de programação linear inteira mista (PLIM) e, dada a complexidade do problema, foi aplicado um Algoritmo Genético denominado *Biased Random-Key Genetic Algorithms* (BRKGA), que utiliza chaves aleatórias para codificar e decodificar as soluções. Os resultados computacionais obtidos e as comparações com os limitantes superiores indicaram um bom desempenho do BRKGA.

PALAVARAS-CHAVE: programação de técnicos, roteirização, janelas de tempo, *Biased Random-Key Genetic Algorithms* (BRKGA).

1. Introdução

Um tópico frequentemente encontrado em empresas prestadoras de serviço é o problema de escalonamento de técnicos de campo (PETC), que consiste em associar um número de tarefas (em diversos locais de uma região, com diferentes prioridades e com janelas de tempo) a uma quantidade de técnicos (com diferentes horários de expediente e com habilidades distintas), que devem retornar para o local de origem no final do expediente. Cada tarefa é atendida por um único técnico.

Entre os primeiros autores que estudaram o PETC estão Tsang e Voudouris (1997) e Xu e Chiu (2001). Em 2007, a Sociedade de Pesquisa Operacional da França (*French Operational Society*) e a France Telecom lançaram este problema como um desafio e os trabalhos de Cordeau et al. (2010) e Hashimoto et al. (2011) foram premiados. Em outra importante publicação, Kovacs et al. (2011) estudou uma extensão da pesquisa dos dois artigos anteriores.

Recentemente, Pillac et al. (2012) analisaram a similaridade entre o PETC e o problema de roteamento de veículos com janelas de tempo (PRVJT). De fato, segundo Kovacs et al. (2011), Xu & Chiu (2001) e Pillac et al. (2012), o PETC é uma generalização do PRVJT, que é NP-difícil, e não poderá ser resolvido em tempo polinomial por métodos exatos.

O trabalho apresenta um modelo de programação linear inteira mista (PLIM) na seção 2 e, dada a complexidade do problema, um Algoritmo Genético foi aplicado ao problema (seção 3). A seção 4 apresenta os testes computacionais e a seção 5, as conclusões e futuros passos.

2. Descrição do problema e modelo de programação linear inteira mista

Neste trabalho, PETC será modelado da seguinte maneira: seja $J = \{1, \dots, n\}$ um conjunto de tarefas ou serviços e $K = \{1, \dots, m\}$ um conjunto de técnicos disponíveis para executá-los. Cada tarefa $i \in J$ tem duração de tempo estimada em p_i , uma prioridade w_i e deve ser iniciada e concluída dentro de um intervalo de horário específico $[e_i, l_i]$. As tarefas estão distribuídas dentro de uma cidade e c_{ij} será o tempo de deslocamento do local i para o local j , com $i, j \in J \cup 0$, onde 0 representa a origem ou a sede da empresa.

Cada técnico $k \in K$ tem seu horário de trabalho diário definido por $[a_k, b_k]$. A habilidade de um técnico k executar uma tarefa i é dada por s_{ik} , que pode ser 0 ou 1, onde 1 indica ser apto e 0, incapaz de executá-la. O objetivo será alocar o maior número ponderado de tarefas e, secundariamente, maximizar a ociosidade dos funcionários ao retornar à origem (no fim do dia).

O modelo tem quatro tipos de variáveis: y_{ik} é uma variável binária que assume 1 quando uma tarefa i foi associada a um técnico k e 0, caso contrário; x_{ijk} também é uma variável binária com valor 1 se uma tarefa j é executada imediatamente após a tarefa i pelo técnico k . A variável t_i indica o horário de início da execução da tarefa i e z_k , o tempo ocioso de cada técnico após retornar à base.

A seguir é apresentada a formulação linear inteira mista, baseada em Xu e Chiu (2001) para o problema estudado.

$$\max \sum_{i \in J} \sum_{k \in K} \frac{w_i s_{ik} y_{ik}}{MW} + \sum_{k \in K} \frac{z_k}{MZ} \quad (1)$$

Sujeito a:

$$\sum_{k \in K} y_{ik} \leq 1 \quad i \in J \quad (2)$$

$$\sum_{j \in J \cup \{0\} \setminus \{i\}} x_{ijk} = \sum_{j \in J \cup \{0\} \setminus \{i\}} x_{jik} = y_{ik} \quad k \in K, i \in J \quad (3)$$

$$\sum_{i \in J} x_{0ik} \leq 1 \quad k \in K \quad (4)$$

$$e_i \leq t_i \leq l_i - p_i \sum_{k \in K} y_{ik} \quad i \in J \quad (5)$$

$$t_i + p_i + c_{ij} \leq t_j + M(1 - x_{ijk}) \quad k \in K, i \neq j \in J \quad (6)$$

$$a_k + c_{0j} \leq t_j + M(1 - x_{0jk}) \quad k \in K, j \in J \quad (7)$$

$$t_i + p_i + c_{i0} \leq b_k - z_k + M(1 - x_{i0k}) \quad k \in K, i \in J \quad (8)$$

$$0 \leq z_k \leq b_k - a_k \quad k \in K \quad (9)$$

$$x_{ijk}, y_{ik} \in \{0, 1\} \quad i, j \in J, k \in K \quad (10)$$

$$t_i, z_k \in R^+ \quad i \in J, k \in K \quad (11)$$

Onde:

$$M > \max \left(\max_{i \in J} l_i, \max_{k \in K} a_k \right) + \max_{i, j \in J \cup \{0\}} c_{ij} \quad (12)$$

$$MW = \min(w_i s_{ik}) \quad i \in J, k \in K, s_{ik} \neq 0 \quad (13)$$

$$MZ = \max(b_k - a_k) + \varepsilon \quad k \in K \quad (14)$$

Na função objetivo (1), o primeiro termo maximiza a soma ponderada das tarefas realizadas e o segundo, a soma dos tempos de disponibilidade dos técnicos após o retorno à base. Os denominadores têm duas funções: tornar adimensionais os valores somados e garantir que sempre será melhor executar uma tarefa, reduzindo a ociosidade de um técnico, e nunca o contrário. Ao mesmo tempo, o objetivo secundário procurará minimizar o tempo de trabalho externo dos técnicos e, conseqüentemente, reduzir o tempo de deslocamento total e os tempos de espera para o início da execução de uma tarefa.

A restrição (2) evita que uma tarefa seja designada a mais de um técnico e a (3) garante que haverá apenas uma antecessora e uma sucessora a uma tarefa programada para um técnico. Cada técnico deve deixar a origem, no máximo, uma vez, como assegura a restrição (4). A restrição (5) garante que as janelas de tempo das tarefas serão respeitadas. As restrições (6) a (8) vinculam o início da execução de cada tarefa com o término da sua antecessora, os tempos de deslocamento e o horário de trabalho do técnico; ao mesmo tempo, essas restrições previnem o *subtour*. O tempo disponível (ou ocioso) máximo de um técnico k após o retorno à origem é determinado pela restrição (9). As restrições (10) e (11) estipulam os tipos de variáveis e as restrições (12) a (14), as constantes M , MZ e MW .

3. Algoritmo Genético com chaves aleatórias viesadas

O Algoritmo Genético com chaves aleatórias, denominado *Random-Key Genetic Algorithms* (RKGA), foi proposto por Bean (1994). Trata-se de um AG específico para problemas de otimização combinatória que garante a factibilidade das novas soluções geradas ao longo da busca. Alguns anos depois, foram introduzidos novos conceitos a esse AG, aumentando o protagonismo das melhores soluções (as mais aptas) de cada geração. Denominada por Gonçalves e Resende (2011) como *Biased Random-Key Genetic Algorithms* (BRKGA), essa nova versão do RKGA apresentou um melhor desempenho (Gonçalves et al., 2012).

No BRKGA, cada indivíduo da população é representado por um vetor com números reais aleatórios (chaves aleatórias) no intervalo $[0; 1)$, que deverá ser decodificado por um algoritmo para transformá-lo em uma solução factível do problema.

A população de uma geração é dividida em dois grupos: um primeiro grupo menor com as melhores soluções (elite) e um segundo grupo maior com as demais soluções (não elite). A geração seguinte é formada por três grupos: todas as soluções de elite são copiadas, algumas novas soluções são geradas aleatoriamente (aumentando a diversidade da população) e o restante da população é formado por cruzamentos dos indivíduos da geração anterior.

O cruzamento é feito por um *crossover* parametrizado: dois cromossomos são selecionados, sendo um do grupo de elite e outro do grupo não elite. Para a escolha de cada gene do filho, é feito um sorteio onde a probabilidade de uma solução de elite transmitir o seu gene é maior do que a da outra solução. Não há mutações após o cruzamento.

Duas versões do BRKGA foram desenvolvidas para o PETC. Na primeira, as chaves aleatórias foram utilizadas para selecionar um técnico para cada tarefa e construir o roteiro de cada técnico. Desse modo, os cromossomos terão uma chave aleatória para cada tarefa, que serão designadas para um técnico da seguinte maneira: se há dois técnicos capazes de realizar uma tarefa e a chave aleatória for um número no intervalo $[0; 1/2)$, então ela será associada ao primeiro técnico; caso seja um número entre $[1/2; 1)$, será designada ao segundo. Um procedimento análogo é realizado quando uma tarefa tem 3 ou mais técnicos aptos. Uma vez distribuídas todas as tarefas, para construir a rota de cada técnico foi extraído um *índice* de cada chave aleatória, que é a porcentagem de distância entre o valor da chave aleatória e o início do subintervalo: por exemplo, se a chave aleatória tem valor 0,25, para o exemplo anterior, então ela está exatamente na metade do subintervalo $[0; 1/2)$ e tem índice de 50%. Desse modo, as tarefas de cada técnico são ordenadas em ordem crescente do índice e, da primeira à última, serão introduzidas no roteiro do técnico; caso não haja solução factível (por conflitos de janelas de tempo, etc.) para introduzir uma tarefa na rota, ela não será realizada. Essa versão foi denominada BRKGA-A.

Na segunda versão, foi criado um cromossomo também com uma chave aleatória para cada tarefa, que serão ordenadas em ordem crescente. As tarefas serão introduzidas nas rotas dos técnicos a partir da tarefa com menor chave aleatória até a tarefa com maior valor da chave aleatória. Para cada tarefa, todos os técnicos aptos são avaliados e o técnico que puder realizar aquela tarefa com menor tempo de deslocamento total no seu dia será escolhido. Essa versão foi chamada de BRKGA-STT.

4. Experimentos computacionais e resultados

Todos os códigos foram programados em linguagem C e os experimentos realizados em um computador Intel com 2.93 GHz e processador 16GB de memória RAM. Para o modelo PLIM utilizou-se o *software* ILOG CPLEX, versão 12.2, com os parâmetros *default*.

Foram construídas 260 instâncias: 20 para cada um dos casos apresentados na Tabela 1 (a segunda e terceira colunas da tabela apresentam, respectivamente, o número de tarefas e técnicos em cada caso). Os parâmetros dessas instâncias foram gerados por uma distribuição uniforme discreta dos seguintes valores: prioridade das tarefas de 1 (baixa) a 10 (alta); tempo de execução das tarefas de 30 a 120 min.; início da janela das tarefas de 8 a 16h; tamanho da janela das tarefas de 2 a 8h; início da janela dos técnicos de 6:30 a 11h; fim da janela dos técnicos somando 9h ao início da janela de tempo; habilidade dos técnicos para executar cada tarefa será 0 ou 1; todas as tarefas estão em uma região onde o tempo de deslocamento entre quaisquer dois locais (diâmetro) é menor ou igual a 1,5 hora.

A população inicial do BRKGA foi gerada aleatoriamente, com exceção de alguns cromossomos incluídos, que representam as soluções de heurísticas construtivas apresentadas na ONPCE XV (Damm e Ronconi, 2013).

Para os casos 1 a 4 (dimensões pequenas) foram encontradas as soluções ótimas pelo software Cplex e, para os casos maiores, não foram obtidas soluções ótimas em até uma hora de processamento. O BRKGA-A encontrou 95% das soluções ótimas dos casos 1 a 4 e o BRKGA-STT, 98%.

A tabela 1 apresenta a distância média percentual com relação aos limitantes superiores, calculados pela expressão:

$$GAP_{UB} = 100 \cdot \frac{BR - UB}{UB}$$

onde BR é o valor da função objetivo obtida pelo BRKGA e UB o valor da função objetivo do limitante superior.

Caso	Tarefas	Técnicos	BRKGA-A	BRKGA-STT
1	12	2	-2,4	-2,4
2	16	2	-3,4	-3,4
3	26	2	-8,2	-8,2
4	40	3	-9,0	-8,9
5	45	7	-5,0	-4,4
6	64	5	-11,6	-11,2
7	80	13	-5,7	-4,8
8	100	10	-12,0	-10,9
9	120	20	-5,1	-4,0
10	150	25	-5,0	-3,4
11	200	33	-5,6	-3,7
12	500	83	-4,4	-3,1
13	999	166	-3,9	-2,7
		Média dos casos 1-4 (%)	-5,7	-5,7
		Média dos casos 5-13 (%)	-6,5	-5,3

Tabela 1: Diferença percentual média entre os resultados obtidos e os limitantes superiores. Em negrito, os melhores valores obtidos.

Os resultados obtidos pelo BRKGA-STT em média superaram os resultados do BRKGA-A nos casos 4 a 13. Para dimensões médias e grandes (casos 5 a 13), os resultados do BRKGA-STT ficaram em média a 5,3% dos limitantes superiores calculados, o que parece um bom desempenho, uma vez que os valores ótimos das instâncias pequenas estão a uma distância média de 5,7% dos limitantes superiores. Nos casos em que há uma maior proporção de tarefas por técnicos (casos 3, 4, 6 e 8), os resultados obtidos ficaram em média mais distantes dos limitantes superiores.

5. Conclusões e perspectivas futuras

A principal contribuição deste trabalho foi a forma de aplicar o BRKGA ao PETC. A diferença percentual do desempenho entre as duas versões do BRKGA foi em média de 1% e, portanto, nota-se que as formas de codificar e decodificar as soluções tem um impacto significativo nos resultados. Outras versões do BRKGA foram desenvolvidas (por exemplo, adotando estratégias de busca local ou enumerações completas para determinar o roteiro de um técnico), obtendo desempenhos ligeiramente inferiores aos apresentados neste trabalho.

Como perspectivas futuras, pretendem-se estudar o problema com diferentes distribuições de pontos numa cidade (distribuição uniforme, agrupamento de clientes em clusters, etc.) e também desenvolver modelo multiobjetivo para o PETC.

Referências

- Bean, J. C.** (1994). Genetic algorithms and random keys for sequencing and optimization. *ORSA journal on computing*, 6 (2), 154-160.
- Cordeau, J. F.; Laporte, G.; Pasin, F.; Ropke, S.** (2010). Scheduling technicians and tasks in a telecommunications company. *Journal of Scheduling*, 13, 393-409.
- Damm, R. B.; Ronconi, D. P.** (2013). O problema de escalonamento de técnicos de campo. *XV Oficina Nacional de Problemas de Corte, Empacotamento, Dimensionamento de Lotes e Programação da Produção*, Caderno de resumos, 18-19.
- Gonçalves, J. F.; Resende, M. G.** (2011). Biased random-key genetic algorithms for combinatorial optimization. *Journal of Heuristics*, 17(5), 487-525.
- Gonçalves, J. F.; Resende, M. G.; Toso, R.** (2012). An experimental comparison of biased and unbiased random-key genetic algorithms. *Technical report, AT&T Labs Research*.
- Hashimoto, H.; Boussier, S.; Vasquez, M.; Wilbaut, C.** (2011). A GRASP-based approach for technicians and interventions scheduling for telecommunications. *Annals of Operations Research*, 183, 143-161.
- Kovacs, A. A.; Parragh, S. N.; Doerner, K. F.; Hartl, R. F.** (2011). Adaptive large neighborhood search for service technician routing and scheduling problems. *Journal of Scheduling*, 15, 579-600.
- Pillac, V.; Guéret, C.; Medaglia, A. L.** (2012). A parallel matheuristic for the technician routing and scheduling problem. *Optimization Letters*, 1-11.
- Tsang, E.; Voudouris, C.** (1997). Fast local search and guided local search and their application to British Telecom's workforce scheduling problem. *Operations Research Letters*, 20, 119-127.
- Xu, J.; Chiu, S. Y.** (2001). Effective Heuristic Procedures for a Field Technician Scheduling Problem. *Journal of Heuristics*, 7, 495-509.

UM ESTUDO DE PROBLEMAS DE OTIMIZAÇÃO COMBINATÓRIA EM HOSPITAIS

Tiago Tiburcio da Silva
DMAp – UNESP/IBILCE
ttsilva@gmail.com

Socorro Rangel
DMAp – UNESP/IBILCE
socorro@ibilce.unesp.br

RESUMO

A área de *scheduling* tem por objetivo sequenciar, escalonar, tarefas em processos industriais, militares, educacionais e na área da saúde. No que tange à área da saúde, este problema surge em diversas situações que inclui, entre outras, escalonamento de enfermeiros, de cirurgias e de exames.

O Problema de Escalonamento de Enfermeiros (PEE) consiste em determinar os horários em que cada enfermeiro irá trabalhar num dado período de tempo respeitando condições trabalhistas (Burke *et al* (2004)). Na literatura são propostos diversos métodos de solução para o problema: Maenhout e Vanhoucke (2007) usam algoritmos genéticos; Carello *et al* (2004) propõem um algoritmo que mescla busca local, míope e tabu; Constantino *et al* (2013) utiliza um algoritmo heurístico multi-estágio. Já o Problema de Escalonamento de Cirurgias (PEC) consiste em determinar em que período de tempo cada médico irá operar cada paciente e em que sala de operação. Há muitos trabalhos na literatura relacionados a este tema, e diferentes técnicas são utilizadas para a sua resolução e vão desde programação linear, quadrática e dinâmica, até métodos de simulação e heurísticas. O problema é tratado tanto na forma determinística como estocástica. Uma revisão bibliográfica sobre o tema é feita em Cardoen *et al* (2008). Este problema também foi tema da *AIMMS/MOPTA Optimization Modeling Competition* em 2013 (Mopta 2013).

O objetivo deste projeto é estudar o Problema de Escalonar Exames Médicos (PEEM). O problema será tratado no contexto de um hospital onde vários tipos de exames são realizados, por exemplo, exames de peças cirúrgicas (mama, útero, ovário, próstata, estômago, cólon, pulmão, entre outros), exame oncogenético. São considerados que existem m tipos de exames; n_i , $i=1,2,\dots,m$ lugares/aparelhos onde cada um desses m exames pode ser realizado; uma lista com p pacientes, cada um com suas preferências/restrições, por exemplo, horário de realização do exame; e uma lista com os responsáveis por realizarem os exames. Cada paciente pode (ou deve) realizar mais de um exame. Deseja-se atribuir um horário e um responsável para cada par exame/paciente de acordo com as preferências/restrições dos paciente, responsáveis e lugares/aparelhos. O critério de otimização é a maximização de uso dos locais/equipamentos. Até o momento não foram encontrados na literatura trabalhos que tratam especificamente do problema PEEM. Foi observado, no entanto, que existe uma relação entre a modelagem do PEEM e o PEC. Neste trabalho, apresentaremos estudos preliminares desenvolvidos. É um projeto de mestrado, e o foco é em hospitais públicos, dada a necessidade urgente de otimização dos processos hospitalares do sistema de saúde brasileiro.

PALAVARAS-CHAVE: Scheduling, Exames Médicos, Otimização Combinatória, PROJETO.

Referências

- Burke, E. K., Causmaecker, P., Berghe, G. V. e Landeghem, H.** (2004), *The state of the art of nurse rostering*, Journal of Scheduling, 7(6), 441-499.
- Cardoen, B.; Demeulemeester, E.; Beliën, J.** (2008). *Operating room planning and scheduling: A literature review*. Katholieke Universiteit Leuven, Bélgica.
- Carello, F., Croci, F. e Tadei, R.** (2004). A greedy-based neighborhood search approach to a nurse rostering problem, European Journal of Operational Research, 153(1), 28-40.
- Constantino, A.A.; Silva, D.L.; Melo, E.L.; Mendonça, C.F.X.M; Rizzato, D.B; Romão, W.** (2013). *A heuristic algorithm based on multi-assignment procedures for nurse scheduling*. Annals of Operation Research, v.1, p. 1-23, 2013.
- Maenhout, B. e Vanhoucke, M.** (2007), *An electromagnetic meta-heuristic for the nurse scheduling problem*, Journal of Heuristics, 13(4), 359-385.
- MOPTA** (2013), *Modeling and Optimization: Theory and Applications (MOPTA) conference, Department of Industrial and Systems Engineering at Lehigh University*. <http://coral.ie.lehigh.edu/~mopta2013/> (última visita: 07 de março de 2014).
- Maenhout, B. e Vanhoucke, M.** (2008). *Comparison and hybridization of crossover operators for the nurse scheduling problem*, Annals of Operations Research, 159(1), 333-35
- Meskens N.; Duvivier, David.; Hanset, Arnould.** (2012). *Multi-objective operating room scheduling considering desiderata of the surgical team*, Decision Support Systems 55 (2013) 650-659.
- Pinedo, M.** (1995). *Scheduling: theory, algorithms and systems*. Prentice-Hall, Upper Saddle River, NJ.
- Pinedo, M.** (2005). *Planning and scheduling in manufacturing and services*. New York, Springer.

Sessão 3

Um algoritmo memético no problema do Corte Unidimensional Inteiro.

Autores: Angelo Aliano Filho, Antônio Carlos Moretti.

Planejamento na divisão de talhões para o plantio da cana-de-açúcar.

Autores: Adriana Cristina Cherri, Andréa Vianna, Helenice Florentino, Rômulo Ramos.

Uma heurística para um problema de carregamento de caixas em um contêiner compartimentado.

Autores: Rodolfo Ranck Junior, Horacio Hideki Yanasse, Reinaldo Morabito.

Problema de Localização-Roteamento-Empacotamento com Demanda Incerta.

Autores: Thiago Alves de Queiroz, José Fernando Oliveira, Maria Antónia Carravilla, Flávio Keidi Miyazawa.

Um modelo compactado para o problema de corte de peças irregulares.

Autores: Marcos Okamura Rodrigues, Aline A. S. Leão, Franklina M. B. Toledo.

Um método heurístico para o problema de corte de peças irregulares.

Autores: Luiz Henrique Cherri, Franklina Maria Bragion de Toledo; Maria Antónia Carravilla.

UM ALGORITMO MEMÉTICO NO PROBLEMA DO CORTE UNIDIMENSIONAL INTEIRO

Angelo Aliano Filho¹ e Antônio Carlos Moretti²

Universidade Estadual de Campinas - IMECC

E-mail: ¹angeloaliano@hotmail.com ²moretti@ime.unicamp.br

RESUMO

Este trabalho visou solucionar o PCUI utilizando uma metaheurística híbrida, um Algoritmo Genético seguido seguida de uma Busca Local. Juntamente com o método desenvolvido, duas heurísticas foram desenvolvidas (i) uma que constroi padrões de corte e (ii) outra que elabora uma solução para o problema. Afim de comparar e avaliar a solução da metaheurística, implementou-se o clássico algoritmo de Geração de Colunas, e em seguida, resolveu-se o Problema Linear Inteiro com as colunas obtidas, usando-se o *Branch-and-Bound*. Os resultados e teste preliminares demonstraram eficiência e robustez do método aqui desenvolvido, tendo em vista a qualidade das soluções encontradas e o tempo de processamento.

PALAVRAS-CHAVE: Problema do Corte, Otimização Combinatória, Algoritmo Memético.

1. INTRODUÇÃO

O PCUI é um dos problemas combinatórios mais estudados, devido, principalmente, a sua aplicabilidade no mundo da engenharia de produção, fazendo parte do planejamento de uma diversidade de indústrias que cortam papel, móveis, vidro, plásticos, tecido, entre outras. Apesar de ser muito simples de ser entendido, este problema têm um elevado nível de complexidade, sendo classificado na literatura como \mathcal{NP} -difícil Garey e Johnson (1996).

A tarefa de se obter uma solução ótima exata para o PCUI utilizando métodos clássicos de otimização, é desafiante. Isso se deve, principalmente, à integrabilidade e o elevado número de variáveis decisórias envolvidas neste problema. Alguns poucos algoritmos exatos para encontrar a solução ótima inteira do PCUI são conhecidos na literatura, como Carvalho (1999) e Vanderbeck (1996). No entanto, estes métodos podem ser aplicados apenas a problemas de pequeno porte.

Sendo assim, metodologias aproximativas têm sido desenvolvidas nas últimas cinco décadas para este problema. O primeiro trabalho a tratar o PCUI de maneira não exata foi Gilmore e Gomory (1961), que propuseram uma técnica de GC para obtenção de uma solução ótima contínua. Outros métodos não-exatos têm sido desenvolvidos e uma breve revisão é dada em Wascher e Gau (1996). Ressalta-se também o uso de Heurísticas de Arredondamento, proposto por Poldi e Arenales (2006), e algumas metaheurísticas específicas, desenvolvidas por Gehring e Bortfeldt (1998), Golfeto et al. (2008), Golfeto et al. (2009), entre tantos.

Mesmo que a otimalidade não seja garantida, o objetivo deste trabalho é apresentar um método heurístico de solução alternativo que forneça soluções “boas” e rápidas para este problema.

2. MODELAGEM MATEMÁTICA

Para modelar o PCUI, considere uma bobina mestre em estoque, de tamanho L (comprimento padrão), e m o número de itens demandados. Cada item tem comprimento

$l_i < L$ e ao menos d_i unidades precisam ser produzidas afim de atender a demanda necessária, $i = 1, \dots, m$. O objetivo consiste em minimizar o número de peças a serem cortadas afim de atender à esta demanda mínima exigida. Do ponto de vista operacional, aqui apenas faz sentido cortar um número inteiro de vezes desta bobina mestre, o que torna este problema difícil de ser resolvido.

Uma instrução que lista os itens demandados a serem cortados da bobina principal é chamada de *padrão de corte*, que pode ser associado ao vetor m -dimensional

$$\mathbf{a}_j = (a_{1j}, a_{2j}, \dots, a_{mj})^T,$$

onde cada entrada $a_{ij} \in \mathbb{N}$, denota a quantidade do item i presente no padrão j , $i = 1, \dots, m$.

Definição 2..1. Um padrão de corte $\mathbf{a}_j \in \mathbb{N}^m$ é admissível se satisfaz as seguintes condições:

$$\sum_{i=1}^m l_i \cdot a_{ij} \leq L, \quad (1)$$

$$\sum_{i=1}^m l_i \cdot a_{ij} \geq L - \Delta, \quad (2)$$

$$\sum_{i=1}^m a_{ij} \leq F, \quad (3)$$

onde $\Delta = \min_{1 \leq i \leq m} \{l_i\}$ e F é o número de facas máximo permitido para produzi-lo.

Suponha que existam n maneiras distintas de se formar todas as combinações de corte possíveis atendendo as restrições (1)-(2). Como visto, mesmo para m da ordem de dezenas, o valor de n pode facilmente atingir a casa dos milhões ou bilhões, desde que a razão L/l_i fique suficientemente pequena.

O modelo para o PCUI é apresentado a seguir:

$$\text{Minimize } z = \sum_{j=1}^n x_j \quad (4)$$

$$\text{sujeito a } \sum_{i=1}^m a_{ij} \cdot x_j \geq d_i, \quad i = 1, \dots, m, \quad (5)$$

$$x_j \in \mathbb{N}, \quad j = 1, \dots, n. \quad (6)$$

A variável decisória deste problema é x_j , e denota a frequência do padrão j a ser utilizado, $j = 1, \dots, n$. O conjunto de restrições (5) diz respeito ao atendimento mínimo da demanda exigida para cada item i .

Como salientado na seção anterior, este modelo tem dois agravantes, e que dificultam a obtenção da solução ótima exata $\mathbf{x}^* = (x_1^*, x_2^*, \dots, x_n^*)^T$: (i) as condições de integrabilidade (6) e (ii) o elevado valor de n , o que torna inviável resolver o PLI (4)-(6) usando os métodos exatos clássicos como *Branch-and-Bound, Cut, Price*.

Afim de encontrar um limitante inferior \underline{z} para o valor da função objetivo e avaliar a qualidade das soluções heurísticas, aplicou-se o algoritmo clássico GC e em seguida resolveu o PLI com as colunas de Gilmory-Gomory encontradas.

3. O ALGORITMO MEMÉTICO

Em virtude da elevada complexidade do PCUI e na impossibilidade de solucioná-lo via métodos exatos, apresenta-se uma metaheurística alternativa, afim de obter soluções admissíveis

que aproximam \mathbf{x}^* num tempo computacional aceitável. Para isto, foram desenvolvidas duas heurísticas: uma para construir padrões admissíveis, respeitando as restrições (1)-(3) e a outra para construir soluções factíveis para o problema atendendo (5).

A codificação de uma solução \mathbf{S} para o PCUI é composta por uma matriz $\mathbf{A} \in \mathbb{N}^{m \times m}$, onde a coluna \mathbf{a}_j é o j -ésimo padrão de corte, acoplada um vetor linha $\mathbf{x}^T \in \mathbb{N}^{1 \times m}$, cuja componente j indica a frequência do padrão \mathbf{a}_j nesta solução, ou seja, $\mathbf{S} = \begin{bmatrix} \mathbf{A} \\ \mathbf{x}^T \end{bmatrix}$.

Sendo assim, esses algoritmos fazem com que o método desenvolvido busque uma solução para o problema somente no espaço admissível. Os passos destes dois algoritmos são apresentados na próxima subseção.

3.1. HEURÍSTICAS CONSTRUTIVAS

O primeiro procedimento constroi padrões admissíveis e tem como dados de entrada m, F, l_i, L . Para evitar um laço infinito, um vetor \mathbf{v} é inicializado, e indica a ordem dos itens que têm preferência de escolha para entrar no padrão a ser formado. Um pseudocódigo deste algoritmo é apresentado na Figura (1).

Procedimento 1: construção de padrões

Input: $\{m, F, l_i, L\}$

Sobra = L

Enquanto *Sobra* > Δ

Inicialize \mathbf{v}

Para $i = 1$ até m

Tome o i -ésimo item de \mathbf{v}

Calcule a_{ij} : quantas vezes este item pode ser adicionado em uma fração da sobra deste padrão

Se o número de facas não foi excedido, aloque a_{ij} na posição correspondente **Fim-Se**

Atualize $Sobra = L - \sum_{k=1}^i l_k \cdot a_{kj}$

Fim-Para

Fim-Enquanto

Output: \mathbf{a}_j

Figura 1: Algoritmo para construir padrões admissíveis

Apresenta-se na Figura (2) o procedimento que constroi soluções admissíveis para o problema.

