UNIVERSIDADE DE SÃO PAULO

Instituto de Ciências Matemáticas e de Computação

BRANCH-AND-BOUND ALGORITHMS FOR CAPACITATED LOT-SIZING IN PARALLEL MACHINES

FRANKLINA MARIA BRAGION DE TOLEDO
VINÍCIUS AMARAL ARMENTANO

Nº 89

NOTAS

São Carlos - SP

UNIVERSIDADE DE SÃO PAULO

Instituto de Ciências Matemáticas e de Computação

ISSN 0103-2577

BRANCH-AND-BOUND ALGORITHMS FOR CAPACITATED LOT-SIZING IN PARALLEL MACHINES

FRANKLINA MARIA BRAGION DE TOLEDO
VINÍCIUS AMARAL ARMENTANO

Nº 89

NOTAS

Série Computação

São Carlos – SP
Mar./2007

Branch-and-Bound Algorithms for Capacitated Lot-Sizing in Parallel Machines

Franklina Maria Bragion de Toledo
Departamento de Matemática Aplicada e Estatística
Instituto de Ciências Matemáticas e de Computação
Universidade de São Paulo - Campus de São Carlos
Caixa Posta 668 - 13560-970 - São Carlos, SP – Brasil
fran@icmc.usp.br

Vinícius Amaral Armentano
Faculdade de Engenharia Elétrica e de Computação
Universidade Estadual de Campinas
CP 6101 – 13083-970 – Campinas – SP – Brasil
vinicius@densis.fee.unicamp.br

Abstract

This paper deals with the capacitated lot-sizing problem for the production of several items on unrelated parallel machines over a finite planning horizon. The aim is to find a production plan that minimises the sum of set-up, production and inventory costs without violating capacity constraints. Two branch-and-bound algorithms based on Lagrangian and Linear programming relaxations, respectively, are developed and computational tests are reported.

***Keywords**: Production Planning; Lot-sizing; Branch-and-bound; Parallel machines.*

1. Introduction

The capacitated lot-sizing problem (CLSP) consists of planning the production of several items over a finite horizon of discrete time periods. The problem involves determining how much to produce of each item in order to meet the demand in each period in order to minimise the sum of set-up, production and inventory costs without violating capacity constraints.

The CLSP can model single stage and multistage production structures. Many papers have addressed this problem and surveys are presented by Bahl *et al.* (1987) and Kuik *et al.* (1994). In a recent paper, Drexl and Kimms (1997) discuss several lot sizing and scheduling models other than CLSP.

This paper considers the single stage CLSP in the environment of unrelated parallel machines. All items can be produced on any machine and a set-up time is incurred before starting production. The presence of set-up times makes it difficult to find a solution that satisfies the capacities constraints, let alone an optimal solution (Trigeiro *et al.*, 1989). In fact, Maes *et al.* (1991) have shown that the problem of finding a feasible solution is NP-complete.

Focusing on the literature for the single stage, single machine CLSP with set-up times, few optimal methods have been proposed. A branch-and-bound (B&B) algorithm is proposed by Diaby *et al.* (1992) in which a lower bound is generated by Lagrangian relaxation and subgradient optimisation. For small problems with set-up cost (8 items, 8 periods) and larger problems without set-up cost (99 items, 8 periods) the algorithm obtained solutions within 1% of the lower bound. The optimal cross decomposition method suggested by van Roy (1983) was applied by Souza and Armentano (1994) to solve small problems (15 items, 8 periods) with bounded production variables. Armentano *et al.* (1999) propose a generalised network flow model and develop a branch-and-bound algorithm with a lower bound generated by a linear programming problem, which is solved by a network flow algorithm. Computational tests show that the method is able to solve problems with 25 items and 12 periods within 1% of the lower bound. To deal with larger problems, heuristic methods have been developed by Trigeiro (1989), Lozano *et al.* (1991) and Diaby *et al.* (1992b).

The literature on lot sizing in parallel machines is quite limited. Lasdon and Terjung (1971) propose a heuristic for the Discrete Lot-Sizing and Scheduling Problem (DLSP) for identical parallel machines without set-up time. In the DLSP only one item can be produced per machine and per period and, if so, production uses the full capacity. Carreno (1990) presents a heuristic for the Economic Lot Scheduling Problem (ELSP) for identical parallel machines with set-up times. The ELSP is a continuous time and infinite horizon model with constant demand rate for each item.

More recently, Kang *et al.* (1999) and Meyer (1999) have proposed heuristics for the simultaneous lot-sizing and scheduling on parallel machines. The models in such papers include sequence dependent set-up costs and for this reason there is the need to determine the item production sequence in each period. In this paper, set-up costs and times are sequence independent and thus we are concerned only with the lot-sizing problem. To the best of our knowledge the CLSP for non-identical parallel machines, denoted CLSPP, has not been addressed in the literature, except for the work of Toledo *et al.* (2006) which suggests a Lagrangian heuristic for such a problem.

