A polynomial algorithm for a lot-sizing problem with backlogging, outsourcing and limited inventory

Chengbin Chu, Feng Chu, Jinhong Zhong, Shanlin Yang

School of Economics and Management, Tongji University, China
Laboratoire Génie Industriel, Ecole Centrale Paris, France
Laboratoire IBISC, Université d’Evry-Val d’Essonne, France
School of Management, Hefei University of Technology, China

Article history:
Received 11 December 2010
Received in revised form 28 June 2012
Accepted 20 August 2012
Available online 25 September 2012

Keywords:
Lot-sizing
Production planning
Dynamic programming
Polynomial algorithm

Abstract
This paper addresses a real-life production planning problem arising in a manufacturer of luxury goods. This problem can be modeled as a single item dynamic lot-sizing model with backlogging, outsourcing and inventory capacity. Setup cost is included in the production cost function, and the production level at each period is unbounded. The holding, backlogging and outsourcing cost functions are assumed to be linear. The backlogging level at each period is also limited. The goal is to satisfy all demands in the planning horizon at minimal total cost. We show that this problem can be solved in $O(T^4 \log T)$ time where $T$ is the number of periods in the planning horizon.

© 2012 Elsevier Ltd. All rights reserved.

1. Introduction

This paper addresses a real-life production planning problem coming from a company of luxury goods, which is a single item capacitated lot-sizing problem over a finite planning horizon of $T$ periods. A production plan should be established based on the forecast demand. The demand of each period can be satisfied by production, and/or through inventory from previous periods, as in classical industry, but also by backlogging from subsequent periods, or it can be partially or entirely outsourced, should this lead to cost savings. In fact, the customers of luxury goods are willing to wait for their favorite items in case of shortage. As a consequence, shortage does not mean lost sales. Outsourcing is also a common practice in luxury goods. Of course, backlogging and outsourcing also cost money and should be taken into account in the model. Furthermore, the inventory, backlogging and outsourcing levels are limited. The problem consists of determining the amount to be produced and outsourced in each period so that all demands are satisfied (not necessarily in time) and the total cost of production, inventory holding, backlogging and outsourcing is minimized. This problem can be modeled as a lot-sizing model with backlogging and outsourcing. Even though this problem comes from a company of luxury goods, it also arises in other industries with a high degree of monopoly.

The capacitated lot-sizing problem with inventory capacity is usually known as a bounded/limited inventory model. Up to now, there are very few works on this aspect in the existing literature. In most of the studies, the inventory is often assumed to be unbounded. Nonetheless, for many real-life applications, bounded inventory models are relevant.

Nowadays, outsourcing has become a common practice for those firms that deal with highly variable demands to reduce the cost. It is more and more popular both to customers and firms since in this way, the former can obtain what he wants while the latter can avoid the goodwill loss or lessen the cost. However, outsourcing decision (when and how much) is not always easy to make for firms. Many scientific issues still need to be addressed. Nevertheless, academic research on this aspect still is scarce. Some authors considered immediate lost sales models (Aksen, Altinkemer, & Chand, 2003; Sandbothe & Thompson, 1990, 1993). Absi, Kedad-Sidhoum, and Dauzère-Pérès (2011), Aksen (2007) considered models involving backlogging and lost sales. In the model considered by Aksen (2007), in case of shortage, the portion of the lost sales are input data while it is a decision variable in the model considered by Absi et al. (2011). It should be noticed that the inventory is unbounded in all these models. In this paper, we prefer talking about outsourcing instead of lost sales, since in the current increasingly competitive environment, all companies make their best to satisfy customer requirements and to avoid lost sales.
lost sales are unrealistic especially when demands are assumed to be deterministic. However, from the modeling point of view lost sales are equivalent to outsourcing a part or whole of the demand of customers for various reasons (e.g., for small quantities, outsourcing from subcontractors is cheaper than in-house production). Deliberate lost sales must be avoided. As we show hereafter, the model studied in this paper is the most general one ever investigated.

The uncapacitated version of lot-sizing model was first introduced by Wagner and Whitin (1958), they presented an $O(T^3)$ dynamic programming method. Zangwill (1966) generalized their work to the backlogging case, and proposed a polynomial algorithm. Blackburn and Kunreuther (1974) and Lundin and Morton (1975) developed an $O(T^2)$ algorithm for the unbounded inventory model with time-varying linear production cost functions and concave holding/backlogging cost functions. Later, Morton (1978) gave an improved algorithm running in $O(T^2)$ time to avoid the inner search for an even more special case where there is unlimited inventory and the cost is linear and stationary over time. Aggarwal and Park (1993), Federgruen and Tzur (1991), Wagelmans, Van Hoesel, and Kolen (1992) improved the complexity to $O(T)$ and provided simple forward algorithms in $O(T \log T)$ time for the problems with general non-stationary linear production, holding and backlogging cost functions. Ganas and Papachristos (2005) devised a new $O(T^2)$ algorithm for the uncapacitated lot-sizing problem with linear holding/backlogging cost functions, fixed ordering cost (independent of the quantity ordered), constant demand, and constant cost parameters.

Lot-sizing models with production capacity have been studied by many authors. The computational complexity was investigated by Bitran and Yanasse (1982), Florian, Lenstra, and Rinnooy Kan (1980). Florian et al. (1980) assumed that production and holding cost functions were continuous, nondecreasing and time-varying. They showed that this problem was NP-hard for quite general objective functions and provided a pseudo-polynomial dynamic programming algorithm. Some research has been done to identify polynomially solvable special cases. Bitran and Yanasse (1982) studied the computational complexity of the problem with linear production and holding cost functions under constant production capacity. Florian and Klein (1971) proposed an $O(T^3)$ dynamic programming algorithm to solve the production planning problem with constant production capacity and concave and inventory holding cost functions. Janannathan and Rao (1973) extended their results to a more general production cost function which was neither concave nor convex. Van Hoesel and Wagelmans (1996) proposed an $O(T^3)$ algorithm for a model with constant production capacity, concave production cost and linear holding cost functions. Baker, Dixon, Magazine, and Silver (1978) devised an $O(2^T)$ algorithm for the case of time-varying production capacity and setup cost, time-varying holding and production cost functions. Chung and Lin (1988) obtained an $O(T^2)$ dynamic programming algorithm for the problem with linear production and holding cost functions where setup cost and unit production cost are nonincreasing in time and the capacities are nondecreasing in time. Recently, for the same problem, Van den Heuvel and Wagelmans (2006) also proposed a more efficient algorithm based on the geometrical interpretation of a dynamic programming algorithm.

In production planning context, concave cost functions arise much more frequently than convex cost functions. Problems with convex cost functions are also much more complex. This perhaps explains why most of the papers deal with linear and/or concave functions. Lee and Nahmias (1993) indicated that concavity occurs when there are declining marginal production costs or other scale economies. Veinott (1964) developed a parametric algorithm for the problem with convex production and holding cost functions. Backlogging was not permitted in his model. He showed that an optimal plan could be obtained by satisfying each unit of demand in turn as cheaply as possible. Recently, Feng, Chen, Kumar, and Lin (2011) showed that a problem with convex inventory cost can be solved in $O(T \log T)$ time if the setup cost is nonincreasing and the production capacity is constant.

