8^{th} *International Conference of Modeling and Simulation - MOSIM'10 - May 10-12, 2010 - Hammamet - Tunisia*

"Evaluation and optimization of innovative production systems of goods and services"

MATHEMATICAL MODELS AND LAGRANGIAN HEURISTICS FOR A TWO-LEVEL LOT-SIZING PROBLEM WITH BOUNDED INVENTORY

N. BRAHIMI

University of Sharjah
Department of Industrial
Engineering and Management
P.O. Box 27272 Sharjah
United Arab Emirates
NBrahimi@sharjah.ac.ae

N. ABSI S. DAUZERE-PERES

École des Mines de Saint-Étienne
Centre Microélectronique de Provence
880 route de Mimet
F-13541 Gardanne, France
{absi, dauzere-peres}@emse.fr

S. KEDAD-SIDHOUM

Université Pierre
et Marie Curie – LIP6
4 Place Jussieu
F-75252 Paris Cedex, France
safia.kedad-sidhoum@lip6.fr

ABSTRACT: *We consider a two-level lot-sizing problem where the first level consists of N end products competing for a single type of raw material (second level), which is supposed to be critical. In particular, the storage capacity of raw materials is limited and must be carefully managed. The goal is to simultaneously determine an optimal replenishment plan for the raw material and optimal production plans for the end products on a horizon of T periods. The problem is modeled as an integer linear program and solved using both a Lagrangian relaxation-based heuristic and a commercial optimization software. The results obtained using the Lagrangian heuristic are promising and new ideas are generated to further improve the quality of the solution.*

KEYWORDS: *Production Planning, Lot-Sizing, Lagrangian, Heuristic, Multi-level.*

1 INTRODUCTION

In this paper, we address the two-level multi-item lot-sizing problem with bounded inventory (2LLSP-BI). It is a production planning problem where a time varying demand of N end products has to be produced over a planning horizon of length T. The end products require a critical raw material. Indeed, the storage capacity for the raw material can be limited or the raw material is shipped over long distances. Both situations imply that the consumption by the end products of the raw material must be carefully managed. In this work, we focus on both determining an optimal replenishment plan for the raw material and optimal production plans for the end products over the planning horizon. The problem is close to the disassembly planning problem with the objective of minimizing the amount of waste by means of recycling and remanufacturing (see Lambert and Gupta (2005)).

Drexl and Kimms (1997) give a detailed review of the definition of different lot-sizing problems from simple single-level uncapacitated to complex multi-level capacitated problems. Eftekharzadeh (1993) as well as Rizk and Martel (2001) provide a review paper on multi-stage lot-sizing problems. We also quote the survey papers of Brahimi *et al.* (2003) and Karimi *et al.* (2003) for single-level problems. Crowston and Wagner (1997) and Kim *et al.* (2007) present a literature review for the planning problem in disassembly systems.

For the two-level disassembly problem, Kim *et al.* (2008) present a polynomial algorithm where a product (usually mechanical or electronic equipment) is to be disassembled while satisfying the demand of leaf items over a given planning horizon with the objective of minimizing the sum of setup and inventory holding costs. The differences between our problem and the one suggested in Kim *et al.* (2008) are presented in the following assumptions of their paper. There is no shortage for the used product (to be disassembled). Whenever the disassembly process is set up, there are always enough used product to process. Moreover, no partial extraction of components is allowed, when a certain amount of the used product is processed all its components are extracted in a proportional way (based on the gozinto factor).

In the problem addressed in this paper, the equivalent of the used product to disassemble is a Raw Material (RM) which can be involved in the production of different types of derived products. The whole quantity of processed RM can be used to make one single type of derived product. We note that the RM can be purchased and kept in stock. This is different from the two-level disassembly problem presented by Kim *et al.* (2008) where they assume that the raw material (the product to be processed) is always available. In this paper, we must decide on when to order (produce) the RM. Additionally, they do not consider the cost related to purchasing and/or holding the product, the only costs considered in their model are the setup related to the disassembly process and the holding cost of the components. We consider (at least initially) production (purchasing), holding and setup cost of the RM and all derived products. Moreover, the production variables in their model must be integer (they usually represent equipment to be disassembled).

The structure of the paper is as follows. Mathematical formulations of the problem are presented in Section 2 as well as a complexity analysis. Section 3 proposes a Lagrangian-based heuristic to solve the problem. The efficiency of the solving method is illustrated by computational experiments in Section 4. Finally, some concluding remarks and research issues are discussed in Section 5

2 MATHEMATICAL PROGRAMMING MODELS

In this section, we present two mathematical formulations, namely a straightforward aggregate model and a more efficient disaggregate formulation.

2.1 Aggregate formulation

The elements of the mathematical programming model are given in Table 1.

