Les cahiers Leibniz

A polynomial time algorithm to solve the single-item capacitated lot sizing problem with minimum order quantities and concave costs

Bertrand Hellion, Bernard Penz, Fabien Mangione

Laboratoire G-SCOP
46 av. Félix Viallet, 38000 GRENOBLE, France
ISSN : 1298-020X

n^{o} 193

September 2011

Site internet : http://www.g-scop.inpg.fr/CahiersLeibniz/

A polynomial time algorithm to solve the single-item capacitated lot sizing problem with minimum order quantities and concave costs

Bertrand Hellion[†], Fabien Mangione[†] and Bernard Penz[†]

[†]G-SCOP, Université de Grenoble / Grenoble INP / UJF Grenoble 1 / CNRS
46 avenue Félix Viallet, 38031 Grenoble Cedex 01, France.

Abstract

This paper deals with the single-item capacitated lot sizing problem with concave production and storage costs, and minimum order quantity (CLSP-MOQ). In this problem, a demand must be satisfied at each period t over a planning horizon of T periods. This demand can be satisfied from the stock or by a production at the same period. When a production is made at period t, the produced quantity must be greater than a minimum order quantity (L) and lesser than the production capacity (U). To solve this problem optimally, a polynomial time algorithm in $O(T^5)$ is proposed and it is computationally tested on various instances.

Keywords: Lot-sizing, polynomial time algorithm, minimum order quantity, capacity constraint.

1 Introduction

This paper deals with a generalization of the single-item capacitated lot sizing problem (CLSP) with fixed capacity. The CLSP consists in satisfying a demand at each time period t over a planning horizon T. The demand is satisfied from the stock or by a production. Costs incur for each production and when an item is stored between two consecutive periods. A fixed maximum production capacity (U) must be respected. The problem we consider in this paper contains a minimum order quantity constraint. This constraint imposes that if a production is done at a given period, the quantity must be greater or equal than a minimum level L. The U and L values are constant for the T periods. This problem is noted CLSP-MOQ in the following.

The single-item capacitated lot sizing problem is known to be NP-Hard [2]. However, some cases are polynomially solvable. This is the case when the capacity is fixed over the T periods. Florian and Klein [6] considered a case where production and holding cost functions are concave. They proposed an exact method with a time complexity in $O(T^4)$. Later van Hoesel and Wagelmans [11] improved the complexity of the algorithm in $O(T^3)$ when the holding costs are linear. A complete survey on the single-item lot sizing problem can be found in [3].

The CLSP-MOQ is relevant in some industrial contexts. Lee [7] studied an industrial problem where a manufacturer imposes a minimum order quantity to its supplier. The author took an example where the buyer has to choose a supplier among a manufacturer using MOQ constraints and a local dealer. The local dealer supplies the products in just-in-time with a higher cost per unit. The author designed an $O(T^4)$ algorithm which has been tested on industrial data. Porras and Dekker [10] studied an industrial case where the producer imposes a minimum order quantities (MOQ) to produce the items. The company uses containers to ship the products, and set-up costs were not specified explicitly. Consequently, in order to save fixed costs, the producer imposes the MOQ constraint, which plays the role of set-up cost. Zhou et al. [12] analyzed a class of simple heuristic policies, to control stochastic inventory systems with MOQ constraints. They also developed insights into the impact of MOQ constraints on repeatedly ordered items to fit in an industrial context.

The first studies on the MOQ constraints were from Constantino [5] and Miller [8]. They analyzed these constraints from a polyhedral point of view. Constantino derived strong inequalities which described the convex hull of the solutions, considering the production level as a continuous variable. Miller [8] replaced the production level by the amount of product that is produced in excess of the lower bound. Thus he studied the facets of the solutions's convex hull. He focused on multi-item problems and he proved that the single period relaxation is NP-hard. Chan and Muckstadt [4] studied a production-inventory system in which the production quantity is constrained by a minimum and a maximum level in each period. However, the production level cannot be zero. They characterized the optimal policy for finite and infinite time horizons. The first exact polynomial time algorithm was recently developed by Okhrin and Richter [9]. They solved a special case of the problem in which the unit production cost is constant over the whole horizon, and then, can be discarded. Furthermore, they assumed that the holding costs are also constant over the T periods. Considering this restriction, they derived from a polynomial time algorithm in $O(T^3)$.