This paper proposes two B&B algorithms with bounds generated by Lagrangian relaxation and Linear programming (LP) relaxation, respectively. The first algorithm is based on the Lagrangian relaxation of the capacity constraints and subgradient optimisation. The Lagrangian problem decomposes into subproblems, one for each item, which are solved by an extension of the Wagner-Whitin algorithm (1958). The second algorithm is an extension of the algorithm proposed by Armentano *et al.* (1999) for the single machine case. Computational experiments show that for some problem classes the LP-based B&B performs better. The paper is organised as follows. Section 2 contains the problem formulation. The Lagrangian-based and the LP-based B&B algorithms are described in Sections 3 and 4, respectively. Computational results are presented in Section 5 and concluding remarks are given in Section 6.

2. Problem Formulation

Consider a planning horizon of T periods with m machines available for processing n items. The objective is to determine a production plan for the items, which meets the demand of items without backorder and minimises the set-up, production and inventory costs. The amount of production in period t on machine j of the item i is denote by x_{ijt}, and one unit of an item incurs a production cost c_{ijt}, and uses b_{ij} units of capacity. Each time that production for an item begins in a period on a machine, a set-up must take place and the existence of a set-up is indicated by a binary variable y_{ijt}. A set-up entails a set-up time, f_{ij}, on the capacity constraint resource (C_{jt}) and a set-up cost, s_{ijt}. End of period inventory, I_{it}, incurs unitary cost, h_{it} The mathematical formulation of the problem is:

The mathematical formulation of the problem is:

$$
\begin{array}{llll}
(CLSPP) & \textit{Minimize} & \sum_{t=1}^{T}\sum_{j=1}^{m}\sum_{i=1}^{n}\left(s_{ijt}y_{ijt}+c_{ijt}x_{ijt}\right)+\sum_{t=1}^{T}\sum_{i=1}^{n}h_{it}I_{it} & \\
& \textit{subject to} & \sum_{j=1}^{m}x_{ijt}+I_{i,t-1}-I_{it}=d_{it} & i=1,\ldots,n;\quad t=1,\ldots,T; \quad (1)\\
& & \sum_{i=1}^{n}\left(b_{ij}x_{ijt}+f_{ij}y_{ijt}\right)\le C_{jt} & j=1,\ldots,m;\quad t=1,\ldots,T; \quad (2)\\
& & x_{ijt}\le M\,y_{ijt} & i=1,\ldots,n;\quad j=1,\ldots,m;\quad t=1,\ldots,T; \quad (3)\\
& & y_{ijt}\in\{0,1\},\ x_{ijt}\ge 0, I_{it}\ge 0 & i=1,\ldots,n;\quad j=1,\ldots,m;\quad t=1,\ldots,T. \quad (4)
\end{array}
$$

Constraints (1) represent the inventory balance equations (d_{it} is the demand). Constraints (2) indicate that the amount of capacity used for production is limited. Constraints (3) ensure the incidence of a set-up cost and a set-up time when x_{ijt} is positive, and where M denotes an upper bound on x_{ijt}. Backlogging is not allowed as indicated by the nonnegative restriction on I_{it} in (4).

3. The Lagrangian-based Branch-and-Bound

The B&B Lagrangian algorithm suggested in this section uses at each node of the tree a lower bound generated by Lagrangian relaxation and subgradient optimisation.

3.1. Lagrangian Relaxation

At each node of B&B tree, consider the Lagrangian relaxation with respect to the capacity constraints (2). For Lagrangian multipliers $\mu_{jt} \geq 0,\ \ j = 1,\ldots,m;\ \ t = 1,\ldots,T$, the resulting Lagrangian problem is given by

$$g(\mu_{jt}) = \underset{x_{ijt}, I_{it}, y_{ijt}}{Minimize} \ \ \sum_{t=1}^{T}\sum_{j=1}^{m}\sum_{i=1}^{n}\left(s_{ijt}y_{ijt} + c_{ijt}x_{ijt}\right) + \sum_{t=1}^{T}\sum_{i=1}^{n} h_{it}I_{it}$$
$$+ \sum_{t=1}^{T}\sum_{j=1}^{m}\mu_{jt}\left[\sum_{i=1}^{n}\left(b_{ij}x_{ijt} + f_{ij}y_{ijt}\right) - C_{jt}\right]$$

$$subject\ to \quad (1), (3)\ and\ (4).$$

The Lagrangian problem decomposes into the sum of the constant $-\sum_{t=1}^{T}\sum_{j=1}^{m}\mu_{jt}C_{jt}$ and n independent subproblems, one for each item, with the form

