



Autor(es): RAMÓN SOUZA SILVA RODRIGUES, VICTOR DE FREITAS ARRUDA, MATEUS FELLIPE ALVES LOPES, DIEGO LEAL MAIA, JOÃO BATISTA MENDES

Proposta e Investigação de Algoritmo Evolucionário para Problemas de Alocação de Recursos

Introdução

Este trabalho estuda o Problema de Alocação de Recursos entre máquinas e tarefas e propõe um algoritmo evolutivo para sua resolução baseado na metaheurística, ou heurística inteligente, *Variable Neighborhood Search* – VNS explorando algumas particularidades da representação por vetores e sinais que permitem a manipulação dos operadores de movimento.

Neste trabalho é apresentado um modelo matemático para o problema identificando um objetivo principal e as restrições mais comumente encontradas na literatura. O tema em questão tem bastante aplicação prática, seja nas indústrias como, por exemplo, o problema de planejamento da produção ou nas instituições de ensino como o problema de alocação de salas de aula.

Material e métodos

Foi realizada uma pesquisa bibliográfica acerca do problema de alocação de recursos, identificando os principais trabalhos encontrados na literatura especializada. Posteriormente, foi feita uma modelagem matemática do problema, onde foram identificados a função objetivo e as principais restrições encontradas. No passo seguinte, será desenvolvido um algoritmo evolutivo para resolução do problema modelado anteriormente.

Resultados e discussão

A. Referencial teórico

O Problema de Alocação de Recursos é um problema de otimização combinatorial que pertence à classe dos problemas *NP-Hard*. De forma simplificada o problema consiste em alocar máquinas (*machines*, recursos) e trabalhos (*jobs*, atividades) de maneira a otimizar o tempo de execução de todo o conjunto. Sendo que cada máquina pode processar no máximo, uma tarefa de cada vez e cada tarefa possui um tempo específico de execução para cada máquina. Este problema vem sendo estudado, principalmente por profissionais da Engenharia, Computação e Matemática e tem sido objeto de uma quantidade significativa de literatura na área de Pesquisa Operacional. Um estudo detalhado sobre o problema de alocação pode ser visto em Yamada e Nakano (1997a) e também em Soares (2011).

Propõe-se neste trabalho apresentar uma descrição do algoritmo VNS para o problema de alocação de recursos. Especificamente, será detalhada uma estrutura de dados para representação da solução implementada pelo algoritmo *Variable Neighborhood Search* - VNS para ser aplicado à resolução do Problema de Alocação. A estrutura proposta é descrita na seção seguinte e mostrada nas Fig. 1 e Fig. 2. O algoritmo em questão foi estudado por Yamada e Nakano (1997b) e Hansen e Mladenovic (2001) e sua eficiência está na principal característica de possuir ferramentas que procuram superar os resultados dos ótimos locais quando ainda estamos distantes de um ótimo global em problemas de otimização, expandindo gradativamente sua busca, sendo classificado na classe dos Algoritmos Genéticos. Seu pseudocódigo pode ser visto na Fig. 3.

Algoritmos Genéticos são largamente aplicados a problemas de otimização combinatorial como no Problema do Caixeiro Viajante (GUEDES, 2009), no Problema de Roteamento de Veículos (HEINEN; OSÓRIO, 2006) dentre outros e baseiam-se não somente na teoria Darwiniana da evolução natural de espécies como também na genética, sendo que a maior parte deles, e no caso do VNS, manipula, a cada iteração (geração), um conjunto de pontos (população), gerando a partir destes uma nova população, que, na média, aproxima-se cada vez mais da região onde é provável que esteja o ponto ótimo e é nesse contexto que utilizamos o VNS. (PARREIRAS, 2006, p. 23)

B. Modelo de representação

Dentre as diversas representações matemáticas que modelam os problemas de natureza combinatorial foi escolhida a representação por vetores que descrevem cada instante de funcionamento de cada *machine*. Esta representação permite a visualização das máquinas através de pulsos e a ausência de pulsos representa a máquina parada, nosso objetivo é aplicar o algoritmo VNS afim de reduzir esse tempo de espera de cada máquina e consequentemente de todo o conjunto.



