

MODELO MATEMÁTICO DE PLANEJAMENTO DA PRODUÇÃO EM INDÚSTRIA DE EMBALAGENS DE VIDRO¹

Flaviana M. S. Amorim^{a*}, Claudio F. M. Toledo^a,
Márcio S. Arantes^b, Magna P. S. Ferreira^a

^a Instituto de Ciências Matemáticas e de Computação
Universidade de São Paulo - USP, São Carlos-SP, Brasil

^b Instituto SENAI de Inovação em Sistemas Embarcados, Florianópolis-SC, Brasil

Recebido 04/09/2018, aceito 10/12/2018

RESUMO

Este trabalho utiliza técnica de programação matemática para um problema de planejamento da produção em Indústria de Embalagens de Vidro (IEV), aplicando um método exato e dois algoritmos genéticos. A principal decisão é definir qual o conjunto ideal de configurações de máquinas a ser determinado junto a um forno construído que representa o planejamento estratégico para atender as previsões de demanda para os recipientes de vidro, dentro de um horizonte de tempo. O problema resolvido considera o cenário onde um novo forno e as máquinas relacionadas devem ser adicionados a uma planta industrial já existente. Uma formulação matemática é introduzida para definir objetivos e restrições para tal problema. Conjuntos de instâncias são gerados com base em dados fornecidos por IEVs localizadas em Portugal e no Brasil. Os resultados obtidos indicam que o modelo e os algoritmos genéticos propostos são capazes de fornecer soluções de boa qualidade.

Palavras-chave: Indústria de embalagem de vidro, Modelagem matemática, Planejamento de produção.

ABSTRACT

This work uses a mathematical programming technique for a problem of production planning in the Glass Container Industry (GCI) by applying one exact method and two genetics algorithms. The main decision is to define the ideal set of machine configurations to be determined next to a built furnace that represents strategic planning to meet demand forecasts for glass containers within a time horizon. The problem solved considers the scenario where a new furnace and related machines are to be added to an existing industrial plant. A mathematical formulation is introduced to define objectives and constraints for such a problem. Instance sets are generated based on data provided by GCIs located in Portugal and Brazil. The obtained results indicate that the model and the proposed genetics algorithms can provide solutions of good quality.

Keywords: Glass container industry, Mathematical modeling, Production planning.

* Autor para correspondência. E-mail: joflaviana@yahoo.com.br
DOI: 10.4322/PODes.2018.010

¹Todos os autores assumem a responsabilidade pelo conteúdo do artigo.

1. Introdução

O processo de produção em uma Indústria de Embalagens de Vidro é composto por duas etapas principais. Na primeira etapa, os componentes que constituem o vidro como sílica, barrilha, calcário e vidros recicláveis (cacos) são derretidos no forno. Na segunda etapa, os produtos finais (recipientes) são produzidos por uma máquina de moldagem. Uma IEV pode apresentar um ou vários fornos com diferentes configurações de máquinas ligadas a eles. Novos fornos podem ser adicionados à planta industrial para aumentar a atual capacidade produtiva ou para substituir fornos cujo prazo de uso já foi atingido. O problema estudado neste trabalho considera o processo de inclusão de um novo forno em uma IEV, onde as configurações das máquinas ligadas ao novo forno devem ser otimizadas e os custos relacionados devem ser minimizados. Esse problema será referenciado por Problema na Indústria de Embalagens de Vidro - Novo Forno (PIEV-NF).

A principal contribuição deste artigo é a proposta de um modelo matemático para o PIEV-NF, seguida pela aplicação de um método exato e dois algoritmos genéticos híbridos para resolver o PIEV-NF. O modelo foi implementado e resolvido no *solver* CPLEX para solucionar instâncias do PIEV-NF de forma ótima, ou obter limites inferiores e superiores dentro de um tempo computacional razoável. Um Algoritmo Genético Multi-Populacional (AGMP) com estrutura em grade e o mesmo AGPM combinado a uma heurística de Filtro (AGPM+F) são aplicados ao problema. A heurística proposta faz uma pré-avaliação dos indivíduos a fim de detectar infactibilidades, a ideia é evitar que indivíduos infactíveis sejam avaliados pelo modelo. Para avaliar a capacidade desses métodos em encontrar soluções de boa qualidade, as soluções são comparadas com resultados ótimos conhecidos. Um conjunto de instâncias foi gerado a partir de dados fornecidos por uma IEV localizada em Portugal e outra localizada no Brasil.

Este artigo está organizado da seguinte forma: a Seção 2 revisa trabalhos relacionados. Detalhes sobre o problema estudado são apresentados na Seção 3 e a Seção 4 introduz o modelo matemático. Os métodos são descritos na Seção 5, enquanto os resultados computacionais são reportados na Seção 6 e as conclusões seguem na Seção 7.

2. Trabalhos Relacionados

O PIEV-NF estudado neste artigo inclui decisões de dimensionamento de lotes. Existem várias revisões na literatura sobre o problema de dimensionamento de lotes e planejamento da produção, como em Drexl e Kimms (1997), Karimi et al. (2003) e Copil et al. (2017). Uma revisão sobre modelos para problemas de dimensionamento de lotes é apresentada por Jans (2009), enquanto que uma revisão sobre meta-heurísticas aplicadas a esta classe de problemas foi realizada por Jans e Degraeve (2007).

Almada-Lobo et al. (2010) estudaram o planejamento da produção em curto prazo de uma IEV, onde o objetivo é satisfazer as ordens diárias de produção e atender a demanda sem atrasos (*backlogging*). Uma formulação matemática é proposta e uma heurística baseada em decomposição lagrangiana é definida para resolver instâncias de tamanhos pequeno e médio. Uma nova formulação matemática e um algoritmo genético multi-populacional híbrido são propostos por Toledo et al. (2013), para os mesmos cenários produtivos de curto prazo. Os autores consideram novas restrições na formulação do problema e o método híbrido aplicado foi capaz de solucionar instâncias classificadas como sendo de pequeno, médio e grande porte.

