LI Simpósio Brasileiro de Pesquisa Operacional
Limeira–SP, 02 a 06 de setembro de 2019

Limitantes inferiores para o problema de minimização de pilhas abertas

Calvin R. Costa
calvin.costa@unifesp.br

Mariá C. V. Nascimento
mcv.nascimento@unifesp.br

Horacio H. Yanasse
horacio.yanasse@unifesp.br
Universidade Federal de São Paulo - UNIFESP
Av. Cesare Monsueto Giulio Lattes, 1201 - Eugênio de Melo,
São José dos Campos - SP, 12247-014

RESUMO

O problema de minimização de pilhas abertas (MOSP) consiste em determinar uma sequência de corte dos padrões de corte de itens de modo a minimizar o número de itens que devem ser produzidos simultaneamente, de modo a atender as demandas dos itens. Diversos métodos heurísticos para esse problema podem ser encontrados na literatura, bem como algumas modelagens matemáticas. Neste artigo, um método para se obter limitantes inferiores para o MOSP é introduzido. Para avaliar a qualidade dos limitantes encontrados, o método proposto foi comparado com uma estratégia simples de obtenção de limitantes inferiores, que avalia o menor grau mais um do grafo MOSP, e com um algoritmo exato da literatura, utilizando-se algumas instâncias *benchmark*. Os resultados do experimento indicaram que o método proposto obteve um melhor comportamento para instâncias esparsas, para as quais a obtenção de uma solução exata é computacionalmente mais onerosa.

PALAVRAS CHAVE. Grafos, Limitantes Inferiores, Detecção de Comunidades.

TAG – Teoria e Algoritmos em Grafos

ABSTRACT

The Minimization of Open Stacks Problem (MOSP) consists of determining a sequence of cutting patterns in order to minimize the number of items that must be produced simultaneously in order to meet their demands. Several heuristic methods for this problem can be found in the literature, as well as some mathematical models. This paper presents a method to obtain lower bounds to the MOSP. In order to evaluate the quality of the obtained bounds, the proposed method was compared with a simple strategy to obtain lower bounds, which evaluates the lowest degree plus one of the MOSP graph, and with an exact algorithm of the literature, using some benchmark instances. The results of the experiment indicated that the proposed method obtained a better behavior for sparse instances, for which the obtaining of an exact solution is computationally more costly.

KEYWORDS. Graphs. Lower Bounds. Detection of Communities.

TAG - Theory and Algorithms in Graphs

https://proceedings.science/p/106958?lang=pt-br

1. Introdução

O problema de minimização do número máximo de pilhas abertas, do inglês *Minimization of Open Stacks Problem* (MOSP) é um problema de otimização combinatória que foi inicialmente proposto por [Yuen, 1991]. Tal problema consiste em determinar uma sequência de corte dos padrões de corte de itens de modo a minimizar o número de itens sendo produzidos simultaneamente e atender a demanda dos itens. Cada um dos tipos de itens cortados forma uma pilha e uma pilha é dita aberta se ainda restam itens a serem cortados do mesmo tipo correspondente à pilha. Se não mais existem itens de um mesmo tipo a serem cortados, a pilha é dita fechada.

O MOSP é, portanto, um problema *min-máx*, o que, consequentemente, implica em uma maior chance de soluções degeneradas. Como consequência dessa característica, há uma grande dificuldade em provar a otimalidade de soluções geradas para o problema por algoritmos enumerativos. A literatura sobre métodos exatos para o MOSP não é extensa. Dentre os trabalhos existentes, destaca-se aqui o trabalho de [La Banda e Stuckey, 2007], que obtiveram o ótimo para a maioria das instâncias *benchmark* do *Constraint Modelling Challenge* [Smith e Gent, 2005]. Destaca-se também o algoritmo exato de [Chu e Stuckey, 2009] que, posteriormente, soluciona na otimalidade todas as instâncias de [Smith e Gent, 2005]. Por outro lado, um número considerável de heurísticas está presente na literatura como a de [Gonçalves et al., 2014].