Symbols	Description
Ranges	
$i = 0, 1, .., N$	Index of products. $i = 0$ corresponds to the raw material
$t = 1, .., T$	Index of time periods
Parameters	
d_{it}	Demand for product i in period t
p_{it}	Production or purchasing cost of product i in period t
h_{it}	Inventory holding cost of product i in period t
s_{it}	Setup cost of product i in period t
g_i	Number of units of raw material required to make one unit of item i
I_{0t}^{max}	Warehouse capacity for raw material
Variables	
x_{it}	Production level of item i in period t
I_{it}	Inventory level of item i in period t
y_{it}	Binary setup variable equal to 1 if $x_{it} > 0$ and zero otherwise

Table 1: Elements describing the aggregate model

The aggregate formulation of the 2LLSP-BI problem is as follows:

$$\min \sum_{i=0}^{N}\sum_{t=1}^{T}(p_{it}x_{it} + s_{it}y_{it} + h_{it}I_{it}) \tag{1}$$

s.t.

$$I_{i,t-1} + x_{it} = I_{it} + d_{it}, \quad i = 1,\dots,N, t = 1,\dots,T \tag{2}$$

$$I_{0,t-1} + x_{0t} = \sum_{i=1}^{N} g_i x_{it} + I_{0t}, \quad t = 1,\dots,T \tag{3}$$

$$I_{0t} \le I_{0t}^{\max}, \quad t = 1,\dots,T \tag{4}$$

$$x_{0t} \le y_{0t}\sum_{s=t}^{T}\sum_{i=1}^{N} g_i d_{is}, \quad t = 1,\dots,T \tag{5}$$

$$x_{it} \le y_{it}\sum_{s=t}^{T} d_{is}, \quad i = 1,\dots,N, t = 1,\dots,T \tag{6}$$

$$y_{it} \in \{0,1\}, \quad i = 0,\dots,N, t = 1,\dots,T \tag{7}$$

$$x_{it},\, I_{it} \ge 0, \quad i = 0,\dots,N,\, t = 1,\dots,T \tag{8}$$

The objective (1) is to minimize the total production, purchasing, setup and inventory holding costs. Constraints (2) and (3) are the inventory balance equations for the end products and raw material, respectively. Constraint (4) restricts the inventory level to be less than the maximum warehouse capacity reserved for the raw material. Constraints (5) and (6) link the continuous production variables x_{it} to the binary setup variables y_{it}. Both of them translate the condition $y_{it} = 1$ if $x_{it} > 0$ and 0 otherwise for $i = 0, \cdots, N$ and $t = 1, \cdots, T$. Constraint (5) is written for the raw material separately because the best value of the *big-M* associated with this constraint is different from that of Constraint (6). Finally, the integrity and non-negativity Constraints are (7) and (8).

2.2 Disaggregate formulation

The aggregate formulation presented in the previous section is very intuitive and easy to understand. However, we expect an integer linear programming solver to take too much time to find an optimal solution if this model is used. This is why we derived a supposedly more efficient formulation where the production variables x_{it} are disaggregated into variables z_{ist}^0 as follows for $i = 1, \dots, N$, $t = 1, \dots, T$ and $s = 1, \dots, t$:

$$x_{it} = \frac{1}{g_i}\sum_{t=s}^{T} z_{ist}^0$$

Here, the variable z_{ist}^0 represents the quantity of raw material processed in period s to satisfy (part of) the demand of product i in period t. This is why we have added the “dummy” index 0 to z_{ist}^0.

For a more elegant presentation of the model, we introduce another continuous variable χ_{ks}^0 which corresponds to the quantity of raw material produced (or ordered) in period s to satisfy the demand of period t. Figure (1) illustrates the relationship between χ^0_{st}, z^0_{ist} and the demand d_{it}. The raw material received in period k will be used in part of the production in period s. This production will be used to partially satisfy the demand in period t. A new parameter in the disaggregate model is the cost a_{ist} for $i = 0, .., N$, $t = 1, .., T$ and $s = 1, .., t$ where $a_{ist} = p_{is} + \sum_{l=s}^{t-1} h_{il}$. The disaggregate formulation of the 2LLSP-BI problem is as follows:

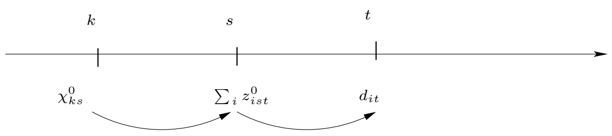

Figure 1: Relationship between continuous variables and demands

$$\min \sum_{i=1}^{N}\sum_{t=1}^{T} s_{it}y_{it} + \sum_{t=1}^{T}\sum_{s=1}^{t} a_{0st}\chi^0_{st} + \sum_{i=1}^{N}\sum_{t=1}^{T}\sum_{s=1}^{t} a_{ist}g_i z^0_{ist} \quad (9)$$

s.t.

$$\sum_{s=1}^{t} z^0_{ist} = g_i d_{it}, \qquad i = 1, \cdots, N,\ t = 1, \cdots, T \quad (10)$$

$$\sum_{s=1}^{t} \chi^0_{st} = \sum_{l=t}^{T}\sum_{i=1}^{N} z^0_{itl}, \qquad t = 1, \cdots, T \quad (11)$$

$$\sum_{j=t}^{T} \chi^0_{tj} - \sum_{i=1}^{N}\sum_{j=t}^{T} z^0_{itj} \le I^{\max}_{0t}, \qquad t = 1, \cdots, T \quad (12)$$

$$\chi^0_{st} \le y_{0s}\sum_{l=t}^{T}\sum_{i=1}^{N} g_i d_{il}, \quad s = 1, \cdots, t,\ t = 1, \cdots, T \quad (13)$$

$$z^0_{ist} \le y_{is}g_i d_{it}, \ \ i = 1, \cdots, N,\ s, t = 1, \cdots, T,\ s \le t \quad (14)$$

$$y_{it} \in \{0, 1\}, \qquad i = 0, \cdots, N,\ t = 1, \cdots, T \quad (15)$$

$$\chi^0_{st} \ge 0, \qquad t = 1, \cdots, T,\ s = 1, \cdots, t \quad (16)$$

$$z^0_{ist} \ge 0, \qquad i = 1, \cdots, N,\ t = 1, \cdots, T,\ s = 1, \cdots, t \quad (17)$$

To our surprise, the standard solver solves the aggregate formulation in Section 2.1 faster than the disaggregate formulation presented here. A comparison between the results obtained using the two models is shown in Section 4.2.