Para tanto, o algoritmo trabalha primeiramente com o carregamento de um *benchmark* de Taillard (1993) onde os tempos de processamento dos *jobs* variam de 1 a 99 unidades de tempo e são gerados usando um gerador de números aleatórios para diferentes sementes. Em seguida uma população inicial aleatória é criada a partir da relação *jobs-machines* e há uma junção do *benchmark* com a população inicial, gerando uma matriz onde cada linha representa uma solução para uma determinada *machine* e é composta por tuplas T(J,D), onde J representa o *job* a ser executado e D a duração de tempo de execução deste *job* (em unidade de tempo). Com estes dados já é possível visualizar o escalonamento dos *jobs* através de pulsos representados por sinais, ver Fig. 1. Dessa forma representa-se a junção das matrizes e funcionamento do conjunto de máquinas, respectivamente, da seguinte maneira:

$$M_1 = [(1,3), (2,2), (3,2)] \rightarrow [1, 1, 1, 2, 2, 3, 3, \mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{0}]$$

$$M_2 = [(3,2), (1,3), (2,4)] \rightarrow [3, 3, 3, 1, 1, 1, \mathbf{0}, \mathbf{0}, 2, 2, 2, 2]$$

$$M_3 = [(2,3), (1,3), (3,1)] \rightarrow [\mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{0}, \mathbf{0}, 2, 2, 2, 1, 1, 1, 3]$$

No exemplo dado acima, a máquina M_1 executa o *job* J_1 por 3 unidades de tempo, J_2 por 2 unidades de tempo, J_3 por 2 unidades e os valores nulos representam os tempos de atraso (máquina em espera) por 4 unidades, analogamente temos a execução das máquinas M_2 e M_3 e ainda na Fig. 1 como uma representação por sinais dos pulsos que indicam a execução de cada *job*. Este modelo poderá ainda ser visualizado na forma binária onde o estado '1' representa a máquina parada enquanto o estado '0' significa que a máquina está em operação. A Fig. 2 ilustra a indicação dos pulsos que representam os tempos de atraso de cada máquina de forma binária, como a seguir:

$$M_1 = [0, 0, 0, 0, 0, 0, 0, \mathbf{1}, \mathbf{1}, \mathbf{1}, \mathbf{1}, \mathbf{1}]$$

$$M_2 = [0, 0, 0, 0, 0, 0, \mathbf{1}, \mathbf{1}, 0, 0, 0, 0]$$

$$M_3 = [\mathbf{1}, \mathbf{1}, \mathbf{1}, \mathbf{1}, \mathbf{1}, 0, 0, 0, 0, 0, 0, 0]$$

Dessa forma, o tamanho de qualquer um dos vetores representa o tempo de execução de todo o conjunto de máquinas e o tempo de atraso total é a soma dos '1'.

A partir de então será aplicado o método VNS para minimização dos tempos de atraso e conseqüentemente de todo o tempo de execução. Para esta representação, utiliza-se 2 principais operadores de movimento (realocação e troca) e são detalhados em Souza et al. (2002) de forma análoga para o problema de alocação de salas.

C. Resultados esperados

Uma bateria preliminar de testes realizada por (SOUZA et al., 2002) indicou que o algoritmo VNS utilizado separadamente de outros métodos não possui resultados satisfatórios, por isso passaremos a construir uma melhor solução inicial a utilizar uma solução completamente aleatória.

A construção de uma melhor solução inicial pode ser criada baseada em procedimentos construtivos parcialmente gulosos (SOUZA et al., 2002), esses métodos serão analisados e selecionados ao que melhor se adaptar à modelagem do problema proposto seguindo os diversos *benchmarkings* propostos por Taillard (1993).