In this study, we extend Okhrin and Richter result's [9] to the problem with concave production and holding costs. We proposed an optimal algorithm called HMP with a time complexity in $O(T^5)$. The paper is organized as follows. Section 2 describes the problem and introduces the notations. In section 3, we give some definitions and present the properties that allow us to solve the problem in polynomial time. The algorithm and its time complexity study are given. In section 4, the efficiency of the method is tested on various instances. Finally, concluding remarks and perspectives are given in section 5.

2 Problem description and notations

The single-item lot sizing problem consists in satisfying the demands over T consecutive periods. At each period t, the demand d_t must be satisfied by production at period t (X_t) and/or from the inventory available at the end of the period $t-1$ (I_{t-1}).

The production at each period is constrained by a constant capacity U. If a production is done at period t, it must be greater or equal than a non-zero minimum order quantity L. We also consider production and inventory costs. The production cost is a concave function of the quantity produced ($p_t(X_t)$) and the inventory costs is a concave function of the inventory level ($h_t(I_t)$). Notice that concave cost functions may include set-up costs. The notations are summarized in Table 1.

T	number of periods
d_t	demand at period t
X_t	production at period t
I_t	inventory level at the end of period t
U	production capacity
L	minimum order quantity
$p_t(X_t)$	concave production cost function
$h_t(I_t)$	concave storage cost function

Table 1: Notations

The CLSP-MOQ can be easily modeled by a mathematical program. The decision variables are X_t, I_t and a decision variable Y_t defined as follows:

$$Y_t = \begin{cases} 1 & \text{if the } X_t > 0 \\ 0 & \text{otherwise.} \end{cases}$$

The mathematical formulation of the CLSP-MOQ is then:

$$\text{Min} \sum_{t=1}^{T} p_t(X_t) + \sum_{t=1}^{T} h_t(I_t) \tag{1}$$

$$X_t + I_{t-1} - I_t = d_t \qquad \forall t \in T \tag{2}$$

$$LY_t \leq X_t \leq UY_t \qquad \forall t \in T \tag{3}$$

$$X_t, I_t \in \mathbb{R} \qquad \forall t \in T \tag{4}$$

$$Y_t \in \{0, 1\} \qquad \forall t \in T \tag{5}$$

The objective function (1) minimizes the total production and storage cost. Constraint (2) is the flow constraint. Constraints (3) insures that the maximum capacity and the minimum order quantity are satisfied. Constraints (4) and (5) defines the validity domain of the variables.

Without loss of generality, we assume that $I_0 = 0$. Unfortunately, I_T can be strictly positive in an optimal solution. These two cases ($I_T = 0$ and $I_T > 0$) are considered in the following section.

3 An optimal algorithm

In this section, we introduce some definitions and we prove some properties. Based on these properties, we will be able to derive a polynomial time algorithm to solve the CSLP-MOQ problem.

Definition 1. *Regeneration points*
A period t is called a regeneration point if $I_t = 0$.

Definition 2. *Fractional production period*
A period t is called a fractional production period if $L < X_t < U$.

Definition 3. *UL-capacity-constrained sequence*
S_{uv} is a UL-capacity-constrained sequence if the following conditions are verified:

- *u and v are regeneration points i.e. $I_u = I_v = 0$;*
- *The demand d_t for $t = \{u+1, \cdots, v\}$ is satisfied;*
- *For all $t \in \{u+1, \cdots, v-1\}$, $I_t \neq 0$ i.e. t is not a regeneration point;*
- *The production X_t for $t \in \{u+1, \cdots, v\}$ is equal to 0, U or L, except for at most one period which can be a fractional production period.*

At this time, we consider that $I_T = 0$. The case for which $I_T > 0$ will be considered at the end of this section.

Property 1. *A solution of the CLSP-MOQ problem can be seen as succession of subsequences such as both the starting period and the ending period are regeneration points.*

Proof. Assuming that $I_k = 0$ for some $k \in \{1...n-1\}$. An optimal solution can be found by independently finding solutions to the problems for the first k periods and for the last $T - k$ periods. Consequently, a production plan can be seen as a sequence of consecutive periods in such a way that the stock is empty at the beginning and at the end of each sequence. □

The problem is now to know if the production plan of each sub-sequence is easy to compute. Fortunately, these sub-sequences have good properties that allow us to find polynomially the optimal production plan.