Although it is not detailed in this paper, it can be shown that this problem is as hard as the classical Capacited Lot-Sizing Problem (CLSP), which is known to be strongly NP-hard (see Chen and Thizy (1990)).

2.3 Problem complexity

The Uncapacitated Dynamic Demand Joint Replenishment Problem (UDJRP) consists in determining the replenishment/production schedule of products belonging to the same family. Whenever there is production of a subset of items in a given period, a joint (family) setup is incurred in addition to the setups of the individual items. One of the earliest publications on the the subject is Zangwill (1966). For a recent literature review of the problem, the reader is referred to Robinson et al. (2009). The UDJRP is an NP-hard problem (Arkin et al. (1989)) that has been treated extensively in the literature. We will show that the UDJRP is a special case of the 2LLSP-BI and thus that the latter is NP-hard.

The classical MILP formulation of the UDJRP is shown below. S_t and Z_t represent the family setup cost and binary setup variable respectively.

$$\min \sum_{t=1}^{T}(S_t Z_t + \sum_{i=1}^{N}(p_{it}x_{it} + s_{it}y_{it} + h_{it}I_{it})) \quad (18)$$

s.t.

$$I_{i,t-1} + x_{it} = I_{it} + d_{it}, i = 1, \dots, N, t = 1, \dots, T \quad (19)$$

$$x_{it} \le y_{it}\sum_{s=t}^{T} d_{is}, \qquad i = 1, \dots, N, t = 1, \dots, T \quad (20)$$

$$\sum_{i=1}^{N} y_{it} \le NZ_t, \qquad t = 1, \dots, T \quad (21)$$

$$Z_t \in \{0, 1\}, \qquad t = 1, \dots, T \quad (22)$$

$$y_{it} \in \{0, 1\}, \qquad i = 1, \dots, N, t = 1, \dots, T \quad (23)$$

$$x_{it},\ I_{it} \ge 0, \qquad i = 1, \dots, N,\ t = 1, \dots, T \quad (24)$$

This model can be obtained from the aggregate formulation of the 2LLSP-BI through the following steps:

1. Set the raw material inventory capacity to zero.
2. Set the production and inventory holding costs of the raw material to zero.
3. Let $Z_t = y_{0t}$, $S_t = s_{0t}$, $\forall t$.
4. Considering Constraints (3), (4) and $I_{0t} \ge 0$ (now $\mathbf{=0}$); and knowing that the received raw material in a period t must be totally used in the production in the same period, we obtain the following constraint: $x_{0t} = \sum_{i=1}^{N} x_{it}$. Though this constraint becomes redundant in the model (it just calculates x_{0t}), it states that "'If raw material is received in period t, then there must be production of at least one product."' This is translated by $\sum_{i=1}^{N} y_{it} \le NZ_t, t = 1, \dots, T$

and constraint (5) ($x_{0t} \leq y_{0t}\sum_{s=t}^{T}\sum_{i=1}^{N} g_i d_{is}$ or $x_{0t} \leq Z_t\sum_{s=t}^{T}\sum_{i=1}^{N} g_i d_{is}$) can be removed from the model.

Thus, we conclude that the UDJRP is a special case of the 2LLSP-BI and consequently the latter is NP-hard.

Theorem. *The 2LLSP-BI is NP-hard.*

3 LAGRANGIAN RELAXATION BASED HEURISTIC

A Lagrangian relaxation approach is often used to solve multi-item lot-sizing problems with coupling constraints. The main idea of this method is to decompose a multi-item lot-sizing problem into several easy to solve single item lot-sizing problems by relaxing the complicating constraints. When considering the two-level lot-sizing problem defined by Constraints (1)-(8), the coupling constraints can be identified as Constraints (3) and (4). If we consider the relaxation of Constraints (3) and (4) simultaneously, we obtain a problem that decomposes into N classical single-item Uncapacitated Lot-Sizing problem (ULS) and a particular single item problem that can be solved analytically. Each ULS sub-problem is solved using an $O(T \log T)$ dynamic programming algorithm (see Wagelmans *et al.*, 1992). In the remainder of this section, we present the general skeleton of the Lagrangian heuristic, the models resulting from the Lagrangian relaxation based on the aggregate formulation of the 2LLSP-BI problem, and some valid inequalities that can improve the Lagrangian based lower bound.

3.1 Lagrangian relaxation approach

The following algorithm summarizes the general scheme of our Lagrangian relaxation.

3.2 Restructuring and decomposing the problem

We relax in the following model Constraints (3) and (4). We can combine Constraints (3) and the non-negativity constraints of the inventory variables for raw material to derive an aggregate formulation without inventory variables for the raw material. The Lagrangian relaxation will be based on this model.