For the single-item lot-sizing problem with outsourcing, very few papers have been published. Sandbothe and Thompson (1990) presented a capacitated lot-sizing model with stockouts, in which all cost parameters are time-independent. They proved necessary conditions for an optimal solution with constant costs and presented an $O(T^3)$ algorithm for the problem with time-varying production capacity for the first $k+1$ periods with computable $k$. Later, Sandbothe and Thompson (1993) considered the same problem but with inventory capacities. A forward algorithm was proposed and the amount of computational effort was only multiplied by a factor of 4 in contrast to the computational complexity of the earlier model. Aksen et al. (2003) considered an uncapacitated single-item lot-sizing problem with outsourcing and proposed an $O(T^2)$ forward recursive dynamic programming algorithm with time-varying linear production, inventory and lost sales costs. Atamtürk and Hochbaum (2001) showed that the model with concave production and outsourcing cost functions, constant production capacity, unlimited outsourcing level and no backlogging can be solved in $O(T^3)$ time. They also investigated other polynomial solvable models with non-speculative cost functions. Chu and Chu (2007) developed an $O(T^2 \log T)$ algorithm to solve the models with no backlogging.
linear holding and outsourcing cost functions, fixed charge linear production cost functions. The problem considered in this paper involves inventory capacity. Furthermore, both outsourcing and backlogging are allowed. To the best of our knowledge, this is the first paper that simultaneously considers the outsourcing and backlogging decisions with inventory capacity. It extends the model of Chu and Chu (2007). In fact, when the limit of backlogging is set to zero we obtain the model of Chu and Chu (2007). Thereby, the complexity of the algorithm to solve this model should also be higher. We show that the model can be solved in $O(T \log T)$ time, if the production cost functions are linear but with a fixed cost component, and the holding/backlogging and outsourcing cost functions are linear. At present, production, holding, backlogging and outsourcing are common managerial policies for firms.

The remainder of this paper is outlined as follows. Section 2 gives the mathematical formulation of the model under study, shows related properties. Section 3 presents mathematical description of subplans, and develops an algorithm to compute optimal subplans and to solve the global problem. Section 4 ends the paper with some concluding remarks.

2. Mathematical formulation and general property

2.1. Mathematical formulation

The following notation will be used throughout the paper.

- $T$: Number of periods, indexed from 1 to $T$, involved in the planning horizon
- $d_t$: Demand of period $t$
- $I_t$: Inventory level at the end of period $t$
- $S_{t-1}$: Lower limit of the storage level ($S_{t-1} \leq 0, -S_{t-1}$ is the upper limit of the backlogging level) at period $t$
- $S_{t,1}$: Upper limit (capacity) of the storage level at period $t$ ($S_{t,1} \geq 0$)
- $\phi(\cdot)$: Inventory holding/backlogging cost function at period $t$
- $K_t$: Set-up cost associated with a production at period $t$
- $x_t$: Production level at period $t$
- $\lambda_t$: Unit out-sourcing cost at period $t$
- $L_t$: Outsourcing level at period $t$

where $x_t$’s and $I_t$’s are decision variables and $I_t$’s are state variables, $t = 1, 2, \ldots, T$.

In the remainder, we set $S_{t,0} = 0$ for $t = 0, 1, \ldots, T$ in order to uniform the analysis. As a result, we have $S_{t-1} \leq S_{t,0} \leq S_{t,1}$ for all $t = 0, 1, \ldots, T$. A time period $t$ is called backlogging period if $S_{t-1} \leq I_t - S_{t,0}$, and holding period if $S_{t,0} \leq I_t \leq S_{t,1}$.

We also make the following realistic assumptions:

- The inventory levels in periods 0 and $T$ are equal to zero, which is equivalent to setting $S_{0,0} = S_{0,1} = 0$ and $S_{T,0} = S_{T,1} = 0$.
- $d_t \geq 0, t = 1, 2, \ldots, T$.

Chu and Chu (2007) showed that the conservation outsourcing models with unbounded outsourcing level can be transformed into an inventory/backlogging model without outsourcing. In this paper, we assume that both backlogging and outsourcing are permitted, but outsourcing level at each period cannot exceed the unmet demand until that period. In other words, the backlogged quantities can also be met by outsourcing, not only by production. The mixed backlogging and outsourcing model with bounded inventory can be formulated as follows.

Define $\sigma_{t,v} \leq S_{t,v} + d_{0,t}$ and $S_{t,v}$ is called actuated inventory bound.

2.2. A general property

We now introduce some definitions and prove a general property similar to those in Love (1973).

**Definition 1.** If $x_t = 0$, $t$ is called a production period.

**Definition 2.** If $I_t = S_{t,v}, t \in [-1,0,1], t$ is an inventory point.
Note that by the previous assumptions, periods 0 and $T$ are inventory points. The following theorem plays a basic role in developing the algorithm.

**Theorem 1.** There is an optimal solution such that between two adjacent production periods there is at least one inventory point. Therefore, between two adjacent inventory points, there is at most one production period.

**Proof.** Similar to the proof of Property 3 in Chu and Chu (2007).

From this theorem, an optimal plan can be decomposed into subplans, each subplan being defined between two (not necessarily adjacent) inventory points involving exactly one production period. The problem can be solved by first computing all possible subplans and then searching the best combination or concatenation of subplans.

Note that it is possible that, in an optimal solution, there is no production within the entire planning horizon, and all of the demand is outsourced. This special case can be solved easily in $O(T^2)$ time, since the problem can be transformed into an inventory/backlogging model by considering a virtual production quantity which is actually equal to the outsourcing level. This transformed problem can be solved with the algorithm presented in Chu and Chu (2007). In this paper, we will focus on the more complex situation with at least one production period in the planning horizon.

In the remainder, let $(i, u, j, k, v)$ with $0 \leq i < j < k \leq T$ and $u, v \in \{-1, 0, 1\}$ denote a subplan with a production period $j$ between two inventory points $i$ and $k$ such that $I_i = S_{u, v}$ and $I_k = S_{k, v}$.

### 3. A polynomial algorithm

In this section, we show how to obtain an optimal solution of subplan $(i, u, j, k, v)$. To reduce the complexity, we first prove some properties of an optimal solution.

#### 3.1. Mathematical description of subplans

In any feasible solution of subplan $(i, u, j, k, v)$, state variables $l_i$ have the following property proved in Appendix A.

**Property 1.** If there exists a period $\tau$ such that $i < \tau < j$ (resp. $j < \tau < k$) and $l_\tau \leq 0$, then for any $\tau$ such that $i < \tau < j$ (resp. $\tau < t < k$) we have $l_t \leq 0$.