Property 2. *Let us consider an interval of periods $[u, v]$ such as $I_u = I_v = 0$. UL-capacity-constrained sequences are dominant.*

Proof. To prove this result, we will show that if a solution is not an UL-capacity-constrained sequence, this one has to be a convex combination of two other feasible solutions. Let us consider a solution S_{uv} such as $I_u = I_v = 0$, $I_t \neq 0$ for $t \in \{u+1, \cdots, v-1\}$ and in such a way that there exists at least two *fractional production periods i.e.* i and j are such as $u+1 \leq i < j \leq v$ and $L < X_i, X_j < U$. Consequently, we can relocate a small value of production between X_i and X_j as follows. Let us define ω as the biggest production quantity we can relocate keeping the solution feasible, and without changing other production levels. Then:

$$\omega = \min\{U - X_j \ ; \ U - X_i \ ; \ X_i - L \ ; \ X_j - L \ ; \ \min_{t=i}^{j-1} I_t\}$$

By relocating $\frac{1}{2}\omega$ from i to j, we obtain a solution S'_{uv}. The production plan S'_{uv} is obviously feasible. Symmetrically, by relocating $\frac{1}{2}\omega$ from j to i, we obtain a valid solution S''_{uv}. However $S_{uv} = \frac{1}{2}S'_{uv} + \frac{1}{2}S''_{uv}$, proving that S_{uv} is not an extreme point. Then S_{uv} is not the unique optimal solution, and it is dominated. □

From now on, we have to verify if finding an optimal UL-capacity-constrained sequence can be done in polynomial time. Let us define α (resp. β) as the number of periods in which the production is equal to U (resp. L). The fractional production is noted ε. Using $D_{uv} = \sum_{t=u+1}^{v} d_t$, the total demand for the sequence, we can write:

$$\alpha U + \beta L + \epsilon = D_{uv} \tag{6}$$

In a first place, we will prove that the number of triplets $(\alpha, \beta, \varepsilon)$ is in $O(T)$. Then, we will prove that it is possible to find an optimal plan in $O(T^3)$.

Property 3. *The number of triplets $(\alpha, \beta, \varepsilon)$ is in $O(T)$*

Proof. Let us consider the maximal interval length with $u = 0$ and $v = T$. Define D as the total demand ($D = \sum_{t=1}^{T} d_t$). The triplet $(\alpha, \beta, \varepsilon)$ verifies equation (6), and obviously the following ones:

$$\alpha + \beta \leq T \tag{7}$$

$$L < \varepsilon < U \qquad \text{or} \qquad \varepsilon = 0 \tag{8}$$

The problem is now the enumeration of the distinct feasible triplets $(\alpha, \beta, \varepsilon)$. Let us define α_{min} (resp. α_{max}) the minimum (resp. the maximum) feasible values for α. In the same way, let us define β_{min} and β_{max}.

Bounds on α: From equation 6, we derive immediately:

$$\alpha U + TL + U > D$$

This leads to the following inequality:

$$\alpha > \frac{D - LT - U}{U} \Leftrightarrow \alpha > \frac{D - LT}{U} - 1 \text{ thus } \alpha_{min} = \lceil \frac{D - LT}{U} - 1 \rceil$$

Furthermore we immediately have:

$$\alpha \leq \frac{D}{U} \text{ and then } \alpha_{max} = \lfloor \frac{D}{U} \rfloor$$

And then:

$$\frac{D - LT}{U} - 1 \leq \alpha_{min} \leq \alpha \leq \alpha_{max} \leq \frac{D}{U}$$

The number of possible values for α is at most:

$$\frac{D}{U} - (\frac{D - LT}{U} - 1) + 1 = \frac{LT}{U} + 2$$

Bounds on β: Let us consider a feasible α. Using equations (6) and (8), we have:

$$\beta \geq \frac{D - \alpha U - U}{L} \text{ and then } \beta_{min} = \lceil \frac{D - \alpha U - U}{L} \rceil$$

and

$$\beta \leq \frac{D - \alpha U}{L} \text{ and then } \beta_{max} = \lfloor \frac{D - \alpha U}{L} \rfloor$$

The maximum number of values for β when α is fixed is:

$$\frac{D - \alpha U}{L} - \frac{D - \alpha U - U}{L} + 1 = \frac{U}{L} + 1$$

Maximum number of distinct triplets $(\alpha, \beta, \varepsilon)$: The maximum number of triplets is given by:

$$\left(\frac{LT}{U} + 2\right)\left(\frac{U}{L} + 1\right) = \left(1 + \frac{L}{U}\right)T + 2\frac{U}{L} + 2$$

We must now bound this quantity. We have two cases.