O problema da produção de garrafas de bebidas a longo prazo foi apresentado por Guimarães et al. (2012). Este problema consiste em atribuir e programar lotes de produção em um ambiente com diversas plantas, em que cada planta tem um conjunto de linhas de enchimento para embalar as bebidas. As plantas também consideram a transferência de produtos finais entre elas. O planejamento da produção a longo prazo na IEV é estudado em Toledo et al. (2016), onde um planejamento da produção e distribuição de recipientes de vidro com múltiplas instalações devem ser definidos. O problema considera as tomadas de decisões relacionadas à transferência de produtos entre plantas, além de dimensionar e programar decisões para cada forno. Um modelo de pro-

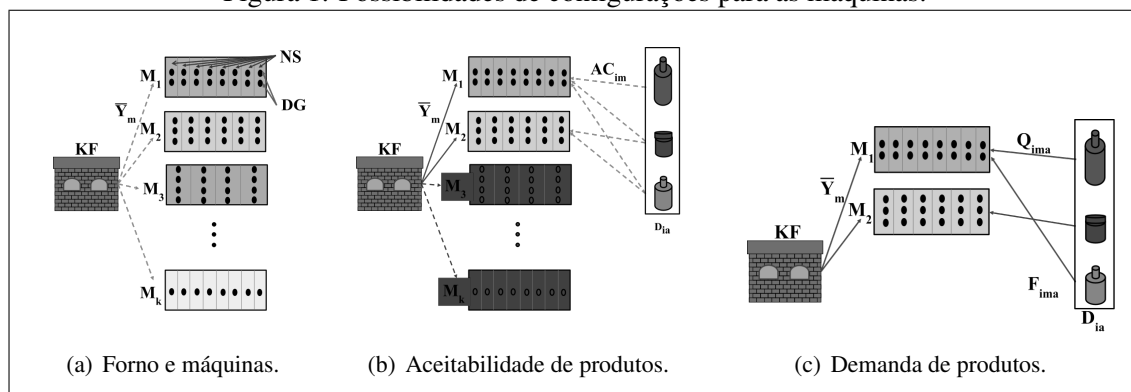
gramação linear inteiro misto é proposto para descrever e solucionar de forma exata o problema, assim como heurísticas e meta-heurísticas combinadas a técnicas de programação matemática são utilizadas.

De forma semelhante aos métodos utilizados por Toledo et al. (2016), o presente trabalho combinará meta-heurísticas com técnicas de programação matemática na resolução do PIEV-NF. A meta-heurística utilizada neste trabalho é um algoritmo genético multi-populacional (AGMP) semelhante ao descrito por Toledo et al. (2016, 2013) e L. Buriol e Moscato (2004). Todavia, o AGMP proposto será responsável por evoluir variáveis inteiras, enquanto as demais variáveis contínuas do problema são determinadas pela resolução do modelo linear relacionado. Trata-se de uma abordagem semelhante a utilizada por Almeder (2010) para solucionar o problema capacitado de dimensionamento de lotes multi-nível. No qual, o autor aplica um algoritmo de colônia de formigas para definir as variáveis binárias, enquanto a solução exata do modelo matemático define as variáveis contínuas.

3. Problema da Indústria de Embalagens de Vidro para um Novo Forno

Indústrias que fabricam recipientes (embalagens) de vidro geralmente apresentam diversas plantas de produção, onde cada uma pode ter um número diferente de fornos com capacidade variável e configurações complexas (Garavelli, 2003). Cada máquina está ligada a um único forno a partir do qual a pasta de vidro é recebida. Além disso, um forno pode alimentar simultaneamente várias máquinas ligadas a ele. As configurações de máquinas conectadas a um forno precisam ser definidas seguindo a previsão de demanda para o processo de produção dentro de um horizonte de tempo. A Figura 1 ilustra os aspectos relacionados a uma única planta e a instalação ou construção de um novo forno.

Figura 1: Possibilidades de configurações para as máquinas.



Fonte: Elaborada pelos autores.

A Figura 1(a) ilustra as decisões envolvidas para a construção de um novo forno que será integrado a uma fábrica de recipientes de vidro. Estas decisões são a capacidade de fusão KF e um subconjunto de máquinas a serem instaladas \bar{Y}_m . Primeiramente, deve-se decidir um subconjunto de várias configurações de máquinas a serem instaladas no forno. Estas máquinas possuem diferentes configurações com relação ao número de seções $NS \in \{6, 8, 10, 12, 16, 20\}$, ao tipo de gota $TG \in \{\text{Single-Gob (SG), Double-Gob (DG), Triple-Gob (TG), Quadruple-Gob (QG)}\}$, ao processo de fabricação soprado-soprado (BB, do inglês, *Blow and Blow*), prensado-soprado (PB, do inglês, *Press and Blow*) e prensado-soprado-gargalo-estrito (NNPB, do inglês, *Narrow Neck Press and Blow*) e as distâncias entre os centros das seções. Esta configuração implica no dimensionamento da produção e nas restrições de produtos possíveis de serem fabricados.

A Figura 1(b) exemplifica um forno que será construindo com duas máquinas (M_1 e M_2) instaladas, onde as máquinas possuem 8 e 6 seções com tipo de gota DG e TG, respectivamente.

Observe que o primeiro recipiente pode ser produzido somente na máquina M_1 , isto significa que algumas máquinas não são capazes de fabricar determinados tipos de produtos AC_{im} .

Na Figura 1(c) a máquina M_2 fabricará apenas o segundo produto e a máquina M_1 ficará dedicada à fabricação do primeiro e terceiro produto. Ao mesmo tempo, a capacidade KF do novo forno deve ser capaz de derreter vidro suficiente para atender a demanda produtiva Q_{ima} .

4. Modelagem Matemática

Esta seção descreve o modelo matemático proposto para o PIEV-NF.

Parâmetros:

- m : Índice das máquinas disponíveis ($m = 1, \dots, M$).
- i : Índice dos produtos a serem fabricados ($i = 1, \dots, I$).
- a : Índice do horizonte de tempo anual ($a = 1, \dots, A$).
- NS_m : Número de seções por máquina m .
- TG_m : Tipo de gota por máquina m .
- AC_{im} : 1 se o produto i for aceito na máquina m .
- C_m : Custo para instalar a máquina m (\$).
- D_{ia} : Demanda esperada do produto i no período a (ton).
- W_i : Peso do produto i (ton).
- R_i : Eficiência da cavidade para o produto i (Garrafas/min).
- \bar{M} : Máximo de máquinas suportada pelo novo forno, normalmente é um valor de 1 até 6.
- CF : Custo para instalar a capacidade de fusão no forno (\$/ton).
- η_m : Eficiência da máquina m (%).