Este algoritmo constroi soluções admissíveis ao PCUI quanto às restrições de demanda e que contenham, no máximo, m padrões. Caso a demanda tenha sido atendida mesmo antes de terem usado as m colunas de \mathbf{S} , completa-se a dimensão desta matriz com colunas nulas. Isto pode ser útil quando se interessa em soluções que utilizem um menor *setup*.

Usando-se os **Procedimentos 1 e 2**, pode-se descrever o método Memético. Inicialmente, aplica-se um Algoritmo Genético (AG) clássico, iniciando-o com uma população com P indivíduos e G gerações. Em seguida, avalia-se cada indivíduo \mathbf{S} segundo a função f :

$$f(\mathbf{S}) = \sum_{i=1}^m x_j + 10 \cdot \left(\sum_{i=1}^m \sum_{j=1}^m a_{ij} \cdot x_j - d_i \right). \quad (7)$$

Procedimento 2: construção de soluções

Input: $\{m, F, l_i, L, \mathbf{d}\}$

$\mathbf{S} = \emptyset$

producao = 0

Falta = **producao** – **d**

Enquanto **Falta** < 0

Certifique os itens que ainda faltam para completar a demanda

Aloque, nas primeiras posições do vetor **v**, os itens cujas demandas não foram atendidas

Construa um padrão \mathbf{a}_j utilizando o **Procedimento 1**

Calcule

$$x_j = \min_i \left\{ \left\lceil \frac{-Falta_i}{a_{ij}} \right\rceil : a_{ij} \neq 0 \wedge sobra_i > 0 \right\}$$

producao = **producao** + $x_j \cdot \mathbf{a}_j$

Falta = **producao** – **d**

$\mathbf{S} = \mathbf{S} \cup \begin{bmatrix} \mathbf{a}_j \\ x_j \end{bmatrix}$

Fim-Enquanto

Output: **S**

Figura 2: Algoritmo para construir soluções admissíveis

Nesse sentido, prioriza-se as soluções com a menor superprodução possível. A melhor solução encontrada (elite) é preservada dos próximos operadores deste método. Em seguida, seleciona-se 80% destes indivíduos, usando-se a roleta viciada, para realizar o **Crossover Uniforme**.

Dados dois indivíduo pareados, para obter novos indivíduos (filhos), realizaram-se cortes verticais em posições aleatórias nestas matrizes, afim de sempre preservar a factibilidade dos padrões de corte construídos. Entretanto, a factibilidade com relação à demanda pode ser perdida ao se gerar um filho. Se isto acontecer, a linha que contém as frequências (vetor \mathbf{x}^T) é recalculada utilizando os padrões que formam o novo indivíduo.

Após esta etapa, realiza-se o operador **Mutação** com probabilidade $\sigma(g)$

$$\sigma(g) = \frac{0,01}{0,01 + e^{-g/10}}, \quad (8)$$

onde $g = 1, \dots, G$ é a geração corrente. Isso significa que este operador atuará mais intensamente nas gerações finais, impedindo e adiando a formação de uma população com características similares. A mutação, se ocorrer numa dada geração, escolhe 10% da população, e com probabilidade de 50% modifica os gens (que é uma coluna do tipo $(\mathbf{a}_j | x_j)^T$) destes indivíduos. Caso a factibilidade seja perdida, a mesma é recuperada no mesmo procedimento.

Após o AG finalizar sua busca, a elite encontrada serve como solução inicial para uma Busca Local clássica (BL) clássica. A única peculiaridade é a topologia da estrutura de vizinhança aqui empregada. Dada uma solução corrente **S**, define-se como um vizinho **V** uma outra solução que tem os $m - 1$ padrões original e o remanescente é gerado pelo **Procedimento 1**, e nesse caso, o vetor das frequências para esta nova solução é calculado. Nessa BL encerra-se a busca quando o número de iterações (comparações) for igual à θm , θ um parâmetro.

A principal ideia de uma metaheurística híbrida é formar um método conjunto, capaz de pesquisar soluções mais eficientes do que se apenas um método (AG ou BL) fosse empregado.

4. PERSPECTIVAS FUTURAS

Este trabalho visou solucionar o PCUI utilizando a metaheurística Memética com duas heurísticas construtivas. Afim de comparar e avaliar a solução da metaheurística, implementou-se o clássico algoritmo de Geração de Colunas, e em seguida, resolveu-se o Problema Linear Inteiro com as colunas obtidas, usando-se o *Branch-and-Bound*. Os resultados preliminares demonstraram eficiência e robustez do método aqui desenvolvido, tendo em vista a qualidade das soluções encontradas e o tempo de processamento. Além disso, o método implementado obteve soluções mais interessantes que as fornecidas pelo concorrente, no tocante à superprodução, ao número de padrões utilizados e o desperdício. Isto pode ser entendido como um sub-produto do método apresentado, sendo uma ferramenta simples, fácil de ser implementada e do ponto de vista computacional, viável.

Como perspectiva futura de novos trabalhos, propõe-se utilizar estes algoritmos heurísticos para tratar o PCUI com múltiplos objetivos.

O trabalho presente está ainda em andamento e é fruto das primeiras pesquisas do aluno envolvido no seu programa de Doutorado. Na apresentação do trabalho, caso seja aceito, resultados mais conclusivos e detalhados serão apresentados.

AGRADECIMENTOS

Este trabalho tem apoio da Fundação de Amparo à Pesquisa do Estado de São Paulo, FAPESP, processo 2013/06035-0.

Referências

- Carvalho, J. M. V.** (1999). Exact Solution of Bin-packing Problems Using Column Generation and Branch-and-bound. *Annals of Operations Research*, 86, 629–659.
- Garey, E. C. M.; Johnson, D.** (1996). *Approximation Algorithms for Bin Packing: A Survey*. Boston. PWS. Approximation algorithms for NP-hard problems.
- Gehring, H.; Bortfeldt, A.** (1998). A Genetic Algorithm for Solving the Container Loading Problem. *International Transactions in Operations Research*, 4, 401–418.
- Gilmore, P. C.; Gomory, R. E.** (1961). A Linear Programming Approach to the Cutting-Stock Problem. *Oper Res*, 9, 848–859.
- Golfeto, R. R.; Moretti, A. C.; Sales, L. L. N.** (2008). A Grasp Metaheuristic for the Ordered Cutting Stock Problem. *Revista Chilena de Ingeniería (En línea)*, 16, 421–427.
- Golfeto, R. R.; Moretti, A. C.; Sales, L. L. N.** (2009). A Genetic Symbiotic Algorithm Applied to the One-dimensional Cutting Stock Problem. *Pesquisa Operacional (impresso)*, 9, 365–382.
- Poldi, K. C.; Arenales, M. N.** (2006). Heurísticas para o Problema do Corte Unidimensional Inteiro. *Pesquisa Operacional*, 26, 473–492.
- Vanderbeck, F.** (1996). An Exact Algorithm for ip Column Generation. *Operations Research Letters*, 19, 151–159.
- Wascher, G.; Gau, T.** (1996). Heuristic for the Integer One-dimensional Cutting Stock Problem: a Computational Study. *OR Spektrum*, 18, 131–144.

PLANEJAMENTO NA DIVISÃO DE TALHÕES PARA O PLANTIO DA CANA-DE-AÇÚCAR

Andréa Vianna, Adriana Cherri

Faculdade de Ciências – UNESP, Bauru
vianna@fc.unesp.br, adriana@fc.unesp.br

Helenice Florentino, Rômulo Ramos

Instituto de Bioestatística – UNESP, Botucatu
helenice@ibb.unesp.br, krpr@fca.unesp.br

RESUMO

Nos últimos anos, a cana-de-açúcar se destaca como uma fonte alternativa de energia, tanto pela geração do álcool como pela cogeração de energia através do bagaço da cana. Um bom planejamento no seu plantio acarreta em vários benefícios, principalmente no aumento da sua produção. Planejar o plantio é um processo complexo e requer cuidados dos gestores do setor, pois as tomadas de decisões nessa fase vão influenciar todo o ciclo da cultura. Com isso surge necessidade de técnicas que auxiliem esses gestores na construção do planejamento do plantio e a primeira fase deste planejamento é a divisão adequada da área de plantio em talhões. Os talhões de cana-de-açúcar mais recomendados são os de forma retangular e de maior comprimento possível, para expressar toda a eficiência da colhedora e carregadores locados de forma estratégica para não ter curvas mortas e evitar manobras excessivas e conseqüentemente a perda de tempo e de consumo de combustível. Dessa forma, este trabalho propõe a alocação de talhões retangulares na área de plantio da cana-de-açúcar de forma otimizada. Para fazer esta alocação dos talhões, o problema será tratado como um problema de corte bidimensional e, devido a irregularidades das regiões e de locais em que não é possível realizar o plantio da cana (lagos, áreas de preservação, etc.), algumas particularidades do problema de corte será explorada.

PALAVARAS-CHAVE: Cana-de-açúcar, Planejamento de Plantio, Otimização.

1. Introdução

O Brasil é o maior produtor mundial de cana-de-açúcar e o maior exportador mundial de açúcar, o que vem influenciar diretamente a economia brasileira. No mercado nacional, a cana tem ganhado espaço, pois recentemente houve um aumento de 5% na mistura de álcool à gasolina. Além disso, o Brasil conta com 16,9% de participação na matriz energética, considerando-se o álcool combustível e a cogeração de eletricidade, a partir do bagaço (EPE, 2012).

Na safra de 2013/2014 a cultura da cana-de-açúcar continua em expansão. A previsão é que o Brasil tenha um acréscimo na área de cerca de 314 mil hectares, equivalendo a 3,7% em relação à safra 2012/13. Esse acréscimo é reflexo do aumento de área da Região Centro-Sul. A Região Norte/Nordeste praticamente se manterá com a mesma área para a próxima safra. São Paulo, Minas Gerais, Goiás e Mato Grosso do Sul deverão ser os estados com maior acréscimo de áreas com 95,9 mil hectares, 60,1 mil hectares, 92,5 mil hectares e 81,4 mil hectares, respectivamente. Já a área de renovação prevista para esta safra deverá alcançar 969 mil hectares. Na Região Centro-Sul, a área de renovação deve chegar a 872,8 mil hectares, correspondendo a 11 % das lavouras atuais (CoNAb, 2013).

Em decorrência dessa expansão, o planejamento do plantio da cana-de-açúcar em novas áreas e em áreas de reforma torna-se essencial, a fim de trazer benefícios econômicos, sociais e ambientais para o setor sulcroatcooleiro. Um plantio bem planejado implica no aumento da produção da cana, no bom aproveitamento da área, na melhoria das logísticas de transporte e

recebimento da cana na indústria, na facilidade de administração das atividades industriais, no melhor atendimento das demandas das indústrias, no melhor aproveitamento da mão de obra, nos planejamentos de custos, no controle de pragas e ervas daninhas, e muitos outros.

De acordo com Pereira e Torrezan (2006), os talhões mais recomendados são os de forma retangular e de maior comprimento possível, para expressar toda a eficiência da colhedora e carregadores locados de forma estratégica para não ter curvas mortas e evitar manobras excessivas e conseqüentemente a perda de tempo. Deve-se escolher uma área sem a presença de árvores, cercas, tocos e pedras que venha a impedir o deslocamento das máquinas.

Para Ripoli (2006), na implantação da colheita mecanizada em cana-de-açúcar é preciso gerenciar um novo sistema, que começa pelo preparo de solo e a sistematização da área. Para ter o máximo de rendimento da colhedora, é preciso ter talhões retangulares e com no mínimo de 600m de comprimento, além de se ter um terreno bem nivelado, paralelismo entre fileiras e um espaçamento adequado de plantio.

Segundo Benedini e Conde (2008), para um melhor aproveitamento da área há necessidade de um planejamento, considerando o formato dos talhões e o traçado dos carregadores, de acordo com o relevo e o solo da área de reforma. Os talhões deverão ter comprimento de sulco em torno de 500 a 700 metros. A largura deverá ser de 150 a 400 m, dependendo da declividade.

O maior custo no plantio e colheita da cana-de-açúcar é com o combustível utilizado para as manobras dos caminhões. Desta forma, este trabalho tem por objetivo planejar de forma otimizada a divisão da área de plantio (terreno) em talhões retangulares, visando a redução de tráfego e manobras das máquinas e caminhões nos talhões de cana-de-açúcar. Para resolver este problema, o mesmo será tratado como um problema de corte bidimensional. As regiões de plantio da cana serão consideradas “*placas*” em estoque e os talhões “*itens*” que devem ser alocados nas placas. Devido a irregularidades das regiões e de áreas em que não é possível realizar o plantio da cana, o corte a ser realizado nas placas deve considerar defeitos, ou seja, nestas regiões não haverá alocação de talhões. Este trabalho é conhecido na literatura como problema de corte de estoque com placa defeituosa.

2. Descrição do problema

Dado que a área disponível para plantio da cana-de-açúcar apresenta uma medida de H hectares, desejamos dividir este terreno cujas dimensões serão (L, W) , em talhões retangulares de dimensões (l_j, w_j) , em que l_j é o comprimento e w_j é a largura do talhão j , $j = 1, \dots, k$. Para aumentar o rendimento da máquina colhedora da cana e reduzir o tráfego e manobras nos talhões durante a colheita, estima-se que as dimensões dos talhões devem satisfazer as restrições $150m \leq l_j \leq 400m$ e $500m \leq w_j \leq 700m$ (Benedini e Conde, 2008).

O problema da determinação e alocação de talhões retangulares na área de plantio da cana-de-açúcar é tratado como um problema de corte de placas defeituosas (Vianna e Arenales, 2006). Nesta abordagem de alocação de talhões à área de plantio, as placas serão o terreno para plantio e os itens são todos os possíveis tipos de talhões que pode ser construído.

De modo geral, a área de alocação dos talhões é irregular, entretanto, será tratada como uma placa retangular. As “falhas” ou “ausência” de terreno serão vistas como defeitos na placa.

Os defeitos também podem representar uma área de preservação, uma estrada, um lago ou qualquer área em que não é permitida a alocação dos talhões. Vianna e Arenales (2006) trataram as áreas defeituosas de forma retangular (Figura 1).



Figura 1. Placa defeituosa e placa a ser tratada.

A Figura 2 a seguir ilustra um terreno para plantio de cana-de-açúcar com defeitos.



Figura 2. Representação de defeitos na alocação de um talhão.

Para evitar tráfego sobre a linha da cana, o espaçamento ideal para a colheita mecanizada é de 1,50m. Assim, sabe-se que o número de manobras das máquinas no talhão é dado por $\frac{\ell_i}{1,5} - 1$.

Utilizando a proposta de Vianna e Arenales (2006) para problema de corte com placas defeituosas, o problema será resolvido com utilizando a abordagem Grafo E/OU (Morabito, 1989).

3. Métodos e abordagens

Para a resolução do problema, inicialmente deve-se gerar um conjunto de talhões com dimensões (l_j, w_j) , respeitando as restrições $150m \leq l_j \leq 400m$ e $500m \leq w_j \leq 700m$.

Com um conjunto de talhões previamente definido, o problema será tratado como um problema de corte com placas defeituosas e resolvido utilizando a abordagem em grafo E/OU (Morabito, 1989) apresentada por Vianna e Arenales (2006) para placas com defeitos.

Um grafo E/OU pode ser definido para representar todos os possíveis padrões do problema de corte, onde os nós representam retângulos (no caso do problema de corte bidimensional, com objeto e peças retangulares) e os arcos representam cortes. De acordo com Vianna e Arenales (2006) padrões de corte numa placa defeituosa são representados num grafo E/OU, análogo ao caso sem defeitos, com algumas modificações. Uma sequência de cortes na placa defeituosa pode ser observada na Figura 3.

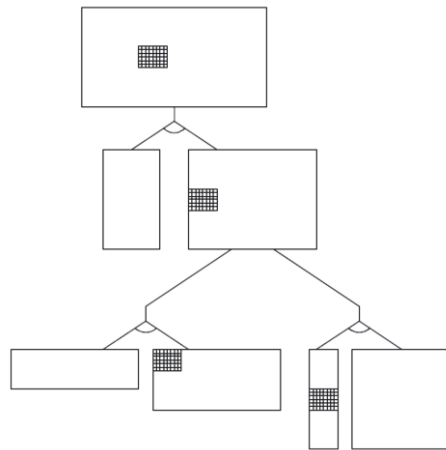


Figura 3. Padrão de corte numa placa defeituosa

Se a região “placa” apresentar mais de um defeito, a sequência de cortes é feita de maneira análoga (Figura 4).

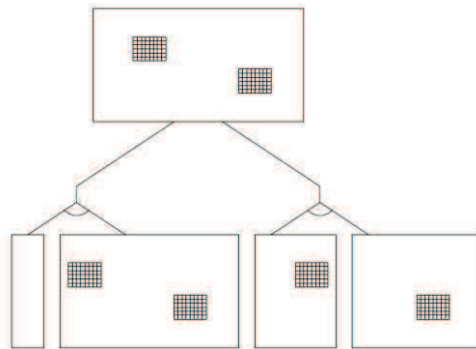


Figura 4. Possíveis cortes numa placa com mais de um defeito

Observe que os defeitos nas placas sempre são reduzidos a retângulos formados ao redor do defeito.

Basicamente, a estratégia para a resolução do Grafo E/OU com placas defeituosas é a mesma apresentada em Morabito (1989). A diferença está no procedimento para a geração do conjunto de discretização da placa e no cálculo do limitante inferior utilizado na ramificação do grafo. Detalhes desta implementação estão em Vianna e Arenales (2006).

4. Resultados

Este trabalho está em fase inicial e encontra-se em estágio de implementação computacional. Serão realizados testes com dados reais.

5. Conclusões e perspectivas futuras

Neste trabalho pretende-se planejar de forma otimizada a divisão da área de plantio de cana-de-açúcar (terreno) em talhões retangulares, visando a redução de tráfego e manobras das máquinas e caminhões nos talhões. Para resolver este problema, o mesmo será tratado como um problema de corte bidimensional com placa defeituosa.

Atualmente teve início a implementação computacional, porém ainda não é possível

apresentar resultados que serão realizados com dados reais.

Outras estratégias para resolução do problema de divisão da área de plantio de cana-de-açúcar em talhões devem ser investigadas e implementadas.

Referências

Companhia Nacional de Abastecimento (CoNAb). Acompanhamento de safra brasileira: cana-de-açúcar, safra 2013/2014 segundo levantamento, agosto. Brasília, 2013.

Empresa de Pesquisa Energética (EPE). Balanço Energético Nacional 2012: Ano base 2011. Rio de Janeiro. 2012.

Benedini, M. S.; Conde, A. J. (2008). Sistematização de área para colheita mecanizada da cana-de-açúcar. Coplana. Guariba, SP, n. 53, p. 23 – 25.

Morabito, R. (1989). Corte de estoque bidimensional. Dissertação de Mestrado, ICMC - USP, SP, Brasil.

Pereira, L. L.; Torrezam, H. F. (2006). *Colheita mecanizada da cana-de-açúcar*. In: Atualização em produção de cana-de-açúcar / organizado por Silvelena Vanzolini Segato [et al.]. Piracicaba. p 333-334.

Ripoli, T. C. C. P.; Casagrandi, D. V.; Ide, B. Y. (2006). *Plantio de cana-de-açúcar: Estado da arte*. Piracicaba: TCC, v. 1, 163-172.

Vianna, A. C. G.; Arenales, M. N. (2006). O problema de Corte de Placas Defeituosas. *Pesquisa Operacional*, 26, 185-202.

UMA HEURÍSTICA PARA UM PROBLEMA DE CARREGAMENTO DE CAIXAS EM UM CONTÊINER COMPARTIMENTADO

Rodolfo Ranck Jr.

Instituto Nacional de Pesquisas Espaciais – INPE
rodolforanck@gmail.com.br

Horacio Hideki Yanasse

Universidade Federal de São Paulo – UNIFESP
horacio.yanasse@unifesp.br

Reinaldo Morabito

Universidade Federal de São Carlos – UFSCAR
morabito@ufscar.br

RESUMO

Neste trabalho aborda-se um problema em que caixas de diferentes tipos têm de ser carregadas nos compartimentos de um contêiner para serem entregues em diferentes destinos de uma rota pré-definida. Este problema aparece, por exemplo, na distribuição de bebidas (empacotadas em “caixas”) usando caminhões com carroçarias compartimentadas. As caixas podem ser empacotadas sem um padrão específico e/ou em camadas horizontais separadas por divisórias, e deve-se atender a aspectos práticos como balanceamento de peso no contêiner, estabilidade do carregamento e resistência a empilhamento das caixas. Busca-se um empacotamento viável que reduza o remanejamento das caixas ao longo da rota. Para resolver este problema, apresenta-se um método de solução heurístico baseado na geração *a priori* de camadas horizontais, e auxiliado pela solução de modelos matemáticos simplificados do problema.

PALAVRAS-CHAVE: Problemas de Carregamento de Contêineres. Problemas de Corte e Empacotamento. Múltiplos Compartimentos.

1. Introdução

O Problema de Carregamento de Contêineres (PCC) é um problema de empacotamento tridimensional em que caixas (itens) precisam ser colocadas em contêineres observando um ou mais critérios. Na literatura, com algumas poucas exceções, os autores consideram itens retangulares e que devem ser arranjados ortogonalmente em contêineres também retangulares. Veja, por exemplo, Morabito e Arenales (1997), Lins et al. (2002), Silva e Soma (2003), Cecilio e Morabito (2004), Araujo e Armentano (2007), e Junqueira et al. (2010).

Nos PCCs, frequentemente, devem-se considerar diversos aspectos práticos de carregamento como o balanceamento de peso no contêiner, a entrega para múltiplos destinos, a estabilidade e a resistência dos itens ao empilhamento. Uma discussão recente das diversas considerações de carregamento pode ser encontrada em Bortfeldt e Wäscher (2012).

Neste trabalho investigamos um problema de carregamento de contêineres motivado de uma indústria de bebidas que precisa distribuir diariamente seus produtos a clientes. Um tipo de veículo largamente utilizado pela empresa é um caminhão que possui, em sua carroçaria, um contêiner multicompartimentado. Este tipo de contêiner é útil, por exemplo, quando se deseja aumentar a estabilidade da carga e/ou separá-la. O caminhão deve cumprir uma rota conhecida e todos os itens demandados pelos clientes dessa rota devem ser empacotados nele. Ao empacotar a carga, devem-se atender diversas restrições práticas como orientação dos itens, estabilidade do

carregamento, resistência dos itens ao empilhamento e balanceamento de peso no contêiner. Deseja-se encontrar um empacotamento viável que minimize o remanejamento dos itens no caminhão ao longo da sua rota.

2. Descrição do problema

Considere um veículo (ou simplesmente um contêiner) que dispõe de um conjunto de compartimentos destinados ao armazenamento da carga, e que podem ser acessados de maneira independente dos seus lados externos. Esses compartimentos são dispostos em fileiras justapostas, possuem a mesma largura e comprimento, porém podem ter alturas diferentes. Há um número variável de fileiras de compartimentos (depende do comprimento do contêiner) no lado maior do contêiner e duas fileiras de compartimentos definidas no lado menor do contêiner.

Os compartimentos e os itens são hexaedros retangulares. Os itens são rígidos, têm dimensões variadas e devem ser colocados ortogonalmente dentro dos compartimentos sem obedecer a um padrão específico e/ou em camadas horizontais. Uma camada é formada por itens de um mesmo tipo e já possui um padrão de empacotamento conhecido para a quantidade máxima de itens que ela pode conter. As camadas são separadas de outras camadas, ou itens, por uma divisória rígida que possui o comprimento e a largura de um compartimento.

O contêiner parte carregado de uma célula logística e sua rota é definida por uma sequência de *destinos* que ele deve cumprir para atender todos os clientes. A rota e as demandas de todos os clientes são conhecidas. Todos os itens demandados devem ser empacotados no contêiner e as seguintes restrições práticas devem ser contempladas:

- a) Orientação dos itens. Os itens têm orientação vertical fixa;
- b) Estabilidade dos itens. Os itens devem ser estabilizados, ao longo de toda rota, por outros itens e/ou pelas paredes presentes no contêiner;
- c) Resistência dos itens a empilhamento. Em qualquer empilhamento, um item abaixo deve ter resistência suficiente para suportar a pressão exercida nele pelos itens acima;
- d) Balanceamento de peso no contêiner. Deseja-se que o contêiner esteja balanceado, ao longo de toda a rota, nos eixos x e y . Diferentemente das restrições (a)-(c), essa restrição é fraca, ou seja, o fato do contêiner estar desbalanceado não o impede de ser utilizado para atender os clientes.

Consideramos como objetivo minimizar o remanejamento dos itens nos compartimentos do contêiner ao longo da sua rota. Um item descarregado no destino k' precisa ser remanejado em um destino k somente se ele estiver sobre itens desse destino k , com $k' > k$. Como as restrições (d) são fracas, também propomos penalizar os desvios de balanceamento de peso no contêiner na função objetivo.

3. Método de solução proposto

O método de solução proposto é uma heurística que busca empacotar os itens em camadas horizontais, separadas por divisórias, com o auxílio de modelos matemáticos. Esta seção descreve resumidamente as partes principais dessa heurística.

Consideramos dois tipos de camadas horizontais: *completas*, camadas repletas de itens de um mesmo tipo e que já possuem um padrão de empacotamento pré-definido para esses itens. Os itens na camada completa não são empilhados; e *falsas*, arranjos de itens que não possuem um padrão de empacotamento pré-definido. Os itens contidos em uma camada falsa (itens *residuais*) podem ser de tipos diferentes e não precisam seguir algum arranjo em específico (podendo, por exemplo, serem empilhados). Por questões de estabilidade, não permitimos que as camadas falsas sejam colocadas abaixo de alguma camada completa. A Figura 1 ilustra um exemplo de camada completa (a) e de camada falsa (b).

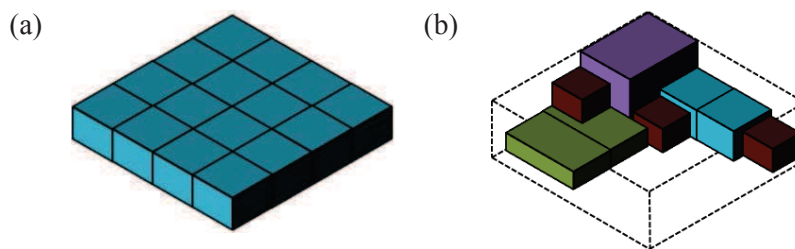


Figura 1 - Exemplo de uma camada completa (a) e falsa (b).

A heurística proposta é baseada na maneira com que a empresa, motivadora deste estudo, coloca os itens no contêiner: empacotando eles em camadas completas sempre que possível.

Inicialmente, geram-se as camadas completas atribuindo itens a elas em ordem inversa a de entrega. Desta maneira, os itens residuais, que só podem ficar no topo dos compartimentos, tendem a serem aqueles demandados nos destinos iniciais. Em seguida, tenta-se gerar as camadas falsas com um procedimento construtivo. Para isso, primeiramente, os itens residuais são ordenados de maneira inversa à ordem de entrega e, como um primeiro critério de desempate utiliza-se a ordem decrescente por altura do item. Como um segundo critério de desempate, utiliza-se a ordem decrescente por resistência a empilhamento do item. Em seguida, tenta-se empacotar todos os itens residuais sequencialmente em uma mesma camada falsa. Após essa tentativa, se ainda existirem itens residuais não empacotados, uma nova camada falsa é gerada e o processo é repetido com essa nova camada e com os itens residuais ainda não empacotados. Para tentar definir a orientação e a posição de um item residual em uma camada falsa utilizam-se duas heurísticas, que são variações das heurísticas de Tarantilis et al. (2009) adaptadas para o nosso problema:

- a) *First Fit* (FFT). As posições disponíveis na camada falsa são mantidas em uma lista em ordem lexicográfica nos eixos z , y e x , nesta ordem. Seleciona-se a primeira posição dessa lista e tenta-se empacotar o próximo item residual nela. Caso nenhuma restrição for violada, o item é empacotado com a sua orientação atual na posição selecionada. Caso contrário, tenta-se empacotá-lo novamente na posição selecionada com a sua outra orientação possível. Caso o item ainda não tiver sido empacotado, seleciona-se a próxima posição da lista, se houver, e repete-se o procedimento para tentar empacotá-lo. Com essa heurística, tenta-se empacotar cada item residual primeiramente nas posições de altura (eixo z) mais baixa da camada falsa, visando minimizar a altura final dessa camada;
- b) *Best Fit* (BFT). Tenta-se empacotar o próximo item residual em todas as posições disponíveis e com todas as orientações possíveis. Se houver, a combinação (posição e orientação) viável que maximizar a área de contato entre as faces laterais desse item e as faces laterais dos itens já empacotados é selecionada. Em seguida, o item é empacotado com essa combinação selecionada. Com essa heurística, busca-se favorecer um empacotamento mais compacto dos itens e, ao mesmo tempo, o empacotamento dos itens em posições mais baixas da camada.

Geradas todas as camadas, tenta-se empacotá-las nos compartimentos resolvendo um Problema de Programação Linear Inteira Mista - PPLIM (PPLI1) proposto em Ranck et al. (2013) que contempla todas as restrições e o objetivo do problema. Este problema pode ser visto como um problema de empacotamento unidimensional com restrições adicionais de empacotamento. Caso uma solução viável para este problema não possa ser obtida, aplica-se um procedimento iterativo que busca gerar um novo conjunto de camadas e, em seguida, empacotá-las resolvendo um novo problema PPLI1.

Se uma solução viável para o problema puder ser obtida, tenta-se melhorá-la por um procedimento iterativo de *busca local*, que consiste em procurar um empacotamento alternativo para os itens residuais. Com esse procedimento, tenta-se atribuir os itens residuais aos

compartimentos resolvendo-se outro PPLIM (PPLI2) e, em seguida, empacotá-los nos seus respectivos compartimentos com um procedimento construtivo similar ao utilizado na geração inicial das camadas falsas, novamente usando as heurísticas FFT e BFT.

4. Resultados

Para avaliar a heurística proposta, 210 exemplares de tamanhos variados foram gerados aleatoriamente, a partir de dados realistas, por um procedimento desenvolvido para esse estudo. Consideraram-se exemplares com até 10 compartimentos e 20 destinos. Todos os PPLIMs foram resolvidos com auxílio do software IBM ILOG CPLEX versão 12.5.1. O tempo máximo permitido para a resolução de qualquer um desses problemas foi de 1200 segundos. O código das implementações foi escrito em C Sharp, compilado em uma plataforma de 64 *bits*, e executado em um computador com processador Intel Core i7-2860QM, 3.6 GHz, com 8GB de memória DDR3 SDRAM.