Algorithm 1 Lagrangian relaxation algorithm

Step 1: **Initialization.**
a. Initialize all multipliers to 0.
b. Set iteration number $k = 1$.
c. Initialize step length, λ^1 (set to 2 in our case).
d. Initialize the lower bound $LB = -\infty$ and the upper bound $UB = \infty$.
Step 2: **Solving the relaxed problem.** Solve Lagrangian problem L^k (See Section 3.3) and calculate current lower bound LB^k.
*Step 3:***Incumbent saving.** If $LB < LB^k$, then $LB := LB^k$.
Step 4: **Smoothing heuristic.** Use the values of X, I and Y obtained in *Step 2* and heuristic procedure to determine a feasible solution and an upper bound UB^k. The procedure described in Section 3.5 is used in our case. If $UB > UB^k$, then $UB := UB^k$
Step 5: **Updating multipliers.** Lagrangian multipliers are updated using the subgradient optimization method (See Parker and Rardin, 1988)
Step 6: **Stopping conditions.** If any stopping condition is met, then stop. The stopping conditions are either an optimal solution is found, i.e. $LB = UB$, or the number of iterations k reaches 150.
Step 7: **Update step length.** $\lambda^{k+1} = \lambda^k/2.0$ if the best lower bound is not improved during the last 10 iterations
*Step 8:***Increment k and go to Step 2.**

$$\min \sum_{i=1}^{N}\sum_{t=1}^{T}(p_{it}x_{it} + s_{it}y_{it} + h_{it}I_{it}) + \sum_{t=1}^{T}(p_{0t}x_{0t} + s_{0t}y_{0t}) + h_{0t}(\sum_{k=1}^{t} x_{0k} - \sum_{k=1}^{t}\sum_{i=1}^{N} g_i x_{ik}) \quad (25)$$

s.t.

$$I_{i,t-1} + x_{it} = I_{it} + d_{it}, i = 1,\cdots,N,\ t = 1,\cdots,T \quad (26)$$

$$\sum_{k=1}^{t}\sum_{i=1}^{N} g_i x_{ik} \leq \sum_{k=1}^{t} x_{0k}, \qquad t = 1,\cdots,T, \quad (27)$$

$$\sum_{k=1}^{t} x_{0k} - \sum_{k=1}^{t}\sum_{i=1}^{N} g_i x_{ik} \leq I_{0t}^{max}, \qquad t = 1,\cdots,T, \quad (28)$$

$$x_{0t} \leq y_{0t}\sum_{k=t}^{T}\sum_{i=1}^{N} g_i d_{ik}, \qquad t = 1,\cdots,T \quad (29)$$

$$x_{it} \leq y_{it}\sum_{k=t}^{T} d_{ik}, \qquad i = 1,\cdots,N,\ t = 1,\cdots,T \quad (30)$$

$$y_{it} \in \{0,1\}, \qquad i = 1,\cdots,N,\ t = 1,\cdots,T \quad (31)$$

$$x_{it},\ I_{it} \geq 0, \qquad i = 1,\cdots,N,\ t = 1,\cdots,T \quad (32)$$

The non-satisfaction of Constraints (27) and (28) is penalized in the objective function using Lagrangian multipliers. We associate the positive Lagrangian multipliers α_t and β_t with Constraints (27) and (28), respectively. When using the Lagrangian relaxation of Constraints (27) and (28) to derive a lower bound for the above problem, we obtain the following problem denoted 2LLSP-BI-LAG that must be solved optimally:

$$\begin{aligned}\min TC^L =& \sum_{i=1}^{N}\sum_{t=1}^{T}(s_{it}y_{it} + h_{it}I_{it}) \\ &+ \sum_{i=1}^{N}\sum_{t=1}^{T}x_{it}\left(p_{it} + g_i\sum_{k=t}^{T}(\alpha_k - h_{0k} - \beta_k)\right) \\ &+ \sum_{t=1}^{T}x_{0t}\left(p_{0t} + \sum_{k=t}^{T}(h_{0k} - \alpha_k + \beta_k)\right) \\ &+ \sum_{t=1}^{T}(s_{0t}y_{0t}) \\ &+ \sum_{t=1}^{T}\beta_t\,(-I_{0t}^{max}) \end{aligned} \tag{33}$$

s.t. (26), (29), (30), (31), (32).

Note that $\sum_{t=1}^{T}\beta_t\,(-I_{0t}^{max})$ is a constant.

3.3 Solving the sub-problems

The 2LLSP-BI-LAG problem defined by (33) and Constraints (26), (29), (30), (31), and (32) can be decomposed into two types of problems:

- The first problem is defined by the following model:

$$\begin{aligned}\min TC_1^L =& \sum_{i=1}^{N}\sum_{t=1}^{T}(s_{it}y_{it} + h_{it}I_{it}) \\ &+ \sum_{i=1}^{N}\sum_{t=1}^{T}\left(p_{it} + g_i\sum_{k=t}^{T}(\alpha_k - h_{0k} - \beta_k)\right)x_{it}\end{aligned} \tag{34}$$

  s.t. (26), (30), (31), (32). This problem decomposes into N ($i = 1, \ldots, N$) independent classical single-item uncapacitated lot-sizing subproblems with T periods. Each sub-problem can be solved using an $O(T \log T)$ dynamic programming algorithm (see Wagelmans *et al.* (1992)).