Variáveis:

- KF : Capacidade de fusão necessária para o forno (ton).
- Q_{ima} : Tamanho do lote do produto i na máquina m no período a (ton).
- F_{ima} : Tempo gasto no período a em que a máquina m foi dedicada para produzir o produto i (anos).
- \bar{Y}_m : 1 se a máquina m estiver instalada, 0 caso contrário.

Formulação Matemática:

$$\text{Min } f(KF, \bar{Y}_1, \dots, \bar{Y}_M) = CF * KF + \sum_{m=1}^M C_m \cdot \bar{Y}_m \quad (1)$$

s.a:

$$\sum_{m=1}^M \bar{Y}_m \leq \bar{M} \quad (2)$$

$$F_{ima} \leq AC_{im} \quad \forall(i, m, a) \quad (3)$$

$$\sum_i F_{ima} = \bar{Y}_m \quad \forall(m, a) \quad (4)$$

$$Q_{ima} = F_{ima} \cdot (R_i \cdot W_i \cdot NS_m \cdot TG_m \cdot \eta_m) \quad \forall(i, m, a) \quad (5)$$

$$\sum_{\tau=1}^a \sum_m Q_{im\tau} \geq \sum_{\tau=1}^a D_{i\tau} \quad \forall(i, a) \quad (6)$$

$$\sum_i \sum_m Q_{ima} \leq KF \quad \forall(a) \quad (7)$$

$$KF, Q_{ima}, F_{ima} \geq 0 \quad (8)$$

$$\bar{Y}_m \in \{0, 1\} \quad (9)$$

A função objetivo (1) define os principais custos envolvidos na tomada de decisão, que ocorre com base na capacidade de fusão KF do forno e custo de implantação C_m das máquinas \bar{Y}_m . A restrição (2) representa o limite de máquinas que podem ser instaladas no forno, sendo no mínimo uma e no máximo seis. A restrição (3) permite produção apenas se o produto i é aceito na máquina m , como descrito na Figura 1 (b). A restrição (4) descreve o tempo de produção que é distribuído entre os produtos i em cada máquina m e ano a , caso a máquina tenha sido instalada $\bar{Y}_m = 1$. A restrição (5) apresenta o cálculo de produção, ou seja, o tamanho do lote do produto i na máquina m no período a em toneladas Q_{ima} . A restrição (6) garante o atendimento da demanda, onde a produção de cada item Q_{ima} acumulada até o período a deve ser maior ou igual a demanda D_{ia} acumulada do item até o mesmo período. A restrição (7) define a capacidade KF do forno necessária para atender a demanda. As restrições (8) e (9) apresentam os respectivos domínios das variáveis de decisão.

5. Métodos

Um total de três métodos são aplicados para resolver instâncias do PIEV-NF: um método exato e dois algoritmos genéticos. O método branch & cut disponível no *solver* comercial *CPLEX*[®], versão 12.6, foi considerado como método exato e executado com a configuração padrão e uma única *thread*. Os dois algoritmos genéticos são multi-populacionais, sendo que um apresenta apenas a estrutura em grade (AGMP) e outro inclui uma heurística de filtro (AGMP+F). Todos os algoritmos genéticos codificam as variáveis binárias do modelo PIEV-NF como indivíduos e a função objetivo (1) é definida como função de aptidão. Assim, as outras variáveis no PIEV-NF são definidas otimamente pela resolução do modelo de programação linear relacionado para cada codificação binária (indivíduo no AGMP). Os algoritmos genéticos AGMP e AGMP+F são baseados no algoritmo genético multi-populacional apresentado por Toledo et al. (2016), como descrito no Algoritmo 1. A diferença está nos operadores que foram implementados especificamente para o PIEV-NF e numa nova forma de estruturar os indivíduos de cada população.

Algoritmo 1: Algoritmo Genético Multi-Populacional (AGMP).

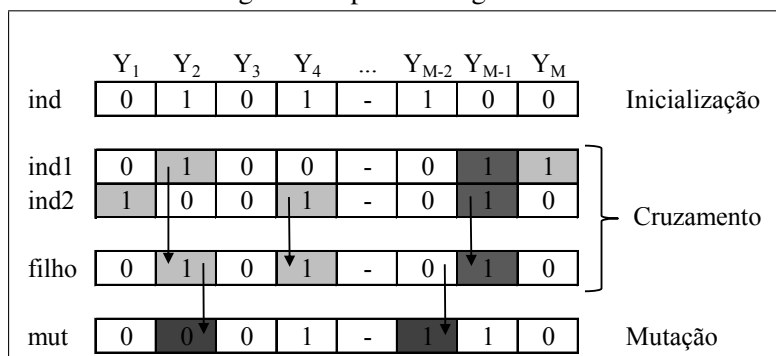
```

1 início
2   repita
3     para pop na população faça
4       para ind na pop faça
5         inicialize(ind);
6         evolua(ind);
7       repita
8         para j = 1 até número de cruzamento faça
9           seleccione(pais);
10          filho ← cruzamento (pais);
11          λ ← um valor aleatório ∈ [0.0 , 1.0];
12          se λ ≤ taxa de mutação então
13            mutacao(filho);
14          evolua(filho);
15          adicione(filho na pop);
16        até convergir(pop);
17      para pop na população faça
18        migracao(pop);
19   até alcançar(tempo limite);

```

Cada indivíduo representa uma solução baseada no vetor de máquinas Y_m . As máquinas para instalação são aleatoriamente escolhidas (atribuindo o valor 1) no procedimento *inicialize(ind)* de acordo com a quantidade máxima de máquinas a serem instaladas (\bar{M}). Isso garante a instalação de pelo menos uma e no máximo \bar{M} máquinas. A Figura 2 exemplifica isso, onde as máquinas Y_2 , Y_4 and Y_{M-2} são escolhidas para serem instaladas.

Figura 2: Operadores genéticos.