Dentre os trabalhos que abordam este problema, vale destacar [Yanasse, 1997a] que propõe uma interpretação do problema em um grafo, conhecido por grafo MOSP, definido por $G = (V, E)$ não orientado e não ponderado, e seu conjunto de nós (ou vértices), V, representa os itens demandados, enquanto o conjunto de arestas E representa a relação entre os itens que é se existe ao menos um padrão de corte com ambos os itens.

Neste artigo é apresentada uma breve revisão da literatura acerca do MOSP na Seção 2 enquanto que propõe-se o desenvolvimento de limitantes inferiores para o MOSP por meio do estudo do modelo proposto em [Yanasse, 1997a] na Seção 3. Com esta proposta pretende-se, por meio do agrupamento do grafo MOSP, identificar simetrias para avaliações menos custosas de limitantes inferiores para o problema. Na Seção 4 são apresentados alguns resultados computacionais nos quais, para as instâncias de baixa densidade *benchmark*, que são de difícil solução, foi possível identificar limitantes de boa qualidade para este problema. Por fim, as conclusões sobre os experimentos e as pretensões para pesquisas futuras são apresentados na Seção 5.

2. *Minimization of Open Stacks Problem* - MOSP

Como já dito, Yuen [1991] introduziu o MOSP e para resolvê-lo propôs duas heurísticas para sequenciamento de padrões, que o autor diferencia por Heurística 1 e Heurística 2, cujo funcionamento divide-se em duas fases.

Na primeira, comum às duas heurísticas, um item é escolhido do conjunto de itens com pilhas abertas, usando o seguinte critério: a menor quantidade de padrões não sequenciados que contém aquele tipo de item. Em caso de empate, seleciona-se o tipo de item cuja pilha falta menos itens para fechar; caso o empate persista, é selecionado o tipo de item disponível que foi primeiramente avaliado.

Na Heurística 1, primeiro seleciona-se um conjunto de padrões que repetem mais vezes dentre os itens cujas pilhas não foram fechadas. No caso da seleção de mais de um padrão, seleciona-se um conjunto de padrões com menos itens. Se este conjunto ainda possuir mais de um padrão, o de menor índice é selecionado. A segunda fase da Heurística 2 diferencia-se da Heurística 1 pela diferença na ordem dos critérios de seleção dos padrões.

Posteriormente, Yuen [1995] apresenta quatro novas heurísticas denominadas por Heurísticas 3, 4, 5 e 6. Novas medidas são propostas, sendo a de compatibilidade o principal indicativo de como

estas heurísticas devem se comportar. Como na Equação 1, para um padrão j, a medida de compatibilidade (MC) é obtida por meio da diferença entre a quantidade de itens que j possui em comum com as pilhas abertas (CP) e a quantidade de itens que j não possui em comum com as pilhas abertas (NCP).

$$MC_j = CP_j - NCP_j \tag{1}$$

Dessas quatro heurísticas, a Heurística 3 merece maior destaque pela obtenção de boas soluções em um tempo computacional menor que as demais. A fase de seleção de padrões da Heurística 3 escolhe o padrão com maior MC, maior CP, menor NCP, menor índice.

Em seu trabalho, Becceneri [1999] apresenta duas propostas de pré-processamento das instâncias MOSP. A primeira considera a dominância entre padrões enquanto a segunda considera um algoritmo que colapsa as arestas do grafo MOSP. Por meio desse colapso de arestas, Becceneri [1999] visa obter um limitante inferior para o número de pilhas abertas. Para o grafo $G^{'} = (V^{'}, E^{'})$ obtido por meio dos colapsos, o limitante é obtido pela Expressão 2 apresentada por [Yanasse et al., 1997].

$$min(d(v^{'})) + 1 \tag{2}$$

Em [Yanasse e Pinto, 2003], os autores propõem uma modelagem para o MOSP baseada no método de enumeração de Yanasse [1997b]. Uma adaptação desse modelo é apresentada a seguir, sendo os parâmetros e variáveis do problema:

n	Número de itens (parâmetro)
e	Vetor n-dimensional de 1's (parâmetro)
R_i	Vetor n-dimensional que assume valor 1 em sua j-ésima posição se os itens i e j aparecem juntos em ao menos um padrão e 0, caso contrário (parâmetro)
C	Número máximo de pilhas abertas (variável)
W_i	Vetor coluna n-dimensional que assume valor 1 na posição i se a pilha do item i estiver aberta ou fechada no instante t.
Y_i	Vetor coluna n-dimensional que assume valor 1 em sua posição i se a pilha referente ao item i é a t-ésima a ser fechada e zero caso contrário.