- If $\frac{U}{L} \leq T$, the maximum number of triplets is bounded by $4T$.
- If $\frac{U}{L} > T$ we have $LT < U$. Let us consider $\alpha_{max} = \alpha_1 = \lfloor \frac{D}{U} \rfloor$. α_1 is a possible value for α. Conversely, $\alpha_1 - 2$ is not possible because $LT + \varepsilon < 2U$. Then only two values for α can be chosen: $\lfloor \frac{D}{U} \rfloor$ and $\lfloor \frac{D}{U} \rfloor - 1$. In each case, there are at most T possible values for β, and then at most $2T$ distinct triplets.

That concludes the proof. □

The number of triplets $(\alpha, \beta, \varepsilon)$ is in $O(T)$. We must now prove that the best production plan can be found in polynomial time.

Property 4. *A UL-capacity-constrained sequence can be computed in polynomial time.*

Proof. Let us consider u and v, two regeneration points. From property 3, we can compute each valid triplets $(\alpha, \beta, \varepsilon)$. For each α and β, we have a unique ε. At most, we can have K different ε with K in $O(T)$. In the following, we note ε_i for $\{i = 1, \cdots, K\}$.

We can now build a directed acyclic graph $(\mathcal{UV})$ as follows. This graph is divided into levels, for convenience from level u to level v. At level u only one node labeled 0 is put. At level $u+1$, the nodes are labeled with a feasible cumulative production level $(0, L, \varepsilon_1, \cdots, \varepsilon_K, U)$. At each level, several non feasible cumulative production could be discarded (see Section 4). The weight of an arc is the cost (production and storage) to have the two cumulative productions at the extremity of the arc. The next levels are built the same way (see Figure 1).

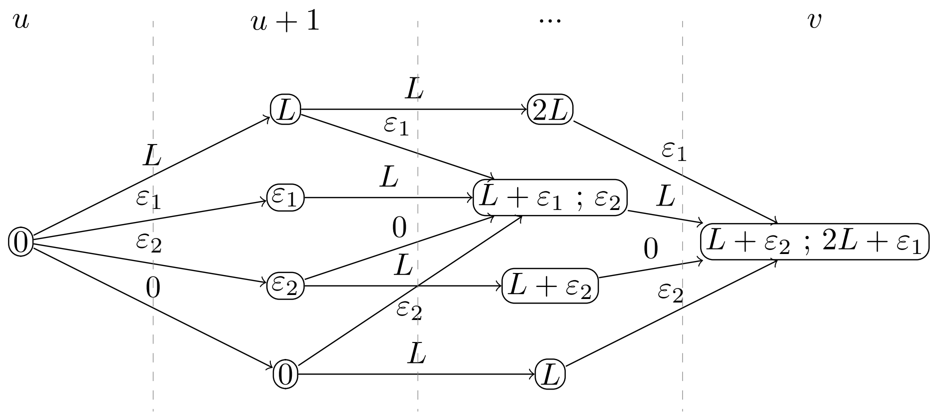


Figure 1: An example of graph $\mathcal{UV}$ with 4 periods

Considering a given triplet $(\alpha, \beta, \varepsilon)$, the number of nodes at each level of the graph is in $O(T)$. The number of triplet is in $O(T)$ therefore at each level, the number of nodes is thus in $O(T^2)$. We have at most T levels, consequently the number of nodes in the graph is in $O(T^3)$. A shortest path in this graph gives us the optimal production plan. As each node has at most four predecessors, the evaluation of the node can be made in constant time, and consequently, the time complexity for finding an optimal solution is in $O(T^3)$. □

Unfortunately, due to the MOQ constraint, an optimal solution can have items in stock at T. Let us assume that the storage cost $h_{T-1}(I_{T-1})$ for a positive value of I_{T-1} is very high. If the production cost at period T is low and if the demand at period T respects $0 < d_T < L$, the best strategy could be to produce L at the last period leading to a storage of $I_T = L - d_T$ items. Consequently, we must study the sequences $\widehat{S}_{uT}$ where u is a *regeneration point* and such as $I_T \neq 0$.