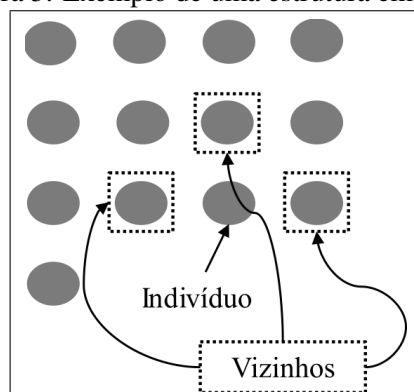


Fonte: Elaborada pelos autores.

O operador de cruzamento verifica nos pais escolhidos (ind1 e ind2) as entradas que já têm máquinas instaladas. O novo indivíduo (filho) manterá as máquinas comuns de ind1 e ind2. As outras máquinas disponíveis em ind1 e ind2 são selecionadas aleatoriamente, mas sem ultrapassar o valor máximo \bar{M} . A Figura 2 ilustra a união de máquinas de ind1 e ind2, entradas Y_1 , Y_2 , Y_4 , Y_{M-1} e Y_M , onde a interseção é apenas uma entrada Y_{M-1} . O filho terá a interseção Y_{M-1} e um subconjunto selecionado aleatoriamente da união, que neste exemplo são Y_2 e Y_4 . O operador de mutação (mut) seleciona aleatoriamente duas entradas e muda seus valores como visto na Figura 2. Nesse caso, as entradas selecionadas são Y_2 e Y_{M-2} com seus valores alterados de 0 para 1 e 1 para 0, respectivamente. A aptidão é calculada resolvendo o modelo linear quando as variáveis binárias são fixadas a partir das máquinas selecionadas na codificação (indivíduo). Assim, os valores ótimos das variáveis contínuas são encontrados para o conjunto binário de valores (configuração

da máquina). Este novo indivíduo (filho) é adicionado à população, se for melhor do que seu pai. O método converge quando um filho não é inserido na população atual. Neste caso, um operador de migração envia o melhor indivíduo da população atual para a próxima a ser evoluída.

Figura 3: Exemplo de uma estrutura em grade.



Fonte: Elaborada pelos autores.

Na natureza é esperado que um par de indivíduos localizados geograficamente próximos tenham maiores chances de acasalarem do que dois indivíduos separados por grandes distâncias. Com base nesse princípio, é implementada uma estrutura em grade para organizar os indivíduos como exemplificado na Figura 3. Uma ligação entre indivíduos é assumida e, a partir dela, se estabelece uma estrutura de vizinhança entre os indivíduos. Neste caso, cada indivíduo tem no máximo quatro vizinhos localizados nas posições superior, inferior, esquerda e direita como ilustrado na Figura 3. Uma das vantagens nesse tipo de estrutura é a baixa complexidade computacional ao se fazer as operações de seleção e adição de novos indivíduos na população.

Assim, o procedimento *selecione* escolhe aleatoriamente um indivíduo e um de seus vizinhos com complexidade $O(1)$ e o procedimento *adicione* insere o novo filho no lugar do pior pai, caso o filho seja melhor, também com complexidade $O(1)$. A critério de comparação, a complexidade de utilizar uma seleção por roleta seria $O(m)$, onde m é o número de indivíduos da população, enquanto na seleção por torneio seria $O(t)$ onde t é o tamanho do torneio. Estruturar os indivíduos hierarquicamente em árvores como proposto por Toledo et al. (2016) tem uma eficiência também de $O(1)$, mas apresenta uma complexidade $O(\log|m|)$ para manter a estrutura hierárquica.

Algoritmo 2: Heurística de Filtro (F)

```

1 início
2   se não existe máquina instalada então
3     └─ retorne penaliza;
4   se quantidade de máquinas instaladas não respeita o limitante ( $\bar{M}$ ) então
5     └─ retorne penaliza
6   para cada produto; faça
7     └─ se existe demanda  $D_{ia} > 0$  mas não há aceitabilidade  $AC_{im} = 0$  então
8       └─ └─ retorne penaliza
9     └─ resolva o modelo no AGMP;
```

Por fim, a heurística de Filtro (F) descrita no Algoritmo 2 apresenta as condições para penalizar uma solução antecipadamente sem a necessidade de solucionar o modelo linear relacionado. Isso ocorrerá quando não existe máquina instalada, a quantidade de máquinas supera o limitante \bar{M} ou não há máquina disponível para atender demanda. Caso a solução codificada em um indivíduo não satisfaça tais condições, temos uma codificação de variáveis inteiras promissora que será

fixada no modelo matemático para resolução do modelo linear resultante.

6. Resultados e Discussões

O modelo de programação linear inteira mista proposto é codificado utilizando a ferramenta *Professional Optimization Framework* (ProOF) (Arantes, 2013, 2014) integrada com o IBM ILOG *CPLEX*[®] 12.6 (IBM, 2018). O ProOF também é utilizado para desenvolver os algoritmos genéticos. Os testes computacionais foram realizados em um computador com processador Intel Xeon E5-2680v2, frequência de 2,8 GHz, memória RAM de 128 GB e sistema operacional Linux. A Tabela 1 apresenta os valores dos parâmetros para os AGMP e AGMP+F. Estes valores são baseados nos melhores relatados por Toledo et al. (2013).

Tabela 1: Valores dos parâmetros dos operadores.

Parâmetros dos Operadores				
Métodos	Tamanho da Populações	Número de Indivíduos	Taxa de cruzamento	Taxa de Mutação
AGMP	3	3x13	5.0	0.7
AGMP+F	3	3x13	5.0	0.7

Fonte: Elaborada pelos autores.

As instâncias avaliadas foram elaboradas utilizando um gerador aleatório a partir de dados fornecidos por uma IEV localizada em Portugal e outra no Brasil. Dois conjuntos de instâncias são elaborados e nomeados, respectivamente, *Factory-Machine* (FM) e *Horizon- Time* (HT). O número de máquinas aumentam no conjunto FM, enquanto o horizonte de tempo aumenta no conjunto HT. Cada conjunto tem cinco subconjuntos com 10 instâncias. O número de variáveis e restrições aumentam em ambos os conjuntos.