- The second problem denoted by 1LLSP-RM is a particular single item lot-sizing problem for $i = 0$ defined by the following mathematical formulation:

$$\min \sum_{t=1}^{T}s_{0t}y_{0t} + \sum_{t=1}^{T}x_{0t}p_{0t}^L \tag{35}$$

s.t.

$$x_{0t} \le y_{0t}\sum_{k=t}^{T}\sum_{i=1}^{N}g_i d_{ik} \qquad t = 1, \cdots, T \tag{36}$$

$$y_{0t} \in \{0,1\} \qquad t = 1, \cdots, T \tag{37}$$

$$x_{0t} \ge 0 \qquad t = 1, \cdots, T \tag{38}$$

where $p_{0t}^L = p_{0t} + \sum_{k=t}^{T}(h_{0k} - \alpha_k + \beta_k)$ is the Lagrangian production cost.

This problem can be further decomposed into T single period problems where the objective is to minimize $s_{0t}y_{0t} + x_{0t}p_{0t}^L$ subject to Constraints (36), (37), and (38) for every period t.

Note that x_{0t} will have a positive value (hence $y_{0t} = 1$) if $s_{0t}+x_{0t}p_{0t}^L < 0$, which can only happen if $p_{0t}^L < 0$. This implies that $x_{0t} > -\dfrac{s_{0t}}{p_{0t}^L}$.

Combining this result with Constraint (36), we conclude that the solution of the single period problem is for each period $t = 1, \cdots, T$:

$$x_{0t} = \begin{cases} 0, & \text{if } \sum_{k=t}^{T}\sum_{i=1}^{N}g_i d_{ik} \le -\dfrac{s_{0t}}{p_{0t}^L} \\ \sum_{k=t}^{T}\sum_{i=1}^{N}g_i d_{ik}, & \text{otherwise} \end{cases}$$

3.3.1 Improving the Lagrangian lower bound

Proposition. *The following constraints are valid inequalities for the problem defined by (25)-(32).*

$$\sum_{k=1}^{t}x_{0k} \ge \sum_{i=1}^{N}g_i\sum_{k=1}^{t}d_{ik}, \qquad \forall t = 1, \cdots, T \tag{39}$$

Proof. We first rewrite Constraint (27):

$$\sum_{k=1}^{t}x_{0k} \ge \sum_{i=1}^{N}g_i\sum_{k=1}^{t}x_{ik}, \qquad \forall t = 1, \cdots, T$$

We know, from the inventory balance equations and the non-negativity constraints of the inventory variables that, for every item i,

$$\sum_{k=1}^{t}x_{ik} \ge \sum_{k=1}^{t}d_{ik}, \qquad \forall t = 1, \cdots, T$$

Hence:

$$\sum_{k=1}^{t}x_{0k} \ge \sum_{i=1}^{N}g_i\sum_{k=1}^{t}d_{ik}, \qquad \forall t = 1, \cdots, T$$

Which can also be written in a more convenient way as:

$$\sum_{k=1}^{t} x_{0k} \geq \sum_{k=1}^{t}\sum_{i=1}^{N} g_i d_{ik}, \qquad \forall t = 1, \cdots, T$$

This constraint will be added to the subproblems represented by (35) to (38) which will improve the Lagrangian lower bounds.

3.3.2 Solving the resulting raw material problem

After adding valid inequalities (39) to the 1LLSP-RM model, the derived model denoted by 1MLLSP-RM becomes a single item lot-sizing problem with zero inventory cost. The $O(T \log T)$ algorithm of Wagelmans *et al.* (1992) is well adapted to solve problems with this structure. In this case, the input of the algorithm of Wagelmans *et al.* (1992) are for $t = 1, \cdots, T$:

- Demand: $d_{0t} = \sum_{i=1}^{N} g_i d_{it}$
- Setup cost: s_{0t}
- Marginal cost: $= p_{0t}^{L}$

3.4 Obtaining an initial feasible solution

In this section, we present three construction heuristics in the chronological order of their development. The three heuristics build an initial solution that will eventually be improved in the iterations of the Lagrangian heuristic. At the end of the tests, we adopted the third heuristic. The aim of presenting the two first heuristics here is to present the steps that we followed before building the most efficient heuristic WHK-BI.

3.4.1 L4L-L4L heuristic

This heuristic implements first applies the Lot-for-Lot (L4L) algorithm to derive the production plans of end products. The replenishment levels of the raw material are equal to the aggregation of the production quantities of the end products in each period. This can be summarized as: This is equivalent to say that the solution of the problem using the L4L-L4L is calculated using the formulas:

$$x_{it} = d_{it}, \qquad i = 0, \ldots, N, t = 1, \ldots, T$$

and

$$x_{0t} = \sum_{i=1}^{N} g_i d_{it}, \qquad t = 1, \ldots, T$$

3.4.2 WHK-L4L heuristic

Before calculating the raw material production using the L4L algorithm, the production plan of the end products is calculated using the algorithm of Wagelmans *et al.* (1992) (WHK) which runs in $O(T \log T)$.

Note that in the WHK-L4L and L4L-L4L heuristics we are sure that the resulting solution is feasible; since all demands are respected and the inventory level of the raw material is always equal to zero. Thus the inventory capacity constraint is always respected. The main drawback of these two heuristics is that the L4L solution for the raw material generates too many setups and thus considerably increases the total costs.