Ao menos uma solução foi encontrada pela heurística para todos os 210 exemplares avaliados. Em 205 destes 210 casos, uma solução pôde ser obtida com o procedimento de busca local proposto. Em 117 destes 205 casos, as soluções obtidas com esse procedimento de busca local são melhores do que as soluções iniciais (fornecidas com a resolução da formulação PPLI1). Em 107 destes 117 casos, essas soluções são melhores somente para o balanceamento de peso no contêiner, enquanto que nos 10 casos restantes elas também são melhores para o remanejamento de itens. A menor frequência de casos em que as penalizações com remanejamento dos itens são diminuídas com a busca local pode ser explicada pelo fato de os itens que são remanejados serem, na maioria dos casos, de camadas completas, que já têm suas posições definidas na solução inicial. Observe que este fato está relacionado com a estratégia da heurística em definir os itens residuais como os de destinos mais recentes e também em buscar empacotá-los, inicialmente, em um menor número de camadas. A Figura 2 apresenta um exemplo de solução para um dos exemplares avaliados em que 1818 itens, de 11 tipos e de 16 destinos, foram empacotadas em um contêiner com 8 compartimentos. Nesse exemplo, o padrão de empacotamento (arranjo dos itens) dentro de uma camada completa não é explicitado.

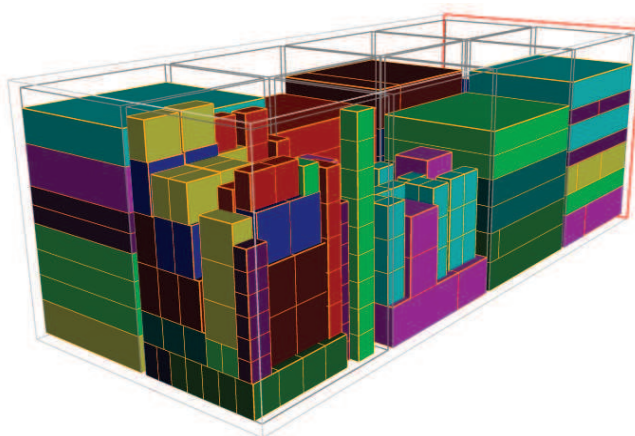


Figura 2 - Visualização de um exemplo de solução para o problema.

5. Conclusões e perspectivas futuras

Apresentou-se uma heurística para a resolução de um PCC encontrado em situações reais, por exemplo, na distribuição de bebidas por meio de caminhões com carroçarias compartimentadas. Esse é um problema de empacotamento tridimensional em que diversos aspectos práticos de carregamento precisam ser atendidos ao longo de toda a rota do contêiner, características que o tornam bastante complexo. Convém salientar que esse é um primeiro estudo

focalizando um problema de carregamento em contêineres com carroçarias compartimentadas no contexto de distribuição de bebidas, e acreditamos que os modelos e a heurística desenvolvidos no estudo possam ser úteis para motivar outras pesquisas que abordem e estendam esse problema.

Uma possível extensão deste problema é avaliar outros objetivos que focalizem a redução do tempo de descarregamento dos itens ao longo da rota, como, por exemplo, buscar manter itens de um mesmo destino em menos compartimentos e/ou em compartimentos mais próximos. Outro possível estudo futuro é a integração do presente PCC com o problema de roteamento de veículos, para que as decisões de roteamento do veículo e de carregamento da carga no contêiner sejam consideradas de maneira integrada.

Os interessados em obter informações detalhadas dos desenvolvimentos abordados neste resumo, podem contatar os autores por e-mail.

Agradecimentos: Os autores agradecem a empresa de bebidas pela colaboração a esta pesquisa; também às agências CNPq, FAPESP e Capes pelo auxílio financeiro.

Referências

- Araujo, O. C. B.; Armentano, V. A.** (2007). A multi-start random constructive heuristic for the container loading problem. *Pesquisa Operacional*, 27(2), 311-331.
- Bortfeldt, A.; Wäscher, G.** (2012). *Container Loading Problems - A State-of-the-Art Review*, No 120007, FEMM Working Papers, Otto-von-Guericke University Magdeburg, Faculty of Economics and Management.
- Cecilio, F. O.; Morabito, R.** (2004). Refinamentos na heurística de George e Robinson para o problema de carregamento de caixas dentro de contêineres. *Transportes*, 11(2), 32-45.
- Junqueira, L.; Morabito, R.; Yamashita, D. S.** (2010). Modelos de otimização para problemas de carregamento de contêineres com considerações de estabilidade e de empilhamento. *Pesquisa Operacional*, 30(1), 73-98.
- Lins, L.; Lins, S.; Morabito, R.** (2002). An n-tet graph approach for non-guillotine packing of n-dimensional boxes into an n-container. *European Journal of Operational Research*, 141(2), 421-439.
- Morabito, R.; Arenales, M.** (1997). Abordagens para o problema do carregamento de contêineres. *Pesquisa Operacional*, 17(1), 29-56.
- Ranck Junior, R.; Yanasse, H.; Morabito, M.** (2013). Contribuições para um problema de Carregamento de Contêiner com Múltiplos Compartimentos. *XLV Simpósio Brasileiro de Pesquisa Operacional – SBPO*, 2366-2377.
- Silva, J. L. C.; Soma, N. Y.** (2003). Um algoritmo polinomial para o problema de empacotamento de contêineres com estabilidade estática da carga. *Pesquisa O.*, 23(1), 79-98.
- Tarantilis, C. D.; Zachariadis, E. E.; Kiranoudis, C. T.** (2009). A hybrid metaheuristic algorithm for the integrated vehicle routing and three-dimensional container-loading problem. *IEEE Transactions on Intelligent Transportation Systems*, 10(2), 255–271.

PROBLEMA DE LOCALIZAÇÃO-ROTEAMENTO-EMPACOTAMENTO COM DEMANDA INCERTA

Thiago A. de Queiroz, José F. Oliveira, Maria A. Carravilla
INESC TEC, Faculdade de Engenharia, Universidade do Porto, Porto, Portugal.
E-mail: taq@ufg.br, jfo@fe.up.pt, mac@fe.up.pt.

Flávio K. Miyazawa
Instituto de Computação, Universidade Estadual de Campinas, Campinas-SP, Brasil.
E-mail: fkm@ic.unicamp.br.

RESUMO

Neste trabalho é investigado o problema integrado de localização e roteamento para o caso em que há restrições de empacotamento bidimensionais e incerteza na demanda dos clientes. O problema é tratado sobre o ponto de vista de cenários com o objetivo de obter uma solução robusta diante dos diferentes cenários. Para tanto, é apresentado um modelo de programação inteira que envolve decisões a nível estratégico (localização dos depósitos), tático (atendimento de clientes por depósitos) e operacional (rotas partindo dos depósitos para os clientes, incluindo o empacotamento dos itens). Cada rota consiste em um empacotamento bidimensional e é validada por um algoritmo de programação por restrições, de forma que um corte é inserido para cada empacotamento inviável.

PALAVRAS-CHAVE: Problema integrado; Localização de facilidades; Roteamento de veículos; Empacotamento bidimensional; Incerteza na demanda.

1. Introdução

Diversas empresas buscam por um melhor planejamento logístico que pode envolver a localização de depósitos e, em seguida, a distribuição partindo dos depósitos para os clientes finais. Segundo aponta Ribeiro (2009), o custo logístico no Brasil equivale a 17% do seu PIB (Produto Interno Bruto) e é componente básico para o preço final dos produtos. Nesta perspectiva, é necessária a resolução de problemas que possibilitem diminuir, se não minimizar, o custo envolvido nos processos de logística. Para isso, é necessário tratar os diferentes aspectos que surgem a nível estratégico, tático e/ou operacional. Um problema que sistematiza este três níveis consiste no problema integrado de Localização e Roteamento (PLR), em que decisões de nível estratégico (decidir onde localizar depósitos), tático (quais clientes atender a partir de cada depósito) e operacional (quais rotas utilizar) são tomadas simultaneamente.

Autores como Laporte e Norbert (1981); Duhamel et al. (2009) e Jarboui et al. (2013) investigaram o PLR apresentando uma estratégia baseada na resolução de programas inteiros, heurísticas baseada em GRASP e heurísticas de busca em vizinhança, respectivamente. Recentemente, Belenguer et al. (2011) propuseram um algoritmo *branch-and-cut* com eficientes rotinas de separação para a versão capacitada do PLR. A estratégia exata que tem apresentado os melhores resultados para a versão capacitada foi desenvolvida por Contardo et al. (2011) e consiste em um algoritmo do tipo *branch-and-cut-and-price*.

Apesar dos esforços da literatura em lidar com problemas integrados, um problema para o qual não encontramos referências envolve integrar no PLR restrições de carregamento bidimensionais (ou tridimensionais). Neste aspecto, a demanda dos clientes é formada por paletes (variando em largura e comprimento) que devem ser organizadas, sem sobreposição, sobre a superfície do veículo de carga.

Soma-se a isto a volatilidade presente no mercado, de forma que incertezas surgem na demanda dos clientes (Petrovic et al., 2008; Cardoso et al., 2013). Outros tipos de incerteza estão ligadas ao custo da manufatura (Zhou et al., 2008), ou ao custo de armazenamento das mercadorias (Giannoccaro et al., 2003).

Neste trabalho investigamos o PLR com restrições de empacotamento sujeito a incerteza na demanda, sendo a incerteza tratada através de uma abordagem por cenários. O objetivo é fornecer um modelo de programação inteira que possa contribuir no projeto e planejamento de operações logísticas considerando a incerteza como um fator inerente às operações. A Seção 2 formaliza o problema em estudo, enquanto que na Seção 3 é apresentado o modelo inteiro. Por fim, conclusões e direções para trabalhos futuros são dadas na Seção 4.

2. Formalização do Problema

No problema de Localização e Roteamento com restrições de Empacotamento Bidimensionais (PLR2E) é dado: um conjunto de possíveis depósitos I , com capacidade b_i e custo de abertura O_i , para cada $i \in I$; um conjunto de clientes J , com R_j sendo itens retangulares com área total a_j e peso total d_j . Além disso, $r \in R_j$ tem dimensões (l_{jr}, h_{jr}) , área a_{jr} peso d_{jr} ; um conjunto de veículos iguais, de capacidade Q , superfície retangular (L, H) e custo operacional F ; um grafo não direcionado $G = (V, E)$, com $V = I \cup J$ sendo os vértices e E as arestas, cada aresta com custo c associado. O grafo é completo nas relações cliente-cliente e depósito-cliente, porém não existe aresta na relação depósito-depósito.

Uma solução do PLR2E consiste em localizar/abrir um subconjunto de depósitos, dos quais partem rotas limitadas pela capacidade do veículo, em que o número de rotas, por depósito, depende da capacidade do respectivo depósito. Cada rota inicia e termina no mesmo depósito e consiste em um empacotamento viável de itens de clientes na superfície retangular do veículo. Toda a demanda dos clientes deve ser suprida, de forma que cada cliente é visitado uma única vez. O objetivo é minimizar o custo total incorrido da abertura dos depósitos, da determinação das rotas e do uso dos veículos. Consequentemente, o número de rotas é uma decisão a ser tratada dentro do problema.

Para lidar com a incerteza na demanda dos clientes é utilizada uma abordagem por cenários. Cada cenário s possui uma probabilidade p_s de ocorrência, satisfazendo $\sum_{s \in \mathcal{P}} p_s = 100\%$, e representa as diferentes demandas que os clientes podem ter. Assim, desejamos obter soluções que sejam robustas diante da volatilidade do mercado, ao mesmo tempo que efetivas no projeto e planejamento das decisões.

3. Formulação Inteira

O modelo inteiro para o problema em estudo é descrito na formulação 1. A notação: $\delta(S)$ representa as arestas com um extremo em $S \subseteq V$ e outro em $V - S$; e a notação $D_s(S) = \lceil \frac{\sum_{j \in S_s} \sum_{r \in R_j} d_{jr}}{Q} \rceil$ representa um limitante inferior no número de veículos para atender a capacidade de peso dos clientes $S \subseteq J$ de acordo com o cenário $s \in \mathcal{P}$; igualmente quando se considera a área dos itens, isto é, $A_s(S) = \lceil \frac{\sum_{j \in S_s} \sum_{r \in R_j} a_{jr}}{A} \rceil$, em que $A = L \times C$.

As variáveis de decisão do modelo, x , y e z , são binárias e $y_i = 1$ indica que o depósito $i \in I$ foi aberto/localizado; $x_{ijs} = 1$, caso o depósito $i \in I$ atenda o cliente $j \in J$ no cenário $s \in \mathcal{P}$; e, $w_{jks} = 1$, se a aresta $\{j, k\} \in E$ é atravessada uma única vez no cenário $s \in \mathcal{P}$. Para modelar rotas que atendem um único cliente, denominadas *rota única*, consideramos o conjunto fictício $I' = I$ de depósitos e arestas $\{i', j\}$ para $i' \in I'$ e $j \in J$. Note que a decisão de abrir um depósito permanece válida para todos os cenários, pois na prática representa uma decisão de longo prazo, com maior custo comparada com as demais.

$$\min \sum_{i \in I} O_i y_i + \sum_{s \in \mathcal{P}} p_s \left(\frac{F}{2} \sum_{i \in I \cup I'} \sum_{j \in J} w_{ijs} + \sum_{\{i,j\} \in E} c_{ij} w_{ijs} \right)$$

sujeito a :

- (i) $\sum_{i \in I} x_{ijs} = 1, \quad \forall j \in J, \forall s \in \mathcal{P}$
- (ii) $\sum_{j \in J} d_j x_{ijs} \leq b_i y_i, \quad \forall i \in I, \forall s \in \mathcal{P}$
- (iii) $\sum_{s \in \mathcal{P}} x_{ij} \leq y_i |S|, \quad \forall i \in I, \forall j \in J$
- (iv) $\sum_{e \in \delta(j)} w_{es} = 2, \quad \forall j \in J, \forall s \in \mathcal{P}$
- (v) $\sum_{i \in I} \lfloor \frac{b_i}{Q} \rfloor y_i \geq D_s(J), \quad \forall s \in \mathcal{P}$
- (vi) $\sum_{j \in J} (w_{ijs} + w_{i'js}) \geq 2 \frac{\sum_{j \in J} d_{js} x_{ijs}}{Q}, \quad \forall i \in I, \forall s \in \mathcal{P}$
- (vii) $\sum_{i \in I \cup I'} \sum_{j \in J} w_{ijs} \geq 2 \max\{D_s(J); A_s(J)\}, \quad \forall s \in \mathcal{P}$
- (viii) $\sum_{s \in \mathcal{P}} \sum_{j \in J} x_{ijs} \geq y_i, \quad \forall i \in I$
- (ix) $\sum_{e \in \delta(S)} w_{es} \geq 2 \max\{D_s(S), A_s(S)\}, \quad \forall S \subseteq J, \forall s \in \mathcal{P}$
- (x) $\sum_{e \in \delta(S)} w_{es} \geq 2(x_{ijs} + y_i - 1), \quad \forall S \subseteq J, \forall j \in S, \forall i \in I, \forall s \in \mathcal{P}$
- (xi) $\sum_{i' \in I'} w_{i'js} \leq 2 - (\sum_{k \in J} w_{jks} + \sum_{i \in I} w_{ijs}), \quad \forall j \in J, \forall s \in \mathcal{P}$
- (xii) $\sum_{k \in J} w_{jks} \geq 2 - (w_{ijs} + w_{i'js}), \quad \forall j \in J, \forall i \in I: i' = i \in I', \forall s \in \mathcal{P}$
- (xiii) $w_{jks} + x_{ijs} \leq 1 + x_{iks}, \quad \forall j, k \in J, \forall i \in I, \forall s \in \mathcal{P}$
- (xiv) $w_{ijs} \leq x_{ijs}, \quad \forall j \in J, \forall i \in I, \forall s \in \mathcal{P}$
- (xv) $w_{i'js} \leq w_{ijs}, \quad \forall j \in J, \forall i \in I: i' = i \in I', \forall s \in \mathcal{P}$
- (xvi) $w_{i'js} \leq x_{ijs}, \quad \forall j \in J, \forall i \in I: i' = i \in I', \forall s \in \mathcal{P}$
- (xvii) $\sum_{e \in R} w_{es} \leq |R| - 1, \quad \forall R \in \mathcal{R}_s, \forall s \in \mathcal{P}$
- (xviii) $y_i \in \{0, 1\}, \quad \forall i \in I$
- (xix) $x_{ijs} \in \{0, 1\}, \quad \forall i \in I, \forall j \in J, \forall s \in \mathcal{P}$
- (xx) $w_{es} \in \{0, 1\}, \quad \forall e \in E, \forall s \in \mathcal{P}$

(1)

Observando a formulação (1), a função objetivo busca minimizar o custo total envolvido na localização dos depósitos mais aquele resultante da probabilidade de ocorrência de cada cenário. O último envolve, para cada cenário, o uso dos veículos, que está relacionado com o número de rotas, e o custo propriamente das rotas. As restrições em (i) garantem que cada cliente, em cada cenário, é atendido por exatamente um depósito, enquanto, em (ii), respeitam a capacidade dos depósitos observando cada cenário. Em (iii) um depósito só pode atender clientes se o mesmo estiver aberto e, em (iv), cada cliente deve ter exatamente duas arestas incidentes, para cada cenário. Existe um número mínimo de depósitos a serem localizados conforme (v) e, então, em (vi), para cada um destes depósitos, observando cada cenário, existe um número mínimo de rotas que devem ser estabelecidas conforme a demanda dos clientes atendidos. Por outro lado, existe um número mínimo total de rotas que devem ser estabelecidas para atender a demanda dos clientes, conforme (vii). Em (viii), se um depósito for aberto, então deve atender pelo menos um cliente.

Para respeitar a capacidade de cada veículo, garante-se em (ix) que o número de arestas no corte para um subconjunto de clientes, em um dado cenário, seja maior que duas vezes a demanda requerida por tais clientes. Em (x), se um cliente é atendido por um depósito em um dado cenário, então deve existir uma rota ligando aquele cliente ao depósito. Por outro lado, em (xi), se existe uma aresta entre um cliente e qualquer outro, então tal cliente não pode fazer parte de uma rota única, enquanto que em (xii), se não existir qualquer aresta entre tal cliente e os demais, então o mesmo deve ser atendido por uma rota única. As restrições em (xiii) obrigam um cliente a ser atendido pelo mesmo depósito de um outro cliente que está conectado com ele. As restrições em (xiv) – (xvi) relacionam as variáveis de forma que uma pode ser ativada somente se a outra também estiver ativa, observando as arestas da relação cliente-depósito. Para lidar com a questão do empacotamento bidimensional, as restrições em (xvii) eliminam o conjunto de rotas em \mathcal{R} , de cada cenário, que não têm um empacotamento viável. Por fim, as restrições (xviii) – (xx) especificam que as variáveis são binárias.

Note que o número de restrições em (ix), (x) e (xvii) pode ser muito grande de forma que usamos rotinas de separação para cada uma delas e as inserimos conforme necessário. As rotinas para (ix) e (x), aplicadas em soluções fracionárias e inteiras, são baseadas em cortes mínimos na árvore de *Gomory-Hu* (Gomory e Hu., 1961) e em adaptações das rotinas propostas por Lysgaard et al. (2004) para o problema de roteamento de veículos capacitados. Para separar (xvii), recaímos em um problema que consiste em encontrar um empacotamento viável bidimensional para um dado conjunto de itens. Abordagens com bons resultados têm sido baseadas em resolver modelos de programação por restrições. Neste caso, usamos o algoritmo proposto por Clautiaux et al. (2008), que é aplicado sempre que uma solução inteira é obtida.

4. Conclusões e Trabalhos Futuros

Neste trabalho é proposto um modelo inteiro para um problema integrado de localização e roteamento com restrições de empacotamento bidimensionais e incerteza na demanda dos clientes. O algoritmo para o modelo foi codificado na linguagem C/C++ e o *solver* Gurobi usado para a sua resolução.

O algoritmo está em fase de validação e testes preliminares usando dados adaptados de problemas reais têm sido efetuados. De imediato, a solução ótima tem sido obtida em um tempo aceitável para as aplicações práticas, já que o problema envolve decisões de longo, médio e curto prazo dentro da cadeia de suprimentos. Estamos trabalhando em novas rotinas de separação para as restrições de capacidade e conexidade das rotas. Além disso, no desenvolvimento de limitantes que possam agilizar o algoritmo que lida com o empacotamento de itens.

Agradecimentos

Os autores agradecem o apoio financeiro recebido das agências de fomento CNPq, FAPEG e FAPESP.

Referências

- Belenguer, J.-M.; Benavent, E.; Prins, C.; Prodhon, C.; Calvo, R. W.** (2011). A branch-and-cut method for the capacitated location-routing problem. *Computers & Operations Research*, 38, 931–941.
- Cardoso, S.; Barbosa-Póvoa, A. P.; Relvas, S.** (2013). Design and planning of supply chains with integration of reverse logistics under demand uncertainty. *European Journal of Operational Research*, 226, 436–451.
- Clautiaux, F.; Jouglet, A.; Carlier, J.; Moukrim, A.** (2008). A new constraint programming approach for the orthogonal packing problem. *Computers & Operations Research*, 35, 944–959.
- Contardo, C.; Cordeaub, J.-F.; Gendron, B.** (2011). A branch-and-cut-and-price algorithm for the capacitated location-routing problem. Technical Report, CIRRELT-2011-44.
- Duhamel, C.; Lacomme, P.; Prins, C.; Prodhon, C.** (2009). A graspxels approach for the capacitated location-routing problem. *Computers & Operations Research*, 37(11), 1912–1923.
- Giannoccaro, I.; Pontrandolfo, P.; Scozzi, B.** (2003). A fuzzy echelon approach for inventory management in supply chains. *European Journal of Operational Research*, 149, 185–196.
- Gomory, R. E.; Hu., T. C.** (1961). Multi-terminal network flows. *Journal of the Society for Industrial and Applied Mathematics*, 9(4), 551–570.
- Jarboui, B.; Derbel, H.; Hanafi, S.; Mladenovic, N.** (2013). Variable neighborhood search for location routing. *Computers & Operations Research*, 40, 47–57.
- Laporte, G.; Norbert, Y.** (1981). An exact algorithm for minimizing routing and operating costs in depot location. *European Journal of Operational Research*, 6, 224–226.
- Lysgaard, J.; Letchford, A. N.; Eglese, R. W.** (2004). A new branch-and-cut algorithm for the capacitated vehicle routing problem. *Mathematical Programming: Series A and B*, 100(2), 423–445.
- Petrovic, D.; Xie, Y.; Burnham, K.; Petrovic, R.** (2008). Coordinated control of distribution supply chains in the presence of a fuzzy customer demand. *European Journal of Operational Research*, 185, 146–158.
- Ribeiro, L. S.** (2009). *Um algoritmo branch-and-price para o problema de localização e roteamento de 3 níveis*. PhD thesis, Programa de Pós-graduação de Engenharia da UFRJ, Rio de Janeiro - RJ.
- Zhou, C.; Zhao, R.; Tang, W.** (2008). Two-echelon supply chain games in a fuzzy environment. *Computers and Industrial Engineering*, 55(2), 390–405.

UM MODELO COMPACTADO PARA O PROBLEMA DE CORTE DE PEÇAS IRREGULARES

Marcos O. Rodrigues

Aline A. S. Leão

Franklina M. B. Toledo

Instituto de Ciências Matemáticas e de Computação, ICMC-USP

okamura@icmc.usp.br

aasleao@icmc.usp.br

fran@icmc.usp.br

RESUMO

Neste trabalho é proposta uma formulação matemática compacta para o problema de corte bidimensional irregular. Esta formulação foi desenvolvida com base no modelo de Toledo et al. (2013). Experimentos computacionais indicam que o modelo compactado é eficiente para instâncias com poucas repetições de peças. Além disso, os resultados mostram que a escolha do conjunto de restrições de não sobreposição é fundamental para a qualidade da solução e a redução do tempo computacional de resolução do modelo.

PALAVRAS-CHAVE: Corte e empacotamento, Corte de peças irregulares, Programação inteira mista

1. Introdução

O problema de corte de peças irregulares, também conhecido como problema de *nesting*, consiste em cortar um conjunto de peças bidimensionais irregulares a partir de objetos maiores. Neste trabalho, consideramos um objeto (placa) de altura fixa e comprimento infinito. O objetivo do problema é minimizar o desperdício de material, isto é, o comprimento da placa utilizado para cortar as peças deve ser minimizado. Segundo Wäscher et al. (2007), o problema de *nesting* pode ser classificado como um problema de corte bidimensional, de dimensão aberta com peças irregulares.

Na literatura há quatro modelos de programação inteira mista para representar o problema. Gomes e Oliveira (2006) exploraram um modelo que utiliza tanto variáveis contínuas para definir o posicionamento das peças, quanto variáveis binárias associadas às arestas dos *no-fit polygons* para evitar a sobreposição entre elas. Fischetti e Luzzi (2009) definem o conceito de fatias associadas ao complemento de um *no-fit polygon* e também utilizam variáveis binárias para assegurar a não sobreposição entre duas peças. Alvarez-Valdes et al. (2013) propõem uma revisão da modelagem de Fischetti e Luzzi (2009) e um método *branch-and-cut* para a resolução do problema. Toledo et al. (2013) apresentam um modelo cujas variáveis de decisão são associadas a cada tipo de peça e a cada ponto de uma malha utilizada para discretizar a placa.

O objetivo deste trabalho é estudar o modelo proposto em Toledo et al. (2013), buscando tornar mais eficiente sua resolução. Como o modelo possui um grande número de restrições de não sobreposição, substituiremos este conjunto por outro com um menor número de restrições, resultando em uma nova formulação para o problema.

2. Descrição do problema

O problema de *nesting* abordado neste trabalho pode ser formalmente descrito da seguinte forma. Seja $\mathcal{P} = 1, \dots, P$ um conjunto de peças bidimensionais (não necessariamente distintas) e uma placa retangular de largura fixa W e de comprimento ilimitado. Além disso, considere que cada peça $i \in \mathcal{P}$ seja representada por um polígono e possua um único ponto de referência $v_i = (x_i, y_i)$ associado, que define a sua localização na placa. Assim, o problema de

nesting consiste em encontrar um plano de corte que posicione todas as peças $i \in \mathcal{P}$ no interior da placa sem provocar sobreposições, de tal forma que o comprimento utilizado z da placa seja mínimo.

3. Modelagem matemática

No modelo de Toledo et al. (2013), para cada tipo de peça t e um ponto da placa d pertencente ao *inner-fit polygon* associado ao tipo de peça t , definimos a seguinte variável binária de decisão:

$$\delta_t^d = \begin{cases} 1, & \text{se uma peça do tipo } t \text{ está alocada ao ponto } d \text{ da placa} \\ 0, & \text{caso contrário} \end{cases} \quad (1)$$

O modelo de Toledo et al. (2013) é dado por:

$$\text{minimizar } z \quad (2)$$

Sujeito a:

$$(c_d g_x + x_t^M) \delta_t^d \leq z \quad \forall t \in \mathcal{T}, \forall d \in \mathcal{IFP}_t \quad (3)$$

$$\sum_{d \in \mathcal{IFP}_t} \delta_t^d = q_t \quad \forall t \in \mathcal{T} \quad (4)$$

$$\delta_t^d + \delta_u^e \leq 1 \quad \forall t, u \in \mathcal{T}, t \leq u, \forall d \in \mathcal{IFP}_t, \forall e \in \mathcal{NFP}_{t,u}^d \quad (5)$$

$$\delta_t^d \in \{0, 1\} \quad \forall t \in \mathcal{T}, \forall d \in \mathcal{IFP}_t \quad (6)$$

$$z \geq 0 \quad (7)$$

em que :

c_d é a coluna associada ao ponto d ;

g_x é a granularidade da malha de pontos para o eixo x ;

\mathcal{T} é o conjunto dos tipos de peça;

x_t^M é a coordenada de maior valor no eixo x dos vértices da peça do tipo t ;

q_t é a demanda das peças de tipo t ;

\mathcal{IFP}_t é o *inner-fit polygon* do tipo de peça t ;

$\mathcal{NFP}_{t,u}^d$ é o *nofit polygon* entre os tipos de peças t e u quando uma peça do tipo t é posicionada no ponto d .

A função objetivo (2) em conjunto com as restrições (3) minimizam o comprimento total da placa utilizado (z) para cortar todas as peças. As restrições (4) garantem que todas as peças serão alocadas completamente na placa, enquanto que as restrições (5) impedem a sobreposição das peças. As restrições (6) e (7) definem, respectivamente, o domínio das variáveis de decisão e da variável associada ao comprimento da placa.

No modelo de Toledo et al. (2013), a matriz de restrições do problema possui uma grande quantidade de linhas devido às restrições de não sobreposição das peças (5), que é no pior caso $\mathcal{O}(T^2 D^2)$, em que T é o número de tipo de peças distintas e D é número de pontos da

malha. Assim, caso seja realizada a substituição das restrições (5) por um conjunto equivalente, é possível reduzir a ordem da matriz do problema, o que pode levar a um problema mais fácil de resolver.

O modelo compactado evita a sobreposição das peças considerando simultaneamente todos os pontos de $\mathcal{NFP}_{t,u}^d$ para um determinado ponto d e um par de tipos de peças t e u , ou seja, faz-se um somatório em $\mathcal{NFP}_{t,u}^d$ das restrições (5) para obter as restrições compactas de não sobreposição, dadas por:

$$\sum_{e \in \mathcal{NFP}_{t,u}^d} \delta_u^e \leq M(1 - \delta_t^d) \quad \forall t, u \in \mathcal{T}, t \leq u, \forall d \in \mathcal{IFP}_t. \quad (8)$$

Nas restrições (8) quando a peça t não é alocada ao ponto d tem-se que $\delta_t^d = 0$, logo, o valor de M deve ser suficientemente grande para permitir que seja alocado aos pontos pertencentes ao conjunto $\mathcal{NFP}_{t,u}^d$ o número máximo possível de peças do tipo u . Portanto, um limitante trivial para M é o valor da demanda das peças do tipo u ($M_u = q_u$).

4. Resultados computacionais

Os testes iniciais foram realizados em um computador com processador Intel Core i7-2600 de 3.4 GHz e 16 GB de memória RAM, utilizando o sistema operacional Ubuntu 13.10 64 bits e o solver CPLEX v12.5 para a linguagem C++ (Concert). Foram considerados três grupos de instâncias RCO (Ribeiro et al., 1999), BLAZEWCZ (Oliveira et al., 2000) e SHAPES (Oliveira et al., 2000), com tempo limite de execução de 3600 segundos e a discretização da malha $g_x = g_y = 1$.