Os conjuntos de instâncias FM e HT considerados de pequeno porte são denominados SFM e SHT (S-Small), respectivamente. Os parâmetros aplicados para criar as instâncias dos cinco subconjuntos de SFM são $M \in \{100, 200, 300, 400, 500\}$ com $T = 8$. Os nomes dados aos subconjuntos estão associados aos valores atribuídos a M : SFM0100, SFM0200, ..., SFM0500. Para o subconjunto SHT, os parâmetros são $M = 300$ com $T \in \{4, 6, 8, 10, 12\}$. Os nomes dados aos subconjuntos são: SHT04, SHT06, ..., SHT12.

Os conjuntos de instâncias FM e HT considerados de grande porte são denominados LFM e LHT (L-Large), respectivamente. Neste caso, os parâmetros aplicados para criar 10 instâncias para cada subconjunto são $M \in \{1000, 2000, 3000, 4000, 5000\}$ com $T = 8$ para as LFM, onde os nomes dos subconjuntos são: LFM1000, LFM2000, ..., LFM5000. Para as instâncias LHT, os parâmetros são: $M = 3000$ com $T \in \{4, 6, 8, 10, 12\}$ e os subconjuntos são: LHT04, LHT06, ..., LHT12.

Logo, os métodos são avaliados a partir de um total de 200 instâncias, onde 50 instâncias são do conjunto SFM, 50 do SHT, 50 do LFM e 50 do LHT. O tempo limite para executar cada método foi de 3600 segundos uma única vez. O desempenho do método exato é avaliado com base na diferença (GAP) entre o *Upper Bound* (UB) e o *Lower Bound* (LB) alcançado pelo CPLEX como visto na equação (10).

Para viabilizar este trabalho, as empresas forneceram dados reais sobre custos e configurações de máquinas, mas não forneceram os dados relativos ao planejamentos da produção executada. Logo, não foi possível fazer uma comparação com os resultados reais. Todavia, o modelo proposto não visa refletir determinada produção executada e sim balizar a tomada de decisão, baseada em histórico de demandas ou previsão de demanda futura, quando se fizer necessária a construção de um novo forno.

$$GAP(\%) = 100 * \frac{CPLEX^{UB} - CPLEX^{LB}}{CPLEX^{UB}}. \quad (10)$$

Os algoritmos genéticos também são comparados com base na diferença (GAP) entre o *Upper Bound* (UB) e o *Lower Bound* (LB) como descrito pela equação (11).

$$GAP_{AGMP}(\%) = 100 * \frac{AGMP^{UB} - CPLEX^{LB}}{AGMP^{UB}}. \tag{11}$$

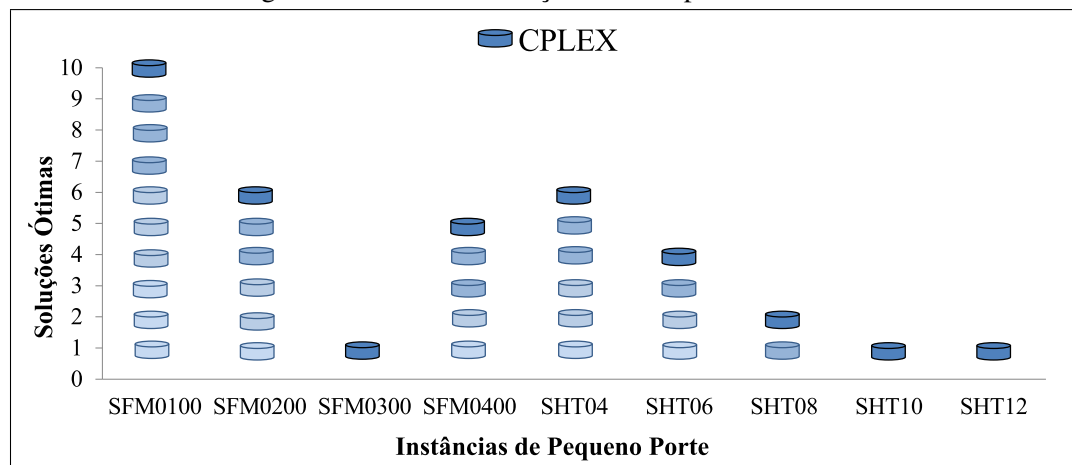
Para comparar melhor os métodos, uma análise estatística foi realizada. As médias dos GAPs, a média dos *Upper Bound* (objetivo) e a média do menor tempo para alcançar a melhor solução foram avaliados. O teste não-paramétrico Friedman (Equipe Estatcamp, 2014) (12) foi aplicado uma vez que as médias dos dados avaliados não seguem uma distribuição normal pelo teste de Anderson-Darling. Esse teste avalia a hipótese nula (H_0) de que todas as amostras têm funções de distribuição iguais, com p-valor = 0,005, versus a hipótese alternativa de que pelo menos duas das amostras têm diferentes funções de distribuição (H_1).

$$Friedman = \begin{cases} H_0 : CPLEX = AGMP = AGMP + F; \\ H_1 : CPLEX, AGMP, AGMP + F. \end{cases} \tag{12}$$

6.1. Resultados das Instâncias de Pequeno Porte

A resolução das instâncias de pequeno porte com o *solver* CPLEX, permitiu alcançar soluções ótimas para alguns casos, como mostra a Figura 4. O CPLEX encontrou 10 soluções ótimas no cenário com 100 máquinas (SFM0100). Foi possível encontrar 6 soluções ótimas em SFM0200. No grupo de SHT, um total de 6 e 4 soluções ótimas foram obtidas para os dois cenários mais simples (SHT04 e SHT06), enquanto temos uma ou duas soluções nos demais casos. Assim, o CPLEX alcança 36 de 100 possíveis soluções ótimas.

