

## Um modelo para o problema de dimensionamento e sequenciamento de lotes com aceitação de pedidos

**Rudivan Paixão Barbosa**

Universidade de São Paulo

Av. Trabalhador São-carlense, 400, Centro, 13566-590, São Carlos - SP, Brasil

rudivanbarbosa@gmail.com

**Willy Alves de Oliveira**

Instituto de Matemática - UFMS

Av. Costa e Silva, Pioneiros, Campo Grande - MS, Brasil

willy.oliveira@ufms.br

**Maristela Oliveira Santos**

Universidade de São Paulo

Av. Trabalhador São-carlense, 400, Centro, 13566-590, São Carlos - SP, Brasil

mari@icmc.usp.br

### RESUMO

Este trabalho apresenta um modelo de programação inteira mista para o problema de dimensionamento e sequenciamento de lotes integrado ao problema de aceitação de pedidos e com a consideração da idade dos produtos em estoque. Neste problema, as demandas dos clientes são agregadas em pedidos, os quais podem ou não serem aceitos. Os pedidos devem ser entregues dentro de uma janela de tempo. Os itens são perecíveis e podem permanecer no estoque por um determinado tempo (*shelf-life*). O objetivo é maximizar o lucro gerado pelo atendimento dos pedidos, descontando os custos de estoque e de preparação da máquina. Para tratar o problema são desenvolvidas duas heurísticas construtivas. Testes computacionais foram realizados e os resultados obtidos foram analisados. As heurísticas tiveram um desempenho, em média, superior ao *branch-and-cut* do CPLEX na obtenção de soluções de boa qualidade, no limite de tempo estabelecido.

**PALAVRAS CHAVE.** Dimensionamento e sequenciamento de lotes, aceitação de pedidos, programação matemática inteira mista.

**Tópicos:** PM - Programação Matemática.

### ABSTRACT

This paper presents a mixed integer programming model with the age of the product in inventory for a lot-sizing and scheduling problem with order acceptance. In this problem, customer demand is aggregated into orders, that can be accepted or not. Orders must be delivered within a time window. Each item can keep in stock for a determinate time (*shelf-life*). The aim of the problem is to maximize the profit obtained by orders acceptance, discounting the costs of inventory and setup costs. Two constructive heuristics are developed, and the branch-and-cut of the CPLEX resolver is also used to obtain a solution to the problem. Computational tests were performed and

the results obtained were analyzed. The heuristics performed, on average, better than the branch-and-cut of the CPLEX in obtaining good quality solutions within the established time limit.

**KEYWORDS.** Lot sizing and scheduling, order acceptance, mixed integer mathematical programming.

**Paper topics:** MP - Mathematical Programming.

## 1. Introdução

O problema de dimensionamento de lotes consiste em planejar a quantidade a ser produzida dos itens em cada período ao longo de um horizonte de tempo finito, de modo a atender uma certa demanda e otimizar uma função objetivo. O problema de decidir, de maneira integrada o tamanho e a sequência dos lotes de produção é conhecido na literatura como problema de dimensionamento e sequenciamento de lotes. Uma revisão recente sobre problemas integrados de dimensionamento de lotes pode ser encontrada em Copil et al. [2017].

A maioria dos trabalhos encontrados na literatura considera o problema com o gerenciamento dos pedidos dos clientes desagregados em demanda, ou seja, procuram atender a demanda dos itens. No entanto, geralmente os clientes solicitam itens distintos agrupados em pedidos. Ao tratar o problema considerando o atendimento dos clientes via itens e permitir atrasos no atendimento da demanda, o pedido do cliente poderá ser entregue parcialmente. Em algumas indústrias alimentícias, os clientes (que podem ser restaurantes e/ou supermercados, por exemplo) usualmente realizam pedidos compostos por vários itens e não possuem interesse no atendimento parcial de seu pedido, pois pode ser necessário buscar os produtos, não entregues, em outros fornecedores, aumentando os custos. Assim, justifica a integração do problema de dimensionamento de lotes com o problema de aceitação de pedidos.

Seresht e Bijari [2013] tratam o problema integrado com possibilidade do não atendimento da demanda dos produtos. O objetivo é maximizar o lucro associado ao atendimento da demanda de cada item. O problema considerado pelos autores não gerencia as demandas via pedidos, mas sim, item por item. Um problema da literatura diretamente relacionado ao nosso trabalho foi abordado em Furtado [2012]. Neste trabalho é abordado um problema multiestágio com gerenciamento de pedidos em fundições de pequeno porte, onde consideram os pedidos como itens finais e os itens solicitados como os componentes. Os modelos propostos permitem atrasos no atendimento dos pedidos, se houver atrasos, estes são penalizados na função objetivo. Estes atrasos, se postergados até o fim do horizonte de planejamento, levam ao não atendimento do pedido.