This property enables us to detail the mathematical description of the subplan. For the subplan $(i, u, j, k, v)$, let $l$ be the last period $t$ such that $i < t < j$ and $l_t \geq 0$. We have $l_0 \leq 0$ for all $l < t < j$. Let $n$ denote the last period $t$ such that $t \leq t < k$ and $l_t \geq 0$. We have $l_k \leq 0$ for all $n < t < k$. In practice, without loss of generality, we set: (i) $l = i$, if $l_m \leq 0$; (ii) $l = j - 1$ if $l_{j-1} \geq 0$, and $n = j - 1$ if $l_j < 0$; (iii) $n = k$ if $l_k \geq 0$. Obviously, we have $l = i$ if $u \in \{0, -1\}$ and $n = k$ if $u \in \{0, 1\}$. Therefore, for a given $h$, we have $0 \leq l < j$ and $j - 1 \leq n \leq T$. Moreover, when $l$ and $n$ are known, there are $0 \leq i \leq a_1$ if $u = 1$ and $0 \leq i \leq a_2$ if $u = 0$ and $n < k \leq T$. If $a_i$ and $b_n$ are undefined, we set $a_i = -1$ and $b_n = T + 1$, without loss of generality. In addition, all $a_i$’s and $b_n$’s can be computed in $O(T)$ time.

To reflect $l$ and $n$ in the notation of subplans, we denote a subplan by $(i, u, j, i, n, k, v)$ in the rest of the paper. Thus, all the subplans can be modeled by the following general mathematical formulation.

$$C_{u, v}(l, j, n, k) = K_j + \min \left\{ \pi_i x_i + \sum_{t=i}^{k} l_t + \sum_{t=i}^{n} g_{l_t} + \sum_{t=m}^{n} h_{l_t} - \sum_{t=m}^{n} g_{l_t} \right\}$$

subject to

$$\begin{align}
I_m &= \sigma_{u,v} - d_{0,m} + \sum_{t=i}^{m} L_t, \quad m = i + 1, \ldots, j - 1 \quad (9) \\
I_m &= \sigma_{j,n} - d_{0,m} + \sum_{t=i}^{m} L_t, \quad m = j, \ldots, n, \ldots, k \quad (10) \\
L_m &\geq 0, \quad m = i + 1, \ldots, k \quad (11) \\
L_m &\leq d_m, \quad m = i + 1, \ldots, l + 1 \text{ or } m = j + 1, \ldots, n + 1 \quad (12) \\
L_m &\leq L_{m-1} + d_m, \quad m = l + 2, \ldots, j - 1 \text{ or } m = n + 2, \ldots, k \quad (13) \\
0 &\leq L_m \leq S_{m,1}, \quad m = i + 1, \ldots, l \text{ or } m = j, \ldots, n \quad (14) \\
S_{m-1} &\leq L_m \leq 0, \quad m = l + 1, \ldots, j - 1 \text{ or } m = n + 1, \ldots, k \quad (15) \\
L_k &= S_{k,1} \quad (16) \\
x_j &\geq 0 \quad (17) \\
L_j &\leq \max(-L_{j-1}, 0) + d_j \quad (18)
\end{align}$$

where $x_j$ and $L_m$ are decision variables, $I_m$ are state variables, $m = i + 1, \ldots, k$. Constraints (10) and (16) imply that $x_j = \sigma_{j,n} - \sigma_{u,v} - \sum_{t=m}^{n} l_t$. Therefore, variable $x_j$ can be eliminated, constraint (16) can be removed and state Eq. (10) can be rewritten as

$$I_m = \sigma_{j,n} - d_{0,m} - \sum_{t=m}^{k} L_t, \quad m = j, \ldots, n, \ldots, k \quad (19)$$

From (9) and (19), constraints (13) can respectively be rewritten as

$$\begin{align}
L_m &\geq -L_{m-1} + d_m = -\sigma_{i,u} + d_{0,m} - \sum_{t=i}^{m-1} L_t, \quad m = l + 2, \ldots, j - 1 \\
L_m &\leq -L_{m-1} + d_m = -\sigma_{k,v} + d_{0,m} + \sum_{t=m}^{k} L_t, \quad m = n + 2, \ldots, k
\end{align}$$

The two inequalities above can be further written as

$$\begin{align}
\sum_{t=m}^{m-1} L_t &\leq d_{0,m} - \sigma_{i,u}, \quad m = l + 2, \ldots, j - 1 \quad (20) \\
\sum_{t=m}^{m-1} L_t &\geq \sigma_{k,v} - d_{0,m}, \quad m = n + 2, \ldots, k \quad (21)
\end{align}$$

In addition, from (9) and (19), the constraints (15) are equivalent to the following:

$$\begin{align}
\sigma_{m-1} - \sigma_{i,u} &\leq \sum_{t=m}^{m-1} L_t \leq d_{0,m} - \sigma_{i,u}, \quad m = l + 1, \ldots, j - 1 \quad (22) \\
\sigma_{k,v} - d_{0,m} &\leq \sum_{t=m}^{m-1} L_t \leq \sigma_{k,v} - \sigma_{m-1}, \quad m = n + 1, \ldots, k \quad (23)
\end{align}$$

It is obvious that the inequalities (22) and (23) make relations (20) and (21) redundant. So constraints (13) are redundant, and can be neglected. From constraints (11) and (18), we have

$$0 \leq L_j \leq \max(-L_{j-1}, 0) + d_j \quad (24)$$

Moreover, at optimality $L_j$ obeys the following theorem proved in Appendix B.
Theorem 2. There is an optimal solution in which $L_j = 0$ if $\pi_j \leq \lambda_j$ and $L_j = \max (-L_{i-1}, 0) + d_t$ otherwise.

From this theorem, the outsourcing level at period $j$ depends on three parameters: the inventory level at period $j - 1$, the unit production and the outsourcing cost at period $j$. Therefore, the objective function can be restated as

$$C_{u,i}(i, l, j, n, k) = A_u(i, l, j) + B_v(j, n, k)$$

where

$$A_u(i, l, j) = \sigma_{iu}(h_{i-1} - g_{i-1,j-1} - \pi_j) - H_{i,j} + G_{i,j} + \left(\pi_j - \sigma_{iu}\right)l_j$$

$$+ \min \left\{ \sum_{t=1}^{l-1} \left(\pi_j + h_{i-1,t} - g_{i-1,j-1}\right) l_t + \sum_{t=1}^{l} \left(\pi_j - \sigma_{iu}\right) l_t \right\}$$

(25)

$$B_v(j, n, k) = K_v + \sigma_{iv}(\pi_j + h_{j-1,n} - g_{n,k}) - H_{j-1,n} + G_{n,k}$$

$$+ \min \left\{ \sum_{t=1}^{n} \left(\pi_j - \pi_j - h_{j-1,t} - g_{n,k}\right) l_t + \sum_{t=1}^{n} \left(\pi_j - \pi_j - h_{j-1,n} + g_{n,k}\right) l_t \right\}$$

(26)

To simplify the notation, let

$$\zeta_u(t, l, j) = \left\{ \begin{array}{ll}
\lambda_t - \pi_j - h_{j-1,t} - g_{i-1,j-1}, & i + 1 \leq t \leq l \\
\lambda_t - \pi_j - h_{j-1,t}, & l + 1 \leq t \leq j - 1
\end{array} \right.$$

$$\eta_v(t, n, j) = \left\{ \begin{array}{ll}
\lambda_t - \pi_j - h_{j-1,n} - g_{n,k}, & j \leq t \leq n \\
\lambda_t - \pi_j - h_{j-1,n} + g_{n,k}, & n + 1 \leq t \leq k
\end{array} \right.$$

After the relaxation of constraint (17), the problem can be decomposed into two independent sub-problems $P_u(i, l, j)$ and $Q_v(j, n, k)$ defined as follows.