$$\min C \tag{3}$$

$$Sujeito\ a:$$

$$e^T.W_t \leq C + t - 1, \quad t = 1, ..., n \tag{4}$$

$$\sum_{i=1}^{n} \sum_{k=1}^{t} Y_{ik} R_i \leq t.W_t, \quad t = 1, ..., n \tag{5}$$

$$\sum_{t=1}^{n} Y_{it} = 1, \quad i = 1, ..., n \tag{6}$$

$$\sum_{i=1}^{n} Y_{it} = 1, \quad t = 1, ..., n \tag{7}$$

$$W_i, Y_i, R_i \in \{0,1\}^n \quad i = 1, ..., n,\ t = 1, ..., n \tag{8}$$

A Função Objetivo (3) visa minimizar o número máximo de pilhas abertas, dentre todos os instantes t, conforme as Restrições (4). Tais restrições asseguram que C seja maior ou igual ao

https://proceedings.science/p/106958?lang=pt-br

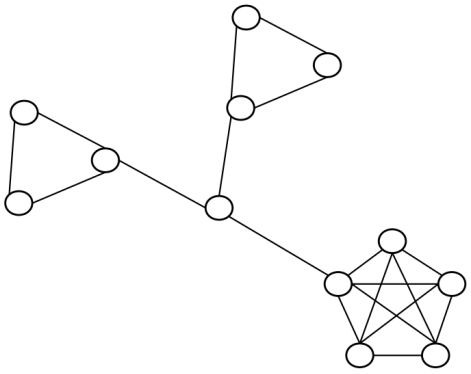
maior número de pilhas abertas no instante t para todos os possíveis t. As Restrições (5) definem que, se a pilha de um dado item i for fechada ou aberta, todas as pilhas de itens que aparecem em padrões aos quais i pertença também devem ter sido abertas ou fechadas. As restrições (5) e (6) garantem que, para cada instante, uma pilha será fechada e que as pilhas de todos os tipos de itens serão fechadas. Por fim, as Restrições (8) definem o domínio das variáveis do problema.

As Restrições (5) são ligeiramente diferentes das propostas em [Yanasse e Pinto, 2003], pois ao invés de t ser o fator multiplicativo de W_t, os autores utilizam uma constante maior ou igual a n. Entretanto, é interessante utilizar um limitante superior de melhor qualidade uma vez que torna a restrição mais restritiva na relaxação linear para o problema.

Note que este modelo procura definir a sequência de fechamento das pilhas de itens para calcular a função objetivo, diferentemente de processar a sequência entre os padrões, realiza-se o sequenciamento dos itens. O número de soluções alternativas possíveis é menor do que se sequenciar as arestas do grafo MOSP pois $m > n$.

Mesmo com essa eliminação de soluções equivalentes, pelo grafo MOSP gerado para algumas instâncias observa-se degenerações, como e o caso apresentado na Figura 1.

Figura 1: Grafo MOSP



Na Figura 1, observa-se que a função objetivo de uma solução ótima seria 5. A degeneração pode ser observada pois, independente, da ordem escolhida dos nós dentro das cliques maximais, desde que a sequência não intercale mais de uma vez nos que não estejam nas cliques na sequência, tais soluções são ótimas. Pode-se calcular possíveis $5!3!3!$ diferentes sequências ótimas considerando apenas soluções que iniciem com nos do $K5$. Além disso, essas sequências invertidas também geram o mesmo valor de solução, resultando em $2 \times 5!3!3!$ possíveis soluções ótimas.