Nos artigos Brahimy et al. [2015a] e Brahimy et al. [2015b] são apresentados modelos para o problema de dimensionamento de lotes com demanda gerenciada via pedidos com janelas de atendimento. No primeiro modelo, cada pedido tem lucro distinto, e não é obrigatório o atendimento dos pedidos, porém, a cada pedido não atendido é descontado um valor na função objetivo. Ainda nos modelos, é considerado carga de trabalho (*workload*) relacionada a produção dos itens nos pedidos, custos relacionados a produção dos itens e estoque são considerados. No segundo modelo, os pedidos possuem janelas de atendimento e as outras características são idênticas ao primeiro modelo. Os modelos nesses artigos permitem a entrega dos pedidos fora da janela de atendimento e penalizam na função objetivo, caso isso ocorra. Nestes artigos são utilizadas duas heurísticas construtivas como método de solução. A decomposição na primeira heurística é por períodos e na segunda é por pedidos. Porém, nestes trabalhos não consideram as decisões de sequenciamento no problema integrado.

Em Brahimy et al. [2018] alguns modelos são apresentados para o problema de planejamento de produção com aceitação de pedidos e incerteza na demanda, porém, sem as decisões das sequências de produção. Exceto no primeiro modelo não é considerado *workload*, e em todos os modelos existe incerteza na demanda dos itens. Como método de solução, é proposta uma heurística MIP de dois níveis. A fase 1, é utilizada uma *relax-and-fix* por período. Na fase 2, a heurística *fix-and-optimize* é utilizada como forma de melhoria.

Outro trabalho que aborda o problema de dimensionamento e sequenciamento de lotes integrado ao problema de aceitação de pedidos é o de Teixeira et al. [2017]. O problema estudado pelos autores foi motivado por uma indústria alimentícia no interior do estado de São Paulo que produz cargas embaladas. O objetivo é decidir quais pedidos serão aceitos a fim de maximizar o lucro total. Os produtos são perecíveis, ou seja, não podem ficar no estoque depois de uma certa quantidade de tempo, pois perdem seu valor nutricional. No modelo proposto para tratar o problema não é considerado idade do produto no estoque. Para a solução desse problema foi proposta uma heurística do tipo *fix-and-optimize*, os resultados obtidos pela heurística e por um resolvedor comercial foram comparados. Neste artigo, propomos uma alteração no modelo proposto por Teixeira et al. [2017] para considerar idade dos estoques dos produtos.

Na Seção 2 apresentamos o modelo matemático, na Seção 3 descrevemos as heurísticas propostas. Na Seção 4 são apresentados testes computacionais que foram realizados as abordagens propostas e as conclusões são apresentadas na Seção 5.

## 2. Modelagem matemática

Para modelar o problema de dimensionamento e sequenciamento de lotes, consideraremos que as demandas dos clientes são agregadas em pedidos. Existe apenas um recurso de produção, os itens são perecíveis e possuem idades (quantidade de períodos que o item fica no estoque). Todos os itens podem ficar apenas um determinado tempo no estoque (*shelf-life*), e esse tempo pode ser distinto para cada item. Os tempos/custos de preparação da máquina são dependentes da sequência de produção dos itens. Cada pedido tem uma janela de atendimento, e podem ser entregues somente dentro dessa janela. Não é permitida entrega parcial dos pedidos, ou seja, todos itens que compõem o pedido são entregues simultaneamente.

O modelo apresentado tem como base trabalho de Teixeira et al. [2017]. O objetivo principal é maximizar os lucros associados ao atendimento dos pedidos, descontando os custos de estoque e preparação da máquina. Neste artigo, utilizamos a ideia de Li et al. [2016] para representar as idades dos produtos no estoque e também para capturar as idades dos itens que são entregues aos clientes. Os índices e parâmetros considerados na modelagem do problema são apresentados a seguir:

---

### Índices e parâmetros

---

$n \in \{1, \dots, N\}$	índice dos pedidos;
$i, j \in \{1, \dots, J\}$	índice dos itens;
$t \in \{1, \dots, T\}$	índice dos períodos;
$N, J$ e $T$	número total de pedidos, produtos e períodos, respectivamente;
$P_{nt}$	lucro associado ao pedido $n$ no período $t$ ;
$F_n$ e $L_n$	primeiro e último período da janela de entrega do pedido $n$ , respectivamente;
$h_j$ e $a_j$	custo de estoque do item $j$ e tempo de produção do item $j$ , respectivamente;
$sc_{ij}$ e $st_{ij}$	custo e tempo de troca da produção do item $i$ para o item $j$ , respectivamente;
$q_{jn}$	demandas do item $j$ no pedido $n$ ;
$sl_j$	tempo máximo de estoque do item $j$ ( <i>shelf-life</i> );
$C_t$	capacidade (tempo) de produção do período $t$ .