A Tabela 1 apresenta os resultados computacionais obtidos para cada modelo. Como esperado, o modelo proposto tem desempenho melhor que o modelo de Toledo et al. (2013) apenas para as instâncias em que o número de repetições de cada peça é pequeno (RCO1, BLAZEWCZ1 e SHAPES2), o que corrobora a importância de escolher um valor adequado para M . Neste sentido, o modelo compactado apresentou resultados tão bons quanto o modelo de Toledo et al. (2013) para quatro instâncias: RCO1, BLAZEWCZ1, SHAPES2 e SHAPES4, mas não foi capaz de encontrar uma solução factível para as instâncias RCO2-5, BLAZEWCZ3-5, SHAPES5-15 ou provar otimalidade para as instâncias BLAZEWCZ2-3.

Tabela 1: Resultados computacionais obtidos para cada modelo.

Instância	Toledo et al.			Compactado		
	z	$GAP(\%)$	Tempo(s)	z	$GAP(\%)$	Tempo(s)
RCO1	8	0	0.19	8	0	0.14
RCO2	15	0	1.43	-	-	TL
RCO3	22	0	17.15	-	-	TL
RCO4	29	3.45	TL	-	-	TL
RCO5	36	2.78	TL	-	-	TL
BLAZEWCZ1	8	0	0.23	8	0	0.19
BLAZEWCZ2	14	0	13.36	15	26.67	TL
BLAZEWCZ3	21	4.76	TL	22	26.36	TL
BLAZEWCZ4	28	22.51	TL	-	-	TL
BLAZEWCZ5	34	14.71	TL	-	-	TL
SHAPES2	14	0	0.58	14	0	0.29
SHAPES4	27	40.74	TL	27	40.74	TL
SHAPES5	32	34.38	TL	-	-	TL
SHAPES7	-	-	TL	-	-	TL
SHAPES9	-	-	TL	-	-	TL
SHAPES15	-	-	TL	-	-	TL

5. Conclusões e/ou perspectivas futuras

Os resultados computacionais mostram que é necessária uma escolha mais adequada para o valor de M nas restrições (8). Assim, como investigação futura, pretendemos resolver um subproblema cujo objetivo é maximizar o número de peças do tipo u que podem ser posicionadas sobre os pontos de $\mathcal{NFP}_{t,u}^d$ sem sobreposição. A Figura 1 (a) ilustra um exemplo de *nofit polygon* entre um quadrado t e um triângulo u , enquanto a Figura 1 (b) ilustra o número máximo de peças que podem ser posicionadas nos pontos de $\mathcal{NFP}_{t,u}^d$ sem sobreposição.

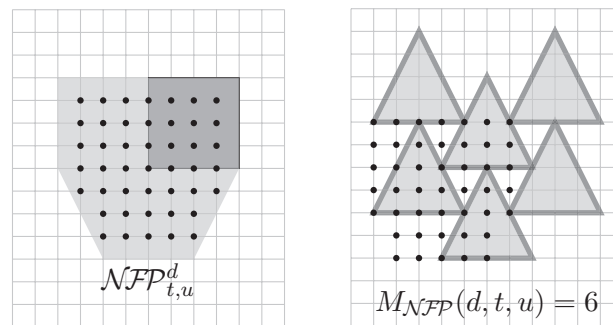


Figura 1: $\mathcal{NFP}_{t,u}^d$ e seu subproblema associado.

Agradecimentos

Este trabalho teve o apoio financeiro do CNPq (300713/2010-0) e da FAPESP (2013/14147-3 e 2010/10133-0).

Referências

- Alvarez-Valdes, R.; Martinez, A.; Tamarit, J.** (2013). A branch & bound algorithm for cutting and packing irregularly shaped pieces. *International Journal of Production Economics*, 145(2), 463 – 477.
- Fischetti, M.; Luzzi, I.** (2009). Mixed-integer programming models for nesting problems. *Journal of Heuristics*, 15(3), 201–226.
- Gomes, A. M.; Oliveira, J. F.** (2006). Solving irregular strip packing problems by hybridising simulated annealing and linear programming. *European Journal of Operational Research*, 171(3), 811 – 829.
- Oliveira, J. F.; Gomes, A. M.; Ferreira, J. S.** (2000). Topos - a new constructive algorithm for nesting problems. *OR-Spektrum*, 22(2), 263–284.
- Ribeiro, C.; Carravilla, M. A.; Oliveira, J. F.** (1999). Applying constraint logic programming to the resolution of nesting problems. *PESQUISA OPERACIONAL*, 19(2), 239–247.
- Toledo, F. M.; Carravilla, M. A.; Ribeiro, C.; Oliveira, J. F.; Gomes, A. M.** (2013). The dotted-board model: A new MIP model for nesting irregular shapes. *International Journal of Production Economics*, 145(2), 478 – 487.
- Wäscher, G.; Haufner, H.; Schumann, H.** (2007). An improved typology of cutting and packing problems. *European Journal of Operational Research*, 183(3), 1109 – 1130.

UM MÉTODO HEURÍSTICO PARA O PROBLEMA DE CORTE DE PEÇAS IRREGULARES

Luiz Henrique Cherri

Instituto de Ciências Matemáticas e de Computação, ICMC, USP
lhcherri@icmc.usp.br

Franklina Maria Bragion de Toledo

Instituto de Ciências Matemáticas e de Computação, ICMC, USP
fran@icmc.usp.br

Maria Antónia Carravilla

Faculdade de Engenharia, Universidade do Porto, FEUP
mac@fe.up.pt

RESUMO

Os problemas de corte e empacotamento são estudados há mais de cinco décadas. Inserido nesta classe de problemas, o problema de corte de peças irregulares é classificado como irregular e bidimensional. Devido à sua dificuldade, os métodos propostos para sua resolução são predominantemente heurísticos e apesar de diversas pesquisas serem desenvolvidas em torno deste problema, poucas destas heurísticas têm sua base em modelos matemáticos. Neste trabalho, abordamos o problema em que as peças são cortadas a partir de uma placa retangular de altura fixa e comprimento a ser minimizado. Propomos um método que combina heurísticas inspiradas na literatura com métodos exatos para a resolução do problema. Testes computacionais comprovam a eficácia do método proposto na busca por soluções factíveis para o problema.

PALAVRAS-CHAVE: Corte irregular, heurística, programação inteira.

1. Introdução

Os problemas de corte e empacotamento aparecem com frequência no planejamento da produção de muitas indústrias. Nestes problemas um ou mais objetos devem ser divididos em peças menores. O problema de corte de peças irregulares é um caso especial dos problemas de corte e empacotamento. O problema abordado neste trabalho consiste em cortar peças irregulares a partir de uma placa retangular de largura fixa e comprimento ilimitado. O objetivo é minimizar o comprimento total utilizado da placa.

O problema de corte de peças irregulares faz parte do grupo de problemas de otimização combinatória classificados como NP-completo (Fowler et al., 1981). Devido à sua complexidade de resolução poucos trabalhos o abordam de forma exata. Carravilla et al. (2003) propuseram a primeira abordagem exata para o problema utilizando um método de solução baseado em enumeração implícita e programação por restrições. Modelos matemáticos para a representação do problema nos quais o posicionamento das peças é realizado utilizando variáveis contínuas e a análise de sobreposição entre as peças feita com auxílio de variáveis binárias são apresentados em Gomes e Oliveira (2006), Fischetti e Luzzi (2009) e Alvarez-Valdes et al. (2013). Um modelo matemático que trata o posicionamento das peças na placa sobre discretizada por uma malha de pontos preestabelecida foi proposto por Toledo et al. (2013). Com esta abordagem foi possível obter soluções ótimas para instâncias de médio porte, sujeitas à discretização utilizada.

Em contraste ao pequeno número de métodos exatos para a resolução do problema, diversos métodos heurísticos foram propostos. Estes métodos podem ser classificados como heurísticas construtivas ou heurísticas de melhoria. Heurísticas construtivas visam a criação de uma solução inicial para o problema. Heurísticas de melhoria têm por objetivo aumentar a qualidade de soluções já existentes para o problema. Uma revisão detalhada sobre métodos heurísticos é feita em Bennell e Oliveira (2009).

Neste trabalho combinamos métodos heurísticos e exatos para o desenvolvimento de um método heurístico para a resolução do problema com o objetivo de obter um bom compromisso entre qualidade e tempo de solução. A heurística é composta por fases de construção e melhoria da solução. Testes computacionais foram realizados e os resultados obtidos são comparados aos apresentados por Toledo et al. (2013).

2. Descrição do problema

O problema de corte de peças irregulares abordado consiste em cortar peças, convexas e não convexas, a partir de uma placa de altura fixa e comprimento infinito. O objetivo do problema é de cunho econômico e visa minimizar o comprimento total utilizado da placa. Diversas indústrias possuem esse problema em parte de seu planejamento de produção, tais como: confecções; metal mecânica; moveleira; calçadista, entre outras. Uma solução é factível para o problema se: a) as peças não se sobrepõem; b) as peças são alocadas inteiramente dentro da placa; e c) todas as peças são cortadas.

Um modelo matemático inteiro misto para a resolução deste problema foi proposto por Toledo et al. (2013). Neste modelo a placa é representada por uma malha de pontos. Cada peça é definida por um conjunto de vértices e um vértice deste conjunto é selecionado para ser seu ponto de referência. Os pontos de referência das peças podem ser alocados somente sobre os pontos da malha de pontos. Para verificar se há sobreposição entre as peças e garantir que as mesmas estão inteiramente dentro da placa são utilizadas as estruturas *nofit polygon* e *inner fit polygon* (Bennell e Oliveira, 2008).

3. Métodos e abordagens

Neste trabalho, desenvolvemos um método heurístico baseado no modelo de Toledo et al. (2013) para a resolução do problema de corte de peças irregulares com o objetivo de obter soluções de boa qualidade para o problema em tempo computacional aceitável. A heurística proposta pode ser dividida em duas partes: a) construção de solução e b) melhoria da solução. Na primeira parte, o problema é resolvido em etapas. Na primeira etapa, um problema no qual apenas uma parte das peças deve ser alocada na placa é resolvido. O número de peças cuja demanda é atendida cresce a cada etapa. Em cada etapa, é solucionado um problema com parte das peças fixas na placa, parte das peças com movimentação restrita e parte das peças livres. Ao final de cada etapa, o conjunto de peças que era livre passa a ter restrições sobre seus movimentos, um conjunto de peças que tinha a possibilidade de se movimentar parcialmente é fixado e é adicionado um novo conjunto de peças para serem livremente alocadas. No final do processo temos uma solução para o problema.

Na segunda parte, a partir da solução obtida, adicionamos uma nova restrição ao modelo de Toledo et al. (2013). Assim, permitimos que metade das peças alocadas na solução encontrada na primeira parte possa mudar de posição, e então resolvemos o problema em busca de uma solução de melhor qualidade. Se uma solução de melhor qualidade for encontrada a etapa é repetida, caso contrário o processo chega ao fim.

4. Resultados

Testes foram conduzidos em um computador com processador Intel Core i7-2600 3.4 GHz, 16 Gb de memória e sistema operacional Ubuntu 12.04. Os métodos foram implementados em linguagem C/C++ e, para a resolução dos subproblemas, foi utilizado o software de otimização CPLEX 12.5. O modelo de Toledo et al. (2013) foi implementado nesta mesma plataforma para realizarmos uma avaliação justa entre o método heurístico e o método exato. As instâncias utilizadas são as mesmas utilizadas por Toledo et al. (2013). As instâncias Bi e $BP2P4_{j_k}$ são compostas respectivamente por sete e dois tipos de peças.

Na Tabela 1 apresentamos os resultados computacionais. As três primeiras colunas da tabela apresentam respectivamente a identificação das instâncias, o número total de peças e os limitantes superiores utilizados. As colunas 4, 5 e 6, apresentam os resultados obtidos pelo modelo matemático de Toledo et al. (2013), contendo o valor da solução, o tempo computacional para encontrar a melhor solução factível e o tempo decorrido na tentativa de provar de otimalidade. O valor das soluções e os respectivos tempos utilizando somente a primeira fase da heurística são apresentados nas colunas 8 e 9. Por fim, as colunas 10 e 11, apresentam o valor das soluções e tempos computacionais para resolução do problema com heurística completa.

Tabela 1: Resultados computacionais

Instância	Nº total de peças	Lim. superior	Toledo et al. (2013)			Fase construtiva		Fase construtiva com melhoria	
			Sol.	T. (s)	T. (opt.)	Sol.	T. (s)	Sol.	T. (s)
B1	7	8	8	1	1	8	1	8	1
B2	14	15	14	16	19	15	10	14	12
B3	21	33	20	1885	2002	22	30	21	36
B4	28	38	28	336	TL	28	71	28	116
B5	35	37	35	1478	TL	36	217	35	273
BP2P4_4_3	7	60	11	1	1	11	0	11	0
BP2P4_7_7	14	60	19	2	1	19	0	19	0
BP2P4_11_10	21	60	28	2	3	28	2	28	2
BP2P4_14_14	28	60	38	3	50	38	5	38	6
BP2P4_18_17	35	60	47	3	208	47	5	47	8
BP2P4_21_21	42	60	56	3	364	56	6	56	13
BP2P4_28_28	56	100	74	3	TL	75	12	74	39
BP2P4_35_35	70	100	93	3	TL	95	13	94	110

Comparando os resultados obtidos ao final da primeira e da segunda fase da heurística, podemos notar que, para as instâncias com dois tipos de peças, 2 soluções foram melhoradas e, para as instâncias com sete tipos de peças, 3 soluções foram melhoradas. Ainda, o valor das soluções obtidas em Toledo et al. (2013) e pela segunda fase do método heurístico só diferem para B3 e BP2P4_35_35.

Em relação ao tempo computacional, para as instâncias com dois tipos de peças, os resultados obtidos pela primeira e segunda fase do método heurístico podem diferir em mais de 8 vezes (BP2P4_35_35). Para as instâncias com sete tipos de peças, a diferença dos tempos de execução entre a primeira e da segunda fase da heurística é menos significativa.

Analisando os tempos computacionais para o modelo encontrar a melhor solução do processo de busca e para a segunda fase do método heurístico obter uma solução, notamos que para dois tipos de peças o modelo obtém a melhor solução em poucos segundos, o que não justifica o uso da heurística nesses casos. Para as instâncias com sete peças, a heurística obtém a solução mais rápido que o método exato, indicando que a heurística pode ser útil na resolução de problemas com maior número de peças. Vale ressaltar que o método proposto, conseguiu encontrar uma solução de boa qualidade para todas as instâncias apresentadas em menos de

cinco minutos.

5. Conclusões

Um método heurístico para a resolução do problema de corte de peças irregulares foi proposto. O método é composto por duas fases: 1) construção de uma solução; e 2) melhoria da solução. Desta forma, foi possível obter soluções de boa qualidade para o problema em menos de cinco minutos, o que abre os horizontes para a exploração de problemas de maior porte ou então para a utilização de uma malha de cálculo mais precisa para a resolução do problema.

Agradecimentos

Os autores agradecem o apoio financeiro da FAPESP(2012/18653-8 e 2010/10133-0) e CNPq (300713/2010-0).

Referências

- Alvarez-Valdes, R.; Martinez, A.,; Tamarit, J.** (2013). A branch & bound algorithm for cutting and packing irregularly shaped pieces. *International Journal of Production Economics*, 145(2), 463 – 477.
- Bennell, J. A.; Oliveira, J. F.** (2008). The geometry of nesting problems: A tutorial. *European Journal of Operational Research*, 184(2), 397 – 415.
- Bennell, J. A.; Oliveira, J. F.** (2009). A tutorial in irregular shape packing problems. *Journal of the Operational Research Society*, 60(60), S93–S105.
- Carravilla, M. A.; Ribeiro, C.,; Oliveira, J. F.** (2003). Solving nesting problems with non-convex polygons by constraint logic programming. *International Transactions in Operational Research*, 10(6), 651–663.
- Fischetti, M.; Luzzi, I.** (2009). Mixed-integer programming models for nesting problems. *Journal of Heuristics*, 15(3), 201–226.
- Fowler, R. J.; Paterson, M.,; Tanimoto, S. L.** (1981). Optimal packing and covering in the plane are np-complete. *Inf. Process. Lett.*, 12(3), 133–137.
- Gomes, A.; Oliveira, J. F.** (2006). Solving irregular strip packing problems by hybridising simulated annealing and linear programming. *European Journal of Operational Research*, 171(3), 811–829.
- Toledo, F. M. B.; Carravilla, M. A.; Ribeiro, C.; Oliveira, J. F.,; Gomes, A. M.** (2013). The dotted-board model: a new mip model for nesting irregular shapes. *International Journal of Production Economics*, 145(2), 478 – 487.

Sessão 4

Restrições do tipo multi-commodity flow aplicadas ao problema integrado de dimensionamento e sequenciamento da produção.

Autores: Michelli Maldonado, Socorro Rangel.

Planejamento e programação da produção na indústria de polpa moldada.

Autores: Karim Pérez Martínez, Reinaldo Morabito, Eli Vitor Toso

Uma abordagem de solução híbrida baseada em algoritmo genético para o problema de planejamento e sequenciamento da produção em indústria de papel e celulose.

Autores: Marcos Mansano Furlan, Bernardo Almada-Lobo, Maristela Oliveira Santos, Reinaldo Morabito.

Uma formulação não-linear para a restrição de ciclos da serra no problema integrado de dimensionamento de lotes e corte de estoque.

Autores: Gislaine Mara Melega, Silvio Alexandre de Araujo e Socorro Rangel.

Problema integrado de produção e distribuição: uma abordagem heurística.

Autores: Márcio Antônio Ferreira Belo Filho, Pedro Amorim, Bernardo Almada-Lobo.

Modelos híbridos para o Problema da Mochila Bidimensional com conflitos.

Autores: Pedro Hokama, Thiago A. de Queiroz, Flavio Keidi Miyazawa.

Otimização estocástica na programação de bombas de redes de abastecimento urbano.

Autores: Jonathan Justen de La Vega Martínez, Douglas Alem.

Restrições do tipo *Multi-Commodity Flow* Aplicadas ao Problema Integrado de Dimensionamento e Sequenciamento da Produção

Michelli Maldonado, Socorro Rangel

Univ. Estadual Paulista - UNESP/Ibilce Depto. Matemática Aplicada São José do Rio Preto - SP
michelli@ibilce.unesp.br, socorro@ibilce.unesp.br

RESUMO

O objetivo deste trabalho é propor um modelo matemático para o Problema Integrado de Dimensionamento de Lotes e Sequenciamento da Produção (PIDS), considerando um processo de produção que envolve um estágio, uma máquina e com tempos e custos de preparo dependentes da sequência. Um exemplo ilustrativo é utilizado para estudar o comportamento computacional do modelo proposto comparado com um modelo da literatura.

Dimensionamento de Lotes, Sequenciamento da Produção, Problema do Caixeiro Viajante Assimétrico, *Multi-Commodity Flow*.

1. Introdução

Decisões de planejamento da produção sempre exerceram grande influência sobre o desempenho das empresas. Questões como dimensionamento e sequenciamento de lotes, adequação dos níveis de estoque, atendimento da demanda de mercado e a consequente sincronização de todos esses processos são ainda hoje desafios presentes na cadeia de suprimentos.

O processo de dimensionar lotes de produção consiste em determinar quanto produzir de cada produto em cada período a fim de atender a uma demanda prevista sob as condições e capacidades operacionais existentes. Um dimensionamento incorreto dos lotes pode causar excesso de produto acabado em estoque, ou pedidos de venda não totalmente atendidos e eventuais perdas de material perecível. Por sua vez, sequenciar lotes de produção significa determinar em que ordem produzir esses lotes de forma a melhor aproveitar os recursos produtivos e atender aos prazos estabelecidos. Um sequenciamento ineficiente pode provocar o não cumprimento dos prazos, reduzir significativamente a capacidade da linha de produção, acumular antecipadamente estoques de produtos acabados, gerar custos maiores de preparação de máquinas, entre outros (Pinedo (1995)).

Em muitos processos industriais, as decisões de dimensionamento de lotes e de sequenciamento da produção são tomadas simultaneamente para que a utilização da capacidade da máquina seja eficiente. (Clark et al. (2011)).

Uma estratégia para integrar o dimensionamento de lotes com o sequenciamento da produção é incluir restrições do Problema do caixeiro Viajante Assimétrico (PCVA) no modelo de dimensionamento de lotes. Existem diversas formulações para o PCVA que podem ser usadas para modelar as decisões de sequenciamento. Essas formulações diferem-se principalmente na estratégia usada para representar as restrições de eliminação de subrotas (Oncan et al. (2009)). Duas classes de restrições de eliminação de subrotas são consideradas clássicas, uma delas foi proposta por Miller et al. (1960) e a outra foi proposta por Dantzig et al. (1954), denominadas aqui por MTZ e DFJ, respectivamente. A formulação DFJ apresenta um número exponencial de restrições porém fornece uma boa relaxação linear. A atratividade das restrições MTZ é que ela possui uma representação polinomial compacta. No entanto, sabe-se que as restrições MTZ produzem uma relaxação linear fraca. As restrições clássicas já foram usadas em diversas

aplicações para modelar as decisões de sequenciamento do PIDS, por exemplo, container de vidro (Almada-lobo et al. (2007)), fábrica de nutrição animal (Toso e Morabito (2005)), indústria de bebidas (Ferreira et al. (2010) e Defalque et al. (2011)).

Outra classe de restrições de eliminação de subrotas para o PCVA foi proposta por Claus (1984), conhecidas por *Multi-Commodity Flow* (MCF). As MCF tem representação polinomial e em Orman e Williams (2007) é apresentado um estudo que prova que a relaxação linear obtida pela MCF é equivalente a relaxação linear obtida pela DFJ. As *Multi-Commodity Flow* são usadas em Clark et al. (ress) para modelar as decisões de sequenciamento na presença de tempos e custos de preparação não-triangulares. O objetivo desse trabalho é apresentar uma formulação para o PIDS usando as restrições MCF para modelar as decisões de sequenciamento.

Referências

- Almada-lobo, B.; Klabjan, D.; Antónia carravilla, M.; Oliveira, J. F.** (2007). Single machine multi-product capacitated lot sizing with sequence-dependent setups. *International Journal of Production Research*, 45(20), 4873–4894.
- Clark, A.; Almada-Lobo, B.; Almeder, C.** (2011). Lot sizing and scheduling: industrial extensions and research opportunities. *International Journal of Production Research*, 49(9), 2457–2461.
- Clark, A. R.; Mahdieh, M.; Rangel, S.** (2014 in press.). Production lot sizing and scheduling with non-triangular sequence-dependent setup times. *International Journal of Production Research*.
- Claus, A.** (1984). A new formulation for the traveling salesman problem. *SIAM journal on algebraic and discrete methods*, 5(1), 21–25.
- Dantzig, G.; Fulkerson, R.; Johnson, S.** (1954). Solution of a large-scale traveling-salesman problem. *Journal of the Operations Research Society of America*, 2(4), 393–410.
- Defalque, C. M.; Rangel, S.; Ferreira, D.** (2011). Usando o ATSP na modelagem do problema integrado de produção de bebidas. *Tendências em Matemática Aplicada*, 12(3), 195–209.
- Ferreira, D.; Morabito, R.; Rangel, S.** (2010). Relax and fix heuristics to solve one-stage one-machine lot-scheduling models for small-scale soft drink plants. *Computers & Operations Research*, 37(4), 684–691.
- Miller, C. E.; Tucker, A. W.; Zemlin, R. A.** (1960). Integer programming formulation of traveling salesman problems. *J. ACM*, 7(4), 326–329.
- Oncan, T.; Altinel, A. K.; Laporte, G.** (2009). A comparative analysis of several asymmetric traveling salesman problem formulations. *Computers & Operations Research*, 36(3), 637–654.
- Orman, A.; Williams, H.** (2007). A survey of different integer programming formulations of the travelling salesman problem. *E.J. Kontoghiorghes, C. Gatu (Eds.), Optimisation, economics and financial analysis. Advances in computational management science*, 9, 93–106.
- Pinedo, M.** (1995). *Scheduling - Theory, Algorithms and Systems*. Prentice Hall.
- Toso, E. A. V.; Morabito, R.** (2005). Otimização no dimensionamento e sequenciamento de lotes de produção: estudo de caso numa fábrica de rações. *Gestão & Produção*, 12(2), 203–217.

PLANEJAMENTO E PROGRAMAÇÃO DA PRODUÇÃO NA INDÚSTRIA DE POLPA MOLDADA

Karim Yaneth Pérez Martínez

Universidade Federal de São Carlos

karim.1504@gmail.com

Reinaldo Morabito Neto

Universidade Federal de São Carlos

morabito@ufscar.br

Eli Angela Vitor Toso

Universidade Federal de São Carlos - *Campus Sorocaba*

eli@ufscar.br

RESUMO

O objetivo deste projeto de pesquisa é o estudo das decisões envolvidas no planejamento e programação da produção na indústria de polpa moldada. Estas decisões estão relacionadas com a determinação dos processos a partir dos quais são produzidos os diferentes produtos, e o dimensionamento e sequenciamento de lotes de tais processos. Estudos anteriores abordaram o dimensionamento e sequenciamento de lotes neste tipo de indústria, propondo modelos que se mostraram adequados e eficientes para exemplares reais, no entanto, as decisões de geração de processos não foram consideradas. Na tentativa de propor abordagens de planejamento eficientes, que incorporem mais aspectos da realidade e plausíveis de serem implementados na prática, esta pesquisa pretende abordar de forma integrada as decisões de geração de processos e dimensionamento e sequenciamento de lotes na indústria de polpa moldada, propondo novos modelos e estudando métodos de solução híbridos que auxiliem as atividades de planejamento e programação.

PALAVRAS-CHAVE: Geração de processos, dimensionamento e sequenciamento de lotes, planejamento e programação da produção, indústria de polpa moldada.

1. Introdução

A preocupação com a sustentabilidade dos processos industriais, com a não geração de resíduos e a conservação do meio ambiente está em constante aumento no mundo, fazendo com que as atividades relacionadas à recuperação de valor de produtos pós-consumo ganhem cada vez mais espaço. A produção de produtos em polpa moldada, ou fibra moldada, representa um exemplo dos processos de produção que contribuem para o reaproveitamento de materiais fora de uso, especificamente no setor de reciclagem de papel. A polpa moldada é obtida pelo processo de desagregação ou separação das fibras de aparas de papel, que misturadas à água e a produtos químicos, formam uma massa (polpa). Esta polpa pode ser moldada para dar origem a diversos produtos como: bandejas para acondicionamento, transporte e proteção de hortifrutigranjeiros; embalagens; calços para proteção de equipamentos eletrônicos, elétricos e de madeira; entre outras aplicações (COMPROMISSO EMPRESARIAL PARA RECICLAGEM - CEMPRE, 2013). Este processo de fabricação envolve um alto consumo de aparas de papel que são a principal matéria prima, contribuindo para recuperação de valor de papel pós-consumo.

O Brasil está entre os maiores produtores de papel, registrando uma produção de mais de 10 milhões de toneladas em 2012 (BRACELPA, 2014). Em geral, a produção de papel no Brasil tem crescido durante as últimas décadas, registrando um crescimento médio anual de aproximadamente 5,5% desde o ano 1970 até o ano 2012. Na medida em que a produção neste setor cresce, a preocupação para a recuperação dos volumes de papel destinados ao consumo doméstico também aumenta. Atualmente, o Brasil registra uma taxa de aproveitamento de 45,5% do consumo aparente de papel, enquanto a Coréia, por exemplo, registrou em 2010 uma taxa de recuperação de 91,6% do volume consumido (BRACELPA, 2014). De acordo com a Associação Nacional dos Aparistas (ANAP), as atividades de reutilização do papel geram dupla receita para o governo, que recebe também um benefício extra ao não ter custo com o descarte do material (ANAP, 2014). Além disso, os produtos que utilizam polpa moldada consistem em uma alternativa econômica e ecologicamente mais adequada para uma ampla gama de produtos, podendo substituir outros tipos de materiais como EPE (Polietileno Expandido), EPS (Poliestireno Expandido ou Isopor), papelão e madeira (EUROPEAN MOULDED FIBRE ASSOCIATION - EMFA, 2013).

Todas estas características e informações ressaltam a importância econômica e ambiental das atividades de produção próprias da indústria de polpa moldada. Neste contexto, e motivados por tal importância do setor, esta pesquisa procura determinar abordagens que auxiliem as decisões relacionadas ao planejamento e programação da produção de modo eficiente, realista e plausíveis de serem implementados na prática.

2. Descrição do problema

O processo de produção de produtos de polpa moldada envolve as etapas de formação da fibra, moldagem e secagem. Dependendo do uso e finalidade dos produtos finais, o processo de produção completo pode envolver etapas adicionais de acabamento do material e personalização dos produtos, ou condições especiais para garantir higiene no processo de produção. Na etapa de formação da fibra, as matérias-primas geralmente formadas por aparas de papel jornal, revistas, papel sulfite, entre outros, são misturadas no com água, pasta mecânica e outros agentes químicos no equipamento apropriado (*Hydrapulper*) para a formação da polpa. Na etapa de moldagem, a fibra passa por um equipamento apropriado que contém o conjunto de moldes que dão o formato da embalagem à polpa, através de operações de pressão e sucção do excesso de água. Finalmente, o material passa pela etapa de secagem, onde é submetido a altas temperaturas durante um tempo determinado, eliminando a umidade do material, deixando-o apto para seu uso final. Este processo de produção pode ser considerado como um processo monoestágio apesar de envolver diversas etapas, devido à continuidade do material desde a entrada da matéria-prima, até a obtenção dos produtos, sem estoques intermediários.

A característica deste processo de produção que apresenta maiores desafios para as atividades de planejamento e programação da produção encontra-se na etapa de moldagem. A moldagem dos diferentes produtos é realizada a partir de ferramentais específicos que consistem em um conjunto de moldes de diversos formatos, e que são acoplados a um sistema de rotores sincronizados. Estes ferramentais são chamados de padrões de moldagem, os quais podem conter moldes para um, dois ou mais tipos de produtos diferentes, como acontece, por exemplo, no processo de produção embalagens em fibra moldada para acondicionamento de ovos e frutas (PÉREZ, 2013).