Problem $P_u(i, l, j)$

$$A'_u(i, l, j) = \min \sum_{t=1}^{l} \zeta_u(t, l, j) l_t$$

subject to

$$L_m \geq 0, \quad m = l + 2, \ldots, j - 1$$

$$0 \leq L_m \leq d_m, \quad m = i + 1, \ldots, l + 1$$

$$d_{0,m} - \sigma_{iu} \leq \sum_{t=1}^{m} l_t \leq \sigma_{m-1} - \sigma_{iu}, \quad m = i + 1, \ldots, l$$

$$\sigma_{m-1} - \sigma_{iu} \leq \sum_{t=1}^{m} l_t \leq d_{0,m} - \sigma_{iu}, \quad m = i + 1, \ldots, j - 1$$

Problem $Q_v(j, n, k)$

$$B'_v(j, n, k) = \min \sum_{t=1}^{k} \eta_v(t, n, j) l_t$$

subject to

$$L_m \geq 0, \quad m = n + 2, \ldots, k$$

$$0 \leq L_m \leq d_m, \quad m = j + 1, \ldots, n + 1$$

$$\sigma_{j+1} - \sigma_{m-1} \leq \sum_{t=1}^{m} l_t \leq \sigma_{k+1} - d_{0,m}, \quad m = j, \ldots, n$$

$$\sigma_{k+1} - d_{0,m} \leq \sum_{t=1}^{m} l_t \leq \sigma_{j+1} - \sigma_{m-1}, \quad m = n + 1, \ldots, k$$

The only link between these two sub-problems is

$$X_t = \sigma_{j+1} - \sigma_{iu} - \sum_{m=i+1}^{j} l_m$$

$$= \left[ \sigma_{j+1} - d_{0,j+1} - \sum_{m=j+1}^{k} l_m \right] - \left[ \sigma_{iu} - d_{0,j+1} + \sum_{m=i+1}^{j} l_m \right] > 0$$

(27)

Obviously, $P_u(i, l, j)$ and $Q_v(j, n, k)$ are linear programs. In the following, we show that due to their special structure, they can be efficiently solved in polynomial time.

3.2. Solving $P_u(i, l, j)$

For the benefit of clarity, problem $P_u(i, l, j)$ can be rewritten as follows:

$$A'_u(i, l, j) = \min \sum_{m=1}^{l} \zeta(m, l, j) l_m$$

subject to

$$0 \leq L_m \leq \delta_m, \quad m = i + 1, \ldots, j - 1$$

$$R_u(i, l, m) = \sum_{t=1}^{m} L_t < R_u(i, l, m), \quad m = i + 1, \ldots, j - 1$$

where

$$R_u(i, l, m) = \left\{ \begin{array}{ll}
d_{0,m} - \sigma_{iu}, & m = i + 1, \ldots, l \\
\sigma_{m-1} - \sigma_{iu}, & m = i + 1, \ldots, j - 1
\end{array} \right.$$

$$R_v(i, l, m) = \left\{ \begin{array}{ll}
\sigma_{j+1} - \sigma_{m-1}, & m = i + 1, \ldots, j \\
\sigma_{k+1} - d_{0,m}, & m = i + 1, \ldots, j - 1
\end{array} \right.$$

$$\delta_m = \left\{ \begin{array}{ll}
d_{0,m}, & m = i + 1, \ldots, l \\
+\infty, & m = i + 1, \ldots, j - 1
\end{array} \right.$$

By definition, $R_u(i, l, m)$ and $R_v(i, l, m)$ represent respectively the lower and upper bounds of the cumulative outsourcing level from period $i + 1$ to period $m$. Without loss of generality, we set $R_v(i, l, l) = R_u(i, l, l)$.

We now establish some properties of an optimal solution of $P_u(i, l, j)$. The arguments of these properties are similar to those presented in Chu and Chu (2007), so we neglect them.

Property 2. For any $\tau$ such that $i + 1 \leq \tau < j$ and $\zeta_u(\tau, i, j) \geq 0$, there is an $m$ such that $\tau \leq m < \tau + \left(\sum_{t=1}^{m} L_t - R_u(i, l, m)\right)$.

Property 3. For any $\tau$ and $\eta$ such that $i < \tau < \eta$ and $\zeta_u(\tau, i, \eta) \geq 0$, there is an $m$ such that $\tau < m < \eta$ and $\sum_{t=1}^{m} L_t - \sum_{t=1}^{m} L_t \geq 0$.

Property 4. For any $\tau$ such that $i < \tau < j$ and $\zeta_v(\tau, i, j) < 0$, there is an $m$ such that $\tau < m < j$ and $L_m - \delta_m \geq 0$.

Property 5. For any $\tau$ and $\eta$ such that $i < \tau < \eta$ and $\zeta_v(\tau, i, \eta) < 0$, there is an $m$ such that $\tau < m < \eta$ and $\left(\sum_{t=1}^{m} L_t - \sum_{t=1}^{m} L_t\right)$.

Let the solution of $P_u(i, l, j)$ and $P_v(i, l, j)$ be an optimal solution verifying the properties above. We have the following theorem which is proved in Appendix C.

Theorem 3. Given $i$ and $l$, we have $\omega_m(i, l, j) \geq \omega_m(i + 1, l, j)$ for any $m$ such that $i + 2 \leq m \leq j - 1$.

When $u \in [0, -1]$, Theorem 3 can be restated as follows: “For a given $j$, we have $\omega_m(i, l, j) \geq \omega_m(i + 1, l, j)$ for any $m$ such that $i + 2 \leq m \leq j - 1$.”

From this theorem, it is only necessary to consider solutions $P_u(i, l, j)$ such that $L_i \geq 0$ for any $t$ such that $i + 1 \leq t < j$, without loss of generality. In other words, an additional constraint $L_i \geq 0$ for any $t$ such that $i + 1 \leq t < j$ can be added without missing optimal solutions. By a variable substitution $e_t = L_t - \omega_{i+1,l,j}$ to denote the required additional amount to be outsourced at period $t$ in $P_u(i, l, j)$ with
respect to $P_d(i+1,l,j)$, $t = i + 1, \ldots, j - 1$, and knowing an optimal solution to problem $P_d(i+1,l,j)$, namely $\{i_{l-1} + 1,j\}$. Problem $P_d(i,l,j)$ can be reformulated as follows.