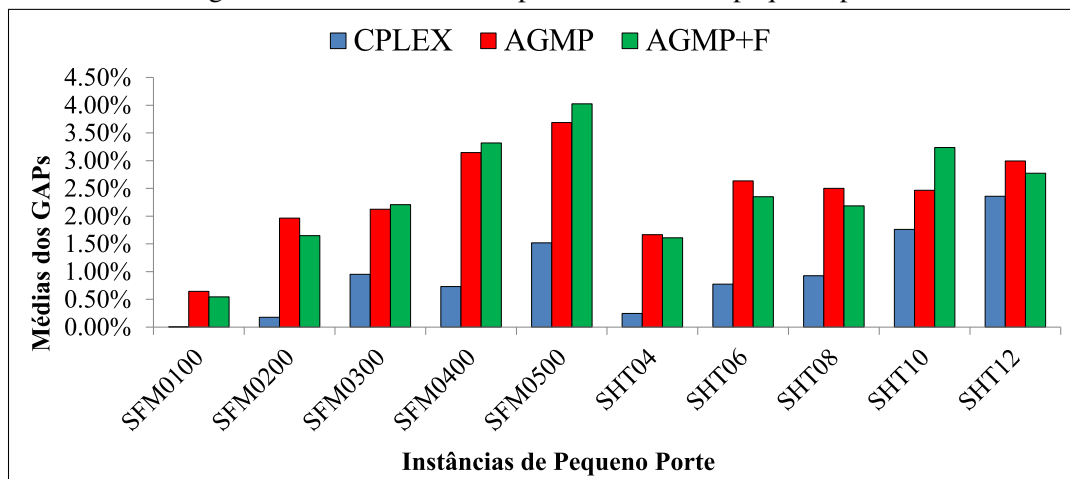
Figura 4: Número de soluções ótimas para o modelo.



Fonte: Elaborada pelos autores.

Os algoritmos genéticos também alcançam soluções ótimas para as instâncias pequenas. O AGMP obteve quatro soluções ótimas, sendo três no conjunto SFM0100 e uma no SHT12. O AGMP+F alcançou três soluções ótimas, sendo duas no SFM0100 e uma no SHT12. A Figura 5 compara as médias dos GAPs para cada conjunto de instâncias. Observe que o método exato tem a menor média de GAP entre todos os métodos. Além disso, o AGMP atinge as melhores soluções em média de GAP nos conjuntos: SFM0300, SFM0400, SFM0500 e SHT10. Enquanto que, o AGMP+F retorna as melhores soluções em média de GAP para os conjuntos: SFM0100, SFM0200, SHT04, SHT06, SHT08 e SHT12.

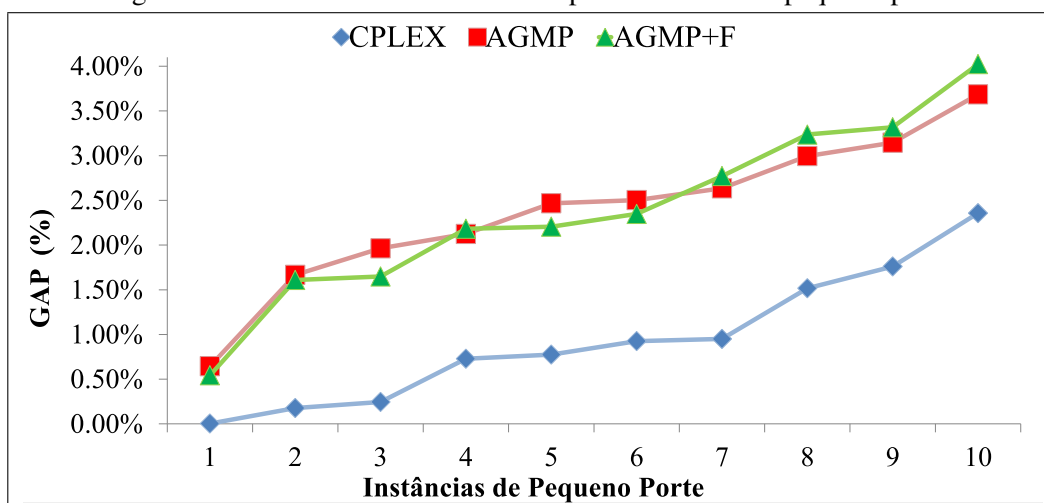
Figura 5: Médias dos GAPs para instâncias de pequeno porte.



Fonte: Elaborada pelos autores.

A Figura 6 apresenta as médias dos GAPs dos métodos ordenados de forma crescente. Pode-se identificar a qualidade do GAP para o método exato, onde quanto menor o GAP melhor é a solução obtida. Nota-se também que os valores de GAP para o AGMP e o AGMP+F são similares.

Figura 6: Médias dos GAPs ordenados para instâncias de pequeno porte.



Fonte: Elaborada pelos autores.

A Tabela 2 apresenta uma análise estatística com o teste de Friedman/Simes-Hochberg para as instâncias de pequeno porte, considerando a média dos GAPs, o Objetivo (LB) e o Tempo (o menor tempo para alcançar a melhor solução) de cada conjunto de instância. Os resultados do teste de Friedman indicam uma diferença significativa entre o CPLEX e os genéticos para os valores de GAP e o Objetivo, uma vez que todos os P-valores são menores que 0,05 (5%). Logo, rejeitamos a hipótese nula de que os métodos são iguais. O teste de Simes-Hochberg é aplicado após o teste de Friedman para comparações múltiplas entre blocos de dados. No teste de Simes-Hochberg, os métodos são classificados do melhor (grupo B) para o pior (grupo A) após a comparação. Nesse caso, a diferença entre os métodos aparece nos grupos diferentes obtidos: **B** para o CPLEX e **A** para os AGMP e AGMP+F.

Tabela 2: Teste de Friedman para instâncias de pequeno porte.

Análise Estatística para Instâncias de Pequeno Porte					
Médias	Teste de Friedman	P-valor	CPLEX	AGMP	AGMP+F
GAP	15.2	0.000500451	B	A	A
Objetivo	15.2	0.000500451	B	A	A
Tempo	2.6	0.272531793	A	A	A

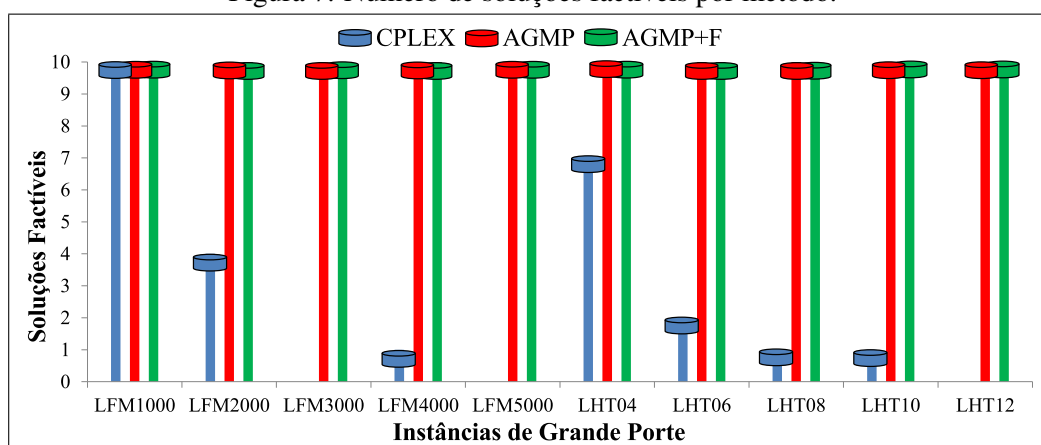
Fonte: Elaborada pelos autores.