Os tempos e custos de preparação neste tipo de indústria são significativos e requerem mão de obra especializada. No caso da produção de embalagens para acondicionamento de ovos e frutas, os tempos de preparação podem ser bastante elevados, variando entre 20 minutos e 48 horas de acordo com a sequência em que os padrões de moldagem são programados, e envolvendo altos custos por hora dedicada a estas operações. Estas características, somado à alta demanda que caracteriza a produção de produtos de polpa moldada, resalta a importância de gerar planos de produção que aproveitem de forma eficiente a capacidade e recursos disponíveis, de modo a ampliar a cobertura do serviço e reduzir os custos envolvidos.

Em pesquisa anterior (PÉREZ, 2013), constatou-se que as atividades de planejamento e programação da produção na indústria de polpa moldada estão principalmente relacionadas com a escolha dos padrões de moldagem a serem utilizados, o tempo durante o qual devem ser usados, e a sequência em que devem ser programados. Para representar e resolver o problema estudado foram propostas duas abordagens de modelagem baseadas no Problema de Dimensionamento de Lotes Capacitado (CLSP) e o Problema de Dimensionamento e Sequenciamento de Lotes Geral (GLSP). Estas abordagens demonstraram vantagens significativas quanto à redução de custos e aproveitamento da capacidade, quando comparados com o planejamento realizado em uma fábrica do setor. Neste trabalho, foram considerados padrões de moldagem definidos *a priori*, ou seja, a decisão sobre a configuração de padrões para moldagem não foi incorporada no problema de planejamento da produção.

Considerando as vantagens que podem ser obtidas através do planejamento eficiente da produção e a importância econômica e ambiental da produção de produtos de polpa moldada, esta pesquisa pretende estudar abordagens de modelagem integradas que representem as decisões de geração de processos, que consiste na geração de padrões de moldagem nesta indústria em particular, e as decisões de planejamento da produção. As decisões relacionadas à geração de padrões de moldagem consistem em determinar a conformação e arranjo dos moldes individuais de diferentes produtos em um único ferramental, assemelhando-se aos problemas de empacotamento, sujeito à disponibilidade limitada dos moldes individuais e às especificações técnicas envolvidas. Já as decisões de planejamento da produção consistem no dimensionamento e sequenciamento de lotes de processo, em que cada padrão de moldagem representa um processo a partir do qual são obtidos diferentes produtos. As decisões envolvidas nestas atividades consistem em definir quais padrões de moldagem utilizar, o tempo de produção de cada um deles, e a designação e sequenciamento dos mesmos em cada linha de produção. Todas estas decisões devem ser abordadas, obedecendo às restrições do sistema de produção estudado e otimizando o recursos disponíveis.

3. Métodos e abordagens

Esta pesquisa se fundamenta no pressuposto de que podemos construir modelos que expliquem total ou parcialmente o comportamento dos processos operacionais de ambientes reais, e que podemos também capturar os problemas de tomada de decisão envolvidos nestes processos. Segundo Bertrand e Fransoo (2002) e Morabito e Pureza (2009), esta pesquisa pode ser caracterizada como empírica prescritiva, visto que propõe principalmente criar um modelo que se ajuste às observações e ações no sistema real, desenvolvendo estratégias, políticas e ações para melhorar a situação atual.

A metodologia de pesquisa proposta para este trabalho se assemelha às etapas definidas por Hillier e Lieberman (2002) para os estudos em pesquisa operacional. O primeiro passo envolve a definição do problema abordado, a caracterização do sistema de produção estudado e a apresentação dos dados e informações relevantes relacionadas.

Em seguida é proposto um modelo matemático para representar o problema, que deve-se adequar corretamente às considerações particulares do sistema de produção. Nesta etapa será tomado como ponto de partida formulações propostas em pesquisas anteriores relacionadas à indústria de embalagens em polpa moldada (PÉREZ, 2013). Nestas pesquisas constatou-se que modelos baseados nas formulações clássicas do Problema de Dimensionamento de Lotes Capacitado (CLSP) baseados nos modelos propostos por Almada-Lobo et. al (2007) e o Problema de dimensionamento de Lotes Geral (GLSP) baseados nos modelos de Meyr (2002) e Ferreira et al. (2012) são adequadas para o problema em estudo, sendo capazes de gerar planos de produção eficientes em tempos computacionais viáveis para as decisões no nível tático-operacional envolvidas no dimensionamento e sequenciamento de lotes. Portanto, estas abordagens fornecem uma base para a incorporação das decisões relacionadas à geração de processos de produção

típicos na indústria de polpa moldada, as quais fazem parte do planejamento e programação da produção do sistema estudado.

A seguir, serão realizados estudos computacionais a partir da resolução do modelo proposto, sobre exemplares reais do sistema de produção particular, verificando a necessidade de efetuar ajustes no modelo e a consideração de métodos de solução mais eficazes. Em seguida, será implementado o método de solução definido para o problema em estudo em uma linguagem de programação de baixo nível. Igual que na validação dos modelos propostos, serão realizados estudos computacionais para o delineamento e experimental e análise do desempenho computacional do método proposto.

Por fim, os resultados dos estudos computacionais serão apresentados, analisados e comparados com informações reais, a fim de verificar a eficiência e eficácia da abordagem proposta em relação ao desempenho computacional e geração de melhores planos de produção viáveis e eficientes em configurações produtivas reais.

4. Análise dos resultados

Os resultados obtidos através desta pesquisa serão analisados em dois aspectos principais que compreendem a validação prática dos resultados obtidos, e a análise do desempenho computacional das abordagens propostas.

A validação prática está relacionada com a análise dos resultados das abordagens propostas com colaboração dos responsáveis do processo de planejamento da produção em empresas do setor. Desta forma, é possível verificar se a abordagem de planejamento proposta nesta pesquisa gera programas de produção adequados para o sistema em estudo, que representam as decisões envolvidas, e que sejam plausíveis para serem implementados na prática. Adicionalmente, os planos de produção resultantes serão comparados detalhadamente com os programas de produção praticados em fábricas do setor, analisando as vantagens e diferenças em relação à flexibilidade de planejamento, aproveitamento da capacidade, atendimento da demanda, tempo dedicado às operações de preparação das linhas de produção, entre outros elementos importantes, característicos da indústria de polpa moldada.

A análise do desempenho computacional das abordagens propostas está relacionada com o estudo das soluções obtidas pela resolução do modelo de programação inteira mista e o método de solução proposto. Em um primeiro momento, serão analisadas as soluções obtidas pela resolução do modelo através de um pacote comercial para diferentes conjuntos de exemplares para verificar a capacidade do modelo de gerar soluções factíveis, a qualidade das mesmas e o tempo computacional envolvido na resolução dos exemplares. Em relação ao método de solução, serão realizados estudos computacionais para conjuntos de exemplares de diferentes tamanhos e características, baseados em informações do sistema de produção real, analisando a qualidade das soluções obtidas e os tempos computacionais envolvidos na resolução dos exemplares. Da mesma forma, os resultados obtidos através da resolução do modelo serão comparados com os resultados fornecidos pelo método de solução, identificando as vantagens e desvantagens de cada abordagem para cada conjunto de problemas, e as melhorias obtidas no desempenho computacional da abordagem proposta.

Referências

Almada-Lobo et al. (2012). Single machine multi-product capacitated lot sizing with sequence-dependent setups. *International Journal of Production Research*, Londres, v.45, n.20, p.4873 - 4894, 2007.

Associação Brasileira de Celulose e Papel - BRACELPA (2014). Dados do setor, Setembro 2013. Disponível em: < <http://www.bracelpa.org.br/>>. Acesso em: 15/02/2014.

Associação Nacional dos Aparistas de Papel - ANAP (2014). Relatório anual 2012. Disponível em: <<http://www.anap.org.br/>>. Acesso em: 15/02/2014.

Bertrand, J. W. M.; Fransoo, F. C. (2002). Operations management research methodologies using quantitative modeling. *International Journal of Operation & Production Management*, Bradford, v.2, n.2, p.241-264.

Compromisso Empresarial para Reciclagem - CEMPRE (2013). Disponível em: <<http://www.cempre.org.br/>>. Acesso em: 07/10/2013.

European Moulded Fibre Association - EMFA (2013). Disponível em: <<http://www.emfa.eu/>>. Acesso em: 01/10/2013.

Ferreira, D. et al. (2012). Single-stage formulations for synchronised two-stage lot sizing and scheduling in soft drink production. *International Journal of Production Economics*, Amsterdam, v.136, p.255-265.

Hillier, F. S.; Lieberman, G. J. (2002) *Introduction to operations research*. 7. ed. New York: McGraw-Hill, 2002. 1197 p.

Meyr, H. (2002). Simultaneous lotsizing and scheduling on parallel machines. *European Journal of Operational Research*, Amsterdam, v.139, p.277–292.

Morabito, R.; Pureza, V. (2009). Modelagem e Simulação In: MIGUEL, P. A. C. (Ed.) *Metodologia de pesquisa em engenharia de produção e gestão de operações*. Rio de Janeiro: Campus/Elsevier, 2009.p.165-195.

Pérez, K. Y. M. (2013). *Planejamento e Programação da Produção na Indústria de Embalagens em Polpa Moldada*. 2013. 136f. Dissertação (Mestrado em Engenharia de Produção) - Departamento de Engenharia de Produção, DEPS-UFSCar, Sorocaba.

Uma abordagem de solução híbrida baseada em algoritmo genético para o problema de planejamento e sequenciamento da produção em indústria de papel e celulose

Marcos Mansano Furlan

Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, Av.
Trabalhador São-carlense 400, São Carlos 13560-970, Brasil
mafurlan@icmc.usp.br

Bernardo Almada-Lobo

INESC-TEC, Faculdade de Engenharia, Universidade do Porto, Rua Dr. Roberto Frias s/n,
Porto 4200-465, Portugal
balobo@fe.up.pt

Maristela Oliveira Santos

Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, Av.
Trabalhador São-carlense 400, São Carlos 13560-970, Brasil
mari@icmc.usp.br

Reinaldo Morabito Neto

Departamento de Engenharia de Produção, Universidade Federal de São Carlos, Rod.
Washington Luís KM 235, São Carlos 13565-905, Brasil
morabito@ufscar.br

RESUMO

Este artigo aborda um problema de planejamento e sequenciamento da produção encontrado em indústria de papel e celulose utilizando um algoritmo genético modificado. A planta industrial considerada tem processo produtivo multi-estágio, onde todos os níveis e unidades produtivas são consideradas de forma integrada. As unidades críticas consideradas são: digestor contínuo, estoques intermediário de celulose e licor negro (resíduos), múltiplas máquinas de papel e linha de recuperação de resíduos. O algoritmo genético proposto agrega um novo processo de diversificação dinâmica que tem início e interrupção de acordo com a dispersão relativa da população. Esta dispersão é calculada com relação a frequência dos genes dentro da população. Apresentamos também novos operadores de mutação, além de verificar operadores da literatura. Os resultados experimentais mostram evidências de que o método proposto é competitivo com outras ferramentas da literatura. Estas evidências são mais claras para instâncias de maior porte (tamanho mais próximo de casos reais).

PALAVRAS-CHAVE: Algoritmo genético; Indústria de papel e celulose; Planejamento e sequenciamento da produção com máquinas paralelas.

1. Introdução

A indústria de papel e celulose é composta em sua maioria por empresas de grande porte e suas linhas produtivas são compostas por diferentes unidades produtivas. Uma indústria básica apresenta: digestor, máquina de papel, tanques de estoque intermediário e planta de recuperação química. Para uma melhor gestão dos recursos produtivos, o planejamento da produção deve ser realizado considerando todas estas unidades de forma integrada, com o objetivo de minimizar custos e garantindo todas as restrições produtivas.

Santos e Almada-Lobo (2012) consideram uma indústria Portuguesa de papel e celulose com uma máquina de papel. O problema é abordado utilizando modelo de programação

inteira mista (MILP) e resolvido por meio de uma abordagem combinada das heurísticas *relax-and-fix* e *fix-and-optimize*. Figueira et al. (2013) considera características adicionais no problema de Santos e Almada-Lobo (2012) como ciclos de produção e garantias de entrega da demanda atrasada. Neste caso foi apresentada uma meta-heurística do tipo VNS e seus resultados foram superiores aos apresentados pelo método de Santos e Almada-Lobo (2012).

Este trabalho teve o objetivo de estender o modelo matemático de Santos e Almada-Lobo (2012) para tratar o problema com múltiplas máquinas de papel e apresentar novo método de solução. O algoritmo genético desenvolvido introduz novos operadores como, por exemplo, a diversificação dinâmica. O processo é iniciado/interrompido sempre que a dispersão da população ultrapassa um dos limites definidos a priori. O intuito é que a dispersão controlada propicie a continuidade da convergência da solução para mínimos locais melhores. Durante o processo de diversificação, a taxa de mutação é incrementada, para re-introduzir valores de genes perdidos no processo evolutivo. Além disso, aumentamos a taxa de aceitação de indivíduos menos adaptados (soluções piores), desde que estes introduzam mais variabilidade a população.

2. Descrição do problema

Neste trabalho considera-se um meio produtivo como apresentado na Figura 1. Três plantas são integradas pelo fluxo de produtos intermediários (licor negro, celulose virgem e reciclada). A planta de polpa é composta de digestor contínuo (produz celulose e licor negro), planta de produção de celulose reciclada e tanques de estocagem de celulose virgem e reciclada. A planta de papel é composta pelas máquinas de papel (produz bobinas jumbo de papel por meio da mistura de celulose virgem e reciclada) e pelas bobinadeiras/cortadoras (produção de bobinas menores e retângulos). Estas últimas não serão consideradas, pois o problema de corte associado às bobinas/jumbos não será abordado. A planta de recuperação contém estoques intermediários de licor negro fraco e concentrado além do evaporador (concentra o teor sólido do licor negro retirando água), da caldeira de recuperação química (produz vapor e resíduos inorgânico sólido) e turbinas geradoras de energia elétrica.

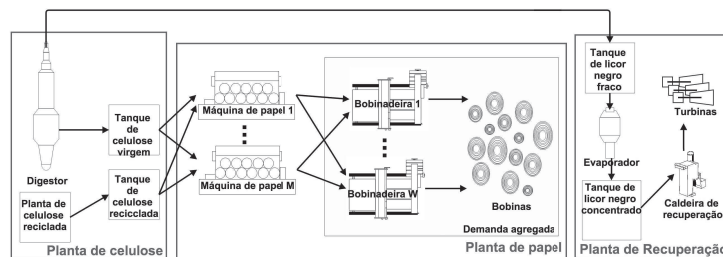


Figura 1: Ilustração de meio produtivo integrado de papel e celulose com múltiplas máquinas.

3. Abordagem de solução

Nesta seção caracterizaremos brevemente o algoritmo genético proposto. Ele é composto por um método de solução exato dentro de um algoritmo genético (AG). O método de solução exato é responsável pela solução dos sub-problemas lineares, enquanto o AG determina o valor das variáveis de decisão (binárias). Ferramentas como o processo de diversificação, ativo por meio da variável *divFlag*, foram desenvolvidos para melhorar o desempenho do método. Outros operadores como, por exemplo, a decodificação do indivíduo e o número variável de micro-períodos entre indivíduos foi determinado para garantir algumas restrições do problema e ajustar melhor a discretização do horizonte de planejamento, respectivamente. Apresentamos a seguir o pseudo-código do método e uma breve descrição de cada procedimento.

População inicial representa a construção das soluções iniciais (*pop*) que pode ser feita

Algorithm 1: Pseudo-código do algoritmo genético híbrido.

```
1 População inicial(pop);
2 Heurística gulosa de sequenciamento(pop);
3 Ajuste de parâmetros(mutprob,  $\varphi$ , divFlag);
4 Avaliação(pop,  $\varphi$ );
5 Seleção(pop);
6 while tempo limite não for atingido do
7   Cruzamento(pop);
8   Ajuste de parâmetros(mutprob,  $\varphi$ , divFlag);
9   Mutação(pop, mutprob);
10  Avaliação(pop,  $\varphi$ );
11  Seleção(pop);
12  Diversificação(pop, divFlag);
13 end
14 Retorna a melhor solução;
```

de forma aleatória ou através de heurística. Neste trabalho testamos as duas opções e optamos pela criação aleatória, pois apresentou melhor desempenho geral do método. A *heurística gulosa de sequenciamento* troca a sequência de produção dentro de cada macro-período do horizonte de planejamento com o intuito de reduzir o custo de preparação de máquina (inicialmente alto). O *ajuste de parâmetros* é responsável pela modificação das probabilidades de mutação (*mutprob*) e pelo ajuste do parâmetro de diversificação φ . O parâmetro de diversificação indica o quanto estamos dispostos a “perder” em qualidade de solução para ganhar em variabilidade. Na *avaliação* são resolvidos os sub-problemas gerados por cada indivíduo e calculadas as aptidões (*fitness*) dos mesmos. Na *seleção* são definidos quais indivíduos sobreviverão para a próxima geração, de acordo com a pressão de seleção ajustada a priori. Neste trabalho optamos por utilizar seleção por torneio de dois, amplamente utilizado em outros trabalhos da literatura.

O *cruzamento* é responsável por misturar duas soluções gerando duas novas soluções. Aqui optamos por outro operador conhecido e amplamente utilizado, o cruzamento de dois pontos. Na *mutação* passamos por um conjunto de seis diferentes mutações: mutação simples, que troca o valor de um único gene; mutação por troca, que muda o valor de dois genes (troca os valores entre eles); mutação por inserção, que adiciona um valor em uma posição aleatória, empurrando todos os genes para frente até uma outra posição escolhida aleatoriamente que é removida; mutação por sequencia, que troca a sequencia de produção de forma aleatória dentro de um macro-período; mutação de micro-períodos, que altera a quantidade de micro-períodos do indivíduo; e mutação por campanha, que tenta aumentar o tamanho das campanhas produtivas de um mesmo item trocando o valor do gene para o mesmo valor de um “vizinho”.

O início e final do processo de diversificação são definidos de acordo com cálculos de espalhamento da população (frequência média de cada gene). Quando a similaridade se torna superior a um determinado limitante, o processo de diversificação é iniciado, aumentando gradativamente *mutProb* e mudando φ para o valor φ^{max} definido a priori. Neste caso, soluções piores podem ser aceitas, desde que adicionem a devida variabilidade a população. Quando o espalhamento da população for suficientemente grande (similaridade menor que um limitante inferior), o processo termina e *mutProb* é reposta ao padrão. O parâmetro φ é reduzido de forma gradativa, até o valor φ^{min} , para que a variabilidade não seja excluída da população instantaneamente.

4. Resultados computacionais

As instâncias foram geradas por meio de uma modificação do gerador proposto por Figueira et al. (2013), que usa como base os dados obtidos por Santos e Almada-Lobo (2012) na indústria Portuguesa. As alterações contam ainda com adaptações feitas para ajustar as instâncias para o problema considerado, por exemplo, o uso de múltiplas máquinas de papel. Os perfis de

demanda das instâncias variam entre 65% e 85% da capacidade das máquinas de papel e foram aleatoriamente gerados de acordo com uma distribuição uniforme. Além disso, os valores de estoque inicial foram gerados numa faixa de 80% a 120% do valor original dos dados.

Os preços foram gerados de acordo com a média dos preços de venda da carteira de pedidos e são utilizados para calcular os custos de estoque, preparação de máquina e atraso, na função objetivo. As instâncias geradas têm horizonte de planejamento de 7, 15 e 30 dias (períodos) e 8 ou 16 itens finais. Foram considerados também 3 ou 4 turnos de produção associados a uma ou duas máquinas de papel, totalizando 3 a 4 micro-períodos por dia para o caso com uma máquina e 6 a 8 micro-períodos por dia no caso de duas máquinas. Para cada combinação de valores foram geradas 10 instâncias, totalizando 160 instâncias.

Os resultados são comparados de forma agregada através de gráficos de desempenho (Dolan e More, 2002). Neste tipo de gráfico um conjunto de soluções é comparado entre dois ou mais métodos através da diferença percentual com a menor solução encontrada por este conjunto de métodos. O eixo das abscissas define a diferença percentual τ das soluções de cada método com relação a melhor solução. O eixo das ordenadas, por outro lado, define a porcentagem de soluções de cada método que tem qualidade melhor ou igual a τ , ou seja, a porcentagem de vezes em que $\frac{t_{ps}}{\min(t_{ps}:s \in S)} \leq \tau$ onde t_{ps} representa o valor da solução do problema p pelo método de solução s e S é o conjunto de todas as abordagens de solução avaliadas.

A Figura 2 apresenta a performance de três métodos (GA, VNS e Cplex) para as instâncias com uma máquina de papel. A versão 12.2 do Cplex foi utilizada com os parâmetros padrão, VNS é uma adaptação do método de solução apresentado por Figueira et al. (2013) o algoritmo genético proposto. Neste caso, podemos verificar que o Cplex tem um número maior de melhores soluções ($\tau = 0$), mas é rapidamente ultrapassado pelo GA em $\tau = 1.02$, ou seja, o método proposto apresenta um número maior de soluções com no máximo 2% pior do que a melhor solução encontrada. De uma forma geral, visualmente verificamos que a curva de performance do GA é superior aos demais métodos. Ambas heurísticas resolvem todas as instâncias, enquanto o Cplex resolve cerca de 90% dentro do limite de 3600 segundos. É importante ressaltar que a adaptação do VNS pode ter reduzido sua eficiência quando comparado com o Cplex.

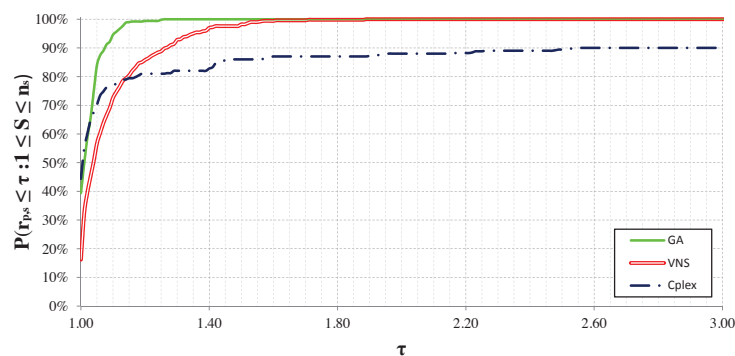


Figura 2: Gráfico de curvas de desempenho para instâncias com uma máquina de papel.

A Figura 3 apresenta as curvas de performance do GA e Cplex para as instâncias com duas máquinas. O VNS de Figueira et al. (2013) não foi testado, pois não foi desenvolvido para esta classe de problemas. A curva de desempenho do GA domina totalmente a curva do Cplex. A porcentagem de melhores soluções do GA é de 74.4% e ele resolve cerca de 81.4% dos casos, enquanto o Cplex inicia com 7.4% de melhores soluções e resolve 62% dos casos testados.

5. Conclusões e perspectivas

Este trabalho aborda o problema de planejamento e sequenciamento da produção na indústria de papel e celulose. O intuito foi estender o modelo matemático de Santos e Almada-

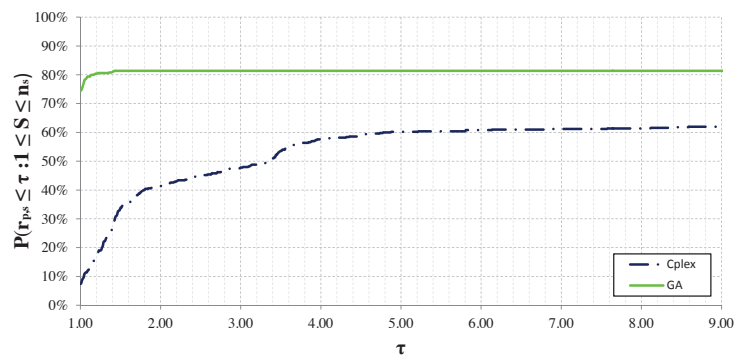


Figura 3: Gráfico de curvas de desempenho para instâncias com duas máquinas de papel.

Lobo (2012) para tratar problemas com múltiplas máquinas de papel e desenvolver método de solução para este caso mais geral. O método de solução é híbrido de algoritmo genético e método de solução exato. Ele conta com um novo processo de diversificação, novos operadores de mutação e representação de indivíduo com tamanho variável. Operadores da literatura também foram implementados e seus resultados foram analisados.

Os resultados demonstraram o bom desempenho do método nos casos testados. Ele obteve melhores soluções na maioria dos casos e resolveu um número maior de problemas, superando os métodos VNS de Figueira et al. (2013) e a ferramenta de solução IBM Cplex 12.2.

Como pesquisa futura, podem ser verificadas melhorias para a modelagem matemática como, por exemplo, reformulações, desigualdade válidas ou outros paradigmas de discretização do horizonte de planejamento. Modelos que misturem planejamento detalhado a curto prazo e agregado no longo prazo também podem ser considerados em combinação com abordagens de solução de horizonte rolante. Com relação ao algoritmo genético, podemos verificar a utilização de outros operadores, a criação de múltiplas populações e a definição de uma quantidade variável de micro-períodos em casa dia do plano de produção, por exemplo.

Agradecimentos

O primeiro autor agradece subsídios concedidos pelo Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) processo número 233210/2012-1 e pela Fundação de Amparo à Pesquisa do Estado de São Paulo (Fapesp) contrato número 2011/00274-8.

O segundo autor agradece ao Fundo Europeu de Desenvolvimento Regional (FEDER) pelo subsídio concedido através do projeto NORTE-07-0124-FEDER-000057, cofinanciado pelo Programa Operacional Regional do Norte (ON.2 - O Novo Norte), ao abrigo do Quadro de Referência Estratégico Nacional (QREN).

Referências

- Dolan, E. D.; More, J. J.** (2002). Benchmarking optimization software with performance profiles. *Mathematical programming*, 91(2), 201–213.
- Figueira, G.; Oliveira Santos, M.; Almada-Lobo, B.** (2013). A hybrid vns approach for the short-term production planning and scheduling: A case study in the pulp and paper industry. *Computers & Operations Research*.
- Santos, M. O.; Almada-Lobo, B.** (2012). Integrated pulp and paper mill planning and scheduling. *Computers & Industrial Engineering*, 63(1), 1 – 12.

UMA FORMULAÇÃO NÃO LINEAR PARA A RESTRIÇÃO DE CICLOS DA SERRA NO PROBLEMA INTEGRADO DE DIMENSIONAMENTO DE LOTES E CORTE DE ESTOQUE

Gislaine Mara Melega

Unesp - Ibilce

Depto de Matemática Aplicada, 15054-000, São José do Rio Preto, SP

E-mail: gislainemelega@gmail.com

Silvio Alexandre de Araujo Socorro Rangel

Unesp - Ibilce

Depto de Matemática Aplicada, 15054-000, São José do Rio Preto, SP

E-mail: saraujo@ibilce.unesp.br/ socorro@ibilce.unesp.br

RESUMO

Neste trabalho aborda-se o problema integrado de dimensionamento de lotes e corte de estoque em um estudo de caso de uma fábrica de móveis de pequeno porte. Um modelo da literatura é discutido, no qual inclui restrições para o controle do número de ciclos da serra. Propomos uma extensão deste conjunto de restrições que resulta em um modelo não linear.

PALAVRAS-CHAVE: Problema de Dimensionamento de Lotes, Problema de Corte de Estoque, Problema Integrado, Ciclos da Serra

1. Introdução

Estudar o problema de dimensionamento de lotes (*PDL*) de forma integrada ao problema de corte de estoque (*PCE*) constitui uma área de relevância na literatura, devido a sua grande aplicabilidade no setor industrial. O objetivo do problema integrado é capturar a interdependência entre as decisões, além de possibilitar economia de matéria-prima e a diminuição dos custos globais no processo de produção. Assim, o problema integrado de dimensionamento de lotes e corte de estoque consiste em, decidir em cada período do horizonte de planejamento a quantidade de produtos finais (itens) a serem produzidos, de maneira a minimizar o custo total de produção. No custo total são incluídos custos de preparo e estocagem referentes ao problemas de dimensionamento de lotes e a quantidade de objetos a serem cortados, este último é objetivo relacionado ao problema de corte de estoque.

Diversos trabalhos da literatura tratam o *PDL* e o *PCE* integrados em diferentes setores industriais, entre eles Farley (1988), Hendry et al. (1996), Nonas e Thorstenson (2008) e Poltroniere et al. (2008). No setor moveleiro, destacamos os trabalhos de Gramani e França (2006), Gramani et al. (2011), Ghidini e Arenales (2009), Santos et al. (2011) e Alem e Morabito (2012). Vanzela et al. (2013) usaram como ponto de partida o modelo apresentado em Gramani et al. (2011) e acrescentaram restrições referentes ao processo produtivo de uma fábrica de móveis de pequeno porte.

Um aspecto importante considerado no modelo proposto em Vanzela et al. (2013) é a possibilidade da máquina de corte cortar vários objetos simultaneamente. O conjunto de todas as operações necessárias para cortar um, ou mais objetos simultaneamente, de acordo com um determinado padrão de corte é definido como um ciclo da serra (Yanasse et al., 1991).

A partir do modelo proposto por Vanzela et al. (2013), apresentamos uma extensão para a restrição referente ao ciclos da serra, com o intuito de se obter um modelo mais realista. Para

o modelo estendido foram propostas abordagens de linearização, já que o mesmo constitui um modelo não linear.

2. Descrição do Problema

A indústria de móveis no Brasil está concentrada em pólos regionais localizados principalmente nas regiões Sul e Sudeste do país. Diversos tipos de matéria-prima são usados na produção de móveis (madeira, metal, plástico, couro). Os pólos da região noroeste do estado de São Paulo são voltados principalmente para a produção de móveis residenciais de madeira (Figueiredo e Rangel, 2008), no qual encontra-se a fábrica deste estudo, aqui denotada por Fábrica L.

A Fábrica L, situada no pólo de Votuporanga, é uma empresa característica do setor e considerada de pequeno porte em função do número de funcionários. A empresa ocupa-se da produção de móveis residenciais de madeira, retilíneos, na sua maioria móveis para dormitório. Para representar o processo de tomada de decisão da Fábrica L Vanzela et al. (2013), fizeram algumas simplificações no processo produtivo.

Simplificações

1. Somente a capacidade do setor de corte é considerada e presume-se que os demais setores podem lidar com as decisões tomadas para o problema de corte de estoque.
2. O tempo de preparo para a máquina de corte e o tempo de troca de padrão de corte não foi levado em consideração. O impacto estimado desta simplificação é levado em consideração indiretamente na capacidade da máquina de corte.
3. O limite de capacidade de produção em termos do número total de ciclos da serra em um dado período é relaxado, de forma que os diferentes padrões de corte podem ser cortados no mesmo ciclo de serra.
4. Atraso na entrega, e horas extras para atender a demanda não são permitidos.
5. As cores dos produtos finais não são considerados.