Property 5. *An optimal production sequence $\widehat{S}_{uT}$ in such a way that u is a regeneration point, and $I_T \neq 0$ can be computed in polynomial time.*

Proof. First of all, $\widehat{S}_{uT}$ cannot contain a fractional production period. If $\widehat{S}_{uT}$ contains a fractional production period at a period k of value ε, it is possible to decrease the production at this period by $min\{\varepsilon - L, \min_{t=k}^{T} I_t\}$.

Furthermore, if $\widehat{S}_{uT}$ contains a period where the production is maximum (U) at a period k, we can easily decrease the production because the storage levels are strictly positive from k to T. Then $\widehat{S}_{uT}$ cannot contain production level at U.

Consequently, the sequence $\widehat{S}_{uT}$ has only production period at L or 0. Furthermore, $I_T < L$ else we can suppress one of the productions, and then we have only to compute the value β as follows:

$$\beta = \lceil \frac{D_{u+1T}}{L} \rceil$$

The best production plan can now be computed in $O(T^2)$ with an algorithm similar to this one presented in Property 4. $\square$

We can now derive a polynomial time algorithm (called HMP) from the previous properties. Let us build a directed acyclic graph ($\mathcal{G}$) as follows. Let us define $T+1$ vertices labeled from 0 to T. A vertex t signifies that t is a *regeneration point.* For each pair (i,j), with $i < j < T$ add an arc. The value on this arc is computed by the algorithm described in Property 4. For each pair (i,T), we have to add in the graph two arcs, one considering that $I_T = 0$ (Property 4), and one considering that $I_T \neq 0$ (Property 5). A shortest path in the graph leads to an optimal production plan.

Theorem 1. *Algorithm HMP gives an optimal solution for the CLSP-MOQ problem with constant capacity, constant minimum order quantity and concave production and storage costs in time in $O(T^5)$*

Proof. An optimal solution is given by a succession of subsequences between to regeneration points (property 1). Then, one of the shortest path in graph ($\mathcal{G}$) is an optimal solution. The construction of the graph ($\mathcal{G}$) is in $O(T^5)$. Indeed, we have $O(T^2)$ arcs in graph ($\mathcal{G}$) and each arc can be computed in $O(T^3)$ (see property 4 and 5), leading to a time complexity of $O(T^5)$. Finding one of the shortest path in ($\mathcal{A}$) can be made in $O(T^2)$. Finally, we conclude that the time complexity of the algorithm HMP is in $O(T^5)$. $\square$

4 Computational experiments

4.1 Computational settings

To test the efficiency of algorithm HMP, we have performed intensive experiments. We aim to define different sets, in which all the instances share the same parameters values. Then we randomly generate different feasible instances for each sets.

Demand values are generated from a normal distribution. As in [9], 4 parameters are selected: the mean and the variance of the demand distribution, the capacity and the MOQ. We choose three different values for the mean μ, say low, medium and large, fixed to 40, 200 and 600. For each of these values, we have three different values for the variance σ which correspond to 10%, 20% and 30% of μ. We set three capacity levels: small, medium and large. Each capacity verifies μ/U equal to 0.75, 0.85 and 0.95 as proposed in [9]. The values of these parameters are given in Table 2.

Mean demand	Demand variance			Capacity		
	Small	Medium	Large	Small	Medium	Large
40	4	8	12	42	47	53
200	20	40	60	211	235	267
600	60	120	180	632	706	800

Table 2: Mean demand, Variance and Production Capacity Parameters

In addition, we choose the MOQ to be equal to the lower quartile L1, median L2 and upper quartile L3 of the normal distribution as in [1]. The different values for L1, L2 and L3 are given in Table 3.

Mean	Variance	L1	L2	L3
40	4	37	40	43
40	8	35	40	45
40	12	32	40	48
200	20	187	200	213
200	40	173	200	227
200	60	160	200	240
600	60	560	600	640
600	120	519	600	681
600	180	479	600	741

Table 3: Minimum Order Quantity Parameters

Hence we have 81 possible sets of instances as shown in Figure 2. However, taking these parameters may lead to infeasible instances. Indeed, the 9 sets with $\mu/U = 0.95$ and L3 are insolvable because of $U < L$. In the same way, in the 3 sets with $\mu/U = 0.85$, L3, and $\sigma = 30\%\mu$, the capacity is smaller than le MOQ. Thus 12 among the 81 sets must be discarded. Finally, we have only 69 feasible sets.