La Banda e Stuckey [2007] apresentam um algoritmo baseado em programação dinâmica. Sejam P o conjunto de padrões existentes; S uma sequência qualquer de padrões $p \in P$; $c(p)$ o conjunto de itens obtidos pelo corte do padrão p; $c(S) = \bigcup_{p \in S} c(p)$; C o número de itens existentes; e $stack[S]$ o número mínimo necessário de pilhas para S. Este algoritmo pode ser representado pelo modelo na Equação (10).

$$a(p, S) = c(p) \bigcup c(S - \{p\}) \bigcap c(P - S - \{p\}), \tag{9}$$

$$stacks(S) = min_{p \in S} max\{a(p, S - \{p\}), stacks(S - \{p\})\} \tag{10}$$

Este algoritmo recebe como parâmetros S, L, U sendo L o limitante inferior e U o superior para o máximo de pilhas abertas. Em [La Banda e Stuckey, 2007] são citadas algumas maneiras

para se obter estes limitantes e, dentre elas, são citados os trabalhos de [Yuen, 1995; Becceneri, 2004].

Para reduzir o número de combinações a serem analisadas, $stacks(S,\ L,\ U)$ realiza respectivamente três verificações: se $S = \{\}$; se S já foi avaliada; e se $\exists p \in S$ tal que $c(p) \subseteq c(P - S) \bigcap c(S)$. Caso $S = \{\}$ seja verdade, $stacks$ retorna 0; se a segunda verificação for verdadeira, então retorna-se $stack[S]$; caso a terceira o seja, $stacks$ retorna o valor obtido pela chamada $stacks(S - \{p\}, L, U)$.

Para as instâncias utilizadas em [Smith e Gent, 2005], o algoritmo de La Banda e Stuckey [2007] obteve soluções ótimas para quase todos os casos[1].

Na Tabela 1 são listados alguns dos trabalhos presentes na literatura do MOSP. Nela o "✓" indicará o tipo de proposta realizada enquanto que na coluna **Descrição** apresentam-se breves explicações sobre o trabalho realizado.

Tabela 1: Trabalhos Relacionados

Referência	**Proposta Exata**	**Proposta Heurística**	**Descrição**
[Yuen, 1991]		✓	Propostas duas heurísticas construtivas que diferem pela ordem como se realiza a montagem da solução.
[Yuen, 1995]		✓	Quatro heurísticas construtivas novas relacionadas às propostas em [Yuen, 1991].
[Yuen e Richardson, 1995]	✓	✓	Uma heurística simples para se determinar um *Lower Bound* e um algoritmo exato do tipo *branch-and-bound*.
[Yanasse, 1997b]	✓		Proposto um modelo com o qual se aplica uma estratégia de *branch-and-bound*.
[Faggioli e Bentivoglio, 1998]	✓	✓	Heurística construtiva que remete a [Yuen, 1991, 1995]. Algoritmo exato espelhando-se em [Yuen e Richardson, 1995].
[Becceneri, 1999]		✓	Propostos dois métodos de pré-processamento e uma heurística construtiva com base neles.
[Fink e Voß, 1999]		✓	Propostas adaptações para heurísticas do tipo *Simulated Annealing* e Busca Tabu.
[Ashikaga, 2001]		✓	Propostas duas heurísticas construtivas. Uma por circuitos hamiltonianos, outra por cliques maximais.

[1] O algoritmo excedeu o tempo estipulado ao resolver três instâncias.

[Linhares, 2002]		✓	Apresentada uma meta-heurística do tipo *Simulated Annealing* e uma heurística construtiva considerando uma métrica de distâncias.
[Becceneri, 2004]	✓		Proposta de um método exato tipo *branch-and-bound*.
[Pinto, 2004]		✓	Proposta uma heurística construtiva para o MOSP definido em um grafo.
[La Banda e Stuckey, 2007]	✓		Algoritmo exato utilizando programação dinâmica.
[Ashikaga e Soma, 2009]		✓	Heurística construtiva utilizando cliques máximos.
[De Carvalho e Soma, 2011]		✓	São apresentadas heurísticas desenvolvidas a partir dos algoritmos propostos em [Ashikaga e Soma, 2009] [La Banda e Stuckey, 2007].
[De Carvalho e Soma, 2014]		✓	Proposta uma heurística construtiva com busca em largura no grafo MOSP.