O problema tratado consiste então em definir o dimensionamento dos lotes de móveis (produtos finais) e o número de objetos a serem cortados (e os respectivos padrões de corte) de forma a minimizar o custo total de produção considerando restrições relativas ao atendimento da demanda dos itens finais e à capacidade da máquina de corte. Neste trabalho a demanda é considerada determinística e a capacidade da máquina de corte é definida em termos do número máximo de ciclos da serra em cada período.

3. Modelagem Matemática

Nesta seção o modelo matemático proposto por Vanzela et al. (2013) para o problema integrado no contexto de fabricação de móveis é apresentado. Para tanto, faz-se o uso dos seguintes índices e dados.

Índices:

- $t = 1, \dots, T$: períodos;
- $f = 1, \dots, F$: produtos finais;
- $p = 1, \dots, P$: peças;
- $e = 1, \dots, E$: espessura da placa;
- $j = 1, \dots, J$: padrões de corte.

Parâmetros:

- c_f : custo de produção para o produto f ;
 h_f : custo de estoque para o produto f ;
 D_{ft} : demanda por produto f no período t ;
 o^e : espessura da placa e ;
 co^e : custo da placa com espessura e ;
 \hat{h}_p^e : custo de estoque da peça p com espessura e ;
 C_t : capacidade máxima de produção no período t , contabilizada em número de ciclos da serra;
 q_{pf}^e : número de peças p de espessura e necessárias para produzir uma unidade do produto f ;
 S : altura da serra;
 cap^e : número máximo de placas de espessura e que podem ser cortadas simultaneamente ($\lfloor \frac{S}{o^e} \rfloor$);
 a_{pj}^e : número de peças p com espessura e presentes no padrão de corte j ;
 I_{f0} : estoque inicial do produto final f ;
 IP_{p0}^e : estoque inicial da peça p com espessura e ;
 ts : porcentagem da demanda utilizada para impor níveis seguros de estoque dos produtos finais;

Variáveis:

- X_{ft} : quantidade do produto f produzida no período t ;
 I_{ft} : quantidade de produto f armazenada no final do período t ;
 IP_{pt}^e : número de peças p de espessura e armazenadas em período de t ;
 y_{jt}^e : número de placas de espessura e , cortadas de acordo com o padrão j no período t .

Formulação do Problema: PIDCM

$$\min Z = \sum_{f=1}^F \sum_{t=1}^T (c_f X_{ft} + h_f I_{ft}) + \sum_{e=1}^E \sum_{j=1}^J \sum_{t=1}^T co^e y_{jt}^e + \sum_{e=1}^E \sum_{p=1}^P \sum_{t=1}^T \hat{h}_p^e IP_{pt}^e \quad (1)$$

Sujeito a:

$$X_{ft} + I_{f,t-1} - I_{ft} = D_{ft}, \quad f = 1, \dots, F; t = 1, \dots, T \quad (2)$$

$$I_{ft} \geq ts D_{ft}, \quad f = 1, \dots, F; t = 1, \dots, T - 1 \quad (3)$$

$$I_{fT} \geq ts \left(\sum_{t=1}^T D_{ft} \right), \quad f = 1, \dots, F \quad (4)$$

$$\sum_{j=1}^J a_{pj}^e y_{jt}^e + IP_{p,t-1}^e - IP_{pt}^e = \sum_{f=1}^F q_{pf}^e X_{ft}, \quad p = 1, \dots, P; t = 1, \dots, T; e = 1, \dots, E \quad (5)$$

$$\sum_{e=1}^E \sum_{j=1}^J \frac{y_{jt}^e}{cap^e} \leq C_t, \quad t = 1, \dots, T \quad (6)$$

$$y_{jt}^e \in \mathbb{Z}_+, \quad j = 1, \dots, J; t = 1, \dots, T; e = 1, \dots, E \quad (7)$$

$$X_{ft}, I_{ft} \in \mathbb{R}_+, \quad f = 1, \dots, F; t = 1, \dots, T \quad (8)$$

$$IP_{pt}^e \in \mathbb{R}_+, \quad p = 1, \dots, P; t = 1, \dots, T; e = 1, \dots, E \quad (9)$$

A função objetivo (1) minimiza a soma dos custos de produção (c_f), estoque final dos produtos (h_f), custos de matéria-prima (co^e) e os custos de estocar as peças (\hat{h}_p^e). As restrições (2) garantem que a demanda dos produtos finais f seja atendida. As restrições (3) e (4) impõem níveis de estoque de segurança para o produto final f como uma porcentagem da demanda de cada produto. Para os primeiros $(t - 1)$ períodos, os níveis de segurança são expressos em termos das demandas individuais ($ts D_{ft}$), e para o período final, os níveis de estoque de segurança são expressos em termos da demanda total. O conjunto de restrições (5) modela a interdependência

entre o *PDL* e o *PCE*, pois considera as decisões relativas ao dimensionamento de lotes (X_{ft}) e a decisão sobre o corte de matéria-prima (y_{jt}^e). O estoque de peças (IP_{pt}^e) é permitido. As restrições (6) garantem o respeito à capacidade de produção (C_t) definida em número de ciclos da serra em cada período t . As restrições (7), (8) e (9) definem domínio das variáveis.

Na simplificação 3 proposta por Vanzela et al. (2013) referente à capacidade da serra, a restrição apresentada constitui uma relaxação para possíveis restrições reais, tais restrições não consideram que padrões de corte distintos não podem compartilhar o mesmo ciclo da serra. Esta restrição foi incluída para evitar que a solução do problema de corte de estoque seja inviável na prática. Na próxima seção será proposta uma reformulação para a restrição (6) a fim de torná-la mais realista.

4. Proposta de Reformulação

O objetivo deste trabalho é apresentar uma reformulação para a restrição referente aos ciclos da serra do problema integrado (restrição 6). Alguns trabalhos na literatura abordam problemas que possuem em sua estrutura o problema do ciclos da serra. Yanasse et al. (1991) desenvolveram uma heurística para resolver o *PCE* bidimensional com o objetivo de reduzir o número de ciclos da serra. Posteriormente Mosquera e Rangel (2007) propuseram 2 variações da heurística apresentada por Yanasse et al. (1991) para um estudo de caso em uma indústria moveleira. Yanasse (2008) mostra que a solução do problema de corte de estoque considerando a redução de ciclos da serra em uma situação de alta demanda é equivalente a resolver o *PCE* com as demandas escaladas pelo número máximo de objetos que a máquina pode cortar ao mesmo tempo. Ranck JR. (2008) propõe um modelo matemático para o problema de corte de estoque que considera a redução do número total de ciclos da serra e o número total de objetos a serem cortados, que pode ser usado tanto no caso unidimensional como no bidimensional, e permite também o uso de ciclos incompletos.

Antes de apresentar a reformulação para a restrição de ciclos da serra consideremos as seguintes definições.

Definição 1. (Toscano et al., 2013): *O número mínimo de ciclos da serra, necessários para cortar y_{jt}^e objetos de espessura e , de acordo com um determinado padrão de corte j e com a capacidade da máquina é calculado de acordo com.*

$$\left\lceil \frac{y_{jt}^e}{cap^e} \right\rceil \quad j = 1, \dots, J; \quad e = 1, \dots, E \quad (10)$$

A fim de tratar situações encontradas na prática, este trabalho reformula a restrição de ciclos da serra (6) para que seja considerado apenas um tipo de padrão de corte em cada ciclo. Isto é feito ao substituir $\frac{y_{jt}^e}{cap^e}$ por $\left\lceil \frac{y_{jt}^e}{cap^e} \right\rceil$ em (6), obtendo uma nova formulação para a restrição de ciclos da serra expressa pela restrição (11). Ao efetuar esta mudança, o número de ciclos é contado exatamente e uma restrição mais apertada é obtida, já que uma solução que era factível com a restrição (6) pode agora ser infactível. Porém, o problema resultante constitui um problema de otimização não linear.

$$\sum_{e=1}^E \sum_{j=1}^J \left\lceil \frac{y_{jt}^e}{cap^e} \right\rceil \leq C_t, \quad t = 1, \dots, T \quad (11)$$

Uma abordagem para lidar com este problema é aplicar uma estratégia linearização na restrição (11), obtendo assim um modelo linear. Uma outra abordagem para lidar com o número de ciclos da serra é proposta em Santos et al. (2011). Os autores dividiram cada período em sub-períodos, em que cada subperíodo corresponde a um ciclo da serra, sendo assim é possível saber

exatamente quantas placas devem ser cortadas de acordo com cada padrão de corte. Pretende-se com os estudos em andamento linearizar a restrição e comparar com outras estratégias da literatura.

Referências

- Alem, D. J.; Morabito, R.** (2012). Production planning in furniture settings via robust optimization. *Computers & Operations Research*, 39, 139 – 150.
- Farley, A. A.** (1988). Mathematical programming models for cutting-stock problems in the clothing industry. *The Journal of Operational Research Society*, 39, 41–53.
- Figueiredo, A.; Rangel, S.** (2008). Geração de padrões de corte produtivos para a indústria de móveis. Anais do XXXVIII Simpósio Brasileiro de Pesquisa Operacional - SBPO 1626 – 1636.
- Ghidini, C. T. L. S.; Arenales, M. N.** (2009). Otimização de processos acoplados na indústria de móveis: dimensionamento de lotes e corte de estoque.
- Gramani, M. C. N.; França, P. M.** (2006). The combined cutting stock and lot-sizing problem in industrial processes. *European Journal of Operational Research*, 174, 509 – 521.
- Gramani, M.; França, P. M.; Arenales, M. N.** (2011). A linear optimization approach to the combined production planning mode. *Journal of the Franklin Institute*, 348, 1523 – 1536.
- Hendry, L. C.; Fok, K. K.; Shek, K. W.** (1996). A cutting stock and scheduling problem in the copper industry. *The Journal of the Operational Research Society*, 47, 38 – 47.
- Mosquera, G.; Rangel, S.** (2007). Redução de ciclos da serra no problema de corte de estoque bidimensional na indústria de móveis. Anais do XXX - Congresso Nacional de Matemática Pura e Aplicada - CNMAC.
- Nonas, S. L.; Thorstenson, A.** (2008). Solving a combined cutting-stock and lot-sizing problem with a column generating procedure. *Computers & operations research*, 35.
- Poltroniere, S. C.; Poldi, K. C.; Toledo, F. M. B.; Arenales, M. N.** (2008). A coupling cutting stock-lot sizing problem in the paper industry. *Annals of Operations Research*, 157.
- Ranck JR., R.** (2008). Desenvolvimento de alguns métodos de solução para o problema de redução de ciclos da serra. Master's thesis, Dissertação de Mestrado, INPE.
- Santos, S.; Araujo, S. A.; Rangel, S.** (2011). Integrated cutting machine programming and lot sizing in furniture industry. *Pesquisa Operacional para o Desenvolvimento*, 3, 249 – 266.
- Toscano, A.; Rangel, S.; Yanasse, H. H.** (2013). A heuristic approach to minimize the number of saw cycles in small-scale furniture factories. *Relatório Técnico, DMAP - UNESP*.
- Vanzela, M.; Rangel, S.; de Araujo, S.** (2013). A study of the integrated lot sizing and cutting stock problem for furniture production. *Relatório Técnico, DMAP - UNESP*.
- Yanasse, H. H.** (2008). A note on the minimization of the number of cutting cycles problem. Livro de resumos do XI Simpósio de Pesquisa Operacional e Logística da Marinha - SPOLM.
- Yanasse, H. H.; Zinober, A. S. I.; Harris, R. G.** (1991). Two-dimensional cutting stock with multiple stock sizes. *The Journal of the Operational Research Society*, 42, 673 – 683.

Problema integrado de produção e distribuição: Uma abordagem heurística

Márcio A F Belo-Filho

Instituto de Ciências Matemáticas e de Computação - Universidade de São Paulo
marciobelof@gmail.com

Pedro Amorim

INESC TEC, Faculdade de Engenharia, Universidade do Porto
amorim.pedro@fe.up.pt

Bernardo Almada-Lobo

INESC TEC, Faculdade de Engenharia, Universidade do Porto
almada.lobo@fe.up.pt

RESUMO

Problemas integrados de produção e distribuição de produtos perecíveis são comuns em muitas atividades industriais. Para perecíveis com prazo de validade particularmente restrito, esta coordenação da cadeia de suprimentos é ainda mais importante. Particularmente a nível operacional, as decisões de tamanho dos lotes e programação da produção devem ser tomadas em conjunto com as decisões de roteamento de veículos para satisfazer os clientes da melhor forma. Entretanto, tais decisões em simultâneo deixam os problemas mais difíceis, principalmente para indústrias com uma grande variedade de produtos. Propomos uma abordagem de solução baseada na metaheurística *Adaptive Large Neighbourhood Search (ALNS)*, conhecidamente eficiente para problemas de roteamento de veículos. O método ALNS proposto utiliza procedimentos exatos e heurísticos e obteve melhores resultados em comparação com métodos exatos puros e a heurística fix-and-optimize padrão, em tempo computacional e qualidade da solução.

PALAVRAS-CHAVE: ALNS, dimensionamento de lotes, programação da produção, produtos perecíveis, VRPTW.

1. Introdução

Muitas indústrias trabalham com produtos perecíveis, cuja utilidade é sensível de maneira decrescente ao tempo, ou seja, a sua utilidade e/ou valor de um produto diminui ao longo do tempo. Uma definição de produtos perecíveis é dada por Amorim et al. (2013): "Um item, que pode ser uma matéria-prima, um produto intermediário ou final, é definido por perecível se durante o horizonte de planeamento considerado pelo menos uma das seguintes condições ocorre: (1) seu estado físico piora sensivelmente (por exemplo por deterioração ou apodrecimento), e/ou (2) seu valor é reduzido na percepção de um cliente, e /ou (3) há um perigo de uma funcionalidade reduzida futuramente, na opinião de alguma autoridade". A entrega de produtos finais perecíveis a clientes não pode demorar, pois pode implicar em perda de qualidade e valor do produto, inclusive no descarte de produtos e, consequentemente, prejuízo para cliente e empresa. Para produtos com prazo de validade pequeno, é interessante, senão indispensável, integrar as decisões de planeamento da produção e da distribuição a nível operacional, já que neste planeamento todas as operações são limitadas de acordo com a perecibilidade do produto.

Neste trabalho, discutimos métodos de solução para o problema integrado de planeamento da produção e distribuição de perecíveis a nível operacional (*operational integrated production and distribution planning problem with perishable products - OIPDPP*). O problema

foi apresentado em Amorim et al. (2013), que desenvolveu dois modelos de programação linear inteiro-mista (*MILP*). O primeiro modelo segue a literatura e impõe que os lotes de produção são indivisíveis enquanto a segunda formulação propõe que os lotes de produção podem ser divididos em múltiplos lotes, consideração até então inédita para este tipo de problema. Como consequência, melhores soluções foram obtidas, por meio da redução de operações de preparação das linhas, de número de veículos necessários e do custo das viagens. Entretanto, os resultados indicaram uma complexidade intrínseca ao problema que não permite a solução de problemas reais em tempo computacional razoável por meio de métodos exatos. Por isso, propomos a metaheurística *ALNS* para preencher esta lacuna da literatura e abordar instâncias práticas maiores do problema. A abordagem proposta é híbrida (utiliza técnicas heurísticas e métodos exatos) e adaptativa em dois sentidos: (1) na escolha das múltiplas maneiras de se buscar novas soluções; e (2) na abrangência destas buscas, o que confere flexibilidade e robustez à abordagem.

2. Descrição do problema

O *OIPDPP* consiste em uma única planta com múltiplas linhas de produção que manufaturam produtos perecíveis variados para múltiplos clientes. A demanda é determinística, as ordens de produção incorrem em custos e tempos de produção e devem ser maiores que o lote mínimo. As linhas são limitadas por restrições de capacidade e são considerados custos e tempos de preparação das linhas dependentes da sequência de produção. No início do horizonte de planejamento a linha já se encontra preparada para a produção de determinado produto/item. Múltiplos veículos são responsáveis por atender a demanda dos clientes dentro de janelas de tempo. Os veículos possuem capacidade limitada, podendo atender a demanda de múltiplos clientes, entretanto um cliente deve ser atendido integralmente por uma única viagem de um único veículo, ou seja, não são permitidas entregas fracionadas. Na frota de veículos há pelo menos um veículo por cliente, porém, a utilização de cada veículo implica em um custo fixo e são considerados os custos e tempos de viagem do veículo. As viagens dos veículos iniciam sempre no depósito em que são carregados. Em seguida, as ordens de demanda são entregues (são considerados tempo de serviço no cliente) e no fim o veículo retorna ao depósito. A entrega dos produtos perecíveis deve ser feita dentro do prazo de vida útil dos mesmos. O tempo de vida útil dos produtos começa a ser contabilizado a partir do momento em que a operação de produção é iniciada, após uma possível operação de preparação da máquina. O objetivo do problema é minimizar os custos das operações de produção e distribuição. A formulação matemática deste problema é discutida em Amorim et al. (2013).

O gráfico de Gantt da Figura 1 ilustra os planos de produção e de distribuição de um exemplo com duas linhas ($L1$ e $L2$), quatro clientes ($C1$, $C2$, $C3$ e $C4$), três produtos ($P1$, $P2$ e $P3$) e quatro veículos ($V1$, $V2$, $V3$ e $V4$). As operações de preparação/produção são representadas pelas barras horizontais em cinza escuro/branco. Os tempos de vida útil dos produtos delimitam a entrega da demanda aos clientes e são representados por barras horizontais menores em tons mais claros de cinza. Os lotes em processamento nas linhas são identificados pela representação (j, x) denotando o produto em processamento e o tamanho do lote. A janela de tempo para a entrega de cada cliente é representada por duas barras verticais paralelas. As operações de entrega são identificadas pelas setas, que indicam as viagens. Por exemplo, o cliente $C1$ tem os lotes $(P3, 50)$ e $(P2, 50)$ processados nas linhas $L1$ e $L2$, respectivamente. Ambas operações necessitam de preparação da linha que iniciam no instante 25, logo após as operações de $P1$ para $C2$. A demanda de $C1$ tem seu processamento concluído no instante 85 e entrega feita pelo veículo $V1$. O veículo $V1$ sai do depósito no instante 160, entrega a $C1$ no instante 180, entrega a $C2$ no instante 200 e retorna ao depósito. O veículo $V4$ faz o trajeto depósito- $C4$ - $C3$ -depósito. Os veículos $V2$ e $V3$ não são utilizados.

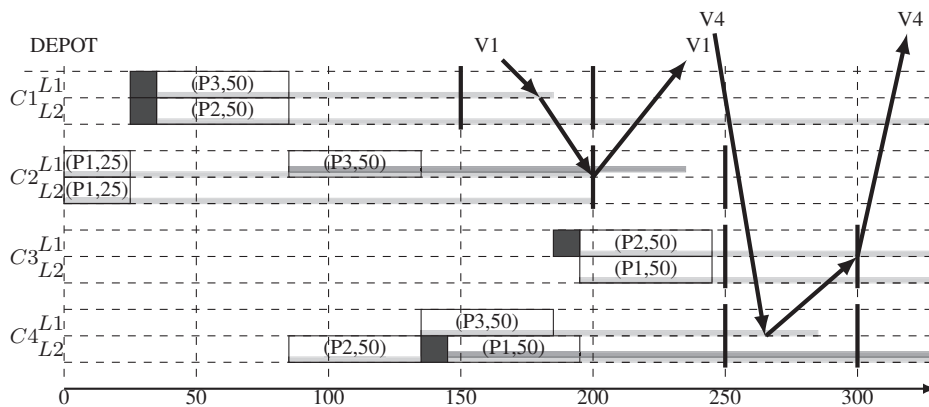


Figura 1: Exemplo de plano de produção e distribuição.

3. Métodos e abordagens

Múltiplas abordagens de solução foram consideradas: (1) métodos exatos por meio da utilização de softwares *MILP-solver*; (2) a heurística *fix-and-optimize (FO)* na versão padrão e; (3) a metaheurística *ALNS*. Uma heurística simples é responsável por fornecer a solução inicial para os métodos *FO* e *ALNS*. Além disso, esta solução inicial foi utilizada também pelo *MILP-solver* devido à dificuldade do método em encontrar soluções factíveis.

A heurística de solução inicial (*Heur*) possui determinadas regras que restringem o espaço de solução inicial: (1) a indivisibilidade das ordens de produção; (2) a sequência de produção respeita a sequência das janelas de tempo dos clientes; (3) a entrega ocorre no fim da janela de tempo do cliente por um veículo exclusivo e; (4) as ordens de produção são distribuídas às linhas de forma sequencial. A partir destas regras, reduz-se o problema inicial e este pode ser resolvido rapidamente pelo *MILP-solver*. A solução inicial obtida por *Heur* é injetada em duas heurísticas de melhoria com base em programação matemática *FO* e *ALNS*.

O método *fix-and-optimize* define partições da solução que são submetidas a buscas locais, enquanto o restante da solução é mantido fixo, ou seja, a cada partição definida, uma iteração de busca local ocorre. A sequência de iterações obedece uma regra bem definida para as partições, inclusive com sobreposições entre partições. Neste caso, cada partição está relacionada com as decisões de um número fixo de clientes, definidos em sequência na ordem crescente das janelas de tempo. A expectativa é de que os clientes com janelas de tempo mais próximas possuem maior chance de terem seus pedidos produzidos e entregues conjuntamente, o que implicaria em uma busca local mais efetiva. Seja *FO_{x-y}* o método *fix-and-optimize*, em que *x* é o número de clientes pertencentes à partição e sujeitos à busca local e *y* o número de clientes na partição sobrepostos entre duas iterações consecutivas. Portanto, *FO₃₋₁* utiliza uma partição de três clientes (de acordo com a sequência pré-estabelecida) e com a sobreposição do último cliente na iteração seguinte.

O algoritmo *large neighbourhood search (LNS)* busca promover a melhoria de soluções por meio de procedimentos (agentes) de destruição e reparação (*AgD* e *AgR*, respectivamente). Dada uma solução inicial, os *AgD* destroem parte da solução, conservando o restante. Em seguida, os *AgR* reparam a solução na expectativa de melhores soluções. Os agentes vão sendo utilizados iterativamente. A versão adaptativa do algoritmo (*ALNS*) estabelece diversos agentes de destruição e reparação e a escolha entre eles é feita por meio de estatísticas relativas ao sucesso de cada agente durante a busca.

O *ALNS* desenvolvido possui múltiplos agentes heurísticos de destruição e um único de reparação, o *MILP-Solver* com tempo limitado. Dois aspectos do algoritmo são adaptativos: (1) a escolha dos agentes e; (2) o tamanho da vizinhança dos agentes. A probabilidade de um agente

ser escolhido em uma iteração é consequência dos sucessos deste agente no decorrer da execução do algoritmo. O tamanho da vizinhança é definido adaptativamente, buscando um equilíbrio entre o tempo estabelecido e o número de soluções visitadas pelo o agente de reparação. Os oito agentes de destruição representam uma vizinhança em que o agente de reparação busca novas soluções. O primeiro agente *Cst*, foca nas decisões conjuntas de produção e distribuição para múltiplos clientes. Três agentes *Cst-P*, *Cst2-P* e *Cst3-P* dão ênfase às decisões de produção para um número fixo de clientes, variando o número de produtos escolhidos (todos, 2 e 3, respectivamente). Três agentes *Sl1-P*, *Sl1L-P* e *Sl1T-P* buscam novas soluções a partir da reprogramação de múltiplas ordens de produção por diferentes critérios de seleção. O agente *Sl1-P* escolhe ordens de produção aleatórias de uma única máquina. O agente *Sl1L-P* seleciona ordens de produção consecutivas de todas as linhas ao mesmo tempo. O agente *Sl1T-P* define um período de tempo e seleciona todas as tarefas pertencentes a este período. O agente *Dst* busca melhores soluções apenas no plano de distribuição. Por exemplo, o agente *Cst* permite alterações em todas as decisões de determinados clientes, fixando-se todas as outras decisões. A busca destas novas soluções é promovida pelo agente de reparação.

4. Resultados

O gerador de instâncias de Amorim et al. (2013) é estendido dado que não temos conhecimento de outro gerador de instâncias para o *OIPDPP*. Foram geradas 100 instâncias de 20 combinações com a nomenclatura $lL_pP_cN_ppP^*$, onde L é o número de linhas, P de produtos, N de clientes e P^* de produtos perecíveis. A tabela 1 apresenta todas as combinações e uma classificação das instâncias de acordo com o número de variáveis binárias.

Tabela 1: Diferentes combinações e classificação das instâncias.

Muito Pequenas	Pequenas	Médias	Grandes
<i>l01_p03_c05_pp01</i>	<i>l01_p05_c10_pp02</i>	<i>l01_p05_c15_pp02</i>	<i>l01_p10_c15_pp03</i>
<i>l01_p03_c05_pp02</i>	<i>l01_p05_c10_pp03</i>	<i>l01_p05_c15_pp03</i>	<i>l01_p10_c15_pp05</i>
	<i>l02_p05_c10_pp02</i>	<i>l02_p05_c15_pp02</i>	<i>l02_p10_c15_pp03</i>
	<i>l02_p05_c10_pp03</i>	<i>l02_p05_c15_pp03</i>	<i>l02_p10_c15_pp05</i>
	<i>l04_p05_c10_pp02</i>	<i>l04_p05_c15_pp02</i>	<i>l04_p10_c15_pp03</i>
	<i>l04_p05_c10_pp03</i>	<i>l04_p05_c15_pp03</i>	<i>l04_p10_c15_pp05</i>

Todos os testes computacionais foram executados em uma estação de trabalho Linux com dois processadores de quatro núcleos Intel Xeon E5504 a 2.00 GHz, 24 GB RAM. A versão 12.4 do IBM ILOG CPLEX foi utilizado como *MILP-solver* em modo paralelo oportunístico e limitado a 4 threads. O tempo computacional de todos os métodos foi limitado a 3600 segundos.

Seis métodos *fix-and-optimize* foram testados, de acordo com o tamanho das partições e sobreposições: *FO_1_0*, *FO_2_0*, *FO_2_1*, *FO_3_0*, *FO_3_1* e *FO_3_2*. Como o *ALNS* utiliza funções aleatórias, executamos o algoritmo cinco vezes para cada instância para obter uma performance média. O tempo limite do agente de reparação é de 5 segundos, pois o *ALNS* apresentou melhor comportamento quanto maior o número de iterações. A Figura 2 mostra uma comparação entre as soluções dos melhores métodos relativamente à solução inicial (*Heur*) ao longo do tempo de execução. O gráfico mostra claramente o melhor comportamento do *ALNS*, principalmente no início da execução do método e com uma melhoria de mais de 50% em relação à solução inicial. Os métodos *FO* mais eficientes possuem períodos em sobreposição, pois assim uma mesma parte da solução pode ser modificada por diferentes vizinhanças. O método *FO_2_1*, por possuir uma vizinhança menor, apresenta uma melhoria mais intensa no início, entretanto possui menos diversificação. Já o comportamento de *FO_3_1* e *FO_3_2*, por terem maior vizinhança, apresentam uma melhoria inicial mais suave, entretanto a diversificação pode indicar melhores

oportunidades de melhoria. O método *ALNS*, por meio de seus diferentes agentes de destruição, diversifica a busca por novas soluções, enquanto o tamanho das vizinhanças é escolhido adaptativamente em função da maior intensidade da busca, o que justifica a superioridade do método.

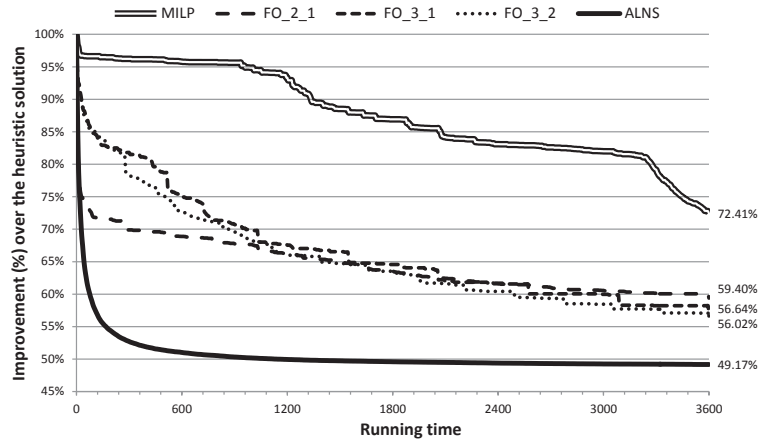


Figura 2: Melhoria média das soluções relativa à solução inicial (*Heur*), ao longo do tempo.

5. Conclusões e/ou perspectivas futuras

Neste estudo, o problema integrado de planejamento de produção e distribuição a nível operacional com produtos perecíveis foi considerado. A perecibilidade dos produtos promove um aumento de complexidade ao problema, reforçando o planejamento conjunto das operações de distribuição e produção. O *OIPDPP* é um problema difícil e não resolvível utilizando métodos exatos, mesmo para problemas com poucos produtos e clientes. Desenvolvemos diferentes métodos de solução com base em programação matemática: (1) heurística construtiva *Heur*; (2) *fix-and-optimize* padrão; e (3) *ALNS*. O *ALNS* apresentou desempenho superior, mostrando que abordagens com procedimentos de intensificação e diversificação adaptativos podem levar a boas soluções. Os vários agentes promovem buscas em diferentes vizinhanças, resultando em soluções mais robustas. A melhoria do *ALNS* é mais rápida em comparação aos outros métodos.

Agradecimentos

Os autores agradecem aos subsídios concedidos pela FAPESP (2010/06901-1), CNPq (590003/2009-7) e projeto NORTE-07-0124-FEDER-000057, cofinanciado pelo Programa Operacional Regional do Norte (ON.2 O Novo Norte), ao abrigo do Quadro de Referência Estratégico Nacional (QREN), através do Fundo Europeu de Desenvolvimento Regional (FEDER).

Referências

- Amorim, P.; Belo-Filho, M.; Toledo, F.; Almeder, C.; Almada-Lobo, B. (2013a). Lot sizing versus batching in the production and distribution planning of perishable goods. *International Journal of Production Economics*, 146(1), 208–218.
- Amorim, P.; Meyr, H.; Almeder, C.; Almada-Lobo, B. (2013b). Managing perishability in production-distribution planning: a discussion and review. *Flexible Services and Manufacturing Journal*, 25(3), 389–413.