Minimize $A_{l,j}(i,l,j) = \sum_{t=i+1}^{j-1} \zeta_{\bar{i},(l,j)}(t,l,j) \epsilon_t + \sum_{t=i+1}^{j-1} \zeta_{\bar{i},(l,j)}(t,l,j) \sigma_{\bar{i},(l+1,j)}$ (28) subject to

$$0 \leq \epsilon_t \leq \bar{\epsilon}, \quad m = i + 1, \ldots, j - 1$$

$$\sum_{t=i+1}^{j-1} \epsilon_t \geq R(i,l,m) - \sum_{t=i+1}^{j-1} \sigma_{\bar{i},(l+1,j)}(t,l,j), \quad m = i + 1, \ldots, j - 1$$

$$\sum_{t=i+1}^{j-1} \epsilon_t \leq R(i,l,m) - \sum_{t=i+1}^{j-1} \sigma_{\bar{i},(l+1,j)}(t,l,j), \quad m = i + 1, \ldots, j - 1$$

where $\epsilon_t, i + 1 \leq t \leq j - 1$ are decision variables and

\[
\bar{\epsilon} = \eta - \sigma_{i,l} \geq 0, \quad i + 1 \leq t \leq j - 1
\]

denotes the quantity not yet outsourced at period $t$ in $P_d(i+1,l,j)$. In this formulation, the second term of the objective function is independent of the decision variables. Therefore, it is dropped in the sequel of the optimization process. By definition, for any $m$ such that $i < m < j$, we have

$$R(i,l,m) - \sum_{t=i+1}^{j-1} \sigma_{\bar{i},(l+1,j)}(t,l,j) \leq R(i,l,m) - R(i+1,l,m)$$

$$= \sigma_{i-1,l} - \sigma_{i,l} = R(i,l,m) - R(i+1,l,m) - \sum_{t=i+1}^{j-1} \sigma_{\bar{i},(l+1,j)}(t,l,j)$$

This relation leads to the following property which is similar to Property 8 in Chu and Chu (2007) and therefore can also be proved in a similar way.

Property 6. There is an optimal solution such that $\sum_{t=i+1}^{j-1} \epsilon_t \leq \sigma_{i-1,l} \leq \sigma_{i,l}$.

From this property and relation (32), constraint (31) can be replaced by

$$\sum_{t=i+1}^{j-1} \epsilon_t \leq \sigma_{i-1,l} \leq \sigma_{i,l}$$

Let $W$ be the set of $t$'s such that $\epsilon_t > 0$ and $q$ a permutation of these $t$'s such that $\zeta_{\bar{i},(q(1),l,j)} \leq \zeta_{\bar{i},(q(2),l,j)} \leq \cdots \leq \zeta_{\bar{i},(q(W),l,j)}$. As usual, let $q^{-1}(t)$ be the position of $t$ in this permutation. The problem defined by (28)-(31) is equivalent to the following:

Minimize $\sum_{m=1}^{W} \zeta_{\bar{i},(q(m),l,j)}(m) \epsilon_{q(m)}$ subject to $0 \leq \epsilon_{q(m)} \leq \bar{\epsilon}, \quad 1 \leq m \leq |W|$ $\sum_{t=1}^{W} \epsilon_t \geq R(i,l,m) - \sum_{t=1}^{W} w_t(i+1,l,j), \quad 1 \leq m \leq |W|$ $\sum_{t=1}^{W} \epsilon_t \leq \sigma_{i-1,l} \leq \sigma_{i,l}$

Now we can present an algorithm to solve problem $P_d(i,l,j)$. The basic idea of the algorithm is to first ensure that the lower bound of cumulative outsourcing level at each period is satisfied, and then seek the optimal policies so that the total cost is minimal. Moreover, from (32), it cannot alter the feasibility of the problem that the lower bound of cumulative outsourcing level from period $i + 1$ to $m$ is satisfied first as long as the first subproblem is feasible at the beginning of the iteration. It can be easily seen that when $u \in [0,-1)$ the first subproblem $P_d(j - 2,j - 2,j)$ must be feasible due to the fact that $\zeta_{\bar{i},(j-2,j-2,j)}(i,j) = \sigma_{i-1,l} \leq \sigma_{i-1,l}$ to ensure the feasibility. For a given $j (2 \leq j \leq T)$ and a given $t (0 < l < j)$, the following algorithm can simultaneously solve all $P_d(i,l,j)$'s such that $0 \leq i < a_l$ or $0 < l < j - 2$ in $O(T^2 \log T)$ time. In this algorithm, $w$ is the head of the list $W$ of $t$'s arranged in nondecreasing order of $\zeta_{\bar{i},(q,l,j)}$'s and such that $\epsilon_t > 0$. $TOSL[m]$ (which stands for total outsourcing level) is the current cumulative outsourcing level from period $i + 1$ to $m$. $\text{Sum E}$ is the current value of $\sum_{m=1}^{W} \zeta_{\bar{i},(q,m),l,j)}(m) \epsilon_{q(m)}$ and $\text{TotalCost}$ is the current value of $A_{l,j}(i,l,j)$.

Algorithm 1.

1. $\epsilon_t := \bar{\epsilon}$, for any $t = 1, \ldots j$; $\text{TotalCost} := 0$. If $u \in [0,-1)$, $i:= 2$; otherwise $i := \min\{j-2,a_l\}$. $\rho(i,l,j)$ for $i = i + 1, \ldots, j - 1$.

2. If $u = 1$ then

while $i \geq 0$ and $\max\{R(i,l,m) \zeta_{\bar{i},(i,l,j)} - \sigma_{i+1,l} \zeta_{\bar{i},(i,l,j)}\} > \sigma_{i+1,l} - \sigma_{i,l}$ do

i. if $\sigma_{i} > 0$ then insert $m$ into $W$ in nondecreasing order of $\zeta_{\bar{i},(q,l,j)}$'s.

ii. while $\text{Sum E} \leq R(i,l,m) - \text{TOSL}[m]$ do

A. $\text{de} := \min\{\epsilon_t, R(i,l,m) - \text{TOSL}[m] - \text{Sum E}\}$ (note that we have necessarily $w \leq m$).

B. $w := w - \text{de}$, $\epsilon_{q(m)} := \epsilon_{q(m)} + \text{de}$, $\text{Sum E} := \text{Sum E} + \text{de}$, $\text{TotalCost} := \text{TotalCost} + \epsilon_{q(m)} \text{w}$. $\text{de}$

C. if $\epsilon_{q(m)} = 0$ then remove $m$ from $W$. end while

end for

(c) while $W \neq \emptyset$ and $\zeta_{\bar{i},(q,l,j)} < 0$ and $\text{Sum E} < \sigma_{i+1,l} - \sigma_{i,l}$ do

i. $\epsilon_t := \min\{\epsilon_t, \sigma_{i+1,l} - \sigma_{i,l} - \text{Sum E}\}$

ii. $\epsilon_t := \epsilon_t - w$, $\epsilon_{q(m)} := \epsilon_{q(m)} + \text{de}$, $\text{Sum E} := \text{Sum E} + \text{de}$, $\text{TotalCost} := \text{TotalCost} + \epsilon_{q(m)} \text{w}$. $\text{de}$

iii. if $\epsilon_{q(m)} = 0$ then remove $m$ from $W$. end while

(d) $i_{l,j} := \sigma_{i+1,l} - \sigma_{i,l} + \text{TOSL}[j] + 1 - \text{Sum E}$

(e) if $\rho_{l_{l,j}} > \rho_{l_{j-1}}$, then $L_j := \max\{L_{j-1} - 1, 0\}$; otherwise $L_j := 0$.