3.4.3 WHK-BI heuristic

In this heuristic the plans of the end products are also calculated using the WHK algorithm. The raw material demands correspond to the aggregation of the resulting production levels of end products. The raw material production plan is determined by solving a single item problem with bounded inventory (explicit consideration of the inventory capacity for raw material). The single item problem with bounded inventory was first solved by Love (1973) using an $O(T^3)$ algorithm. Recently, Liu (2008) designed an $O(T^2)$ algorithm based on the geometric techniques of Wagelmans *et al.* (1992).

On the other hand, it was shown by Wolsey (2006) that the single item problem with bounded inventory is equivalent to the single item problem with non-customer specific time windows (introduced by Dauzère-Pérès *et al.* (2002)), which was first solved in $O(T^4)$ by Dauzère-Pérès *et al.* (2002), and then in $O(T^2)$ by Wolsey (2006). It is very easy to convert the data of the bounded inventory problem into a non-customer specific time window problem using a simple heuristic with linear time complexity.

Our heuristic is based on the latter approach. The algorithm that converts the data of the bounded inventory problem into a non-customer specific time window problem and is summarized below. This procedure first calculates the total available and due demand for all periods (Wolsey (2006)), then uses these aggregate demands to calculate the individual time window demands (Dauzère-Pérès *et al.* (2002)). Note that the algorithm mostly adopts the notation used in Wolsey (2006). Once the time window demands are built, the $O(T^2)$ algorithm proposed by Wolsey (2006) is used to solve the problem.

Algorithm 2 Procedure to convert demands d_{0t} to non-customer specific time windows problems.

Declarations:
d_{0t}: Aggregate demand of raw material
$D_{0t} = \sum_{k=1}^{t} d_{0t}$: Cumulative demand of RM from period 1 to period t
v_{0t}: Aggregate demand available for processing in period t
$V_{0t} = \sum_{k=1}^{t} v_{0t}$: Cumulative available demand from period 1 to period t
d_{0k}^{w}: demand of time window (interval) k with availability date b^k and delivery date e^k

Compute $d_{0t} = \sum_{i=1}^{N} g_i x_{it}$, $\forall t$
Compute D_{0t}
Compute the available demand based on the inventory upper bound:
$V_{0,t} = D_{0,t} + I_{0,t}^{max}$, for t = 1 to T
$v_{0,t} = D_{0t} + I_{0t}^{max} - V_{0,t-1}$ for $t = 1$ to T (we suppose that $V_{0,0} = 0$)

Compute the time window demands
Let $L_t = V_t$, for $t = 1$ to T
Let $R_t = D_t$, for $t = 1$ to T
$L_0 = R_0 = 0$ and $k = 1$
While L, $R \neq 0$
    Set $\sigma = \min_t \{L_t > 0\}$, $\tau = \min_t \{R_t > 0\}$
    Set $d_{0k}^{w} = \min\{L_\sigma, R_\tau\}$, $b^k = \sigma$, $e^k = \tau$
    $L_\sigma \leftarrow L_\sigma - d_{0k}^{w}$, $R_\tau \leftarrow R_\tau - d_{0k}^{w}$
    $k \leftarrow k - 1$
End While

3.5 Improvement (smoothing) heuristic

The smoothing heuristic takes the solution of the Lagrangian problem and modifies it to build a good *feasible* solution. This heuristic is based on the WHK-BI heuristic presented in Section 3.4.3. The production quantities of the end products ($x_{it}, i = 1, ..., N, t = 1, ...T$) obtained from the solution of the Lagrangian problem are used to calculate a new demand for the raw material as $d_{0t} = \sum_{i=1}^{N} g_i x_{it}$ for every period t.

The production plan of the raw material is built using the same approach in Section 3.4.3. That is, we first convert the data of the problem into a non-customer specific time window problem before using the $O(T^2)$ algorithm of Wolsey (2006) to solve the new problem.

4 EXPERIMENTAL TESTS

4.1 Generated data sets

The tested data sets are built based on Trigeiro *et al.* (1989) data sets (TTM). The TTM data sets were generated for the capacitated single level problem with setup times.

In our problem, we consider the set of 540 problems in TTM. The generated production capacity values are replaced with generated inventory limits I_{0t}^{max}. At each period, I_{0t}^{max} is either equal to zero (zero inventory capacity), to the total average demand of all products, or to twice the total average demand.

$$I_{0t}^{max} = \begin{cases} 0 \\ 1 \times \frac{1}{T} \sum_{i=1}^{N} \sum_{t=1}^{T} d_{it} \\ 2 \times \frac{1}{T} \sum_{i=1}^{N} \sum_{t=1}^{T} d_{it} \end{cases}$$

This approach allows us to keep the same set of problems. However, since the setup time is varied (two different levels) in TTM, we just need 270 problems (the remaining problems become duplicate for us).

The two other parameters that we varied in our data sets were the holding cost of raw material with respect to that of end items and its Time-Between-Orders (TBO).

The holding cost of the raw material is usually smaller than the smallest holding cost of all end products. The values that we have chosen are:

$$h_0 = \begin{cases} 0.2 \times \min_i \{h_i\} \\ 0.6 \times \min_i \{h_i\} \\ 0.9 \times \min_i \{h_i\} \end{cases}$$

Note here that $h_{it} = h_i$, for t in TTM data. We suppose that this is also the case for raw material.