To compute the concave costs, both production and storage costs are defined by a fixed and a linear cost. For each period, these costs are randomly chosen among 3 different values: 10, 20 and 30 for the setup production cost; 1, 2 and 3 for the linear production cost; 1, 2 and 3 for the setup storage cost and 0.1, 0.2 and 0.3 for the linear storage cost.

$$\mu : \begin{pmatrix} 40 \\ 200 \\ 600 \end{pmatrix} \rightarrow U : \begin{pmatrix} \mu/0.95 \\ \mu/0.85 \\ \mu/0.75 \end{pmatrix} \rightarrow \sigma : \begin{pmatrix} 0.1\mu \\ 0.2\mu \\ 0.3\mu \end{pmatrix} \rightarrow L : \begin{pmatrix} \mu - 0.68\sigma \\ \mu \\ \mu + 0.68\sigma \end{pmatrix}$$

Figure 2: Parameter settings

Finally, we created 69 sets of instances and for each set, we randomly generated ten feasible instances.

Algorithm HMP has been implemented in Java 1.6 and has been run on a 3 GHz Intel Core 2 Duo Machine with 4GB memory running Windows 7.

4.2 Improvements of Algorithm HMP

To increase the algorithm efficiency, we implement several improvements. In the construction of graph ($\mathcal{G}$), if an edge (uv) verifies $D_{u+1,v} > U(u-v)$, the production plan from $u+1$ to v is infeasible and then the vertex u cannot be a regeneration point. The vertex u is removed.

In the construction of graph ($\mathcal{UV}$), we have to verify that each node created leads to a solution for which the demand is satisfied and the stock level is strictly positive.

These improvements reduce the number of nodes in graphs ($\mathcal{G}$) and ($\mathcal{UV}$), and consequently save computational time. They strongly affect the efficiency of Algorithm HMP, see further in section 4.4.

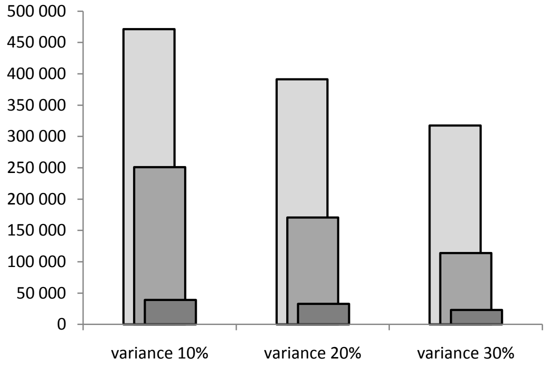
4.3 Performance evaluation of Algorithm HMP on the sets of instances

We now compare the running time of Algorithm HMP to test its efficiency. For that, we fixed the horizon to 40 periods and launch the algorithm on the 69 instance sets. For each instance, we observe the computational time and the number of nodes generated in graph ($\mathcal{UV}$). We denote as *iterations* this number of nodes. Figure 3 shows the mean number of iterations, of each sets satisfying $\mu = 200$.

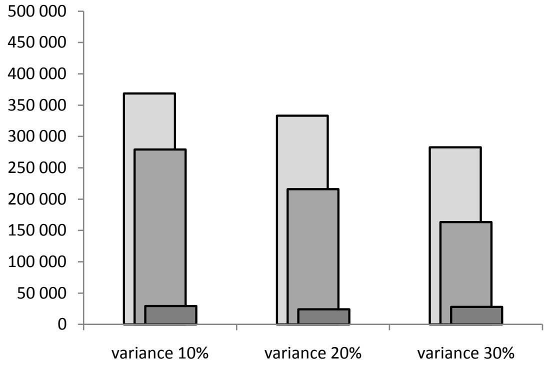
Indeed with another value of μ, the shape of the histograms remains the same.

The value of U hugely affects the number of iterations. For small values of U, the production plan is more constrained and consequently, some periods cannot be regeneration points. Vertices in $\mathcal{A}$ are then removed. For the same reason, large values for σ cause the deletion of many edges and vertices in graph $\mathcal{G}$.