3. Método Proposto

Existem relativamente poucas estratégias que exploram o grafo MOSP para a obtenção de limitantes inferiores, como é o caso de Yanasse et al. [1997], que derivam o limitante calculando o menor grau de G mais 1, Expressão (2). Esse método é referenciado aqui por MD.

O método proposto neste artigo está representado no pseudocódigo no Algoritmo 1 e consiste na decomposição do grafo MOSP por meio de uma estratégia de agrupamento em grafos e posterior resolução dos subgrafos obtidos, subproblemas da instância. A estratégia proposta é referenciada por MP.

Neste algoritmo, LI armazena o valor do limitante inferior, SP , o conjunto de soluções parciais obtidas, e ng indica a quantidade de comunidades obtidas pelo algoritmo de Santos et al. [2016]. Cada comunidade é referenciada por um inteiro no conjunto $\{1 \ldots, ng\}$. A entrada do algoritmo heu_SAN, é o grafo MOSP, G_{MOSP} , sua saída, além de ng, é uma matriz Mg de dimensão $n \times ng$ cujo o elemento na linha i e coluna j, se 1, indica que o item i encontra-se na comunidade j e, se 0, que o item i não se encontra na comunidade j. Esta matriz servirá para a identificação dos grupos e é passada de parâmetro para a resolução dos subproblemas. Dessa forma, detectadas as comunidades, resolve-se cada um dos subproblemas referentes a elas (os subgrafos MOSP induzidos pelos nós de cada comunidade). Estes subproblemas são resolvidos pelo *CPLEX* v.12.6.3 usando-se o modelo (3)-(8).

Por serem subgrafos do grafo MOSP original, a solução de cada subproblema é um limitante inferior para a solução do GMOSP. Assim, a cada iteração, atualiza-se o valor de LI para ele armazenar o melhor limitante inferior para o problema original. Para potencial uso em heurísticas construtivas, toda solução parcial sol é adicionada a SP, de acordo com a linha 9 do algoritmo. Ressalta-se que LI só poderá ser considerado limitante inferior do G_{MOSP} caso seu respectivo

https://proceedings.science/p/106958?lang=pt-br

Algoritmo 1: MP

Entrada: grafo G_{MOSP}
Resultado: LI, SP

```
1  LI = 0;
2  SP = {};
3  {ng, Mg} = heu_SAN(G_MOSP);
4  enquanto i = 1, ..., ng faça
5  |  {L, sol} ← resolve(G_MOSP, i, Mg);
6  |  se L > LI então
7  |  |  LI = L;
8  |  fim
9  |  SP ← add(sol, SP);
10 fim
```

subgrafo seja resolvido na otimalidade. Em testes computacionais tal otimalidade não é atingida sempre devido à dificuldade para resolver alguns dos subproblemas, mesmo de tamanhos menores.

4. Testes Computacionais

Para os experimentos, as instâncias *benchmark* utilizadas foram retiradas do repositório do *Modelling Challenge* [Smith e Gent, 2005]. A solução exata dos subproblemas foi obtida utilizando-se o pacote de otimização *CPLEX* v.12.6.3. As Tabelas 2 e 3 além de reportarem os resultados obtidos, sumarizam as principais características dessas instâncias (o número de itens é o primeiro decimal após '_' que define o nome da classe de instâncias. Na Tabela 2 as instâncias foram agrupadas por classes de acordo com suas respectivas quantidades de itens e padrões sendo que cada classe e especificada na coluna "Classe". Na Tabela 2 'QI' especifica o número de instâncias em cada classe e na coluna 'DM' indicam-se as médias das densidades[2] das instâncias presentes em cada uma das classes. Já as colunas '$\overline{\texttt{MP}}$' indicam as médias por classe dos limitantes inferiores atingidos pela estratégia proposta, MP; as colunas '$\overline{\texttt{MD}}$' indicam as médias por classe dos limitantes inferiores atingidos pelo método MD; e as colunas '$\overline{\texttt{ML}}$' apresentam as médias por classe dos valores das soluções ótimas obtidas por [La Banda e Stuckey, 2007], estratégia esta referida aqui por ML.