The TBO of raw material is usually larger than that of end products, but we have tried to cover most possible cases by setting it equal to:

$$\text{TBOz} = \begin{cases} 1 \times \text{TBOi} \\ 2 \times \text{TBOi} \\ 3 \times \text{TBOi} \end{cases}$$

Where TBOi is the time between orders of end items in TTM. Given TBOz and h_0, we can calculate the setup cost s_0 as follows:

$$s_0 = \frac{1}{2T} h_0 \left(\text{TBOz}\right)^2 \sum_{i=1}^{N} \sum_{t=1}^{T} d_{it}$$

We suppose that the utilization coefficient of raw material g_i is equal to 1 for all products. That is one unit of each product i needs one unit of raw material.

By combining all possible values of h_0 and TBOz with the already existing problems, the total number of tested problems is: $540 \times 3 \times 3 = 4860$ problems where the parameters are fixed for every 10 problems.

4.2 Numerical results

We have implemented our algorithms in the C++ language using Microsoft Visual C++ 6.0. The mathematical models were implemented on Xpress-MP with optimizer version 18.1. Both Xpress-MP and the C++ code were run on a personal computer with CPU Intel Core2 Duo T7300 and 1GB of RAM.

The average CPU time of the Lagrangian heuristic is less than 0.05 sec. and at most 0.08 seconds. We decided to fix the maximum execution time of Xpress-MP to 10 seconds.

The gaps were calculated using the formula:

$$Gap = 100 \times \frac{UB - LB}{LB} \tag{40}$$

It is worth noting here that, by default, the gaps in Xpress-MP are equal to $100 \times \frac{UB-LB}{UB}$. Instead of using this formula, we extracted the upper bounds and lower bounds in Xpress-MP and used formula (40).

In the next paragraphs, we analyze the results in two steps. We first compare the gaps of the Lagrangian heuristics with those obtained by running Xpress-MP on both aggregate and disaggregate formulations. In the second step, we compare the upper bounds and lower bounds of the lagrangian heuristic with the best lower bounds and upper bounds available.

Table 2 summarizes the results of our experimental tests and compares the different (upper bound)-(lower bound) gaps. The columns of the table represent from left to right the changing parameters and their values and the gaps for the disaggregate model (see Section 2.2) MIP-FAL, the aggregate model (see Section 2.1) MIP-AGG and the Lagrangian relaxation LR. The CV parameter represents the coefficient of variation of demand. 0.35 represents a medium variation when 0.59 represents a high variation. The I^{max} parameter represents the coefficient used to calculate inventory capacity.

It is clear from these numerical results that the Lagrangian heuristic gives better (smaller) gaps than ones obtained using Xpress-MP for the aggregate formulation with an average gap of 9.29% for the Lagrangian relaxation and 11.96% for Xpress-MP. The worst gaps were obtained using the solver on the disaggregate formulation (with an average gap of 50.89%).

		Gap		
Parameter	Value	MIP-FAL	MIP-AGG	LR
h_0	0.2	22.28	9.68	**9.10**
	0.6	56.90	11.73	**9.92**
	0.9	73.47	14.48	**8.85**
TBOz	1	13.02	12.37	**7.5**
	2	50.29	13.87	**8.64**
	3	89.35	**9.65**	11.72
N	10	35.18	**0.10**	8.44
	20	56.86	**5.83**	9.51
	30	60.61	29.96	**9.91**
I^{max}	0	4.85	**6.54**	14.30
	1	77.46	16.65	**7.64**
	2	70.35	12.70	**5.92**
TBO	1	12.38	**1.91**	7.54
	2	55.03	16.78	**11.74**
	4	85.25	17.20	**8.58**
CV	0.35	50.94	11.48	**7.28**
	0.59	50.83	12.45	**11.29**
Global Average		50.89	11.96	**9.29**

Table 2: Average gaps for the two solution approaches.

If we compare aggregate formulation gaps with those of the Lagrangian relaxation for the different parameters, we notice that the solver gives better gaps for large TBOs of the raw material, for small size problems (for $N = 10$ and $N = 20$), for problems with zero inventory capacity, and when the TBOs of individual items is small. We have to insist here on the fact that Xpress-MP was run for 10 seconds which gave it time to solve a lot of small size problems and relatively easy problems. An example of the easy problems is when the TBO of individual items is small. This means that the setup costs are small and, even if the MIP solution is suboptimal, the consequences of extra setup costs are less harmful to the solution.

Compared to Xpress-MP results, the gaps of the Lagrangian relaxation are relatively stable even if we vary the different parameters. However, we can notice that increasing the TBO of raw material increases the gap. This is related to the principle of the smoothing heuristic. This heuristic shifts extra amounts of raw material to other periods which generates extra setup costs. They are directly proportional to the value of TBOz. As the inventory capacity increases, the gaps decrease. This is due to the fact that, for larger inventory capacities, fewer constraints are violated during the solution of the sub-problems (relaxed solution). And the smoothing heuristic has to make very few changes to this solution to build a feasible plan.

In Table 3, we report the gap between Lagrangian heuristic lower bounds (LBLR) and upper bounds (UBLR) and the best known lower bounds (LBMIP) and upper bounds (UBMIP), respectively. The latter are obtained by running Xpress-MP for 60 seconds on the aggregate formulation. These results clearly show that the Lagrangian relaxation lower bounds are of good quality. This is further supported by the fact that out of the 4860 problems solved, the LBLR was better than LBMIP on 363 problems (7.47%). This is not the case for the UBLR. This means that more effort has to be made to improve the smoothing heuristic in the Lagrangian relaxation.