$$g_i(\mu_{jt}) = \underset{x_{ijt}, y_{ijt}, I_{it}}{Minimize} \ \ \sum_{t=1}^{T}\sum_{j=1}^{m}\left[\left(s_{ijt} + \ _{jt}f_{ij}\right)y_{ijt} + \left(c_{ijt} + \ _{jt}b_{ij}\right)x_{ijt}\right] + \sum_{t=1}^{T} h_{it}I_{it}$$

$$\begin{aligned} subject\ to \quad & \sum_{j=1}^{m} x_{ijt} + I_{i,t-1} - I_{it} = d_{it} & & t = 1,\ldots,T; \\ & x_{ijt} \leq M\, y_{ijt} & j = 1,\ldots,m; \ \ & t = 1,\ldots,T; \\ & y_{ijt} \in \{0,1\},\ x_{ijt} \geq 0, I_{it} \geq 0 & j = 1,\ldots,m; \ \ & t = 1,\ldots,T. \end{aligned}$$

Note that $g_i\ (\mu_{jt})$ is a minimisation problem with a concave objective function and linear constraints. It follows that the minimum occurs in an extreme point, which in the case of a polyhedral set has at most the number of constraints (T) nonzero variables. If $d_t \geq 0$, $t = 1,\ldots,T$, then either $I_{t-1} > 0$ or $x_{ijt} > 0$ for some $j \in \{1,\ldots,m\}$. Thus, every extreme point must satisfy the following property

$$I_{t-1}x_{ijt} = 0 \qquad j = 1,\ldots,m; \quad t = 1,\ldots,T.$$

The characterisation of an extreme point above is a generalisation of the single machine case which was stated by Wagner and Whitin (1958). Armentano and Toledo (1997) extended the single machine dynamic programming algorithms proposed by Evans (1985) and Wagelmans *et al.* (1992) and reported computational results for the parallel machine case.

The subgradient method was used to maximise the dual function $g_i(\mu_{jt})$ over $\mu_{jt} \geq 0$, $j = 1,...,m$, $t = 1,...,T$. The subgradient algorithm proposed by Held *et al.* (1974) was implemented and 100 steps of the algorithm are applied at each node of B&B tree.

3.2. Node Selection

An active node is chosen for examination if it has the least lower bound. This adaptive rule stops being applied when the tree lower bound remains constant for 10 iterations. In this case we apply the following strategy suggested in (Nemhauser and Wolsey, 1988) which combines the depth-first search with the least lower bound rule. When a node is branched into two descendant nodes, we choose the one with the least lower bound. If such a node is fathomed we choose the other descendant node and in case it is also fathomed, we select the node in the tree with the least lower bound.

3.3.Branching Variable Selection

We used a hybrid strategy for choosing a variable to define two descendant nodes from an active node. The B&B tree is started by using a priority rule which chooses the variable y_{ijt} corresponding to the maximum production cost ($s_{ijt} + c_{ijt} d_{it}$). This rule defines two subproblems with the following characteristics: for y_{ijt} fixed at zero, we generate a subproblem with the aim of reducing the total production cost; for y_{ijt} fixed at one, the objective is to fathom the subproblem by bound. This rule showed a good performance when the incumbent solution value is distant from the node lower bound. However, when the relative gap between these two values falls below 10% we noticed that the use of a penalty to define the branching variable is more effective. The exact penalty used here is an extension of that proposed by Diaby *et al.* (1992a) and corresponds to a degradation in the lower bound of a node when a binary variable is forced to assume a certain value. Let y_{ijt} be a free variable that assumes the value 0 or 1 in the optimal solution of the Lagrangian problem at a given node. The penalty corresponds to an increase in the optimal value of the Lagrangian problem when the problem is solved with y_{ijt} constrained to assume the opposite value. The free variable associated with the largest degradation is then chosen as the branching variable in order to favour the fathoming of nodes by bound.

3.4. Initial Solution

An initial feasible solution for the B&B is provided by a Lagrangian heuristic developed by Toledo and Armentano (2006). At every 20 nodes a smoothing heuristic is applied to the Lagrangian solution corresponding to the last subgradient iteration on the node in an attempt to obtain a capacity feasible solution which updates the incumbent solution. This heuristic is based on production shifts between periods and machines.