Pela Tabela 2, observa-se que a média obtida para os limitantes inferiores obtidos por MP aproxima-se poucas vezes da média das soluções ótimas, caracterizando um limitante fraco para esse tipo de instâncias. Em comparação com MP, MD foi melhor em 98% das classes presentes na Tabela 2 mostrando-se superior ao MP para as instâncias nestas classes.

Em um segundo experimento, consideram-se as instâncias mais desafiadoras de [Smith e Gent, 2005], cujas principais características estão sumarizadas na Tabela 3, além dos resultados atingidos. Na coluna 'Inst' encontram-se os nomes das respectivas instâncias enquanto na coluna '$|V|$' especifica-se a quantidade de itens em cada uma delas. Por fim, na coluna 'Dens.' indica-se a densidade de cada uma dessas instâncias. A coluna 'MP' indica os valores atingidos pelo método proposto; a coluna 'MD', os limitantes inferiores baseado em graus; e, por fim, a coluna 'SO' apresenta as soluções ótimas obtidas por [Chu e Stuckey, 2009].

Na Tabela 3 observa-se que apesar de competitivo, o MP não superou os resultados do MD para a grande maioria das instâncias. Entretanto, vale destacar que o MP superou MD para as instâncias desse *benchmark* conhecidamente mais difíceis: SP2, SP3 e SP4.

[2] A densidade de uma instância é dada por $\frac{|E|}{|V|\times(|V|-1)}$.

https://proceedings.science/p/106958?lang=pt-br

Tabela 2: Tabela com os dados sobre as classes de instâncias apresentadas em [Smith e Gent, 2005] e resultados de MP, MD e ML.

Classe	QI	DM	$\overline{\text{MP}}$	$\overline{\text{MD}}$	$\overline{\text{ML}}$	Classe	QI	DM	$\overline{\text{MP}}$	$\overline{\text{MD}}$	$\overline{\text{ML}}$
P_10_10	550	0,41	4,78	6,75	8,03	wbop_10_10	40	0,33	4,2	6,35	6,75
P_10_20	550	0,45	4,93	8,22	8,92	wbop_10_20	40	0,4	4,35	7,7	8,07
P_15_15	550	0,44	7,87	11,15	12,87	wbop_10_30	40	0,43	4,85	8,27	8,55
P_15_30	220	0,47	8,46	13,2	14,02	wbop_15_15	60	0,34	6,15	9,5	10,37
P_20_10	550	0,41	9,79	12,18	15,88	wbop_15_30	60	0,4	7,18	11,28	12,15
P_20_20	220	0,46	10,71	15,94	17,97	wbop_20_10	40	0,35	8,8	13,07	14,28
P_30_10	550	0,41	15,11	17,73	23,95	wbop_20_20	90	0,36	8,79	13,55	14,87
P_30_15	220	0,44	16,06	21,16	25,97	wbop_30_10	40	0,36	14,03	20,17	22,48
P_30_30	110	0,48	17,31	25,92	28,32	wbop_30_15	60	0,35	13,72	19,55	22,38
P_40_20	110	0,46	23,29	30,48	36,38	wbop_30_30	140	0,38	14,53	21,56	23,84
Shaw_20_20	25	0,33	8,04	7,64	13,68	wbp_10_10	40	0,36	4,6	6,27	7,28
wbo_10_10	40	0,3	3,75	4,57	5,92	wbp_10_20	70	0,44	4,6	8,1	8,71
wbo_10_20	40	0,37	4,28	6,35	7,35	wbp_10_30	100	0,47	4,91	8,94	9,31
wbo_10_30	40	0,41	4,68	7,35	8,2	wbp_15_15	60	0,36	5,14	9,5	11,05
wbo_15_15	60	0,31	5,95	6,57	9,35	wbp_15_30	120	0,44	6,93	12	13,09
wbo_15_30	60	0,38	7,12	9,63	11,58	wbp_20_10	40	0,37	7,94	12,62	15,12
wbo_20_10	70	0,33	7,97	8,27	12,9	wbp_20_20	90	0,38	9,5	13,54	15,41
wbo_20_20	90	0,34	8,34	10,29	13,69	wbp_30_10	40	0,37	9,27	19,4	23,18
wbo_30_10	100	0,34	12,73	11,74	20,05	wbp_30_15	60	0,37	14,78	19,06	22,98
wbo_30_15	120	0,35	12,22	14,08	20,96	wbp_30_30	140	0,39	14,52	21,44	24,46
wbo_30_30	140	0,37	13,89	17,3	22,56						

Tabela 3: Principais informações relativas às instâncias de mais difícil solução dentre as presentes em [Smith e Gent, 2005] e limitantes inferiores obtidos por MP e MD e as soluções ótimas, SO.