(f) $A_{l,j}(i,l,j) := \text{TotalCost} - R_{l,j} + \sigma_{i,l} \rho_{l_{j-1}, (l_{j}, j)} - \rho_{l_{j-1}} - \rho_{l_{j}} \rho_{l_{j-1}} - \rho_{l_{j}}$ (g) $i_{l,j} := i_{l,j} - 1$

end while

At step 3a, the outsourcing levels $\sigma_{i,l+1,j}$, $i + 2 \leq t \leq j$ are initialized to the value of $\rho_{l_{j-1,l,j}}$ according to Theorem 2 (i.e. $\rho_{l_{j-1,l,j}} \rho_{l_{j}, l+1,j}$, $i+2 \leq t \leq j$). This step requires $O(j)$ time. Step 3b ensures that the lower bound of the cumulative outsourcing level at each period is satisfied. If $\text{de} = \epsilon_{q(m)}$, then the new value of $\epsilon_{q(m)}$ is 0, and $W$ will be removed from $W$ forever. If $\text{de} = \epsilon_{q(m)}$ - $\epsilon_{q(m)}$, then $\text{TOSL}[m]$ will be updated. The value of $\text{Sum E}$ is updated and the condition $\text{Sum E} < R(i,l,m) - \text{TOSL}[m]$ becomes false. The insertion
of each new \(m\) into \(W\) and the removal of each \(m\) from \(W\) require \(O(\log j)\) time. There are at most \(j\) insertions or removals. Hence, the complexity is \(O(\log j)\).

At step 3c, if \(c_i(w,l_j) < 0\) and Sum \(E < \sigma_{i+1,u} - \sigma_{i,u}\), the outsourcing level at period \(w\) is increased to reduce the total cost. During the recursive computation, if \(de = e_{w}\), \(w\) will be removed from \(W\) forever; if \(de:=\sigma_{i+1,u} - \sigma_{i,u} - Sum\( E\), the value of \(Sum\ E\) is updated and the condition \(Sum \(E < \sigma_{i+1,u} - \sigma_{i,u}\) becomes false and Problem \(P_0(l_j)\) is solved. Here, the number of removals is at most \(j\). The complexity is also \(O(\log j)\).

To sum up, there are at most \(j\) insertions and removals for each \(i\). As a consequence, a single iteration requires at most \(O(\log j)\) time. Consequently, for given \(j\) and \(l\), computing all \(A_i(l_j)\)'s with \(0 < i < \min\{j, 2a_j\}\) requires \(O(T^2\log T)\) time. As a result, since there are \(O(T^2)\) couples of \((j,\ l)\), the total computational complexity is \(O(T^3\log T)\) to compute all the \(A_i(l_j)\)'s. However, all \(A_0(l_j)\)'s with \(u \in \{0, -1\}\) can be calculated in \(O(T^3\log T)\) time.

In practice, it is possible to produce at the first period (i.e., \(j = 1\)). In this case, we have \(l_i = 0\) and the value of \(A_0(l_j)\)'s can be computed in constant time. In fact, we have \(L_1 = d_1\), if \(x_1 > l_1\), and \(L_1 = 0\) otherwise and \(A_0(l_1) = (\bar{x}_1 - \bar{x}_1)\).

3.3. Solving \(Q_l(j, n, k)\)

We show that all of the results obtained for \(P_0(l_j)\) still hold for problem \(Q_l(j, n, k)\), after adopting the techniques similar to those in Section 3.2. All \(Q_l(j, n, k)\)'s can be solved using a variant of Algorithm 1 in \(O(T^3\log T)\) time.

Recall the definition of problem \(Q_l(j, n, k)\) as follows

Minimize \(\sum_{t=1}^{j} \eta(t, n, j) l_t\)

subject to

\(0 < L_m \leq \delta_m, \quad m = j + 1, \ldots, k\)

\(A_v(m, n, k) = \sum_{t=m+1}^{k} l_t \leq \zeta_v(m, n, k), \quad m = j, \ldots, k - 1\)

where

\(A_v(m, n, k) = \begin{cases} \sigma_{k,v} - \sigma_{m,v}, & j < m \leq n \\ \sigma_{k,v} - d_{m,v}, & n < m \leq k \end{cases}\)

\(\zeta_v(m, n, k) = \begin{cases} \sigma_{k,v} - \sigma_{m,v}, & j < m \leq n \\ \sigma_{k,v} - d_{m,v}, & n < m \leq k \end{cases}\)

\(\delta_m = \begin{cases} d_{m,v}, & j < m \leq n + 1 \\ +\infty, & n + 2 \leq m \leq k + 1 \end{cases}\)

Similarly, \(A_v(m, n, k)\) and \(\zeta_v(m, n, k)\) respectively represent the lower and upper bounds of the cumulative outsourcing level from period \(m + 1\) to period \(k\). Without loss of generality, we also set \(A_v(k, n, k) = \zeta_v(m, n, k) = 0\).

As shown below, problem \(Q_l(j, n, k)\) has a similar structure to \(P_0(l_j)\). Similar properties for \(P_0(l_j)\) also hold for \(Q_l(j, n, k)\) and can be proved in a similar way.

Property 7. For any \(\tau\) such that \(j < \tau < k\) and \(\eta_{\tau}(\tau, n, j) \geq 0\), we have

\(L_1 = \sum_{t=m+1}^{k} l_t - \zeta_v(m, n, k) = 0\)

for some \(m\) such that \(j < m < \tau\).

Property 8. For any \(\tau\) and \(t\) such that \(j < \tau < t < k\) and \(\eta_{\tau}(\tau, n, j) \geq 0\), we have

\(L_1 = \sum_{t=m+1}^{k} l_t - \zeta_v(m, n, k) = 0\)

for some \(m\) such that \(j < m < \tau\).

Property 9. For any \(\tau\) such that \(j < \tau < k\) and \(\eta_{\tau}(\tau, n, j) < 0\) we have

\(L_1 = \sum_{t=m+1}^{k} l_t - \zeta_v(m, n, k) = 0\)

for some \(m\) such that \(j < m < \tau\).

Property 10. For any \(\tau\) and \(t\) such that \(j < \tau < t < k\) and \(\eta_{\tau}(\tau, n, j) \leq 0\), we have

\(L_1 = \sum_{t=m+1}^{k} l_t - \zeta_v(m, n, k) = 0\)

for some \(m\) such that \(j < m < \tau\).

Let \(Q_l(j, n, k) = (\omega_{j+2}(j, n, k), \omega_{j+2}(j, n, k), \ldots, \omega_{j+2}(j, n, k))\) be an optimal solution verifying the properties above. The following theorem is similar to Theorem 3, and can be proved in a similar way.

Theorem 4. Given \(j\) and \(n\) \((j - 1 < n < T)\), we have \(\omega_{m+2}(j, n, k - 1) = \omega_{m+2}(j, n, k)\) for every \(m\) such that \(j < m < k\).

According to this theorem, when computing an optimal solution of \(Q_l(j, n, k)\), it is only necessary to consider those solutions in which \(L_1 \geq \omega_{m+2}(j, n, k - 1)\) for any \(t\) such that \(j < t < k\), where \(\omega_{m+2}(j, n, k - 1) = 0\), without loss of generality. By a variable substitution \(e_t = L_1 - \omega_{m+2}(j, n, k - 1)\) to denote the required additional amount to be outsourced at period \(t\) in \(Q_l(j, n, k)\) compared to \(Q_l(j, n, k - 1), t = j + 1, \ldots, k\), and knowing an optimal solution to problem \(Q_l(j, n, k - 1)\), namely \(U_l(j, n, k - 1)\), problem \(Q_l(j, n, k)\) can be reformulated as follows.