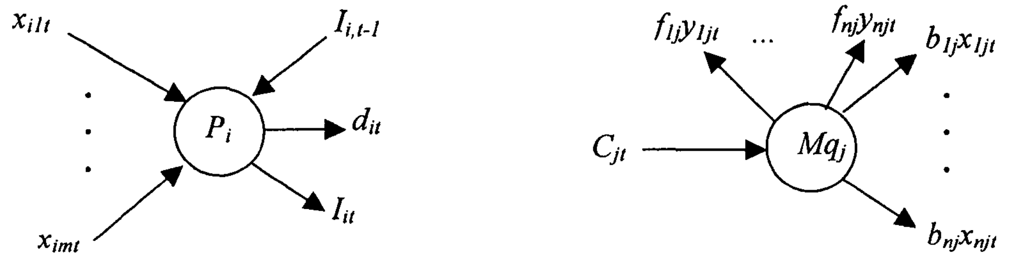
4. The LP-based Branch-and-Bound

The network flow model for the CLSPP that is described next is an extension of the model proposed by Armentano *et al.* (1999) for the single machine case. The authors suggest a B&B method with a lower bound generated by linear programming relaxation, which is solved by a network flow algorithm.

4.1. Network Flow Model

For a given item i and a period t, each equation in (1) can be represented as a production node (P_i). The incoming arcs of this node represent the production amount on each machine and the inventory at the end of period t-1. The outgoing arcs are the demand and the inventory at end of period t (see Figure 1a).

The capacity constraint in the each period t and each machine j can also be represented by a machine node (Mq_j). The incoming arc is the capacity of the machine and the outgoing arcs represent the setup time and the production time of the items (see Figure 1b).



(a) (b)

Figure 1. Network representation of constraints (1) and (2).

The network, which represents the mathematical model, contains all the production and machine nodes. The flow in the production arcs that enter the node P_i is expressed in units of items while the flow of the production arcs that leave the node Mq_j is expressed in units of time. In order to link these nodes, the flow in the production arcs is multiplied by a conversion factor $1/b_{ij}$, which is called the gain of the arc. This characterizes a generalized network (Jensen and Barnes, 1980), which is illustrated in Figure 2. In the network, four parameters [a,b,c,d] are associated with each arc, which represent the arc conversion factor, the flow lower bound, the flow upper bound and the unit cost, respectively.

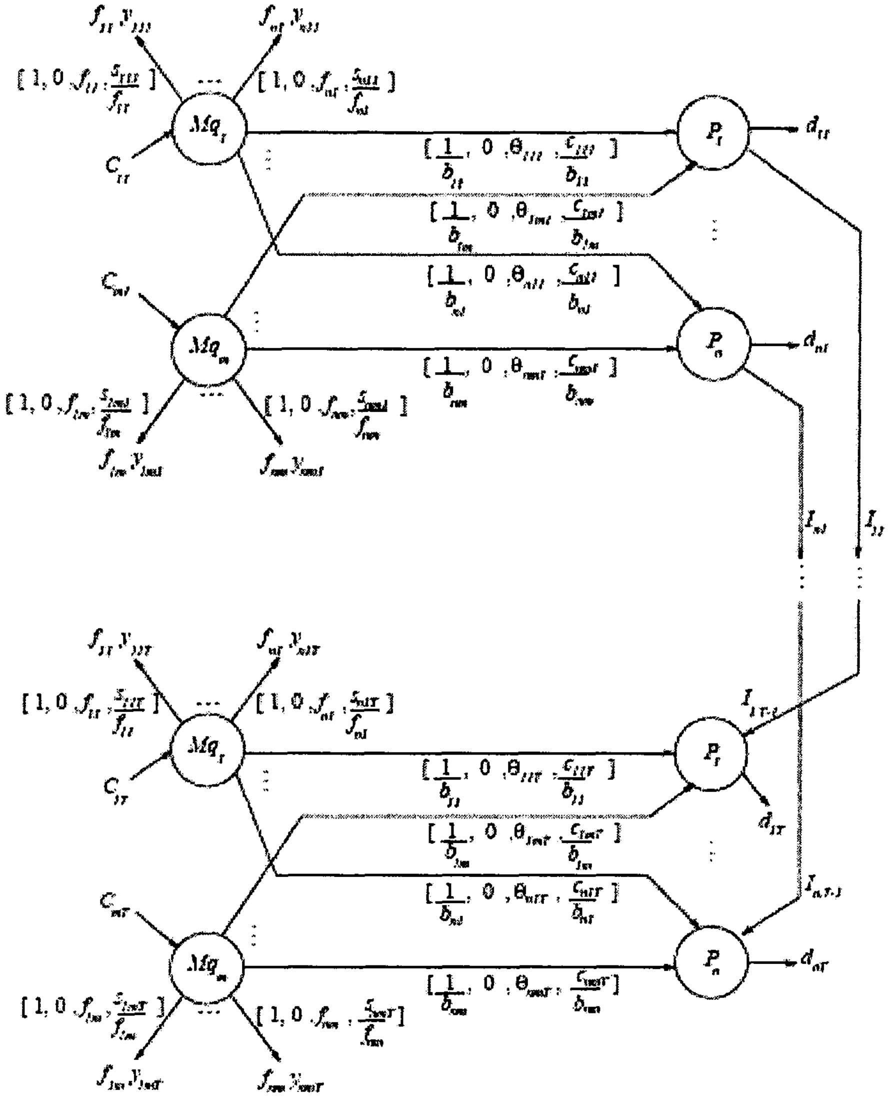


Figure 2. The network of the model *CLSPP*.