### Variáveis de decisão

$x_{jt}$	produção do item $j$ no período $t$ ;
$I_{jt}^k$	estoque do item $j$ com idade $k$ ao final do período $t$ ;
$Q_{jnt}^k$	quantidade de itens $j$ com idade $k$ utilizados para atender o pedido $n$ ao final do período $t$ ;
$y_{jt}$	1 se a máquina está preparada para a produção do item $j$ no início do período $t$ e 0, caso contrário;
$\gamma_{nt}$	1 se o pedido $n$ vai ser atendido no período $t$ e 0, caso contrário;
$z_{ijt}$	1 se ocorre troca da produção do item $i$ para o item $j$ durante o período $t$ e 0, caso contrário;
$V_{jt}$	variável auxiliar que representa a ordem de produção do item $j$ no período $t$ .

$$\max \sum_{n=1}^N \sum_{t=F_n}^{L_n} P_{nt} \cdot \gamma_{nt} - \sum_{t=1}^T \sum_{j=1}^J h_j \cdot \sum_{k=0}^{sl_j} I_{jt}^k - \sum_{t=1}^T \sum_{i=1}^J \sum_{j=1}^J s c_{ij} \cdot z_{ijt} \quad (1)$$

Sujeito a:

$$\sum_{k=0}^{\min(sl_j, t-1)} I_{jt-1}^k + x_{jt} = \sum_{n=1}^N q_{jn} \cdot \gamma_{nt} + \sum_{k=0}^{\min(sl_j, t)} I_{jt}^k \quad \forall j, t; \quad (2)$$

$$I_{jt}^0 = x_{jt} - Q_{jnt}^0 \quad \forall j, t; \quad (3)$$

$$I_{jt}^k = I_{j,t-1}^{k-1} - \sum_{n=1}^N Q_{jnt}^k \quad \forall j \quad k = 1, \dots, sl_j + 1 \quad t = 2, \dots, T; \quad (4)$$

$$\sum_{k=0}^{sl_j} Q_{jnt}^k = q_{jn} \cdot \gamma_{nt} \quad \forall j, n, t; \quad (5)$$

$$I_{jt}^{sl_j} = 0 \quad \forall j, t; \quad (6)$$

$$\sum_{j=1}^J a_j \cdot x_{jt} + \sum_{i=1}^J \sum_{j=1}^J s t_{ij} \cdot z_{ijt} \leq C_t \quad \forall t; \quad (7)$$

$$\sum_{j=1}^J y_{jt} = 1 \quad \forall t; \quad (8)$$

$$y_{jt} + \sum_{i=1}^J z_{ijt} = \sum_{i=1}^J z_{j,i+1} + y_{j,t+1} \quad \forall j, t; \quad (9)$$

$$V_{jt} \geq V_{it} + 1 - J(1 - z_{ijt}) \quad \forall i, j \neq i, t; \quad (10)$$

$$x_{jt} \leq \frac{C_t}{a_j} (y_{jt} + \sum_{i=1}^J z_{ijt}) - \frac{s t_{ij}}{a_j} \sum_{i=1}^J z_{ijt} \quad \forall i, j, t; \quad (11)$$

$$\sum_{t=F_n}^{L_n} \gamma_{nt} \leq 1 \quad \forall n; \quad (12)$$

$$\gamma_{nt} = 0; \quad t < F_n \text{ e } t > L_n; \quad (13)$$

$$\gamma_{nt} \in \{0, 1\} \quad \forall n, t; \quad (14)$$

$$y_{jt} \in \{0, 1\} \quad \forall j, t; \quad (15)$$

$$z_{ijt} \in \{0, 1\} \quad \forall i, j, t; \quad (16)$$

$$x_{jt}, V_{jt} \geq 0 \quad \forall j, t \quad I_{jt}^k \geq 0 \quad \forall j, k, t \quad Q_{jnt}^k \geq 0 \quad \forall j, k, n, t. \quad (17)$$