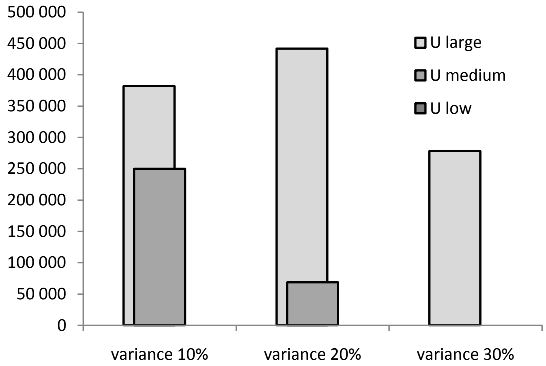
Finally, all the instances are solved in less than 30 seconds.



(a) L1 = lower quartile



(b) L2 = μ



(c) L3 = upper quartile

Figure 3: Number of iterations for $\mu = 200$ with *normal* costs

4.4 Evaluation of the computational time when T increases

The following experiments have two goals. Firstly, we show that the number of nodes in graph $\mathcal{UV}$ and the running time are strongly dependent. Lastly we point out the computational complexity of Algorithm HMP.

To show that, we choose three sets of instance among the 69 previous ones:

- Instance set for which we have the minimum number of iterations among all the sets (set A);
- Instance set for which we have the mean number of iterations among all the sets (set B);
- Instance set for which we have the maximum number of iterations among all the sets (set C).

The parameters for the 3 instance sets are given in Table 4.

sets	μ	σ	L	U
A	40	20%	μ	$\mu/0,95$
B	40	20%	μ	$\mu/0,85$
C	200	10%	upper quartile	$\mu/0,75$

Table 4: Parameters for Instance Sets A, B and C

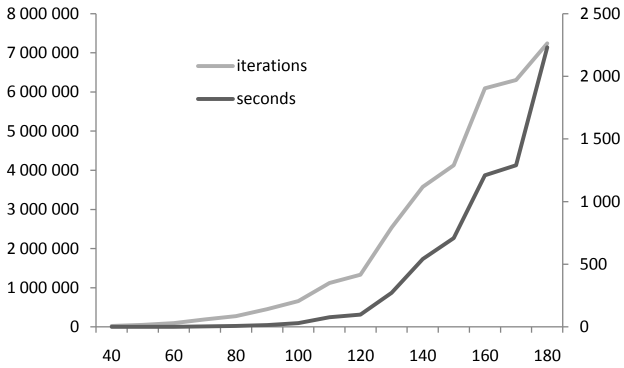
For each set, 10 instances were generated. The horizon length starts from $T = 40$ and increases with a step of 5 for set C and 10 for sets A and B.

The curves given in Figure 4 show the strong dependence of the computational time and the number of nodes in Graph $\mathcal{UV}$ for the 3 instance sets. For the set A, Algorithm HMP solves the instances in a few seconds up to 120 periods in the horizon. When $T = 180$, the computational time is less than 1 hour. For instance set B, the instances are more difficult to solve. The algorithm is quick until $T = 70$ but it needs 1 hour to solve instances with $T = 105$ and almost 2 hours when $T = 120$. Finally, for difficult instances (set C) the algorithm is very efficient until $T = 50$, but it needs almost 1 hour when $T = 73$ and almost 4 hours when $T = 80$.

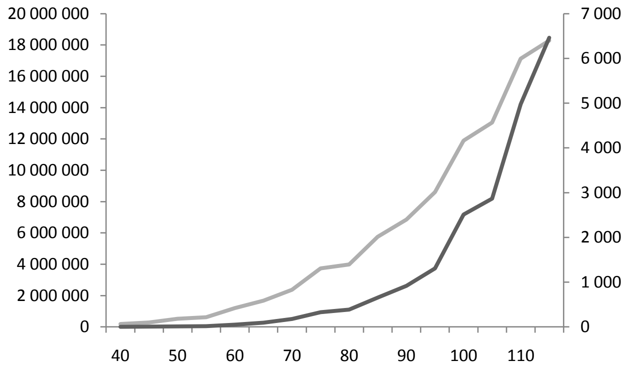
The standard deviations of the number of iterations fluctuate, depending on the instance sets chosen. It can be seen in Table 5. For easy instances (set A), the standard deviations are large: almost equal to the mean value (69%). Conversely, for set C, the standard deviations are low (7% of the mean value). It means that all the set C instances are difficult to solve. However, the solving difficulty of the set A instances is hugely varying. This reinforces the idea that the more constrained the instance is, the easier it is to solve it.