4.2. Solution Method

At each node of the B&B method some variables are free and the remainder are fixed in 0 or 1. The mathematical model at each node ($\overline{CLSPP}$) includes the relaxed production time arcs, represented by dotted lines, and the arcs of setup time and production time associated with the free and fixed variables, respectively. Let x^r_{ijt} represent the amount of production corresponding to the free binary variables, $\psi_{ijt} = f_{ij} y_{ijt}$ and $w_{ijt} = \dfrac{1}{b_{ij} + f_{ij}\dfrac{b_{ij}}{M}}$. The model is given by

$$(\overline{\text{CLSPP}})\quad \textit{Minimize}\quad \sum_{t=1}^{T}\sum_{j=1}^{m}\sum_{i=1}^{n}\left(\frac{\bar{s}_{ijt}}{f_{ij}}\psi_{ijt}+\bar{c}_{ijt}x_{ijt}+\bar{c}^{r}_{ijt}x^{r}_{ijt}\right)+\sum_{t=1}^{T}\sum_{i=1}^{n}h_{it}I_{it}$$

$$\textit{subject to}\quad I_{i,t-1}+\sum_{j=1}^{m}x_{ijt}+\sum_{j=1}^{m}x^{r}_{ijt}-I_{it}=d_{it}\qquad i=1,\dots,n;\quad t=1,\dots,T;\quad (5)$$

$$\sum_{i=1}^{n}\left(b_{ij}x_{ijt}+\psi_{ijt}+\frac{1}{w_{ijt}}x^{r}_{ijt}\right)\le C_{jt}\qquad j=1,\dots,m;\quad t=1,\dots,T;\quad (6)$$

$$b_{ij}x_{ijt}\le M\qquad i=1,\dots,n;\quad j=1,\dots,m;\quad t=1,\dots,T;\quad (7)$$

$$w_{ijt}x^{r}_{ijt}\le M\qquad i=1,\dots,n;\quad j=1,\dots,m;\quad t=1,\dots,T;\quad (8)$$

$$x^{r}_{ijt}\ge 0,\ x_{ijt}\ge 0, I_{it}\ge 0\qquad i=1,\dots,n;\quad j=1,\dots,m;\quad t=1,\dots,T.\quad (9)$$

Model $\overline{\text{CLSPP}}$ can be represented by the generalized network shown in Figure 3. Since the binary variables y_{ijt} are not represented in this network, a large positive penalty (B) is assigned to the arc cost where the flow must be set to zero and a large negative penalty (-B) is assigned to the arc cost where the flow must be positive. The overline in model $\overline{\text{CLSPP}}$ is used to indicate that the parameters have different values according to the status of the variable y_{ijt}. There are three possible situations regarding flow values that correspond to $y_{ijt}=1$, $y_{ijt}=0$ and $0\le y_{ijt}\le 1$. When $y_{ijt}=1$, the item *i* can be produced on machine *j* in period *t* and the flow values should satisfy $x_{ijt}\ge 0$, $x^{r}_{ijt}=0$ and $\psi_{ijt}=f_{ij}$. Correspondingly, the cost values should be set to $\bar{c}_{ijt}=c_{ijt}$, $\bar{c}^{r}_{ijt}=\text{B}$ and $\bar{s}_{ijt}=-\text{B}$. The remaining situations follow similar reasoning (Armentano *et al.*, 1999).

The generalized network $\overline{CLSPP}$ is solved by the network flow algorithm described in (Jensen and Barnes, 1980).

4.3. Outline of the B&B Implementation

The strategy for node selection is identical to that described in the Lagrangian based B&B. Several priority rules for branching variables were adapted from the rules proposed in (Evans, 1985b) and (Armentano *et al.*, 1999). Computational tests showed that the best rule consists of choosing the variable y_{ijt} associated to the largest setup cost (s_{ijt}). This rule defines two subproblems: for y_{ijt} fixed at zero, we generate a subproblem with the objective of reducing the set-up cost; for y_{ijt} fixed at one, the aim is to fathom the subproblem by bound.

The initial solution for the B&B algorithm is also given by the heuristic presented in (Toledo and Armentano, 2006).