A função objetivo (1) refere-se à soma dos lucros obtidos com o atendimento dos pedidos com diversas idades, reduzida pela soma dos custos de estoque e preparação da máquina. As restrições (2) são responsáveis pelo balanceamento de estoque dos itens considerando a idade do produto no estoque, garantindo a existência de itens em quantidade necessária para compor todos os pedidos que serão atendidos em cada período. As restrições (3) garantem que os itens que são produzidos e não são entregues no mesmo período que foram produzidos, são estocados com idade zero. As restrições (4) relacionam os estoques dos itens com diversas idades. As restrições (5) implicam que as demandas dos itens nos pedidos só podem ser satisfeitas com a entrega de itens com as idades possíveis. As restrições (6) garantem que no estoque não terão produtos com idade igual ou maior que o seu tempo de vida máximo. As restrições (7) referem-se aos limites de capacidade produtiva em cada período. As restrições (8) garantem que a máquina está preparada para algum item no início de cada período. As restrições (9) são restrições de fluxos, as restrições (10) capturam a ordem de produção de cada item em cada período, evitando subciclos desconexos. As restrições (11) garantem que um item só pode ser produzido se a linha estiver preparada para a sua produção. As restrições (12) garantem que cada pedido pode ser atendido, no máximo, uma vez durante o horizonte de planejamento. As restrições (13) introduzem as janelas de tempo para atendimento dos pedidos e, por fim, as restrições (14)-(17) definem os domínios das variáveis de decisão.

### 3. Heurísticas

Heurísticas construtivas (Pochet [2006]) são baseadas na decomposição do conjunto das variáveis inteiras em diversos subconjuntos. Considere  $R$  como o conjunto das variáveis binárias do problema. O conjunto  $R$  é particionado em  $L$  subconjuntos distintos, onde cada subconjunto é representado por  $R^l$  com  $l = 1, \dots, L$ .

Assim,

$$R = R^1 \cup R^2 \cup R^l \cup \dots \cup R^L \text{ e } R^1 \cap R^2 \cap R^l \cap \dots \cap R^L = \emptyset.$$

A heurística é inicializada considerando as variáveis inteiras em  $R^k$ , para algum  $k \in \{1, \dots, L\}$ , e relaxando as variáveis dos  $L - 1$  subconjuntos restantes. Caso a solução do sub-modelo seja factível, as variáveis do conjunto  $R^k$  são fixadas nos valores obtidos. O processo é repetido para todos subconjuntos de  $R$ . Se para algum passo do processo da heurística, o sub-modelo for infactível, o processo é interrompido sem solução factível para a instância considerada.

### 3.1 Heurística construtiva 1

A heurística proposta foi baseada na ideia de *Most Profitable First* apresentada no artigo de Brahimi et al. [2015b]. A heurística visa atender os pedidos que retornam os maiores lucros. Inicialmente, todos os pedidos são ordenados em ordem não-crescente em relação ao seu lucro ( $P_{nt}$ ). Em seguida, para o pedido mais lucrativo, adicionamos uma restrição do tipo  $\sum_{t=F_n}^{L_n} \gamma_{nt} = 1$  ao modelo para forçar o atendimento deste pedido, dentro da sua respectiva janela de atendimento. A ordem de inserção das restrições respeita a ordenação dos pedidos.

Para esta heurística faremos a decomposição do conjunto  $R$  das variáveis binárias  $\gamma_{nt} \forall n, t$  em  $N$  (número de pedidos) subconjuntos, ou seja, um conjunto para cada pedido. Neste caso, temos  $R = R^1 \cup R^2 \cup R^n \cup \dots \cup R^N$ . A cada passo da heurística, os conjuntos não considerados até o presente momento têm suas variáveis fixadas em zero ( $\forall \gamma_{nt} \in R^k \cup R^n \cup \dots \cup R^N \implies \gamma_{nt} = 0$ ). Durante o procedimento heurístico, caso a solução de um sub-modelo com a inserção da restrição que força o atendimento de um determinado pedido forneça o limitante primal pior do que o obtido até a iteração anterior, as suas variáveis também são fixadas em zero. Convém salientar que ao adicionar as restrições ( $\sum_{t=F_n}^{L_n} \gamma_{nt} = 1$ ), que forçam o atendimento de algum pedido, pode levar a uma solução infactível. Neste caso, retira-se a restrição recém adicionada.

Como comentado anteriormente, os pedidos que não estão sendo forçados a serem atendidos, não são atendidos na iteração corrente. Isso significa que, no sub-modelo, as variáveis desses pedidos são fixadas em zero. Com fixação das variáveis das várias partições em zero, pelos testes computacionais preliminares, foi observado que a heurística obteve soluções iniciais factíveis em um tempo computacional muito pequeno. A heurística construtiva 1 tende a encontrar soluções melhores do que o *branch-and-cut* do solver Cplex para instâncias com um número maior de variáveis e restrições, isso ocorre pela redução do número de variáveis binárias nos sub-modelos.