Parameter	Value	Gap between LBLR and LBMIP	Gap between UBLR and UBMIP
h_0	0.2	2.30	5.65
	0.6	1.85	7.08
	0.9	1.76	6.22
TBOz	1	2.14	3.81
	2	1.64	6.06
	3	2.12	9.09
N	10	2.53	5.74
	20	2.60	6.62
	30	0.78	6.60
I^{max}	0	2.90	10.69
	1	1.75	4.60
	2	1.26	3.68
TBO	1	1.84	5.56
	2	2.56	8.37
	4	1.50	5.02
CV	0.35	1.14	5.21
	0.59	2.79	7.43
Global Average		1.97	6.32

Table 3: Gaps of lower and upper bounds of Lagrangian heuristic to the best available bounds.

5 CONCLUSION

We have successfully analyzed and solved a two-level lot-sizing problem with bounded storage capacity for raw material. Different models and solution approaches were tested and compared. The best approach was the Lagrangian relaxation heuristic which showed a good stability for varying parameters such as TBO and the inventory capacity.

Though the Lagrangian relaxation heuristic is good compared to the standard solver, we believe that it is possible to obtain better results using this approach. We noticed that the lower bounds of the Lagrangian relaxation are very good. Thus we intend to focus on improved smoothing heuristics.

ACKNOWLEDGMENTS

The authors are grateful to an anonymous referee for his helpful comments that improved the presentation of the theoretical and experimental results.

REFERENCES

Arkin E, Joneja D and Roundy R., 1989. Computational complexity of uncapacitated multi-echelon production planning problems. *Operations Research Letters*, 8, p.61-6.

Boctor, F. F., Laporte, G. and Renaud, J. 2004 Models and algorithms for the dynamic-demand joint replenishment problem, *International Journal of Production Research*, 42: 13, 2667-2678

Brahimi N., Dauzère-Pérès S., Najid N. and A. Nordli, 2003. *Etat de l'art sur les problèmes de dimensionnement des lots avec contraintes de capacité*. Actes de MOSIM'03, p. 385-392.

Chen W.H. and J.M. Thizy, 1990. Analysis of relaxations for the multi-item capacitated lot-sizing problem. *Annals of Operations Research*, 26, p. 29-72.

Crowston W. B. and M. H. Wagner, 1973. Dynamic Lot Size Models for Multi-Stage Assembly Systems. *Management Science*, 20, p. 14-21.

Dauzère-Pérès S., Brahimi N., Najid N. M. and A. Nordli, 2002. Uncapacitated Lot-Sizing Problems with Time Windows. *Research Report, 02/4/AUTO*, Ecole des Mines de Nantes, France.

Drexl A. and A. Kimms, 1997. Lot sizing and scheduling - Survey and extensions. *European Journal of Operational Research*, 99(2), p. 221-235.

Eftekharzadeh R., 1993. A comprehensive review of production lot-sizing. *International Journal of Physical Distribution and Logistic Management*, 23, p. 30-44.

Karimi B., Fatemi Ghomi S.M.T. and J.M. Wilson, 2003. The capacitated lot sizing problem : a review of models and algorithms. *Omega*, 31(5), p. 365-378.

Kim H.-J., Lee D.-H. and P. Xirouchakis, 2007. Disassembly scheduling: literature review and future research directions. *International Journal of Production Research*, 45(18/19), p. 4465-4484.

Kim H.-J., Lee D.-H. and P. Xirouchakis, 2008. An Exact Algorithm for Two-Level Disassembly Scheduling. *Journal of the Korean Institute of Industrial Engineers*, 34(4), p. 414-424.

Lambert A.D.J. and S.M. Gupta, 2005. *Disassembly modeling for assembly maintenance, reuse, and recycling.* Boca Raton, FL.

Liu T., 2008. Economic lot sizing problem with inventory bounds. *European journal of operational research*, 185(1), p.204-215.

Love S.F., 1973. Bounded production and inventory models with piecewise concave costs. *Management Science*, 20(3), p. 313-318.

Narayanan, A. and Robinson, E.P., 2006. More on Models and algorithms for the dynamic-demand joint replenishment problem. *International Journal of Production Research*, Vol. 44, No. 2, 383-397

Parker, R.G. and Rardin, R.L, 1988. Discrete Optimization. *Academic Press, San Diego.*

Rizk N. and A. Martel, 2001. Supply Chain Flow Planning Methods: A Review of the Lot-Sizing Literature. *Centor Working Paper*, Université Laval.

Robinson P., Narayanan A., Sahin F., 2009. Coordinated deterministic dynamic demand lot-sizing problem: A review of models and algorithms, *Omega*, Volume 37, Issue 1, Pages 3-15

Trigeiro W.W., Thomas L.J. and J.O. McClain, 1989. Capacitated lot sizing with set-up times. *Management Science*, 35, p.353-366.

Wagelmans A., van Hoesel S. and A. Kolen, 1992. Economic lot sizing an $O(n \log n)$ algorithm that runs in linear time in the Wagner-Whitin case. *Operations Research*, 40(S1), p.145-156.

Wolsey L.A., 2006. Lot-sizing with production and delivery time windows. *Mathematical Programming, Series A*, 107, P.471-489.

Zangwill W.I. 1966. A deterministic multi-product, multi-facility production and inventory system. *Operations Research*;14:486-507.