Inst	$\|V\|$	Dens	MP	MD	SO	Inst	$\|V\|$	Dens	MP	MD	SO
Miller	20	0,53	7	11	13	NWRS3	15	0,5	7	5	7
GP1	50	0,98	31	38	45	NWRS4	15	0,6	7	7	7
GP2	50	0,94	28	33	40	NWRS5	20	0,74	7	10	12
GP3	50	0,95	30	40	40	NWRS6	20	0,75	10	11	12
GP4	50	0,82	27	25	30	NWRS7	25	0,45	10	6	10
GP5	100	0,99	62	88	95	NWRS8	25	0,7	12	12	16
GP6	100	0,93	60	75	75	SP1	25	0,26	5	2	9
GP7	100	0,93	55	70	75	SP2	50	0,21	9	2	19
GP8	100	0,83	43	53	60	SP3	75	0,19	18	5	34
NWRS1	10	0,38	3	3	3	SP4	100	0,21	33	4	53
NWRS2	10	0,49	3	3	4						

5. Conclusões e Pesquisas Futuras

No problema de minimização de pilhas abertas pretende-se reduzir o número de pilhas de itens trabalhadas simultaneamente para o atendimento das demandas dos itens que são empilhados

por tipos. Ligado ao escopo do planejamento operacional de uma empresa, estudos sobre problemas semelhantes a ele como o problema de minimização de troca de ferramentas e o *linear gate assignment* podem ser beneficiados com novos métodos para resolvê-lo.

Neste artigo representam-se as instâncias MOSP como grafos. Com base nisto, realizou-se um pré-processamento nas instâncias obtendo delas subproblemas e resolvendo-os com uso de uma adaptação do modelo proposto por [Yanasse e Pinto, 2003]. O objetivo desta abordagem resume-se a reduzir o problema a subproblemas e, deles obter limitantes inferiores, além de fornecer soluções parciais ao problema para posterior uso.

Além da existência de degenerações, de acordo com observações presentes em [Chu e Stuckey, 2009], instâncias mais esparsas apresentam maior grau de de dificuldade para resolução. A obtenção de bons limitantes inferiores é essencial para facilitar a prova da otimalidade das soluções assim como o uso de soluções parciais pode facilitar a construção de soluções de boa qualidade, mesmo para instâncias com as características previamente citadas.

O método proposto depende de provar a otimalidade da solução obtida para o subproblema para que esta possa ser usada como limitante inferior, o que lhe é um ponto negativo dependendo da instância ao qual for utilizado. Por exemplo, instâncias muito densas nas quais a fase de detecção de comunidades tende a identificar apenas uma comunidade de modo que na segunda fase trabalha-se com o problema original em vez de um subproblema. Entretanto, o método proposto obteve melhores resultados para instâncias de baixa densidade como as instâncias $SP2$ a $SP4$ que são as mais esparsas dentre as presentes em [Smith e Gent, 2005].

Para trabalhos futuros, pretende-se utilizar as soluções dos subproblemas gerados pelo `MP` para refinar os resultados até então encontrados na literatura.

Agradecimentos

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001, do Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) e da FAPESP (Proc.: 16/01860-1) a quem os autores agradecem.

Referências

Ashikaga, F. M. e Soma, N. Y. (2009). A heuristic for the minimization of open stacks problem. *Pesquisa Operacional*, 29(2):439–450.

Ashikaga, F. (2001). *Um método frugal para o problema de minimização de pilhas abertas. 2001*. PhD thesis, Dissertação (Mestrado em Engenharia Eletrônica e Computação)–Instituto Tecnológico de Aeronáutica, São José dos Campos, 2001.