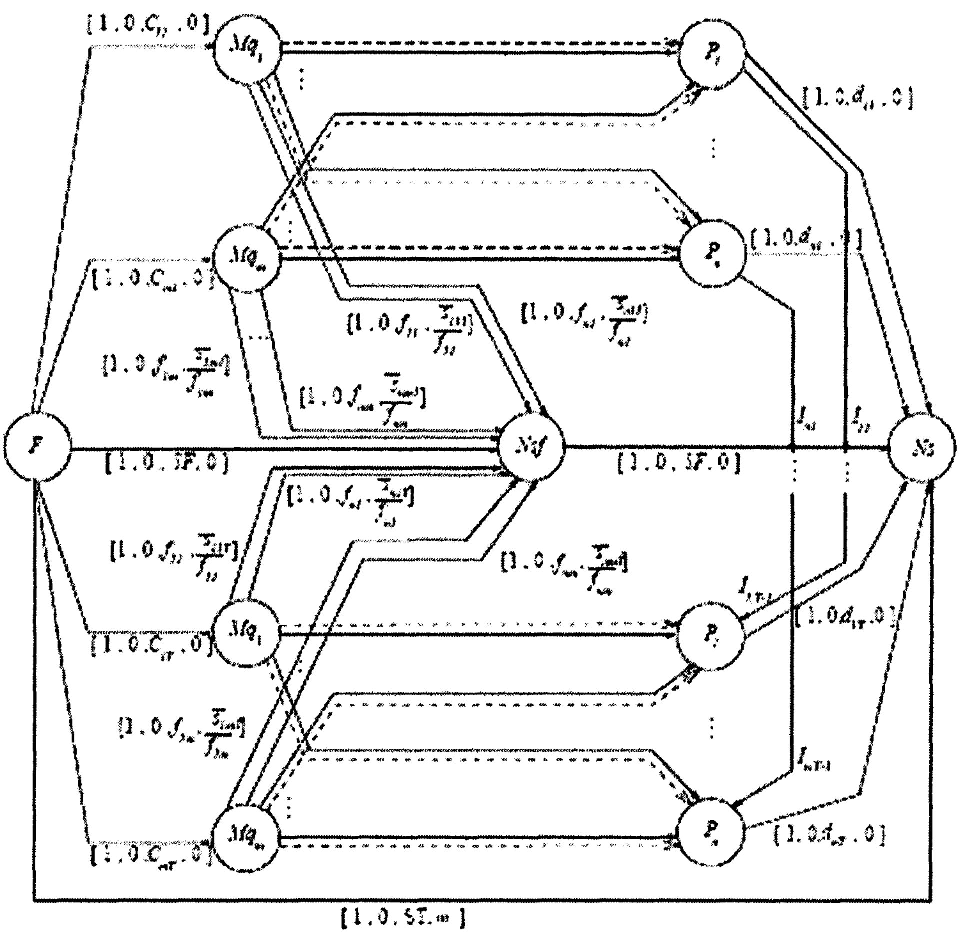


Figure 3. The network of the model $\overline{CLSPP}$.

5. Computational Tests

The B&B algorithms were programmed in C and run on a SUN SPARCstation ULTRA 1. The algorithms were tested on a set of 320 problems and analysed according to three factors: dimension, set-up cost and capacity. The dimension, represented by the number of items (n), machines (m) and periods (T), is indicated on Table 1. Set-up cost can be high or low and capacity can be normal or tight. For each combination of the three factors 10 problems were generated with parameters constant over time. The parameters were generated in an interval [a, b] with uniform distribution, denoted U[a, b]:

- unit production cost (c_{ij}): U[1.5, 2.5]
- set-up cost (s_{ij}): U[5.0, 95.0]; low set-up costs are generated in this interval and high set-up costs are obtained by multiplying low set-up costs by 10.
- unit inventory cost (h_i): U[0.2, 0.4]
- unit processing time (b_{ij}): U[1.0, 5.0]
- set-up time (f_{ij}): U[10.0, 50.0]
- demand (d_{it}): U[0, 180]

Table 1. Dimension of problems.

n	*m*	*T*
4	2	4
		6
		8
	3	4
		6
6	2	4
		6
	3	4

Capacity was generated by splitting the demand of each item in each period among the machines and applying the lot-for-lot policy to each machine. The capacity (*Cap*) is then, the average over the number of machines and periods.

$$Cap = \frac{\sum_{t=1}^{T}\sum_{j=1}^{m}\sum_{i=1}^{n}\left(\frac{d_{it}}{m}b_{ij} + f_{ij}\right)}{mT}$$

Computational tests indicated that *Cap* should be adjusted with the number of machines. For the number of machines considered in this paper the following expression was used to generate the normal capacity

$$C_{jt} = \left(1.4 - \frac{m}{10}\right)Cap.$$

Tight capacity is obtained by reducing C_{jt} by 10%.