set A	set B	set C
69%	29%	7%

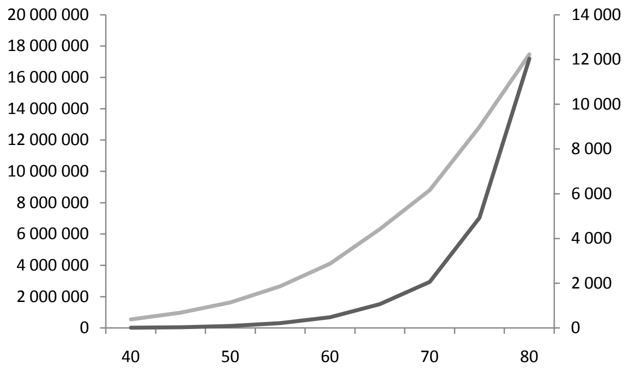
Table 5: Ratio between standard deviation and the mean of the number of iteration for the 3 sets of instances (in average)



(a) set A



(b) set B



(c) set C

Figure 4: Correlation between time and iterations increasing the number of periods

5 Conclusion

This paper has focused on a generalization of the capacitated single item lot sizing problem. In this problem, the production levels must be bounded by a minimum order quantity and a maximum capacity when a production is decided. Both production and storage costs are concave. We proposed an $O(T^5)$ exact algorithm that generalizes Florian and Klein's one [6] (in $O(T^4)$ without the minimum order quantity) and Okhrin and Richter's one [9] (in $O(T^3)$ with a simplified cost structure).

The computational experiments showed that Algorithm HMP solves large instances in reasonable time, even for hard instances. Then, it could be used as a brick to solve more complex lot sizing problems as multi-items ones.

The theoretical complexity of Algorithm HMP seems to be difficult to improve with general concave costs. In future works, it could be interesting to know if this theoretical complexity could be decreased when the cost structure is only a fixed cost plus a linear one as we used in our tests. Also we could only consider linear costs for the storage.

References

[1] E.J. Anderson and B.S. Cheah. Capacitated lot-sizing with minimum batch sizes and setup times. *International Journal of Production Economics*, 30-31:137–152, 1993.

[2] G.R. Bitran and H.H. Yanasse. Computational complexity of the capacitated lot size problem. *Sloan School of Management*, 1981.

[3] N. Brahimi, S. Dauzère-Pérès, N.M. Najid, and A. Nordli. Single item lot sizing problems. *European Journal of Operational Research*, 168(1):1–16, 2006.

[4] E.W. Chan and J.A. Muckstadt. The effects of load smoothing on inventory levels in a capacitated production and inventory system. *School of Operations Research and Industrial Engineering*, 1999.

[5] M. Constantino. Lower bounds in lot-sizing models: A polyhedral study. *Mathematics of Operations Research*, 23(1):101–118, 1998.

[6] M. Florian and M. Klein. Deterministic production planning with concave costs and capacity constraints. *Management Science*, 18(1):12–20, 1971.

[7] C.-Y. Lee. Inventory replenishment model: lot sizing versus just-in-time delivery. *Operations Research Letters*, 32(6):581–590, 2004.

[8] A.J. Miller. *Polyhedral Approaches to Capacitated Lot Sizing Problems.* PhD thesis, Georgia Institute of Technology, 1999.

[9] I. Okhrin and K. Richter. An $O(T^3)$) algorithm for the capacitated lot sizing problem with minimum order quantities. *European Journal of Operational Research*, 211(3):507–514, 2011.

[10] E. Porras and R. Dekker. An efficient optimal solution method for the joint replenishment problem with minimum order quantities. *European Journal of Operational Research*, 174(3):1595–1615, 2006.

[11] C.P.M. van Hoesel and A.P.M. Wagelmans. An $O(T^3)$ algorithm for the economic lot-sizing problem with constant capacities. *Management Science*, 42(1):142–150, 1996.

[12] B. Zhou, Y. Zhao, and M.N. Katehakis. Effective control policies for stochastic inventory systems with a minimum order quantity and linear costs. *International Journal of Production Economics*, 106(2):523–531, 2007.

Les cahiers Leibniz ont pour vocation la diffusion des rapports de recherche, des séminaires ou des projets de publication sur des problèmes liés au mathématiques discrètes.

Responsables de la publication : Nadia Brauner et András Sebő
ISSN : 1298-020X