Becceneri, H. H. N. Y., José Carlos (2004). A method for solving the minimization of the maximum number of open stacks problem within a cutting process. *Computers & Operations Research*, 31 (14):2315–2332.

Becceneri, J. (1999). O problema de sequenciamento de padrões para minimização do número máximo de pilhas abertas em ambientes de corte industriais. *São José dos Campos. Tese (Doutorado em Ciências no Curso de Engenharia Eletrônica e Computação na Área de Informática)–ITA, Centro Tecnológico Aeroespacial*.

Chu, G. e Stuckey, P. J. (2009). Minimizing the maximum number of open stacks by customer search. In Gent, I. P., editor, *Principles and Practice of Constraint Programming - CP 2009*, p. 242–257, Berlin, Heidelberg. Springer Berlin Heidelberg. ISBN 978-3-642-04244-7.

De Carvalho, M. A. M. e Soma, N. Y. (2011). Métodos simplificados para o problema de minimização de pilhas abertas. *Gestão & Produção*, 18(2).

De Carvalho, M. A. M. e Soma, N. Y. (2014). A breadth-first search applied to the minimization of the open stacks. *Journal of the Operational Research Society*, 66(6):936–946.

Faggioli, E. e Bentivoglio, C. A. (1998). Heuristic and exact methods for the cutting sequencing problem. *European Journal of Operational Research*, 110(3):564–575.

Fink, A. e Voß, S. (1999). Applications of modern heuristic search methods to pattern sequencing problems. *Computers & Operations Research*, 26(1):17–34.

Gonçalves, J. F., Resende, M. G., e Costa, M. D. (2014). A biased random-key genetic algorithm for the minimization of open stacks problem. *International Transactions in Operational Research*.

La Banda, M. G. e Stuckey, P. J. (2007). Dynamic programming to minimize the maximum number of open stacks. *INFORMS Journal on Computing*, 19(4):607–617.

Linhares, A. (2002). *Industrial pattern sequencing problems: some complexity results and new local search models*. PhD thesis, Doctoral Thesis, INPE, S. José dos Campos, Brazil.

Pinto, M. J. (2004). Algumas contribuições à resolução do problema de corte integrado ao problema de seqüenciamento dos padrões. *Monografia (Doutorado em Computação Aplicada)–Instituto Nacional de Pesquisas Espaciais, São José dos Campos.[Links]*.

Santos, C. P., Carvalho, D. M., e Nascimento, M. C. (2016). A consensus graph clustering algorithm for directed networks. *Expert Systems with Applications*, 54:121–135.

Smith, B. e Gent, I. (2005). Constraint modelling challenge 2005. In *IJCAI 2005 Fifth Workshop on Modelling and Solving Problems with Constraints*, p. 1–8.

Yanasse, H. H., Becceneri, J. C., e Soma, N. Y. (1997). Lower bounds for the problem of sequencing cutting patterns. In *II Oficina de Problemas de Corte & Empacotamento, 1997*. Anais do II Oficina de Problemas de Corte & Empacotamento.

Yanasse, H. (1997a). A transformation for solving a pattern sequencing problem in the wood cut industry. *Pesquisa Operacional*, 17(1):57–70.

Yanasse, H. H. (1997b). On a pattern sequencing problem to minimize the maximum number of open stacks. *European Journal of Operational Research*, 100(3):454–463.

Yanasse, H. H. e Pinto, M. J. (2003). Uma nova formulaçao para um problema de sequenciamento de padroes em ambientes de corte. *XXXV SBPO-Simpósio Brasileiro de Pesquisa Operacional, Natal, RN*, p. 1516–1524.

Yuen, B. J. (1991). Heuristics for sequencing cutting patterns. *European Journal of Operational Research*, 55(2):183–190.

https://proceedings.science/p/106958?lang=pt-br

Yuen, B. J. (1995). Improved heuristics for sequencing cutting patterns. *European Journal of Operational Research*, 87(1):57–64.

Yuen, B. J. e Richardson, K. V. (1995). Establishing the optimality of sequencing heuristics for cutting stock problems. *European Journal of Operational Research*, 84(3):590–598.