The stopping criterion for the algorithms is the running time of 1800 CPU seconds or 100,000 nodes of the B&B tree, and a solution is declared optimal when the relative gap between the incumbent solution value and the lower bound is less than or equal to 0.5%.

Table 2 shows the performance of the LP-based B&B algorithm and the Lagrangian-based B&B algorithm relative to the average time in seconds, average number of nodes (*NN*) and the number of optimal solutions (*NO*) which were obtained by the algorithms for the stopping criterion described above.

Regarding the number of optimal solutions, the LP B&B algorithm shows a better performance, obtaining the optimal solution for 83% and 74% of the problems with normal and tight capacity, respectively, while the Lagrangian B&B algorithm found the optimal solution for 78% and 67% of the problems. For problems with low set-up cost the LP and Lagrangian B&B algorithms present a similar behaviour and for high set-up cost the first algorithm is superior (84% against 73%). With respect to the dimension, it can be noted that the number of machines is the parameter that mostly affect the behaviour of the algorithms.

The Lagrangian B&B algorithm outperforms the LP B&B algorithm for problems with three machines, but when two machines are available the opposite occurs. The Lagrangian B&B algorithm explores a much smaller number of nodes than the LP B&B algorithm, however the time required to solve a node is much larger for the first algorithm.

Table 2. Computational Results.

C/S [a]	*T*	*m*	*n*	*LP Algorithm*			*Lagrangian Algorithm*		
				Time	*NN*	*NO*	*Time*	*NN*	*NO*
N/L	4	2	4	3	826	10	11	373	10
			6	191	17332	10	15	337	10
			8	396	35439	5	7	96	10
		3	4	8	1676	10	232	5506	6
			6	302	28473	6	281	4722	6
	6	2	4	228	18208	10	57	1094	9
			6	652	52944	1	190	2483	8
		3	4	565	29311	8	1384	19343	1
T/L	4	2	4	7	1433	10	32	408	10
			6	305	29211	9	386	8590	9
			8	403	31809	5	15	247	10
		3	4	23	3474	10	357	8192	4
			6	297	30397	5	1251	20920	4
	6	2	4	332	19463	10	116	2292	8
			6	-----	-----	0	199	2709	7
		3	4	46	4823	6	-----	---	0
N/H	4	2	4	2	469	10	13	479	10
			6	39	5396	10	41	944	10
			8	325	24596	9	89	1577	10
		3	4	7	1209	10	173	4348	10
			6	221	17520	9	364	6214	8
	6	2	4	23	3262	10	349	6747	9
			6	538	32946	4	174	2355	6
		3	4	266	19823	10	1142	16131	2
T/H	4	2	4	3	629	10	22	740	10
			6	68	7605	9	76	1801	9
			8	691	35804	7	375	6753	10
		3	4	14	2183	10	307	7604	6
			6	182	14433	8	588	9970	8
	6	2	4	32	3840	10	518	10314	9
			6	76	7474	1	577	7834	2
		3	4	202	9647	8	1444	20410	2

[a] (C) Capacity – (N) Normal (T) Tight (S) Set-up Cost - (L) Low (H) High

6. Conclusions

This paper presented Lagrangian-based and LP-based branch-and-bound algorithms for the capacitated lot-sizing problem that involves the production of the several items on parallel machines. The performance of the algorithms was satisfactory, since for most of the problems tested the optimal solution was founded in reasonable computational time (less than 15 minutes). Although the Lagrangian relaxation with subgradiente optimisation in general provides better lower bounds than Linear relaxation, the last presents better results for some problem classes.

Acknowledgments

This research was partially funded by the Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), Fundação Coordenação de Aperfeiçoamento de Pessoal de Ensino Superior (CAPES) and Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP).

References

1. Armentano, V. A. and Toledo, F. M. B. Optimal Algorithms for the parallel machine lot sizing problem, Pesquisa Operacional 1997; 17: 137-149.
2. Armentano, V. A., França, P. M. and Toledo F. M. B. A network flow model for the capacitated lot- sizing problem, OMEGA 1999; 27(2): 275-284.
3. Bahl, H. C., Ritzman, L. P. and Gupta, J. N. D. Determining lot sizes and resources requirements: a review, Operations Research 1987; 35(3): 329-345.
4. Carreno, J.J. Economic lot scheduling for multiple products on parallel identical processors, Management Science 1990; 36(3): 348-358.
5. Diaby, M., Bahl, H. C., Karwan, M. H. and Zionts, S. A lagrangean relaxation approach for very-large-scale capacitated lot sizing, Management Science 1992a; 38(9): 1329-1340.
6. Diaby, M., Bahl, H. C., Karwan, M. H. and Zionts, S. Capacitated lot-sizing and scheduling by Lagrangean relaxation, European Journal of Operational Research 1992b; 59: 444-458.
7. Drexl, A. and Kimms, A. Lot sizing and scheduling - survey and extensions, European Journal of Operational Research 1997; 99: 221-235.
8. Evans, J.R. An efficient implementation of the Wagner-Whitin algorithm for dynamic lot-sizing, Journal of Operational Management 1985a; 5: 229-235.
9. Evans, J.R., Network-based optimization algorithms for the capacitated multi-item lot sizing problem, Computers and Industrial Engineering 1985b; 9: 297-305.
10. Held, M., Wolfe, P. and Crowder, P. Validation of subgradient optimization, Mathematical Programming 1974; 6: 62-88.