Para o resultado final deste problema serão realizadas diversas execuções do algoritmo, cada qual com uma semente diferente de números aleatórios. Utilizaremos como critério de parada número suficiente de interações que seja capaz de equilibrar o tempo de execução com uma solução ótima.

Considerações

Este trabalho está contribuindo não somente para o estudo dos problemas de alocação, mas também para o desenvolvimento acadêmico de seus autores, por ter sido a porta de entrada para a iniciação científica.

O algoritmo VNS requer apenas a definição de alguns parâmetros e estruturas de vizinhança, ao contrário de outras metaheurísticas, tais como a Otimização por Enxame de Partículas (PSO) ou a Otimização da Colônia de Formigas (ACO), que requerem a calibragem de parâmetros mais complexos.

Como trabalho futuro pretende-se implementar a construção de soluções iniciais utilizando métodos diversos, aplicar o VNS utilizando as particularidades da representação por sinais, expandir e complementar este trabalho cruzando informações com outros algoritmos genéticos que utilizam operações como mutação e crossover.

Agradecimentos

Aos professores pelo conhecimento e atenção dispensados, em especial ao orientador deste trabalho que deu-nos a oportunidade de iniciar este projeto de Iniciação Científica Voluntária. Aos colegas que compõem esta equipe.



Referências bibliográficas

- GUEDES, A. C. B.; **Um algoritmo genético com infecção viral para o problema do caixeiro viajante**. Natal: UFRN, 2009. 24p. Disponível em <https://periodicos.ufrn.br/publica/article/view/125>. Acesso em: 08 nov. 2016.
- HANSEN, P.; MLADENOVIC, N.; **Variable neighborhood search: Principles and applications**. European Journal of Operational Research. v. 130, n. 3, p. 449-467, mai. 2001. Disponível em <http://www.sciencedirect.com/science/article/pii/S0377221700001004>. Acesso em: 08 nov. 2016.
- HEINEN, M. R.; OSORIO, F. S. **Algoritmos Genéticos Aplicados ao Problema de Roteamento de Veículos**. 2006. Hifen, Uruguiana, vol. 30, num. 58. 2006. Disponível em <http://revistaseletronicas.pucrs.br/fass/ojs/index.php/hifen/article/download/3781/2893>. Acesso em: 08 nov. 2016.
- PARREIRAS, R. O.; **Algoritmos Evolucionários e Técnicas de Tomada de Decisão em Análise Multicritério**. 2006. Tese (Doutorado em Engenharia Elétrica) - Escola de Engenharia - UFMG, Belo Horizonte, MG, Brasil, 2006. Disponível em <http://www.ppgee.ufmg.br/defesas/853D.PDF>. Acesso em: 08 nov. 2016.
- SOARES, H. C. A.; **Um estudo sobre o Problema de Alocação**. São Paulo: Unifesp, 2011. Disponível em <http://www.ft.unicamp.br/docentes/meira/publicacoes/2011henrique.pdf>. Acesso em: 08 nov. 2016.
- SOUZA, M. J. F. et al. **Método de Pesquisa em Vizinhaça Variável aplicado ao Problema de Alocação de Salas**. In: XXII ENCONTRO NACIONAL DE ENGENHARIA DE PRODUÇÃO, 2002, Curitiba. **Anais...** Curitiba: 2002. Disponível em <http://www.abepro.org.br/biblioteca/ENEGERP2002_TR68_0554.pdf>. Acesso em: 08 nov. 2016.
- TAILLARD, E. **Benchmarks for basic scheduling problems**. European Journal of Operational Research. v. 64, n. 2, p. :278-285, jan. 1993. Disponível em <http://mistic.heig-vd.ch/taillard/problemes.dir/ordonnancement.dir/ordonnancement.html>. Acesso em: 08 nov. 2016.
- YAMADA, T.; NAKANO, R.; Job-shop scheduling. Londres: IEE, 1997a. Separata de: ZALZALA, A.M.S.; FLEMING, P.J. **Genetic algorithms in engineering systems**: IEE: Londres, 1997. 134-160. Disponível em <http://www.keel.ntt.co.jp/as/members/yamada/galbk.pdf>. Acesso em: 08 nov. 2016.
- _____. **Variable neighborhood search**. Computers & Operations Research. vol. 24, ed. 11, p. 1097-1100, nov. 1997b. Disponível em <http://www.sciencedirect.com/science/article/pii/S0305054897000312>. Acesso em: 08 nov. 2016.