Assim, o CPLEX se mostrou diferente em relação aos genéticos e seu bom desempenho para as instâncias de pequeno porte é visível no gráfico de médias dos GAPs ordenados, apresentado na Figura 5. Já os genéticos, AGMP e AGMP+F, possuem comportamentos similares com nenhuma diferença significativa como visto na análise estatística.

6.2. Resultados das Instâncias de Grande Porte

O CPLEX não retorna nenhuma solução ótima para instâncias de grande porte dentro do limite de tempo. A Figura 7 mostra o número de soluções factíveis alcançadas pelos métodos. O método exato também tem problemas para retornar soluções factíveis, retornando apenas 26 soluções de 100. Os AGMP e AGMP+F encontraram soluções factíveis para todos os conjuntos de instâncias.

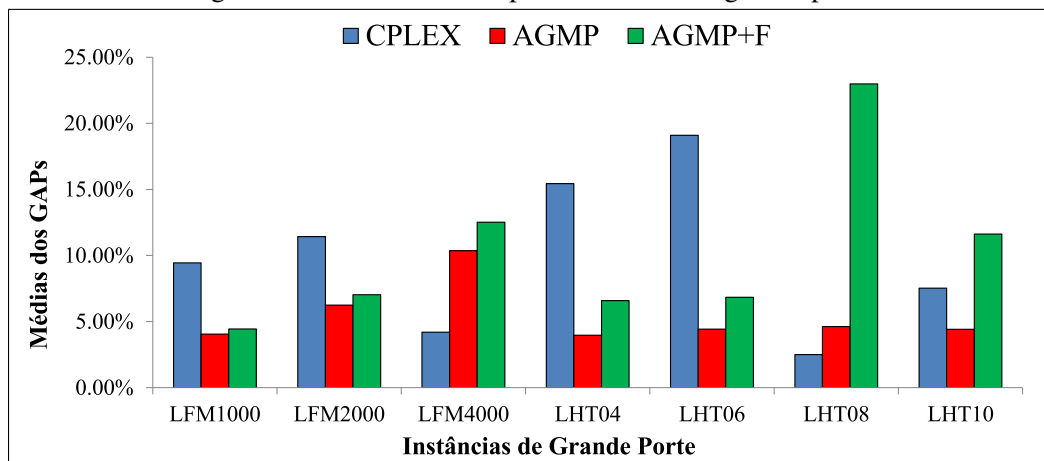
Figura 7: Número de soluções factíveis por método.



Fonte: Elaborada pelos autores.

As médias dos GAPs são comparada na Figura 8, mas somente para os conjuntos que alcançaram alguma solução. Há vários casos de instâncias de grande porte não solucionados pelo CPLEX, onde tal situação ocorre com todas as instâncias do conjuntos LFM3000, LFM5000 e LHT12. O método exato tem a menor média de GAP apenas para LFM4000 e LHT08 como visto na Figura 8. O AGMP apresenta melhores médias dos GAPs para os conjuntos: LFM1000, LFM2000, LHT04, LHT06 e LHT10. Enquanto o AGMP+F se comporta melhor que o CPLEX para os conjuntos: LFM1000, LFM2000, LHT04 e LHT06.

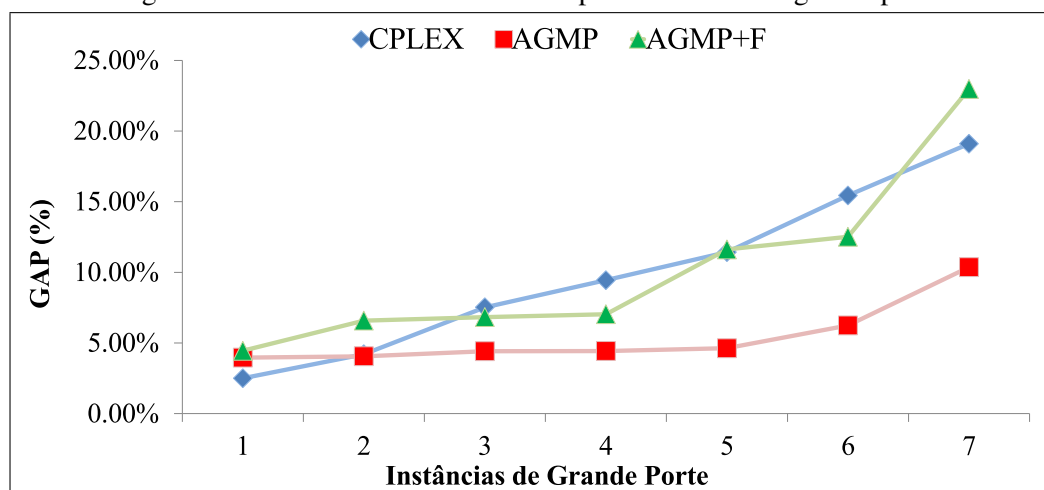
Figura 8: Média dos GAPs para instâncias de grande porte.



Fonte: Elaborada pelos autores.

A Figura 9 apresenta as médias dos GAPs dos métodos ordenadas de forma crescente. Pode-se identificar a qualidade do GAP para o AGMP já que quanto menor o GAP, melhor é a solução obtida.

Figura 9: Médias dos GAPs ordenados para instâncias de grande porte.



Fonte: Elaborada pelos autores.