Minimize \(E_v(j, n, k) = \sum_{t=1}^{k} \eta(t, n, j) e_t + \sum_{t=1}^{k} \eta(t, n, j) e_{t+1}\)

subject to

\(0 \leq e_m = \epsilon_n, \quad m = j + 1, \ldots, k\)

\(\sum_{t=m+1}^{k} e_t \geq \zeta_v(m, n, k) - \sum_{t=m+1}^{k} \omega_{t+1}(j, n, k - 1)\)

\(m = j, \ldots, k\)

where \(e_t, j < t < k\) are decision variables and \(e_t \equiv e_t - \omega_{m+2}(j, n, k - 1) > 0, t = j + 1, \ldots, k\)

denotes the quantity not yet outsourced at period \(t\) in \(Q_l(j, n, k - 1)\).

In this formulation, the second term of the objective function is independent of the decision variables. Therefore, it is dropped in the sequel of the optimization process. By definition, for any \(m\) such that \(j < m < k\), we have

\(\zeta_v(m, n, k) - \sum_{t=m+1}^{k} \omega_{t+1}(j, n, k - 1)\)

\(\leq \zeta_v(m, n, k) - \zeta_v(m, n, k - 1) = \sigma_{k,v} - \sigma_{k-1,v}\)

\(= \zeta_v(m, n, k) - \zeta_v(m, n, k - 1)\)

\(\leq \zeta_v(m, n, k) - \sum_{t=m+1}^{k} \omega_{t+1}(j, n, k - 1)\)

This relation leads to the following property, the proof of which is similar to that of 6.

Property 11. There is an optimal solution such that

\(\sum_{t=m+1}^{k} e_t \leq \sigma_{k,v} - \sigma_{k-1,v}\)

From this property and relation (37), constraint (36) can be replaced by

\(\sum_{t=m+1}^{k} e_t \leq \sigma_{k,v} - \sigma_{k-1,v}\)

The problem defined by (33)–(36) is equivalent to a problem identical to the one studied in Section 3.2. As Algorithm 1, for given \(j\) and \(n\), the above algorithm can solve all \(Q_l(j, n, k)\)'s with \(b_n \leq k \leq T\).
in $O(T^2 \log T)$ time. As a result, computing all $B_j(n,k)$’s requires $O(T^3 \log T)$ time. However, all $B_d(n,k)$ with $v \in [0,1]$ can be computed in $O(T^3 \log T)$ time.

3.4. Handling the link between the sub-problems and solving the global problem

Here let us deal with the relaxed constraint (17). Let $\rho'_u(i,l,j)$ denote the value of $l_{j-1}$ in the optimal solution of $P_d(i,l,j)$. From (9), we have

$$
\rho'_u(i,l,j) = \left\{ \begin{array}{ll}
\sum_{t=1}^{l-1} o_t(i,l,j) - R_u(i,l,j-1) & \text{if } l = j - 1 \\
\sum_{t=1}^{l-1} o_t(i,l,j) - R_u(i,l,j-1) & \text{if } l < j - 1
\end{array} \right.
$$

Also let $\rho_u(i,l,j)$ be the value of $\rho'_u(i,l,j) + l'_u(i,l,j)$ in an optimal solution, where $l'_u(i,l,j)$ represents the optimal outsourcing level at period $j$. Define $r_d(i,l,j)$ as the value of $\delta_{x,v} - d_{o,l-1} - \sum_{k=m+1}^{j} l_m$ in the optimal solution of $Q_d(i,l,j)$. When the subplan is undefined; i.e., if $B_d(n,k) = \infty$, let $r_d(i,l,j) = -\infty$, without loss of generality. Hence, we have $x_j = r_d(i,l,j) - \rho_u(i,l,j)$.

The following Theorem proved in Appendix D shows that it is sufficient to consider the subplan $(i,u,j,k,v)$ with $\rho'_u(i,l,j) < r_d(i,l,j)$ in order to satisfy the constraint (17).

**Theorem 5.** There is an optimal solution in which for any subplan $(i,u,j,k,v)$ with a production period $j$, there are an $l$ and an $n$ such that $\rho_u(i,l,j) < r_d(i,l,j)$.

For given $j$, $n$, $k$, $l$, $u$ and $v$, we need to consider all $i$ with $0 \leq i < j$ to check whether the requirement $\rho_u(i,l,j) < r_d(i,l,j)$ is fulfilled. The following lemma and theorem provide us a way to improve the seeking procedure.

**Lemma 1.** For any $j$, $l$ and $i$ such that $2 \leq j \leq T$, $0 \leq l < j$, $0 \leq i < j - 2$ with $u \in [0,1)$ or $0 \leq i \leq \min(j - 2, a)$ with $u = 1$, we have $\rho_u(i,l,j) \leq \rho_u(i,l+1,j)$.

**Proof.** Similar to the proof of Theorem 5 in Chu and Chu (2007).

**Theorem 6.** For any $j$, $l$ and $i$ such that $2 \leq j \leq T$ and $0 \leq l < j$ and $0 \leq i \leq j - 2$ with $u \in [0,1)$ or $0 \leq i \leq \min(j - 2, a)$ with $u = 1$, we also have $\rho_u(i,l,j) \leq \rho_u(i+1,l,j)$.

**Proof.** See Appendix E.

In this procedure, the computation of the permutation $s_j,v$ requires $O(T \log T)$ time. At step 4b, either $m$ or $i$ is decremented at the end of each iteration. Since the number of $m$ is upper-bounded by $T - k + 1$ and that of $i$ upper-bounded by $j$, and $j$, there are at most $T$ iterations for a given $l$. Therefore, all $\beta_u(i,l,j,k,v)$’s can be computed respectively in $O(T^2 \log T)$, $O(T^3)$, $O(T^3 \log T)$ and $O(T^4)$ for the following cases: (i) $v \in [0,1)$ and $u \in [0,1)$. (ii) $v \in [0,1)$ and $u = 1$. (iii) $v = -1$ and $u \in [0,1)$, (iv) $v = -1$ and $u = 1$. From Theorems 5 and 6, for the subplan $(i,u,j,k,v)$ to belong to an optimal solution, we must have $i = \beta_u(i,l,j,k,v)$.

Therefore, if $E_d(k)$ denotes the minimal cumulative cost from period 0 through period $k$ such that $l_k = S_{k+1}$ with $0 \leq k < T$ and $v \in [1,0)$, the problem can be solved using the following recursive equations:

$$
E_d(k) = \min_{1 \leq j \leq k} \left\{ B_d(i,j,k) + \min_{1 \leq u \leq 1} \left[ E_u(i) + A_u(i,l,j) \right] \right\}
$$

An optimal solution corresponds to $E_d(T)$. By defining

$$
F_u(m,l,j) = \min_{0 \leq i < m} \left[ E_u(i) + A_u(i,l,j) \right]
$$

We have the following recursive equations

$$
F_u(m,l,j) = \min_{0 \leq i < m} \left[ F_u(m-1,l,j) + A_u(m,l,j) \right]
$$

$$
E_d(k) = \min_{1 \leq j \leq k} \left\{ B_d(i,j,k) + \min_{1 \leq l \leq j} \left[ F_u(m-1,l,j) + A_u(m,l,j) \right] \right\}
$$

with $F_u(-1,l,j) = \infty$, $E_u(0) = 0, u \in [1,0)$, $0 \leq l < j$ as starting conditions.