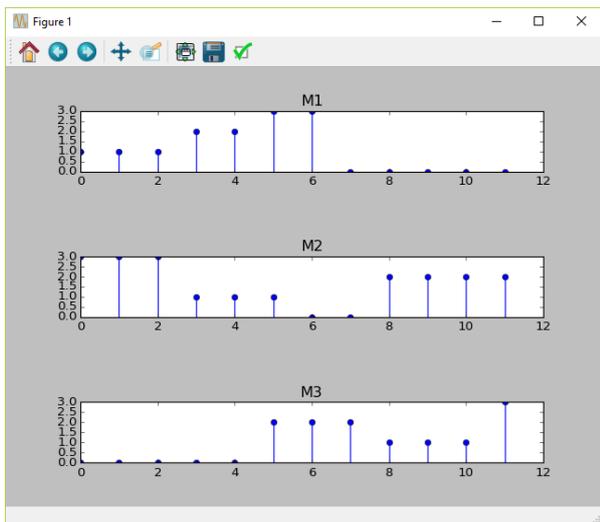


Figura 1 – Representação por sinais dos pulsos que indicam o funcionamento dos jobs

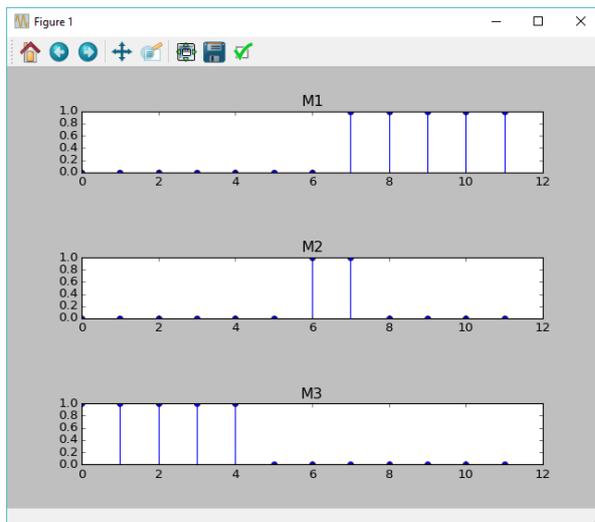


Figura 2 – Representação por sinais dos pulsos binários que indicam os tempos de atraso de cada máquina

```

Procedimento Variable Neighborhood Search
1. Seja  $s_0$  uma solução inicial e  $r$  o número de estruturas de vizinhança;
2.  $s \leftarrow s_0$ ; {Solução corrente}
3. Enquanto (Critério de parada não satisfeito) faça
4.    $k \leftarrow 1$ ; {Tipo de estrutura de vizinhança}
5.   Enquanto ( $k \leq r$ ) faça
6.     Gere um vizinho qualquer  $s' \in N^{(k)}(s)$ ;
7.      $s'' \leftarrow \text{Busca\_Local}(s')$ ;
8.     Se  $f(s'') < f(s)$ 
9.       Então  $s \leftarrow s''$ ;
10.       $k \leftarrow 1$ ;
11.     Senão  $k \leftarrow k + 1$ ;
12.   Fim-se;
13. Fim-enquanto;
14. Fim-enquanto;
15. Retorne  $s$ ;
Fim
  
```

Figura 3 – Pseudocódigo do algoritmo VNS