A Tabela 3 apresenta uma análise estatística com o teste de Friedman/Simes-Hochberg para instâncias de grande porte, baseado na média dos GAPs, no Objetivo e no Tempo de cada conjunto de instância. Os resultados do teste de Friedman mostraram uma diferença significativa entre os métodos AGMP e AGMP+F, com base no GAP e no Tempo, uma vez que os P-valores são menores que 0,05 (5%). Assim, rejeitamos a hipótese nula de que os métodos são iguais. O teste de Simes-Hochberg aplicado após o teste de Friedman para comparações múltiplas entre blocos de dados revela a diferença entre os métodos que são colocados em diferentes grupos: **B** para o AGMP+F e **A** para o AGMP.

Tabela 3: Teste de Friedman para instâncias de grande porte.

Análise Estatística para Instâncias de Grande Porte				
Médias	Teste de Friedman	P-valor	AGMP	AGMP+F
GAP	7	0.008150972	A	B
Objetivo	1.6	0.205903211	A	A
Tempo	6.4	0.011412036	A	B

Fonte: Elaborada pelos autores.

Logo, os genéticos, AGMP e AGMP+F, possuem desempenhos distintos com diferença significativa como visto na análise estatística. Considerando o gráfico de médias dos GAPs ordenados (Figura 9), temos que o AGMP apresenta um comportamento melhor que o AGMP+F.

7. Conclusões

Este trabalho apresenta um modelo matemático para o PIEV-NF e a aplicação de um método exato e dois algoritmos genéticos combinados com técnica de programação matemática. A resolução exata do modelo, através do CPLEX, é capaz de determinar várias soluções ótimas para problemas de pequeno porte. No entanto, o método exato não é capaz de encontrar soluções ótimas ou mesmo factíveis para muitos casos em que as instâncias são de grande porte. Por outro lado, o AGMP e AGMP+F obtiveram um bom desempenho em instâncias de grande porte e retornaram soluções viáveis para todas essas instâncias. Comparando AGMP com AGMP+F, observa-se que o uso da heurística de Filtro (F) não foi capaz de retornar melhores resultados considerando a análise estatística para as médias das funções objetivos. Porém para as médias dos GAPs e dos tempos houve uma diferença significativa. Isso pode indicar que a resolução dos modelos lineares relacionados aos indivíduos foi capaz de avaliar de forma mais consistente esses indivíduos do que a penalização baseada nos critérios de filtragem estabelecidos. O desenvolvimento de outras abordagens que combinam meta-heurística como Greedy Randomized Adaptive Search Procedure (GRASP), Iterated Local Search (ILS) e Simulated Annealing (SA) com técnicas de programação matemática serão propostos como trabalhos futuro.

Agradecimentos. Os autores deste trabalho agradecem a Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), o Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq-grant 483474/2013-4) e a Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP) pelo apoio no desenvolvimento deste trabalho. Esta pesquisa foi desenvolvida com a utilização de recursos computacionais disponibilizados pelo Centro de Ciências Matemáticas Aplicadas à Indústria (CeMEAI), financiados pela FAPESP (proc. 2013/07375-0).

Referências

- Almada-Lobo, B., Klabjan, D., Carravilla, M. A., e Oliveira, J. F. Multiple machine continuous setup lotsizing with sequence-dependent setups. *Computational Optimization and Applications*, v. 47, n. 3, p. 529–552, 2010.
- Almeder, C. A hybrid optimization approach for multi-level capacitated lot-sizing problems. *European Journal of Operational Research*, v. 200, n. 2, p. 599 – 606, 2010.
- Arantes, M. S. *Professional optimization framework (ProOF)*. Version 2.4.0, 2013. Disponível em: <https://github.com/marcio-da-silva-arantes/ProOF/wiki>. Acesso em: 30/11/2018.
- Arantes, M. S. *Ambiente para desenvolvimento de métodos aplicados a problemas de otimização*. 116 pg. Dissertação de Mestrado. Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos, S.P., 2014.

- Copil, K., Wörbelauer, M., Meyr, H., e Tempelmeier, H. Simultaneous lotsizing and scheduling problems: a classification and review of models. *OR Spectrum*, v. 39, n. 1, p. 1–64, 2017. ISSN 1436-6304.
- Drexl, A. e Kimms, A. Lot-sizing and scheduling – survey and extensions. *European Journal of Operational Research*, v. 99, p. 221–235, 1997.
- Equipe Estatcamp. *Software action*, 2014. Disponível em: <http://www.portalaction.com.br/>. Acesso em: 30/11/2018.
- Garavelli, A. C. Flexibility configurations for the supply chain management. *International Journal of Production Economics*, v. 85, p. 141–153, 2003.
- Guimarães, L., Klabjan, D., e Almada-Lobo, B. Annual production budget in the beverage industry. *Engineering Applications of Artificial Intelligence*, v. 25, n. 2, p. 229–241, 2012.
- IBM. *Ibm ilog cplex - Optimization Studio - CPLEX Users' Manual*. Version 12, Release 6, 2018. Disponível em: <https://www.ibm.com/analytics/cplex-optimizer>. Acesso em: 30/11/2018.
- Jans, R. Solving lot-sizing problems on parallel identical machines using symmetry-breaking constraints. *INFORMS Journal on Computing*, v. 21, n. 1, p. 123–136, 2009.
- Jans, R. e Degraeve, Z. Meta-heuristics for dynamic lot sizing: A review and comparison of solution approaches. *European Journal of Operational Research*, v. 177, p. 1855–1875, 2007.
- Karimi, B., Ghomi, S. M. T. F., e Wilson, J. M. The capacitated lot sizing problem: a review of models and algorithms. *Omega*, v. 31, p. 365–378, 2003.
- L. Buriol, P. M. F. e Moscato, P. A new memetic algorithm for the asymmetric traveling salesman problem. *Journal of Heuristics*, Kluwer Academic Publishers, Hingham, MA, USA, v. 10, p. 1381–1231, 2004.
- Toledo, C. F. M., Arantes, M., Hossomi, M. Y. B., e Almada-Lobo, B. Mathematical programming-based approaches for multi-facility glass container production planning. *Computers Operations Research*, v. 74, n. 2016, p. 92–107, 2016.
- Toledo, C. F. M., Arantes, M., Oliveira, R., e Almada-Lobo, B. Glass container production scheduling through hybrid multi-population based evolutionary algorithm. *Applied Soft Computing*, v. 13, n. 3, p. 1352–1364, 2013.