11. Jensen, P.A. and Barnes, J. W. Network flow programming. John Wiley Sons 1980.
12. Kang, S., Malik, K. and Thomas, L. J. Lotsizing and Scheduling on Parallel Machines with Sequence-Dependent Setup Costs, Management Science 1999; 45: 273-289.
13. Kuik, R., Salomon, M., and van Wassenhove, L. N. Batching decisions: structure and models, European Journal of Operational Research 1994; 75: 243-263.
14. Lasdon, L. S. and Terjung, R. C. An efficient algorithm for multi-item scheduling, Operations Research 1971; 19(4): 946-969.
15. Lozano, S., Larraneta, J. and Onieva, L. Primal - dual approach to the single level capacitated lot-sizing problem, European Journal of Operational Research 1991; 51: 354-366.
16. Maes, J., McClain, J. O. and Van Wassenhove, L. N. Multilevel capacitated lotsizing and complexity, European Journal of Operational Research 1991; 53: 131-148.
17. Meyer, H. Simultaneous lotsizing and scheduling for parallel production lines, Technical Paper 1999, Lehrstuhl für Producktion und Logistik, Universität Ausgsburg, 86.135, Ausgsburg - Germany.
18. Nemhauser, G.L., Wolsey, L.A. Integer and Combinatorial Optimization. John Wiley & Sons 1988.
19. Souza, K. X. S. and Armentano, V. A. Multi-item lot-sizing by a cross decomposition based algorithm, Annals of Operations Research 1994; 50: 557-574.
20. Toledo, F.M.B. and Armentano, V.A. A Lagrangean-based Heuristic for Capacitated Lot-Sizing in Parallel Machines, European Journal of Operational Research 2006; 175: 1070-1083.
21. Trigeiro, W. W. Capacitated lot sizing with set-up times, Management Science 1989; 35(3): 353-366.
22. van Roy, T. J. Cross decomposition for mixed integer programming, Mathematical Programming 1983; 25: 46-63.
23. Wagelmans, A., Van Hoesel, S. and Kolen, A. Economic lot sizing: an O(n log n) algorithm that runs in linear time in the Wagner-Whitin case, Operations Research 1992, 40/S1: 145-156.
24. Wagner, H. M. and Whitin, T. M. Dynamic version of the economic lot size model, Management Science 1958; 5(1), 89-96.

NOTAS DO ICMC

SÉRIE COMPUTAÇÃO

088/2007 TOLEDO, F.M.B. ; SANTOS, M.O. ; ARENALES, M.N. ; SELEGHIM JÚNIOR, P. - Logística de distribuição de água em redes urbanas - racionalização energética.

087/2005 GOIS, J.P; ESTÁCIO, K.C.; OISHI, C.M. BERTTONI,V.; BOTTA, V.A.; NAGAMINE, A.; KUROKAWA, F.A.; FEDERSON, F. – Aplicação de volumes finitos na simulação numérica de contaminação em lençóis freáticos.

086/2005 MARQUES, F.P.; ARENALES, M.N. – The constrained compartmentalized knapsack problem.

085/2005 POLDI, K.C.; ARENALES, M.N. – Dealing with small demand in integer cutting stock problems with limited different stock lengths.

084/2005 PRADO, T.A.S.; NUNES, M.G.V. – A statistical generative model for unsupervised learning of web argument structures.

083/2005 POLTRONIERE, S.C.; ARENALES, M.N.; TOLEDO, F.M.B.; POLDI, K.C. – Coupling cutting stock and dot sizing problems in the paper industry.

082/2004 OLIVEIRA, P.R.; ROMERO, R.A.F. – Modelo de misturas ICA aperfeiçoado para classificação não supervisionada.

081/2004 HOTO, R.; ARENALES, M.; MACULAN, N. – The compartmentalized knapsack problem: a case study.

080/2004 KAIBARA, M.K.; FERREIRA, V.G.; NAVARRO, H.A. – Upwinding finite-difference schemes for convection dominated problems – Part I: theoretical results.

079/2004 PAIVA, D. M. B.; FREIRE, A. P.; FORTES, R. P. M. – Web engineering process – a case study from academic development