MODELOS HÍBRIDOS PARA O PROBLEMA DA MOCHILA BIDIMENSIONAL COM CONFLITOS

Pedro Hokama

Instituto de Computação - Universidade Estadual de Campinas
hokama@ic.unicamp.br

Thiago A. de Queiroz

Departamento de Matemática, Universidade Federal de Goiás - Campus de Catalão
th.al.qz@catalao.ufg.br

Flávio Keidi Miyazawa

Instituto de Computação - Universidade Estadual de Campinas
fkm@ic.unicamp.br

RESUMO

Neste trabalho estamos interessados no Problema da Mochila Bidimensional com Conflitos. Apresentamos uma modelagem em programação linear inteira que resolve o problema através da abordagem *branch-and-cut*, em que separações são encontradas por um algoritmo modelado por programação por restrições. Comparamos nossos resultados com a literatura, através de testes computacionais que comprovaram a eficiência do nosso método.

PALAVRAS-CHAVE: Programação Linear Inteira, Programação por Restrições, Branch-and-Cut.

1. Introdução

No problema da Mochila Bidimensional com Conflitos, recebemos um conjunto de itens e um recipiente bidimensional, em que cada item possui um valor associado. Também recebemos uma lista de conflitos, ou seja, conjuntos de pares de itens que não são compatíveis. O objetivo é encontrar um subconjunto dos itens, que maximize o valor e que possam ser empacotados no recipiente, sendo que dois itens conflitantes entre si não podem estar simultaneamente no recipiente. Esse problema foi investigado na versão unidimensional por Yamada et al. (2002), e a versão bidimensional que denotaremos por 2KPC, foi recentemente investigado por Queiroz e Miyazawa (2013). Na Seção 2 descrevemos formalmente o problema e na Seção 3 descrevemos a formulação de Queiroz e Miyazawa (2013). Depois apresentamos uma nova abordagem *branch-and-cut* para o problema na Seção 4. Por fim nas seções 5 e 6 apresentamos os resultados computacionais, conclusões e trabalhos futuros.

2. Descrição do problema

A seguir, definimos o problema da mochila bidimensional com conflitos. São dados como entrada: (i) Um conjunto I de itens bidimensionais, onde cada item $i \in I$ tem dimensões (w_i, h_i) e um valor associado v_i ; (ii) Um recipiente B de dimensões (W, H) ; (iii) Um conjunto de conflitos C , onde cada conflito em C é composto por um par de itens $\{i, j\}$. O objetivo é encontrar um subconjunto $J \subseteq I$ e um empacotamento dos itens de J em B , cujo valor total dos itens em J é maximizado e para todo conflito $\{i, j\}$ pertencente a C , temos que i e j não pertencem a J simultaneamente.

3. Formulação por Programação Linear Inteira

A formulação apresentada por Queiroz e Miyazawa (2013) considera uma malha de pontos, obtidos através dos *reduced raster points* (Scheithauer e Terno, 1996), onde os itens podem ser empacotados. Considere S o conjunto de todos os pontos da malha, e S_i o conjunto de pontos onde o item i pode ser empacotado. Além disso, D_{ip} é o conjunto de pontos q , tal que o item i empacotado em q cobre o ponto p . O número de conflitos em que i está presente é representado por α_i . Considere a variável binária $x_{ip} = 1$ se o item i foi empacotado no ponto p , e 0 caso contrário.

$$\max \quad \sum_{i \in I} \sum_{p \in S_i} x_{ip} v_i, \quad (1)$$

$$\text{sujeito a} \quad \sum_{p \in S_i} x_{ip} \leq 1, \quad i \in I, \quad (2)$$

$$\sum_{i \in I} \sum_{q \in D_{ip}} x_{iq} \leq 1, \quad \forall p \in S, \quad (3)$$

$$\sum_{\{i,j\} \in C} \sum_{q \in S_j} x_{jq} \leq (1 - \sum_{p \in S_i} x_{ip}) |\alpha_i|, \quad \forall i \in I, \quad (4)$$

$$x_{ip} \in \{0, 1\}, \quad i \in I; \forall p \in S_i. \quad (5)$$

Nessa formulação as restrições (2) garantem que um item da solução só está atribuído a no máximo um ponto. As restrições (3) evitam que dois itens empacotados ocupem um mesmo ponto p . As restrições (4) garantem que se um item i for empacotado em qualquer ponto, nenhum dos itens conflitantes estará na solução. Por fim as restrições (5) garantem que as variáveis são binárias. Denotaremos essa formulação como *Grid*.

4. Algoritmo Branch-and-cut com Programação por Restrições

Nessa formulação o problema da mochila com conflitos é resolvida por Programação Linear Inteira e a cada nó verificamos se a solução parcial pode ser empacotada. Essa verificação é feita por um resolvidor utilizando Programação por Restrições. Para essa formulação, seja q_i uma variável binária que indica a pertinência do item i na solução. Seja $s_i = h_i w_i$ o volume do item i e $S = HW$ o volume no recipiente B . Seja \mathcal{R} o conjunto de todas as combinações de itens que não podem ser empacotadas simultaneamente no recipiente. Por exemplo, se o conjunto de itens $\{3, 5, 8\}$ não pode ser empacotado no recipiente, então $\{3, 5, 8\} \in \mathcal{R}$. Dessa forma podemos formular o problema como segue,

$$\max \quad \sum_{i \in I} q_i v_i, \quad (6)$$

$$\text{sujeito a} \quad \sum_{i \in I} q_i s_i \leq S, \quad (7)$$

$$q_i + q_j \leq 1, \quad \forall \{i, j\} \in C, \quad (8)$$

$$\sum_{i \in R} q_i \leq |R| - 1, \quad \forall R \in \mathcal{R}, \quad (9)$$

$$q_i \in \{0, 1\} \quad \forall i \in I. \quad (10)$$

Obviamente, o número de elementos em \mathcal{R} pode ser exponencial, portanto relaxamos a formulação, removendo as restrições (9), e adicionamos as restrições dessa família como cortes. Infelizmente descobrir se uma combinação de itens pertence a \mathcal{R} é um problema NP-difícil, já que é equivalente a resolver um problema de empacotamento bidimensional.

Para resolver o problema de empacotamento bidimensional para uma dada combinação de itens R , utilizamos nessa fase uma modelagem em Programação por Restrições descrita a seguir. Seja X_i a variável que representa a coordenada X em que o item i será empacotado,

e Y_i a variável que representa a coordenada Y em que o item i será empacotado. Iniciamos a formulação atribuindo o domínio as variáveis.

$$\text{Dom}(X_i) = [0, \dots, W - w_i], \quad (11)$$

$$\text{Dom}(Y_i) = [0, \dots, H - h_i]. \quad (12)$$

Como a restrição (8) é respeitada pelos itens de R , não há conflitos e a única restrição necessária é a de que os itens não se sobreponham, dessa forma adicionamos:

$$\begin{aligned} X[i] + w[i] &\leq X[j] \text{ ou } Y[i] + h[i] \leq Y[j] \text{ ou} \\ X[j] + w[j] &\leq X[i] \text{ ou } Y[j] + h[j] \leq Y[i]. \end{aligned} \quad (13)$$

Se o resolvidor de programação por restrições provar que o conjunto de itens R não pode ser empacotado, adicionamos um corte da família (9), senão continuamos o algoritmo. Denotaremos essa formulação como *BNC-CP*.

5. Resultados

Os algoritmos propostos foram implementados em c++ e compilados com g++ versão 4.6.4 em um computador com sistema operacional Linux. Os experimentos foram executados em um Processador Intel Xeon 2.93GHz. O resolvidor de programação linear escolhido foi o CPLEX 12.5.1 e o resolvidor de Programação por Restrições foi o CP solver 1.7, ambos da IBM ILOG.

A Tabela 1 apresenta as 48 instâncias geradas por Queiroz e Miyazawa (2013), a quantidade de itens e a quantidade de conflitos em cada instância. Também é apresentado o valor da solução obtida, o tempo de execução e se foi provada a otimalidade da solução por 3 diferentes formulações. A primeira é a formulação *Grid* apresentada em (Queiroz e Miyazawa, 2013) e descrita na Seção 3., a segunda é a formulação *BNC-CP*, a terceira formulação é idêntica a formulação *BNC-CP*, porém substituindo o modelo que encontra a viabilidade do empacotamento pelo descrito por Clautiaux et al. (2008), esse modelo denominamos por *BNC-CP2D*. Para a formulação *Grid* o tempo de execução foi limitado a 3200 segundos, e nesse tempo conseguiu provar a otimalidade de 31 das 48 instâncias com um tempo médio de 1640,80 segundos. A abordagem branch-and-cut se mostrou muito eficiente. Para os modelos seguintes o tempo de execução foi limitado a 600 segundos. Nesse limite o modelo *BNC-CP* apenas não provou a otimalidade em 2 instâncias, porém encontrou uma solução ótima em todas as 48 instâncias, com um tempo médio de execução de 51 segundos.

Já a formulação *BNC-CP2D* se mostrou a mais eficiente, resolvendo todas as instâncias na otimalidade, com um tempo médio de apenas 0.03 segundos, sendo que a instância que levou mais tempo, foi executada em apenas 0.55 segundos. O resultado desse modelo foi bastante satisfatório.

6. Conclusões e/ou perspectivas futuras

Nesse trabalho investigamos um problema da mochila bidimensional com conflitos, um problema pouco investigado na literatura. Apresentamos um modelo que melhorou consideravelmente o tempo para encontrar soluções ótimas. Estamos investigando novos modelos e melhorias e esperamos apresentá-las em trabalhos futuros.

Referências

Clautiaux, F.; Jouglet, A.; Carlier, J.,; Moukrim, A. (2008). A new constraint programming approach for the orthogonal packing problem. *Computers and Operations Research*, 35(3), 944 – 959.

Tabela 1: Instâncias e Resultados anteriores

Instancia	Itens	Conflitos	Grid			BNC-CP			BNC-CP2D		
			Valor	Tempo	Otimo	Valor	Tempo	Otimo	valor	Tempo	Opt
2kdc01	25	21	6203	91.82	SIM	6203	17.54	SIM	6203	0.05	SIM
2kdc02	25	21	5568	504.55	SIM	5568	5.19	SIM	5568	0.19	SIM
2kdc03	25	21	2354	0.13	SIM	2354	0.14	SIM	2354	0.05	SIM
2kdc04	25	21	7487	3200	NAO	7487	0.54	SIM	7487	0.01	SIM
2kdc05	25	30	4245	1173.77	SIM	4245	2.11	SIM	4245	0.2	SIM
2kdc06	25	30	4516	1484.72	SIM	4516	137.25	SIM	4516	0.24	SIM
2kdc07	25	30	2793	4.26	SIM	2793	0.25	SIM	2793	0.31	SIM
2kdc08	25	30	7905	3200	NAO	7905	0.12	SIM	7905	0.01	SIM
2kdc09	25	38	4663	3200	NAO	4663	12.57	SIM	4663	0.08	SIM
2kdc10	25	38	3826	174.77	SIM	3826	0.77	SIM	3826	0.04	SIM
2kdc11	25	38	3764	1.39	SIM	3764	0.02	SIM	3764	0	SIM
2kdc12	25	38	7701	3109.21	SIM	7701	0.26	SIM	7701	0.01	SIM
2kdc13	25	47	4649	391.32	SIM	4649	14.15	SIM	4649	0.29	SIM
2kdc14	25	47	4754	679.28	SIM	4754	4.28	SIM	4754	0.03	SIM
2kdc15	25	47	3350	0.47	SIM	3350	0.07	SIM	3350	0.02	SIM
2kdc16	25	47	5396	3200	NAO	5396	1.45	SIM	5396	0.26	SIM
2kdc17	25	21	8138	1.69	SIM	8138	137.56	SIM	8138	0.11	SIM
2kdc18	25	21	6923	3.93	SIM	6923	0.16	SIM	6923	0.29	SIM
2kdc19	25	21	6560	0.66	SIM	6560	0.11	SIM	6560	0.03	SIM
2kdc20	25	21	12608	3200	NAO	12608	520.39	SIM	12608	0.58	SIM
2kdc21	25	30	11209	139.14	SIM	11209	0.89	SIM	11209	0.09	SIM
2kdc22	25	30	6199	1.11	SIM	6199	9.09	SIM	6199	0.29	SIM
2kdc23	25	30	4207	0.08	SIM	4207	0.11	SIM	4207	0.24	SIM
2kdc24	25	30	12385	3200	NAO	12385	6.88	SIM	12385	0.08	SIM
2kdc25	25	38	9552	3200	NAO	9552	10	SIM	9552	0.08	SIM
2kdc26	25	38	7045	547.99	SIM	7045	2.35	SIM	7045	0.07	SIM
2kdc27	25	38	4555	0.36	SIM	4555	0.22	SIM	4555	0.41	SIM
2kdc28	25	38	11838	3200	NAO	11838	0.05	SIM	11838	0.01	SIM
2kdc29	25	47	5677	688.7	SIM	5677	2.6	SIM	5677	0.53	SIM
2kdc30	25	47	8402	133.77	SIM	8402	0.67	SIM	8402	0.02	SIM
2kdc31	25	47	7158	3131.89	SIM	7158	0.13	SIM	7158	0.02	SIM
2kdc32	25	47	10228	3200	NAO	10228	6.32	SIM	10228	0.02	SIM
2kdc33	25	21	15516	3200	NAO	15516	648.24	NAO	15516	0.09	SIM
2kdc34	25	21	14258	3200	NAO	14258	2.49	SIM	14258	0.03	SIM
2kdc35	25	21	7545	58.21	SIM	7545	0.17	SIM	7545	0.37	SIM
2kdc36	25	21	23253	3200	NAO	23391	220.34	SIM	23391	0.1	SIM
2kdc37	25	30	13512	0.4	SIM	13512	2.65	SIM	13512	0.02	SIM
2kdc38	25	30	16011	3126.36	SIM	16011	31.02	SIM	16011	0.03	SIM
2kdc39	25	30	6647	0.25	SIM	6647	0.25	SIM	6647	0.5	SIM
2kdc40	25	30	18012	3200	NAO	18210	4.4	SIM	18210	0.01	SIM
2kdc41	25	38	9529	31.33	SIM	9529	5.03	SIM	9529	0.36	SIM
2kdc42	25	38	10849	3131.48	SIM	10849	0.88	SIM	10849	0.48	SIM
2kdc43	25	38	9387	2.15	SIM	9387	0.15	SIM	9387	0.04	SIM
2kdc44	25	38	17559	3200	NAO	17559	0.04	SIM	17559	0.01	SIM
2kdc45	25	47	10920	3200	NAO	10920	1.55	SIM	10920	0.32	SIM
2kdc46	25	47	15501	3200	NAO	15501	2.75	SIM	15501	0.03	SIM
2kdc47	25	47	10633	63.09	SIM	10633	0.12	SIM	10633	0.29	SIM
2kdc48	25	47	17913	3200	NAO	17913	632.2	NAO	17913	0.39	SIM

Queiroz, T. A.; Miyazawa, F. K. (2013). Approaches for the 2d 0-1 knapsack problem with conflict graphs. Em *Anais do CLEI 2013, 2013*, 1–8, Naiguata - Venezuela.

Scheithauer, G.; Terno, J. (1996). The g4-heuristic for the pallet loading problem. *Journal of the Operational Research Society*, 47, 511–522.

Yamada, T.; Kataoka, S.,; Watanabe, K. (2002). Heuristic and exact algorithms for the disjointly constrained knapsack problem. *Information Processing Society of Japan Journal*, 43(9).

OTIMIZAÇÃO ESTOCÁSTICA NA PROGRAMAÇÃO DE BOMBAS DE REDES DE ABASTECIMENTO URBANO

Jonathan Justen De La Vega Martínez

Universidade Federal de São Carlos
jhonatank-2@hotmail.com

Douglas José Alem Junior

Universidade Federal de São Carlos
douglas@ufscar.br

RESUMO

Esse estudo apresenta um problema de programação de bombas para a captação, armazenamento e transferência de água em sistemas de abastecimentos de água em redes urbanas, cujo objetivo é minimizar o custo de energia elétrica associados às operações de bombeamento. Para lidar com a natureza dinâmica e aleatória da demanda por água, foi proposto um modelo de programação estocástica de dois estágios com recurso, em que a variável aleatória é representada por conjunto finito de realizações ou cenários. O modelo matemático desenvolvido é uma extensão de um modelo determinístico da literatura e reflete a suposição básica de que é possível se incorrer a um custo fixo pelas atividades de liga/desliga das bombas hidráulicas. Para controlar as violações das restrições de demanda por água na presença de múltiplos cenários diferentes, considerou-se também uma técnica de robustez na tentativa de gerar soluções “quase” factíveis. Por último, mas não menos importante, adotou-se um critério de aversão ao risco denominado desvio médio absoluto para obter custos de segundo estágio menos dependentes das realizações dos cenários. Os cenários foram gerados de acordo com um procedimento baseado em simulação Monte-Carlo que pode utilizar qualquer distribuição de probabilidade para produzir as probabilidades empíricas das variáveis aleatórias. Como o problema de programação de bombas com custo fixo proposto é um programa estocástico de dois estágios 0 – 1, desenvolve-se uma heurística híbrida eficiente para obter soluções de boa qualidade de instâncias práticas em um tempo computacional plausível.

PALAVRAS-CHAVE: Programação de Bombas em Redes Urbanas; Programação Estocástica; Análise de Robustez; Desvio Médio Absoluto; Simulação Monte-Carlo.

1. Introdução

A preservação dos recursos energéticos e hídricos para a sua disponibilização às gerações futuras é um tema que tem causado uma considerável preocupação mundial. Tal preocupação origina esforços na proposição de novas técnicas de manejo eficiente de água e uso racional de energia elétrica. No Brasil, o setor de abastecimento de água e de tratamento de esgoto é responsável por 3% do consumo de eletricidade, sendo que mais de 90% desse consumo é destinado apenas às operações de bombeamento de água. Além disso, estima-se que 35% a 42% da água tratada é perdida no vazamento da rede (PROSAB, 2009), o que compromete o abastecimento e contribui para o aumento do custo das operações de bombeamento e distribuição de água em redes urbanas. Desta forma, programar a ativação de bombas para a captação de água e planejar as atividades de armazenamento e distribuição de água são essenciais para minimizar os custos de energia elétrica, perda de água e manter a eficiência dos sistemas de abastecimento urbano.

A programação das bombas indica quais delas serão usadas e em que períodos do dia as mesmas serão ligadas. O planejamento da distribuição de água em redes de abastecimento consiste em definir a fração do tempo em que as bombas serão mantidas ligadas para captar a água e atender uma expectativa de demanda. O problema de programar a ativação de bombas hidráulicas e planejar a distribuição de água em redes de abastecimento para otimizar os custos associado à energia elétrica, é conhecido na literatura como *Optimal Control Problem* (OCP), o Problema de Controle Ótimo Ormsbee e Lansey (1994). Para abordar o OCP, geralmente, o horizonte de planejamento é dividido em intervalos discretos de 1 hora de duração. Com isso, em cada intervalo as decisões ótimas de ativação de bombas, de captação, de armazenagem e de transferência de água são determinadas.

Para definir uma política ótima de captação, armazenagem e transferência de água, entretanto, é necessário conhecer a demanda por água da rede de abastecimento. Na prática, é comum que tal demanda seja especificada em horas, pois ela apresenta, indiscutivelmente, um ciclo diário importante e as tarifas são diferenciadas ao longo das horas do dia. Porém, mesmo a demanda de *curtíssimo prazo* apresenta uma natureza dinâmica e difícil de ser prevista, pois depende de diversos fatores inerentemente aleatórios, como temperatura, precipitação e umidade. Além disso, a demanda também depende do dia da semana, do mês, estação, perfil socioeconômico e cultural da população e do próprio custo de utilização da água Shvartser et al. (1993); Odan e Reis (2012).

Tipicamente, a natureza aleatória da demanda por água é tratada via o desenvolvimento de métodos de previsão, que podem ser de longo- ou de curto-prazo (Alvisi et al., 2007; Adamowski, 2008; Herrera et al., 2010; Nasser et al., 2011; Odan e Reis, 2012). Assim, as políticas ótimas de controle podem se acionadas para os perfis de demanda gerados pelos métodos de previsão. Todavia, se em algum momento, for constatada uma diferença significativa entre a demanda estimada e a que de fato se materializou ao longo das horas do dia, é necessário reotimizar o modelo de controle ótimo com os dados atualizados, ainda assim assumindo as estimativas sobre as demandas futuras (Alvisi et al., 2007). Uma possível desvantagem em acionar as políticas de controle ótimo com as demandas estimadas é a ineficiência das atividades de captação, armazenagem e transferência de água se as estimativas não forem muito precisas. Se, por um lado, a estimativa da demanda ignorar as realizações históricas pessimistas, será necessário realizar operações extras de bombeamento nos períodos críticos em que o custo de energia elétrica é mais elevado. Se, por outro lado, a estimativa da demanda for muito pessimista, provavelmente as bombas hidráulicas serão mantidas ligadas em intervalos maiores de tempo do que o necessário, incorrendo num aumento significativo e desnecessário no consumo de energia elétrica.

Uma maneira de tornar as políticas ótimas de controle menos dependentes de *uma única demanda estimada* é considerar um conjunto de possíveis realizações para a mesma, ponderando-as de acordo com a chance de ocorrência. Assim, parte dessas políticas serão “obrigadas” a satisfazer todos os perfis de demanda simultaneamente, gerando decisões *razoavelmente boas* ou mesmo subótimas para o problema. Dentre as técnicas de otimização que são baseadas nesse paradigma, tem-se a Programação Estocástica de dois Estágios, que é uma das técnicas de otimização sob incertezas mais difundidos na literatura para lidar de forma explícita com as incertezas envolvidas em um problema. Nesse caso, as políticas de controle que devem satisfazer todas as realizações da demanda ao mesmo tempo são denominadas decisões de *primeiro estágio*. Ainda, associa-se a cada possível realização da demanda um conjunto de variáveis de decisão denominadas de *segundo estágio*, cujo objetivo é corrigir as inviabilidades que surgem após a materialização das variáveis aleatórias, dada a escolha de primeiro estágio.

2. Descrição do problema

A fim de facilitar a descrição do problema de programação de bombas ou de controle ótimo, considere a rede de abastecimento urbano ilustrada na Figura 1.

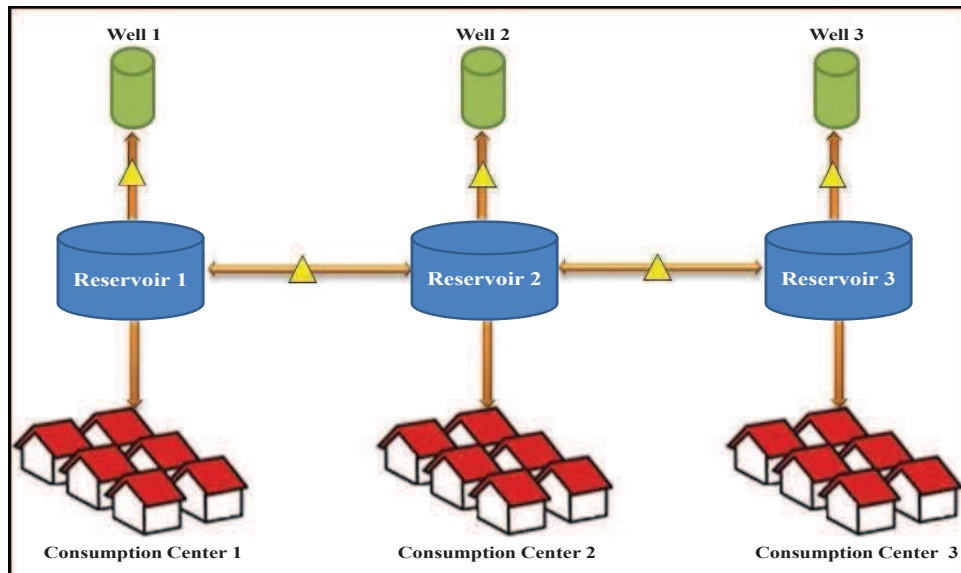


Figura 1: Exemplo de uma rede de abastecimento urbano de água (Adaptado de Toledo et al. (2008)).

Nessa rede, cada poço pode abastecer somente o reservatório com o qual apresenta ligação direta. Do mesmo modo, cada reservatório pode suprir o centro de consumo relacionado. No entanto, como os reservatórios estão localizados em partes altas da cidade, não é necessária a designação de bombas entre eles a seu centro de consumo, uma vez que a distribuição de água é feita por força gravitacional. Além disso, existem bombas entre os diferentes reservatórios para indicar a existência de abastecimento de água entre os mesmos. Porém, como podem existir pares de reservatórios que não apresentam ligação direta, então, necessariamente, a vazão de água entre um par de reservatórios deve passar pelos reservatórios alocados entre eles.

O problema de controle ótimo, abordado neste trabalho, consiste em determinar as políticas de ativação de bombas, de captação, de armazenamento e de transferência de água sobre um período de tempo predefinido, tal que a demanda por água seja atendida sem atrasos, as condições hidráulicas da rede sejam satisfeitas e o custo total de consumo de energia elétrica seja minimizado Zessler e Shamir (1989).

3. Métodos e abordagens

O presente trabalho tem como objetivo principal propor um modelo de programação estocástica de dois estágios para o problema de captação, armazenamento e transferência de água em redes urbanas, considerando a demanda por água uma variável aleatória que pode ser razoavelmente aproximada por um conjunto de realizações. O modelo matemático desenvolvido é uma extensão do modelo determinístico proposto no trabalho de Toledo et al. (2008) e reflete a suposição básica de que pode haver um custo fixo incorrido pelas atividades de liga/desliga das bombas hidráulicas. Na programação estocástica baseado em cenários, um conjunto de realizações da variável aleatória são consideradas de forma simultânea no modelo de

otimização. Por essa razão, medidas de aversão ao risco e medidas para quantificar situações de “déficits” e “superávits” são incorporadas ao problema de otimização a fim de tornar a solução e o modelo do problema de programação estocástica menos sensível às diferentes realizações da variável aleatória. As possíveis realizações da variável aleatória foram geradas por um método de amostragem baseada em simulação de Monte-Carlo. Como o modelo matemático com custo fixo é um problema de programação estocástica 0-1 de grande porte, propôs-se um algoritmo heurístico baseado em Soler (2008) para obter soluções de boa qualidade num tempo computacional aceitável. Convém salientar que o modelo matemático do problema de programação de bombas não será apresentado neste resumo, pois o limite de páginas seria ultrapassado.

4. Resultados

Os resultados evidenciam a estabilidade do método de geração de cenários, a sensibilidade da solução de acordo com parâmetros-chave do modelo matemático, e a eficiência da heurística na resolução de instâncias de grande porte. Finalmente, foi demonstrado que é possível poupar recursos pela resolução do modelo de programação estocástica, em vez de adotar abordagens mais simples baseadas no valor esperado.

5. Conclusões e/ou perspectivas futuras

Resultados da análise dos níveis de serviços sugerem que estratégias para se atender a demanda de água dos centros de consumo devem ser estudadas, pois há uma demanda reprimida relevante, principalmente quando o vazamento da rede excede 10%. Uma possível estratégia é aumentar a capacidade dos reservatórios, o número de reservatórios e a vazão das bombas hidráulicas, com isso, maiores volumes de água são captadas ao longo dos períodos do dia aumentando as chances de atender a demanda faltante.

Há diversas perspectivas de continuidade do presente estudo. Aquelas que parecem mais promissoras são: (1) Usar outros métodos de geração de cenários; (2) Usar métodos de previsão para melhorar o desempenho dos métodos de geração de cenários; (3) Estender a versão de programação estocástica de dois estágios à multi-estágio e (5) Formular o problema de programação de bombas sob o paradigma de otimização robusta preconizado no trabalho de Ben-Tal e Nemirovski (2000).

Referências

- Adamowski, J. F.** (2008). Peak daily water demand forecast modeling using artificial neural networks. *Journal of Water Resources Planning and Management*, 134(2), 119–128.
- Alvisi, S.; Franchini, M.; Marinelli, A.** (2007). A short-term, pattern-based model for water-demand forecasting. *Journal of Hydroinformatics*, 9(1), 39.
- Ben-Tal, A.; Nemirovski, A.** (2000). Robust solutions of linear programming problems contaminated with uncertain data. *Mathematical Programming Series B*, 88.
- Herrera, M.; Torgo, L.; Izquierdo, J.; Pérez-García, R.** (2010). Predictive models for forecasting hourly urban water demand. *Journal of Hydrology*, 387(1-2), 141–150.
- Nasseri, M.; Moeini, A.; Tabesh, M.** (2011). Forecasting monthly urban water demand using extended kalman filter and genetic programming. *Expert Systems with Applications*, 38(6), 7387 – 7395.

- Odan, F.; Reis, L.** (2012). *Hybrid water demand forecasting model associating artificial neural network with fourier series*chapter 115, 1287–1305. *Water Distribution Systems Analysis* 2010.
- Ormsbee, L.; Lansey, K.** (1994). Optimal control of water supply pumping systems. *Journal of Water Resources Planning and Management*, 120(2), 237–252.
- PROSAB** (2009). *Uso racional de água e energia: Conservação de água e energia em sistemas prediais e públicos de abastecimento de água*. Universidade Federal do Espírito Santo. Programa de pesquisa em saneamento básico.
- Shvartser, L.; Shamir, U.,; Feldman, M.** (1993). Forecasting hourly water demands by pattern recognition approach. *Journal of Water Resources Planning and Management*, 119(6), 611–627.
- Soler, E.** (2008). Otimização dos custos de energia elétrica na programação do armazenamento e distribuição de água em redes urbanas. Master's thesis, Instituto de ciências matemáticas e de computação. Universidade de São Paulo.
- Toledo, F. M. B.; dos Santos, M. O.; Arenales, M. N.,; Júnior, P. S.** (2008). Logística de distribuição de água em redes urbanas - Racionalização energética. *Pesquisa Operacional*, 28(1), 75–91.
- Zessler, U.; Shamir, U.** (1989). Optimal operation of water distribution systems. *Journal of Water Resources Planning and Management*, 115(6), 735–752.

Pôsteres

ABORDAGEM GRAFO E/OU PARA SOLUÇÃO DO PROBLEMA DE CORTE BIDIMENSIONAL COM SOBRAS APROVEITÁVEIS

Luiz Carlos Felix Ribeiro

UNESP Bauru - Faculdade de Ciências

luizfelix@yahoo.com

Adriana Cherri, Andréa Vianna

UNESP Bauru - Faculdade de Ciências

adriana@fc.unesp.br, vianna@fc.unesp.br

RESUMO

Este trabalho aborda o problema de corte bidimensional com sobras aproveitáveis (PCESA) estudado por Cherri (2009) que busca a minimização das perdas. Dessa forma, métodos e procedimentos heurísticos da literatura são empregados permitindo gerar retalhos que serão utilizados para atender demandas futuras, não sendo contabilizados como perda.