### 3.2 Heurística construtiva 2

Esta heurística difere da anterior na forma de escolha dos pedidos que serão atendidos em cada iteração. Na heurística 1 os pedidos que são atendidos inicialmente possuem maiores lucros associados, na heurística 2 os pedidos que serão atendidos inicialmente consideram os últimos períodos da janela de atendimento ( $L_n$ ). Isso significa que, a preferência de atendimentos dos pedidos é baseada em qual pedido que pode ser atendimento mais tarde dentro do horizonte de planejamento. Como a única diferença das heurísticas é a escolha da ordem de atendimento dos pedidos, será apresentado apenas um pseudocódigo para ambas.

No algoritmo 1 é apresentado o pseudocódigo das heurísticas, onde VetorLucro captura a ordem não-crescente dos lucros associados a cada pedido, e o VetorTime captura a ordem, também não-crescente, dos  $L_n$ .  $P$  é um conjunto. O parâmetro AtualSolução e MelhorSolução referem-se a atual e melhor solução encontrada pela heurística até a iteração corrente, respectivamente. O Tempolimite é o tempo limite estabelecido para cada iteração.

Consideramos o parâmetro TempoLimite utilizado no Algoritmo 1 com valor de 100 segundos, ou seja, cada subproblema é resolvido por 100 segundos. Observamos que nas primeiras iterações os subproblemas são resolvidos rapidamente, pois com a fixação das demais variáveis  $\gamma_{nt}$  (pedidos) em zero, o subproblema é consideravelmente menor do que o problema original em relação as variáveis binárias.

**início**

```

    AtualSolução, MelhorSolução, N, TempoLimite, VetorLucro, VetorTime, P;
    Initialize MelhorSolução =  $-\infty$ ;
    Initialize P =  $\emptyset$ ;
para  $n = VetorLucro[n](VetorTime[n]) : N$  faz
    |  $P = P \cup n$ ;
    | se  $n \in P$  então
    |   | Adicione  $\sum_{t=F_n}^{L_n} \gamma_{nt} = 1$ ;
    | fim
    | senão
    |   | Adicione  $\sum_{t=F_n}^{L_n} \gamma_{nt} = 0$ ;
    | fim
    | se Problema factível então
    |   | Resolva por TempoLimite (determine AtualSolução);
    |   | se AtualSolução < MelhorSolução então
    |   |   | MelhorSolução = AtualSolução;
    |   |   | Remova  $\sum_{t=F_n}^{L_n} \gamma_{nt} = 0, \forall n \notin P$ ;
    |   | fim
    |   | senão
    |   |   | Remova  $\sum_{t=F_n}^{L_n} \gamma_{nt} = 1$ ;
    |   |   | Remova  $\sum_{t=F_n}^{L_n} \gamma_{nt} = 0, \forall n \notin P$ ;
    |   |   |  $P = P - \{n\}$ ;
    |   | fim
    | fim
    | senão
    |   | Remova  $\sum_{t=F_n}^{L_n} \gamma_{nt} = 1$ ;
    |   | Remova  $\sum_{t=F_n}^{L_n} \gamma_{nt} = 0, \forall n \notin P$ ;
    |   |  $P = P - \{n\}$ ;
    | fim
fim
retorna MelhorSolução.

```

**fim**

**Algoritmo 1:** Pseudocódigo das heurísticas construtivas 1 e 2

#### 4. Testes computacionais

Nesta seção é apresentada a forma como os dados foram gerados e o ambiente de teste. Os resultados obtidos pelos métodos de solução são apresentados, também é realizada uma análise de desempenho com a comparação entre as abordagens.

##### 4.1 Geração dos dados

Nos testes computacionais foram consideradas 25 instâncias geradas aleatoriamente. Os valores utilizados para os parâmetros  $J$  (número de itens),  $T$  (número de períodos),  $N$  (número de pedidos),  $Cut$  (relação entre a capacidade e a demanda) e  $Wdw$  (tamanho da janela de tempo para entrega dos pedidos) estão apresentados na Tabela 1.

Tabela 1: Parâmetros adotados na geração de dados

Parâmetros	Variação
Número de itens ( $J$ )	{25,50}
Número de períodos ( $T$ )	{5,10,15}
Número de pedidos ( $N$ )	{30,50,70,90}
$Cut$	0.8
Tamanho da janela de tempo do atendimento dos pedidos ( $Wdw$ )	{1,2,3,4}
Capacidade produtiva em cada período ( $C_t$ )	equação 18

Assim como em James e Almada-Lobo [2011], o parâmetro  $Cut$  foi introduzido para representar a relação entre a capacidade produtiva e as demandas por itens. Dessa forma, para cada período  $t$ , a capacidade produtiva ( $C_t$ ) foi definida conforme a equação abaixo, onde  $S$  é o conjunto dos pedidos cuja janela de entrega contém  $t$ , isto é,  $S = \{n \mid F_n \leq t \leq L_n\}$ .

$$C_t = \left\lceil \frac{Cut \cdot (\sum_{n \in S} \sum_{j=1}^J a_j \cdot q_{jn})}{Wdw} \right\rceil, \quad (18)$$

onde

$$\overline{Wdw} = \left\lceil \frac{(\sum_{n=1}^N L_n - F_n + 1)}{N} \right\rceil.$$

Outras características importantes para a geração das instâncias são:

- Cada pedido é composto por:  $\{1, \dots, \lceil \frac{J}{2} \rceil\}$ ;
- Quantidade de cada item em cada pedido sorteada entre 40 e 59 e fixada em zero nos demais itens. (Sortear, primeiro o número de produtos que vai compor o pedido. Depois, sortear as demandas para esses itens entre 40 e 59 e fixar as outras demandas em zero);
- Tempo de *setup*: {2, ..., 10};
- Custo de *setup*:  $sc_{ij} = \theta st_{ij}$ , onde  $\theta = 50$ ;
- Tempo de produção de cada item: fixe em 1;
- Tempo de vida (*shelf-life*) do produto no estoque: sorteie no conjunto {1, 2, 3};
- Custo de estoque: sorteie no conjunto {2, ..., 9};
- Lucro dos pedidos:  $500 + 2^*(\text{quantidade total de itens no pedido})$ .

#### 4.2 Ambiente de teste

Os resultados das instâncias executadas utilizando as heurísticas foram obtidos por meio de algoritmos implementados em linguagem C++ com auxílio da ferramenta *Concert Technology* com parametrização padrão. Os algoritmos foram executados em um computador equipado com 2 Processadores Intel Xeon E5-2680v2 de 2.8 GHz e com 128 GB DDR3 1866MHz de memória RAM.

#### 4.3 Resultados

Esta subseção apresenta alguns resultados obtidos pelas abordagens de solução. Utilizamos 3 abordagens de solução, o CPLEX utilizando *branch-and-cut* e as heurísticas construtivas 1 e 2 que foram apresentadas na Seção 3. Definimos C como classe, P como o conjunto de parâmetros em cada classe. O tempo de execução é denotado por TP, e o valor da função objetivo em cada abordagem é FO. O limite superior obtido em pelo CPLEX foi chamado de LS. O GAP é definido como  $GAP = 100 \cdot (\frac{LS - FO}{LS})$ . O tempo máximo de execução (TM) foi definido sendo 3600 segundos.

Tabela 2: Resultados das Abordagens de Solução

C	P	CPLEX				Heurística construtiva 1			Heurística construtiva 2		
		FO	LS	TP	GAP(%)	FO	TP	GAP(%)	FO	TP	GAP(%)
1	J = 25	20680	21574.55	TM	<u>4.14</u>	14979	2199.88	30.57	18338	1275.28	15.00
	T = 5	21828	21830.18	3459.58	<u>0.01</u>	16834	910.35	22.88	19936	1400.98	8.67
	N = 30	23828	25877.04	TM	<u>7.92</u>	18851	1737.05	27.15	22068	1523.33	14.72
	Cut = 0.8	20400	20402.04	2953.40	<u>0.01</u>	15636	1567.26	23.36	18556	1532.84	9.05
	Média	19048	19988.22	TM	<u>4.70</u>	13630	1006.49	31.81	18022	918.32	9.83
2	J = 50	25763	85157.35	TM	69.92	41204	TM	<u>51.90</u>	25538	TM	70.19
	T = 5	32733	91295.01	TM	<u>64.47</u>	37220	TM	<u>59.60</u>	46406	TM	<u>49.63</u>
	N = 70	12062	86408.58	TM	86.20	37357	TM	<u>57.26</u>	28662	TM	67.21
	Cut = 0.8	27215	96690.29	TM	72.15	47186	TM	<u>51.71</u>	38592	TM	60.51
	Média	37795	92292.54	TM	<u>59.48</u>	43043	TM	<u>53.85</u>	34834	TM	62.65
3	J = 50	5998	29347.73	TM	79.56	6628	2758.17	77.41	7391	3016.13	<u>74.81</u>
	T = 10	6206	22549.59	TM	73.39	8357	2851.04	<u>64.16</u>	6710	2207.76	71.22
	N = 30	4817	28426.27	TM	83.05	5867	3108.83	<u>79.36</u>	4494	2749.05	84.19
	Cut = 0.8	4042	25674.98	TM	84.64	8006	2798.43	<u>69.58</u>	6299	2681.04	76.06
	Média	6065	24827.90	TM	<u>76.23</u>	5282	2761.96	79.30	3816	2626.95	85.04
4	J = 50	5425.60	26165.29	TM	79.37	6828	2855.68	73.96	5742	2656.18	78.26
	T = 10	7996	76189.95	TM	<u>89.50</u>	17036	TM	<u>77.64</u>	28023	TM	<u>63.22</u>
	N = 70	18279	84339.38	TM	<u>78.32</u>	16775	TM	<u>80.11</u>	11344	TM	<u>86.55</u>
	Cut = 0.8	9600	80229.98	TM	<u>88.03</u>	24541	TM	<u>69.41</u>	14957	TM	81.35
	Média	15504	88225.73	TM	82.42	18257	TM	<u>79.30</u>	13382	TM	84.83
5	J = 50	6198	84859.02	TM	92.69	16096	TM	81.03	23546	TM	72.25
	T = 15	7521.40	109007.13	TM	86.19	18541	TM	77.50	18250.40	TM	77.64
	N = 90	5059	105259.74	TM	95.19	17262	TM	<u>83.60</u>	12706	TM	87.93
	Cut = 0.8	14899	110937.44	TM	<u>86.57</u>	21920	TM	<u>80.24</u>	13529	TM	87.80
	Média	3954	110749.01	TM	<u>96.43</u>	14002	TM	<u>87.35</u>	12416	TM	88.79
	6010	6010	109649.22	TM	<u>94.52</u>	16868	TM	<u>84.61</u>	10536	TM	90.39
	Média	7685	108440.25	TM	92.91	21222	TM	<u>80.43</u>	11160	TM	89.71
<b>Média Geral</b>		14546.56	66048.87	3568.51	66.49	20162.36	3027.97	63.34	18050.44	2957.26	64.02