PALAVRAS-CHAVE: aproveitamento de sobras, abordagem em Grafo E/OU.

1. Introdução

O estudo que envolve PCESA é recente na literatura e, em grande parte, voltado ao caso unidimensional. Este trabalho, por sua vez, tem foco no caso bidimensional. Portanto, busca-se resolver o PCE minimizando a perda gerada, de forma que os itens que compõem o padrão de corte podem ser rearranjados ou trocados, a fim de se obter perdas nulas, pequenas ou retalhos.

2. Descrição do problema

O PCESA tem por finalidade a obtenção de itens a partir do corte de placas padronizadas (compradas de fornecedores) ou não padronizadas (retalhos gerados durante o processo de corte) disponíveis em estoque, de modo que a sobra de material seja minimizada e simultaneamente permita que uma quantidade limitada de retalhos retorne ao estoque.

3. Métodos e abordagens

Para resolver o PCESA alterações estão sendo realizadas na abordagem proposta por Cherri (2009), que utiliza a abordagem Grafo E/OU (Morabito, 1989). O programa anterior está sendo reescrito em linguagem C++ a fim de possibilitar a utilização do software CPLEX. Além disso, as modificações realizadas visam melhorar o desempenho do software atual.

4. Resultados

O estudo está em fase intermediária e ainda não apresenta resultados concretos.

5. Conclusões e/ou perspectivas futuras

O PCESA apresenta aplicações na indústria moveleira e desta forma, é pertinente o estudo. A estratégia proposta está em fase intermediária de implementação e após sua conclusão, testes computacionais serão realizados com problemas reais.

Referências

- [1] Cherri, A.C., *Algumas extensões do problema de corte de estoque com sobras de material aproveitáveis*. Tese de doutorado, ICMC - USP, São Carlos, Brasil (2009).

Aplicação de metaheurísticas ao problema de dimensionamento de lotes e sequenciamento de produção na indústria de nutrição animal

André Badawi Missaglia

Instituto de Ciências Matemáticas e de Computação - Universidade de São Paulo
andrebm@usp.br

Cláudio Fabiano Motta Toledo

Instituto de Ciências Matemáticas e de Computação - Universidade de São Paulo
claudio@icmc.usp.br

RESUMO

Este trabalho aplica metaheurísticas híbridas para resolver o problema de dimensionamento de lotes e sequenciamento da produção com tempos de setup que ocorre na indústria de nutrição animal. Assim como em outras indústrias, a quantidade de lotes bem como a sequência de produção dentro de um horizonte de planejamento deve ser determinado por tomadores de decisão na indústria de nutrição animal.

Todos os tipos de ração animal seguem a mesma linha de produção, e se utilizam dos mesmos recursos básicos: silos, máquinas de dosagem, pré-misturadores, misturador e empacotamento. O tempo gasto nessa operação varia entre diferentes famílias de produtos. O misturador é considerado o gargalo do sistema e determina a capacidade produtiva da linha.

Uma característica é a existência de produtos contaminantes, fazendo com que toda a linha seja limpa, e portanto o tempo gasto com a preparação da linha faz com que toda a sequência leve mais tempo. No problema estudado, a capacidade de tempo pode ser violada, resultando em custo de overtime, o qual deve ser minimizado. O problema ainda considera a possibilidade de atrasos no atendimento da demanda, mas que deve ser evitado.

Trabalhos anteriores resolveram instâncias reais, onde foi considerado um tempo de preparação igual para todos os casos. Foram utilizados a partir de formulações matemáticas, solvers comerciais como CPLEX para encontrar soluções exatas, ou heurísticas baseadas em programação matemática.

Este trabalho estuda o caso onde diferentes tempos de setup podem ocorrer, sendo mais próximo de situações reais. As abordagens híbridas combinam um algoritmo genético, o qual explora sequências, e tanto uma heurística gulosa quanto soluções exatas de um modelo reduzido determinam o dimensionamento de lotes.

Métodos de busca locais são aplicados em cima dos melhores indivíduos durante a execução do algoritmo genético, onde Busca Tabu e *Simulated Annealing* são analisados. Resultados computacionais mostram a efetividade de tais métodos, levando em consideração também múltiplos processadores.

Os resultados indicam que soluções obtidas pelo solver exato são superadas em diversas das instâncias estudadas por abordagens híbridas, considerando também múltiplos processadores.

PALAVRAS-CHAVE: Metaheurísticas, Indústria de Nutrição Animal, Dimensionamento de Lotes, Sequenciamento de Produção.

HEURÍSTICAS PARA A OBTENÇÃO DE SOLUÇÕES INTEIRAS EM PROBLEMAS DE CORTE DE ESTOQUE COM SOBRAS APROVEITÁVEIS

Douglas Nogueira do Nascimento

Universidade Estadual Júlio de Mesquita Filho – Bauru
douglasnn92@gmail.com

Adriana Cristina Cherri

Universidade Estadual Júlio de Mesquita Filho – Bauru
adriana@fc.unesp.br

RESUMO

Neste trabalho de iniciação científica, desenvolvemos um estudo que envolve problemas de corte de estoque (PCE) unidimensionais e consideraremos que as sobras de material (pedaço cortado não demandado) podem retornar ao estoque para serem utilizadas no futuro.

PALAVRAS-CHAVE: problema de corte de estoque unidimensional, aproveitamento de sobras, modelagem matemática.

1. Introdução

Um problema pouco estudado e comumente encontrado na prática industrial consiste em determinar uma política de aproveitamento das sobras de objetos cortados. Estas sobras, desde que não sejam demasiadamente pequenas, podem retornar ao estoque como retalhos para atender demandas futuras e, portanto, não são consideradas perdas. Neste trabalho revisamos alguns problemas de corte de estoque com sobras aproveitáveis (PCESA) e suas aplicações. Um modelo matemático foi estudado e implementações computacionais estão sendo desenvolvidas.

2. Descrição do problema

O PCESA tem por finalidade a obtenção de itens com dimensões e quantidades especificadas, a partir do corte de objetos padronizados (comprados de fornecedores) ou não padronizados (retalhos gerados durante outros processos de corte) disponíveis em estoque de modo que a sobra de material seja minimizada e simultaneamente, permitindo que uma quantidade limitada de retalhos retorne ao estoque.

3. Métodos e abordagens

O modelo matemático utilizado para resolver o PCESA está sendo implementado utilizando o software OPL. Testes computacionais serão realizados com exemplares da literatura, gerados aleatoriamente e, possivelmente, com problemas práticos.

4. Resultados

O trabalho teve início recentemente e, portanto, não apresenta resultados.

5. Conclusões e perspectivas futuras

O PCESA, apesar de ser recente, apresenta aplicações importantes na prática industrial, sendo pertinente o estudo e aprimoramento acadêmico voltado à sua solução. Após concluir a implementação do modelo, pretendemos alterar e/ou propor procedimentos heurísticos para a obtenção de soluções inteiras para o problema.

HEURÍSTICA *RELAX AND FIX* NA SOLUÇÃO DE UM MODELO MONO-ESTÁGIO DE PROGRAMAÇÃO DA PRODUÇÃO DE BEBIDAS.

Carla Ferreira Andrade Cunha

Universidade Federal do Triângulo Mineiro
carlaacunha@hotmail.com

Deisemara Ferreira

Universidade Federal de São Carlos
deise@usfcar.br

Flavio Molina da Silva

Universidade Federal do Triângulo Mineiro
flavio.molina@icte.ufm.edu.br

RESUMO

Neste artigo propomos a aplicação de uma heurística *relax and fix* na solução de um modelo integrado de dimensionamento e sequenciamento de lotes cujo sequenciamento é feito via restrições do modelo ATSP. Os softwares AMPL/CPLEX versão 12.4 foram utilizados na implementação. Nos testes computacionais foram utilizadas instâncias de pequeno porte da literatura. Os resultados obtidos pela heurística foram comparados aos resultados obtidos pelo CPLEX com parâmetros *default*. Eles mostraram que a heurística escolhida é competitiva, pois, obtém solução inteira e factível de maneira mais rápida em todas as instâncias do conjunto e obtém a solução ótima em um tempo inferior ao do CPLEX para 46% das instâncias do grupo.

PALAVRAS-CHAVE: Programação inteira mista, Programação da produção, Heurística *relax and fix*.

1. Introdução

Para se determinar a programação da produção de uma fábrica é necessário considerar vários fatores como por exemplo tempo disponível para produção, disponibilidade de insumos, demanda a ser atendida, quantidade de itens a serem produzidos etc. Em alguns casos, além desses fatores, a ordem da produção dos itens tem influência nos tempos e/ou custos de preparo das máquinas. Isso acontece na fabricação de bebidas, iogurtes, tintas, injeção de plásticos, dentre outros. Esses casos deram origem a estudos da literatura que tratam simultaneamente o dimensionamento e o sequenciamento do problema considerando tempos e/ou custos de preparo de máquinas. Um destes estudos é sobre o problema de programação da produção de bebidas. Na literatura encontra-se diversos modelos matemáticos e heurísticas para resolver o problema.

O processo de produção de bebidas consiste em duas etapas principais: preparação do xarope e envase da bebida pronta. Em ambas as etapas é necessário definir a programação da produção. Essa programação deve definir quanto, quando e a ordem de produção de cada item em cada linha de envase em um determinado horizonte de planejamento. Envolve, portanto, um número elevado de produtos, tanques/máquinas e, períodos. Para se ter uma ideia da dimensão do problema em uma fábrica de médio-grande porte, por exemplo, são mais de 100 tipos de itens, fabricados, em 7 linhas, 13 tanques na xaroparia, e período de trabalho 24 horas por dia. Dado o número maior de tanques de xarope do que de linha de envase será considerado que um tanque abastece apenas uma linha por vez e que é dedicado a apenas uma linha.

Há ainda restrições de produção que devem ser consideradas como capacidade de tempo disponível para produção, disponibilidade de insumos, capacidades mínimas e máximas de produção na xaroparia e tempos de troca de um item para outro dependentes da sequência de produção tanto nas linhas quanto nos tanques. Além disto, há a sincronia entre os estágios de envase e xaroparia, ou seja, se a xaroparia não estiver com o xarope pronto, a linha de envase deve esperar até que ele fique pronto. Do mesmo modo, se a linha de envase não estiver preparada, a xaroparia deve aguardar para enviar o xarope. Percebemos, portanto, que podem ocorrer esperas entre os dois estágios resultando em perda de tempo de produção. O objetivo da programação é definir um plano que atenda a demanda com o maior lucro possível ou o menor custo possível. Portanto, devem ser considerados os custos envolvidos no processo, como custo de estoque, atrasos e trocas.

Embora a definição do que fazer seja clara, na prática esta tarefa pode exigir mais de 4 horas de dedicação semanal do funcionário, ou seja, mais de meio turno de trabalho. Se for necessário reprogramar a produção outras 4 horas podem ser necessárias. Os modelos propostos na literatura também são difíceis de serem resolvidos. Para instâncias reais, por exemplo, os gaps ainda são altos. Isto se justifica pelo fato do problema ser NP-difícil. Assim a aplicação de heurísticas na solução dos modelos pode colaborar na redução dos gaps, e de tempos computacionais no caso de instâncias com solução ótima.

2. Métodos e abordagens

No presente trabalho foi realizada a implementação de uma heurística *relax and fix* na solução do modelo de otimização linear proposto por Ferreira et al. (2012) e denominado F1. Ele assume que os tanques são dedicados às linhas de envase e os tempos de *setup* não são carregados de um período para o próximo. Além disso, é uma modelagem de estágio único em que a sincronia é feita através da consideração dos tempos de *setup* como sendo os máximos entre os valores de *setup* da xaroparia e do envase. São consideradas restrições de lotes máximos e mínimos, preparo, capacidade da linha, balanceamento de estoque, restrições baseadas no problema do caixeiro viajante para definir o sequenciamento e, restrições de domínio de variáveis. Para maiores detalhes ver Ferreira et al. (2012).

Conforme comentado na seção anterior, resultados da literatura mostram que este modelo ainda é difícil de ser resolvido. Assim uma heurística do tipo *relax and fix* (Wolsey, 1998) foi utilizada para resolvê-lo. A heurística faz a partição da variáveis por período e fixa as variáveis binárias e inteiras. As iterações são feitas do primeiro período do horizonte de planejamento até o último. Na próxima seção são apresentados os resultados computacionais.

3. Resultados Computacionais

Nos testes computacionais foi resolvido um conjunto de 28 instâncias baseadas em dados reais e obtidas na literatura. Estas instâncias possuem as seguintes características: 2 linhas de produção, 4 itens e 3 períodos. Em relação ao número de variáveis e restrições elas possuem 232 variáveis, sendo 24 inteiras, 104 binárias e 104 contínuas, e 216 restrições.

Os estudos computacionais consistem em um primeiro momento em comparar a solução heurística com a primeira solução do software de otimização CPLEX (ILOG, 2008), que é um dos pacotes de otimização mais robustos da atualidade. Em seguida, é feita uma comparação entre o tempo para se encontrar solução ótima utilizando apenas o modelo puro e o tempo para se encontrar essa mesma solução utilizando-se o resultado da heurística *relax and fix* como solução inicial.

Os testes computacionais foram realizados em um computador Intel Core I5 (2,53GHz), 4.0 GB RAM. A linguagem de modelagem AMPL (Fourer et al., 2002) e o software de otimização

CPLEX versão 12.4 (2012) foram utilizados na implementação e solução dos modelos matemáticos. Os parâmetros do CPLEX foram mantidos no modo *default*.

A Tabela 1 a seguir apresenta os resultados obtidos na solução das instâncias E1-E28 com a heurística *relax and fix* e com o *default* CPLEX. A tabela está organizada da seguinte maneira, mostra na primeira coluna o nome da instância, em seguida o valor da solução encontrada pela heurística *relax and fix*, o tempo necessário para obtê-la, o valor da solução ótima obtida pelo CPLEX e seu tempo de obtenção. As últimas colunas fornecem a porcentagem de melhoria da solução e do tempo de processamento da heurística em relação ao pacote de otimização. A última linha contém a média do valor de cada coluna. As melhores soluções estão destacadas em negrito. A Tabela 2 mostra o detalhamento de custos das soluções encontradas.

Tabela 1: Soluções heurística *relax and fix* e método *cut and branch*.

Instâncias	Solução R&F	Tempo R&F	Solução Ótima	Tempo C&B	% Melhoria Solução	% Melhoria Tempo
E1	202,86	0,08	202,86	5,09	0,00	98,43
E2	205,9	0,05	205,78	4,57	-0,06	98,91
E3	223,72	0,05	223,72	5,43	0,00	99,08
E4	208	0,03	207,88	4,96	-0,06	99,40
E5	232,65	0,03	228,13	2,17	-1,98	98,62
E6	257,77	0,02	257,77	4,12	0,00	99,51
E7	252,98	0,03	202,94	0,12	-24,66	75,00
E8	202,86	0,05	202,86	1,31	0,00	96,18
E9	205,78	0,05	205,78	1,12	0,00	95,54
E10	223,72	0,03	223,72	1,22	0,00	97,54
E11	258,73	0,06	226,63	1,76	-14,16	96,59
E12	232,65	0,05	228,13	0,75	-1,98	93,33
E13	257,77	0,03	257,77	0,64	0,00	95,31
E14	252,98	0,03	202,94	0,25	-24,66	88,00
E15	957,15	0,22	212,04	2,57	-351,40	91,44
E16	206,68	0,05	206,68	1,54	0,00	96,75
E17	582,63	0,06	243,37	1,95	-139,40	96,92
E18	677,5	0,23	246,28	2,4	-175,09	90,42
E19	233,55	0,05	229,03	1,29	-1,97	96,12
E20	258,67	0,34	258,67	0,66	0,00	48,48
E21	253,88	0,05	253,88	0,06	0,00	16,67
E22	203,76	0,06	203,76	3,99	0,00	98,50
E23	206,98	0,05	206,68	4,45	-0,15	98,88
E24	224,62	0,03	224,62	4,56	0,00	99,34
E25	209,08	0,03	208,78	3,79	-0,14	99,21
E26	233,55	0,11	229,03	3,88	-1,97	97,16
E27	258,67	0,05	258,67	3,57	0,00	98,60
E28	253,88	0,05	203,84	0,3	-24,55	83,33
Média	284,96	0,07	223,65	2,45	-27,22	90,83

Tabela 2: Detalhamento de custos das instâncias E1-E28.

Instâncias	R&F			C&B		
	Estoque	Atraso	Troca	Estoque	Atraso	Troca
E1	159,47	42,13	1,26	159,47	42,13	1,26
E2	188,32	16,2	1,38	188,32	16,2	1,26
E3	222,46	0	1,26	222,46	0	1,26
E4	206,62	0	1,38	206,62	0	1,26
E5	191,91	39,48	1,26	226,87	0	1,26
E6	236,81	19,7	1,26	236,81	19,7	1,26
E7	167,3	84,95	0,72	202,22	0	0,72
E8	159,47	42,13	1,26	159,47	42,13	1,26
E9	188,32	16,2	1,26	188,32	16,2	1,26
E10	222,46	0	1,26	222,46	0	1,26
E11	171,7	86,76	1,26	225,37	0	1,26
E12	191,91	39,48	1,26	226,87	0	1,26
E13	236,81	19,7	1,26	236,81	19,7	1,26
E14	167,3	84,95	0,72	202,22	0	0,72
E15	125,26	829,72	2,16	194,45	15,43	2,16
E16	188,32	16,2	2,16	188,32	16,2	2,16
E17	187,86	392,62	2,16	241,21	0	2,16
E18	137,16	538,19	2,16	244,12	0	2,16
E19	191,91	39,48	2,16	226,87	0	2,16
E20	236,81	19,7	2,16	236,81	19,7	2,16
E21	167,3	84,95	1,62	167,31	84,95	1,62
E22	159,47	42,13	2,16	159,47	42,13	2,16
E23	188,32	16,2	2,46	188,32	16,2	2,16
E24	222,46	0	2,16	222,46	0	2,16
E25	206,62	0	2,46	206,62	0	2,16
E26	191,91	39,48	2,16	226,87	0	2,16
E27	236,81	19,7	2,16	236,81	19,7	2,16
E28	167,3	84,95	1,62	202,22	0	1,62
Média	184,94	93,36	1,66	208,79	13,23	1,63

Na Tabela 1 nota-se que a heurística *relax and fix* obtém uma solução inteira e factível em menos tempo, 90,83% menos em média, que o CPLEX em todas as instâncias testadas. Além disso, consegue encontrar a solução ótima em 13 instâncias desse conjunto, ou seja, em 46%.

Através da Tabela 2 observa-se que as soluções da heurística têm custos de estoque iguais ou inferiores às soluções apresentadas pelo CPLEX em todo o conjunto de instâncias. Quando não encontrada a solução ótima, o aumento de custos se dá principalmente nos custos de atraso. Já os custos de troca são bem próximos para os dois métodos de solução. Perceba que o custo de estoque médio da heurística *relax and fix* é de 184,94 contra 208,79 do CPLEX. Essa situação se inverte quando é feita uma análise dos custos de atraso, observa-se um custo médio de atraso da heurística de 93,36 contra apenas 13,23 do método exato. Os custos médios de troca são bastante próximos, 1,66 e 1,63, em relação as soluções encontradas, respectivamente, pela *relax and fix* e pelo CPLEX.

4. Conclusões e/ou perspectivas futuras

Concluimos então que a heurística escolhida conseguiu resolver o problema integrado de dimensionamento e sequenciamento da produção de maneira mais rápida em todas as instâncias do conjunto e obtém a solução ótima em um tempo inferior em 46% das instâncias do grupo analisado. Sendo assim, mostrou-se uma estratégia adequada para resolver o problema em estudo. Os próximos passos são testar instâncias de grande porte e outras estratégias de partição e fixação de variáveis.

5. Agradecimentos

Os autores agradecem à Fapemig, FUNEPU, FAPESP e CNPq pelo apoio financeiro.

Referências

- Ferreira, D.; Clark, A. R.; Almada-Lobo, B.; Morabito, R.** (2012). Single-stage formulations for synchronised two-stage lot sizing and scheduling in soft drink production. *International Journal of Production Economics*, 136(2), 255–265.
- Fourer, B.; Gay, D., M.; Kernighan, B., W.** (2002). *AMPL: A Modeling Language for Mathematical Programming*.
- ILOG** (2008). AMPL/CPLEX A user's guide. *Social Semiotics* 1–102.
- Wolsey, L.** (1998). Integer programming. *John Wiley & Sons*.

LOCALIZAÇÃO DE CONTENTORES PARA COLETA DE MATERIAIS RECICLÁVEIS NA ÁREA URBANA

**Lucas Daniel Del Rosso Calache, Osmar Aléssio, Flávio Molina da Silva e Victor C. B.
Camargo**

UFTM (Universidade Federal da Triângulo Mineiro)
lucascalache@hotmail.com

PALAVRAS-CHAVE: programação matemática, localização de facilidades, coleta de material reciclável, PEV.

RESUMO

Este trabalho tem como objetivo geral estudar o problema de localização de contentores (PEVs - Ponto de Entrega Voluntário) para coleta seletiva de material reciclável e como caso teste, utilizamos dados da cidade de Uberaba/MG. Este problema tem especial relevância, uma vez que, os materiais recicláveis possuem grande valor de mercado e toda cidade deve preocupar-se com os resíduos produzidos pela população e o desafio é cumprir o prazo previsto na nova legislação sobre os resíduos sólidos, para que até agosto de 2014 os lixões estejam desativados. Segundo a Lei nº 12.305 (2010), que institui a Política Nacional de Resíduos Sólidos (PNRS), a gestão dos resíduos passa a ser obrigatório, tendo como instrumentos a coleta seletiva, o sistema de logística reversa e a criação de cooperativas.

Inicialmente, detalhes sobre a geração de resíduos de uma população foram estudados. Juntamente com métodos de previsão, definimos o aumento na geração de resíduos para a cidade de Uberaba para os próximos anos.

Através do problema de localização de facilidades podemos analisar sobre a instalação de depósitos regionalizados para receber o material vindo dos PEVs. Essa análise implica em indicar um bairro (ou mais de um) para receber um depósito que consiga receber também o material vindo dos bairros adjacentes. Primeiramente com a ajuda do solver AMPL/CPLEX, testamos um modelo que retorna a quantidade mínima de depósitos para atender toda a cidade e em seguida, aplicamos o modelo para uma análise de sensibilidade na população atendida com o número de depósitos limitados. Com base nos problemas estudados, podemos definir os bairros mais propícios a receber um centro de recolhimento de materiais recicláveis, servindo como base para a decisão de instalação para os depósitos e PEVs.

Desenvolveu-se um método de alocação dos PEVs dentro de um possível bairro, considerando as distâncias entre as intersecções das ruas, buscando um atendimento satisfatório segundo estudos em que a população esteja disposta a aderir. A instalação dos PEVs proporciona facilidade da coleta, reduzindo percursos longos (especialmente em bairros com baixa densidade populacional, como em zonas rurais) e evitando trechos improdutivos na coleta porta a porta; permite a exploração do espaço do PEV para publicidade e eventual obtenção de patrocínio; e também permite a separação e descarte dos recicláveis por tipos, dependendo do estímulo educativo e do tipo de container, o que facilita a triagem posterior.

Cada vez mais é evidente a grande necessidade da implantação de estratégias públicas de recolhimento de material reciclável. Esperamos que o desenvolvimento desse estudo possa auxiliar a administração pública na tomada de decisão de onde alocar os PEVs e depósitos, contribuindo para acordos setoriais entre poder público e fabricantes, onde o titular do serviço público de limpeza urbana e de manejo de resíduos seja devidamente remunerado ao se encarregar das responsabilidades dos fabricantes no manejo correto dos resíduos.

OTIMIZAÇÃO DO USO DE SALAS CIRÚRGICAS

Adelino Carreira, Flávio Molina da Silva

Universidade Federal do Triangulo Mineiro (UFTM)

adelinocarreira@yahoo.com.br

Palavras Chave: centro cirúrgico, otimização, mau uso das salas.

RESUMO

O estudo em questão aborda um tema não muito bem trabalhado em hospitais pelo Brasil: o uso de salas cirúrgicas. Pensando nisso, iniciou-se um estudo de caso da Santa Casa de Misericórdia de Ituverava e o seu respectivo centro cirúrgico. A Santa Casa é um hospital de médio porte e possui um corpo clínico de 50 médicos em que 20 são cirurgiões. O centro cirúrgico é definido como uma área complexa e de acesso restrito que pertence a um estabelecimento assistencial de saúde. Essa unidade é muito importante dentro do contexto hospitalar e exige detalhes minuciosos para facilitar a execução do serviço. O centro cirúrgico possui 5 salas para operação e funciona 24 horas por dia. Devido ao mau uso de horários, escalas, equipamentos e equipe hoje este setor do hospital é deficitário e a administração do hospital precisa resolver essa questão.

A priori, foi feita uma análise dos dados de um ano para começar a fazer tirar as primeiras conclusões. Fora analisados dados como qual a sala que mais recebe cirurgia, qual é o tipo de cirurgia mais realizado, quais médicos realizam qual cirurgia, o tempo de preparo de cada sala, etc. Assim, pode-se fazer uma abordagem mais ampla das salas e seu respectivo uso. Logo após, foi realizado um diagnóstico de cada sala e suas restrições. Ou seja, os aparelhos que a sala possui, se realizam cirurgias de grande, médio ou pequeno porte, etc. Todos esses dados foram fornecidos e conferido pela alta administração do próprio hospital, nos dando mais veracidade no nosso estudo. Há de se ressaltar que o estudo esta em seu começo portanto não há conclusões sobre o caso.

Busca-se métodos heurísticos para melhorar o uso destas salas. Deseja-se o utilizar de ferramentas de gestão, pesquisa operacional e simulação para otimizarmos o uso das mesmas e fazer com que centro cirúrgico não seja mais deficitário para o Hospital.

Planejamento da produção de múltiplos fornos não-simultâneos em uma fundição de pequeno porte

Diego Navarenho

Instituto de Ciências Matemáticas e de Computação - Universidade de São Paulo
dnavarenho@usp.br

Victor C. B. Camargo

Departamento de Engenharia de Produção - Universidade Federal do Triângulo Mineiro
victor@icte.uftm.edu.br

RESUMO

A indústria de fundição é um importante setor da economia brasileira, seus produtos podem ser encontrados em diversos outros setores da indústria bem como em domicílios. Otimizar o planejamento de produção de uma fundição de mercado não é algo simples. Fatores como a grande carteira de produtos e ligas, o tamanho e a complexidade dos pedidos, o uso adequado de fornos, equipamentos e problemas energéticos e de infraestrutura são complicadores na definição de um plano de produção.

Dentre os trabalhos que abordam o planejamento da produção em fundições de pequeno porte, de Araujo et al. (2008) apresentam um modelo com resultados significativos baseado em uma indústria real. O modelo proposto visa a minimização dos custos de atraso e estoque da empresa e é voltado para indústrias que fazem uso de um único forno, algo que não condiz com a realidade de muitas fundições, que dispõem de mais de um forno, mas não podem operar simultaneamente por restrições tecnológicas. Camargo et al. (2012) propõem o modelo MSGLSP—Multi-stage general lot-sizing and scheduling problem - para o problema de planejamento de produção utilizando mais de uma máquina.

O objetivo dessa pesquisa é adaptar o MSGLSP (que representa um ambiente de máquinas paralelas) para o problema de planejamento da produção para fundições de pequeno porte com múltiplos fornos não simultâneos. São consideradas o mesmo objetivo e restrições como em de Araujo et al. (2008).

A implementação de uma biblioteca em linguagem C++ para o acesso aos pacotes de otimização comerciais CPLEX e Gurobi foi utilizada para a validação e comparação dos modelos. Os resultados obtidos com o novo modelo foram comparados com os resultados de Araujo et al. (2008) e se mostram promissores.

PALAVRAS-CHAVE: Fundição, Dimensionamento de Lotes, Planejamento da Produção.

Referências

Camargo, V. C. B.; Toledo, F. M. B.; Almada-Lobo, B. (2012). Three time-based scale formulations for the two-stage lot sizing and scheduling in process industries. *Journal of the Operational Research Society*, 63, 1613–1630.

de Araujo, S. A.; Arenales, M. N.; Clark, A. R. (2008). Lot sizing and furnace scheduling in small foundries. *Computers Operations Research.*, 35, 916–932.

PROBLEMA DE CORTE DE ESTOQUE: APLICAÇÃO A UMA INDÚSTRIA MADEIREIRA

Juliana Rodrigues Silva de Oliveira, Natália Bogaz Trivelato
julianarso@hotmail.com, bogaznatalia@gmail.com
Universidade Estadual Paulista - Campus São José do Rio Preto

Silvio Alexandre de Araujo, Socorro Rangel
saraujo@ibilce.unesp.br, socorro@ibilce.unesp.br
Universidade Estadual Paulista - Campus São José do Rio Preto

RESUMO

Neste trabalho investiga-se o problema de corte de estoque bidimensional (PCE-2D) (*e.g.* Arenales et al. (2004)) aplicado a uma indústria madeireira. Este setor industrial está dividido basicamente em dois tipos de empresas (ABIMCI (2014)). Um tipo, chamado madeireira fornecedora, é aquele que corta os troncos das árvores e depois transformam esses troncos em madeiramento serrado. Esse tipo de empresa tem como cliente outras madeireiras. O outro tipo de indústria, chamada simplesmente de madeireira, recebe as cargas com o madeiramento já serrado e produz itens de dimensões menores de acordo com a procura de cada cliente. O problema de uma indústria madeireira específica é discutido, e é apresentada uma simulação inicial do PCE-2D. Avaliamos a viabilidade da utilização do CorteBi a essa indústria. O CorteBi (Rangel et al. (2013)) é um software em fase de desenvolvimento, de fácil manipulação, que resolve problemas gerais de corte de estoque bidimensional. Os resultados obtidos com o CorteBi são comparados com o método manual utilizado na empresa para gerar padrões de corte.

PALAVRAS-CHAVE: Indústria Madeireira, Problema de corte de estoque bidimensional, Gerações de padrões de corte.

Referências

ABIMCI (2014). *ABIMCI - Normas ABNT*. <http://www.abimci.com.br/>.

Arenales, M.; Morabito, R.; Yanasse, H. (2004). *Mini Curso de Corte e Empacotamento*. Simpósio Brasileiro de Pesquisa Operacional, São João del Rei - Minas Gerais.

Rangel, S.; Barbosa, T.; Lemos, R. B.; Cavali, R.; Castilho, V.; Perin, C. (2013). *Manual do Sistema CorteBi*. Relatório Técnico, UNESP- SJRP.