O *branch-and-cut* do resolvedor Cplex obteve solução de qualidade superior do que as heurísticas em 6 das 25 instâncias, mas atingiu o tempo máximo em 23 instâncias, com tempo médio de 3568,51 segundos. A heurística construtiva 1 obteve solução de melhor qualidade em 14 instâncias e tempo médio foi de 3027,97 segundos. A heurística construtiva 2 obteve solução de melhor qualidade em apenas 4 instâncias e tempo médio geral de 2957,26 segundos. Os valores dos melhores desvios estão sublinhados na Tabela 2.

#### 4.4 Análise de desempenho

A seguir fazemos uma análise de desempenho do modelo (1)-(17) considerando as instâncias geradas, bem como do desempenho das heurísticas construtivas 1 e 2. O gráfico da Figura 1 foi construído de acordo com Dolan e Moré [2002]. Para a interpretação deste gráfico, considere  $NP$  sendo o conjunto das 25 instâncias analisadas,  $S$  o conjunto dos métodos que estão sendo testados e comparados e  $t$  a métrica de avaliação. Além disso, considere:

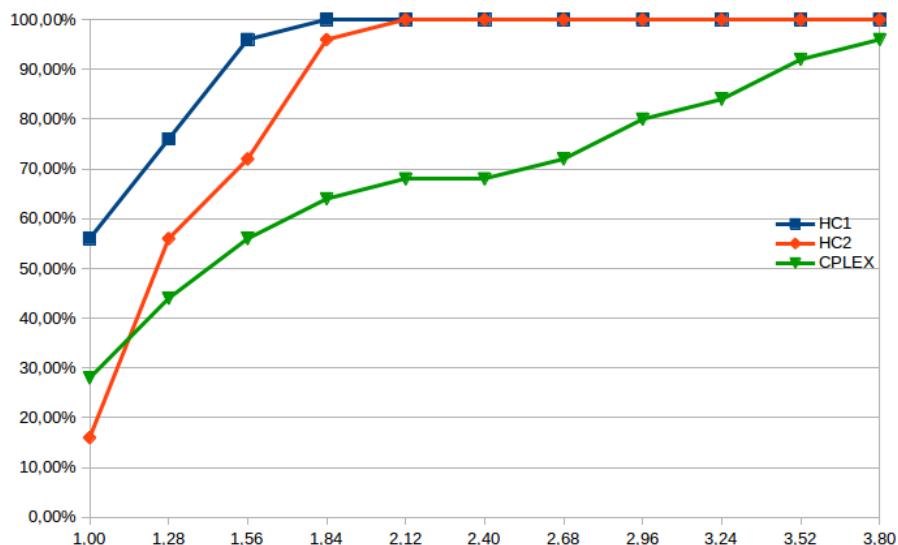
- a comparação do limitante primal (LP). O gráfico da Figura 1 mostra as curvas de desempenho referente aos valores dos limitantes primais encontrados por cada uma das abordagens. Deste modo,  $t_{ps}$  denota o valor LP do método de solução  $s$  ao resolver a instância  $p$  e, definimos a qualidade do limitante primal de um método de solução particular  $s$  para uma dada instância  $p$  como:

$$r_{ps} = \frac{\max\{t_{ps}: s \in S\}}{t_{ps}}.$$

Com estes valores, a probabilidade de que cada modelagem seja melhor ou igual a  $\tau$  pode ser calculada por:  $\frac{|p \in P: r_{ps} \leq \tau|}{NP}$ , onde  $NP$  é o número de instâncias consideradas.

No gráfico da Figura 1, o eixo das abscissas representa os valores de  $\tau$  e o eixo das ordenadas representa as probabilidades de obter uma solução cuja qualidade é melhor ou igual a  $\tau$ . Deste modo, o valor  $\tau = 1$  indica a porcentagem de instâncias nas quais a heurística  $s$  forneceu a melhor solução. As instâncias que estão entre 1 e 1,28 representam os 28% melhores resultados dos métodos de solução. A ideia é a mesma para outros valores. Na análise geral, a heurística construtiva 1, HC 1, obteve o melhor desempenho na obtenção de soluções de melhor qualidade.

Figura 1: Análise de desempenho das heurísticas construtivas



#### 5. Conclusões

Os testes computacionais mostraram que as heurísticas encontraram soluções de boa qualidade em um tempo computacional razoável. Foi observado que para as classes de instâncias com um número inferior de itens e pedidos (Classe 1), o *branch-and-cut* do CPLEX obteve soluções de melhor qualidade do que as heurísticas. Também foi observado que mesmo nas instâncias que o *branch-and-cut* obteve soluções de qualidade superior, as heurísticas obtiveram boas soluções.

Para melhorar os resultados obtidos pela heurísticas podem ser utilizadas outras heurísticas combinadas, como por exemplo, uma heurística de melhoria do tipo *fix-and-optimize*. Para melhorar os resultados no método exato, outras formulações para o problema podem ser investigadas. Trabalhos futuros também podem explorar outras estratégias de decomposição das variáveis via heurística.

### Agradecimentos

Os autores agradecem o apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001. Esta pesquisa foi desenvolvida com utilização dos recursos computacionais do Centro de Ciências Matemáticas Aplicadas à Indústria (CeMEAI), financiados pela FAPESP (proc. 2013/07375-0).

### Referências

- Brahimi, N., Aouam, T., e Aghezzaf, E. (2015a). Integrating order acceptance decisions with flexible due dates in a production planning model with load-dependent lead times. *International Journal of Production Research*, 53(12):3810–3822.
- Brahimi, N., Aouam, T., e Aghezzaf, E. (2015b). Production planning and order acceptance: an integrated model with flexible due dates. In *Xème Conférence Internationale : Conception et Production Intégrées*.
- Brahimi, N., Aouam, T., Geryl, K., e Kumar, K. . (2018). Production planning with order acceptance and demand uncertainty. *Computers and Operations Research*, (91):145–159.
- Copil, K., Wörbelauer, M., Meyr, H., e Tempelmeier, H. (2017). Simultaneous lotsizing and scheduling problems: a classification and review of models. *OR Spectrum*.
- Dolan, D., L e Moré, J. (2002). Benchmarking optimization software with performance profiles. *Math. Program*, (91):201–213.
- Furtado, G. S., M (2012). O planejamento da produção de pedidos em fundições de pequeno porte. Master's thesis, Universidade de São Paulo. URL <http://www.teses.usp.br/teses/disponiveis/55/55134/tde-12062012-161729/pt-br.php>.
- James, R. e Almada-Lobo, B. (2011). Single and parallel machine capacitated lotsizing and scheduling: New iterative mip-based neighborhood search heuristics. *Elsevier*.
- Li, Y., Chu, F., Yang, Z., e Calvo, R. (2016). A production inventory routing planning for perishable food with quality consideration. *IFAC-PapersOnLine*, 49(3):407–412.
- Pochet, L. A., Y.; Wolsey (2006). *Production Planning by Mixed Integer Programming*. New York. URL <https://www.springer.com/series/3182>.
- Seresht, N. e Bijari, M. (2013). Profit maximization in simultaneous lot-sizing and scheduling problem. *Applied Mathematical Modelling*, (37):9516–9523.
- Teixeira, V., Oliveira, W., e Santos, M. (2017). Um problema de dimensionamento e sequenciamento de lotes de produção com gerenciamento da demanda via pedidos e com tempos/custos de preparação dependentes da sequência. *Simpósio Brasileiro de Pesquisa Operacional*.