The equations above show that as soon as $A_u(i,l,j)$’s, $B_d(i,j,k)$’s and $\beta_u(i,l,j,k)$’s are known, it requires an additional amount of $O(T^3)$ time to obtain an optimal solution by computing $E_d(k)$’s and $F_u(m,l,j)$’s alternatively. Since all $A_u(i,l,j)$’s, $B_d(i,j,k)$’s can be computed in $O(T^3 \log T)$, and all $\beta_u(i,l,j,k)$’s can be computed in at most $O(T^4)$ time, the overall complexity is $O(T^4 \log T)$.

4. Implementation and conclusions

The proposed algorithm has been implemented in Visual C++ on a personal computer with an Intel Core Duo CPU of 2 GHz and 1.99 GB RAM. Its performance was compared with Cplex Academic version 12.1 running on the same computer with real-life industrial data in which a production plan should be established over a horizon of 78 weeks ($T = 78$); i.e. 18 months, due to the fact the suppliers need this information to build up their own production plans. All the 200 instances, considered to be representative by our industrial partner, are solved to optimality by both tools within negligible amounts of computation time. The computation time
ranges from 31 to 34 ms for the proposed algorithm and from 62 to 74 ms for Cplex.

In spite of the academic interest of the proposed algorithm, the savings in computation time are very limited with respect to commercial MIP solvers. However, from practical point of view, few people in industry are familiar with such solvers and therefore a training of employees is necessary, in addition to the purchasing cost of the software. Furthermore, the method proposed in this paper can be easily implemented in Microsoft Excel, a popular spreadsheet software which is a front-end office automation standard in the business world. Such an implementation is possible since the values of all functions can be computed with Excel tables. These tables can be filled in at the same time as the problem data is collected. In other words, it is possible to obtain the optimal solution immediately after the problem data acquisition.

This paper extends the results given in Chu and Chu (2007) to a model with backlogging. Our future research consists of investigating multi-item models and production capacity. Another interesting topic is investigating the application of the method in a rolling horizon environment by taking into account the uncertainty of the demand forecast.

Appendix A. Proof of Property 1

From the constraints (5), \( L_k \leq \max(-L_{k-1},0) + d_t \). Since \( L_t \leq 0 \) and if furthermore \( t + 1 < j \) which implies that \( x_{t+1} = 0 \), we would have \( L_{t+1} = x_{t+1} + L_{t+1} - L_t = \max(-L_0,0) + L_t = 0 \). The same argument can be applied for any period \( t \) such that \( t < j \).

Appendix B. Proof of Theorem 2

From the definition of subplans, the outsourcing level at period \( j \) just confines to the constraint (24). When \( \pi_t \leq \lambda_t \), if \( L_j > 0 \) we can improve a solution by shifting \( L_j \) units from outsourcing to production at period \( j \) while satisfying all the constraints. When \( \pi_t > \ lambda_t \), from the assumptions of the problem, if \( 0 < L_j < \max(-L_{j-1},0) + d_t \), then we can reduce the production quantity while increasing the outsourcing quantity at period \( j \).

Appendix C. Proof of Theorem 3

We first consider \( u = 1 \). Assume that there is some \( t \) such that \( i + 1 < t < j \) and \( 0 < o_{i}(i,l,j) < o_{i}(i+1,l,j) \). Let \( t^{*} \) be the largest one among such \( t^* \)‘s. Let \( t^{*} \) be the largest one of those \( t^{*} \) such that \( i + 1 < t < j \) and \( o_{i}(i,l,j) > o_{i}(i+1,l,j) \), if any. If \( t^{*} \) is undefined, i.e., \( o_{i}(i,l,j) < o_{i}(i+1,l,j) \) for any \( t \) such that \( i + 1 < t < j \) with at least one inequality being strict, due to the fact that \( R_{i}(i,l,m) = R_{i}(i+1,l,m) = R_{i}(i,l,m) - R_{i}(i+1,l,m) = \sigma_{i+1,i} - \sigma_{i,i} \geq 0 \), we have

\[
\sum_{t=i+2}^{m} o_{i}(i+1,l,j) = \sum_{t=i+2}^{m} o_{i}(i+1,l,j) > \sum_{t=i+1}^{m} o_{i}(i,l,j) \geq R_{i}(i+1,l,m)
\]

for any \( m \) such that \( i < m < j \), and

\[
\sum_{t=i+1}^{m} o_{i}(i,l,j) \leq \sum_{t=i+2}^{m} o_{i}(i+1,l,j) = \sum_{t=i+2}^{m} o_{i}(i+1,l,j) \leq R_{i}(i+1,l,m) \]

for any \( m \) such that \( t^{*} < m < j \).

The fact that \( 0 < o_{i}(i,l,j) < o_{i}(i+1,l,j) \) implies that \( L(i+1,l,j) \) does not verify Property 2 if \( \varphi^{*}(i,l,j) \geq 0 \) and \( L(i,l,j) \) does not satisfy Property 4 if \( \varphi^{*}(i,l,j) < 0 \).

If \( \tau^{*} > t^{*} \) and \( \varphi^{*}(\tau^{*},l,j) \geq \varphi^{*}(t^{*},l,j) \), the facts that \( 0 < o_{\tau}(\tau+1,l,j) < o_{\tau}(\tau,l,j) \) and \( 0 < o_{\tau+1}(\tau+1,l,j) < o_{\tau+1}(\tau+2,l,j) \) imply, from Property 3, that there must be an \( m^{*} \) such that \( \tau^{*} < m^{*} < \tau^{*} \) and \( o_{m}(i,l,j) = R_{i}(i,l,m^{*}) \). For any \( m \) such that \( \tau^{*} < m < j \), we have

\[
\sum_{t=i+2}^{m} o_{t}(i+1,l,j) = \sum_{t=i+1}^{m} o_{t}(i+1,l,j) \\
= \sum_{t=i+1}^{m} o_{t}(i+1,l,j) + \sum_{t=m+1}^{m} o_{t}(i+1,l,j) \\
= R_{i}(i+1,l,m) \quad \text{(Property 3)} \\
> R_{i}(i,l,m) + \sigma_{i+1,i} - \sigma_{i,i} \geq \sum_{t=m+1}^{m} o_{t}(i,l,j) \\
= \sum_{t=m+1}^{m} o_{t}(i,l) \geq \sigma_{i+1,i} - \sigma_{i,i} = R_{i}(i+1,l,m) \\
= R_{i}(i+1,l,m)
\]

Consequently, from Property 2, we obtain \( \varphi^{*}(\tau^{*},l,j) < 0 \). From Property 4, there is an \( m \) such that \( \tau^{*} < m < j \) and \( o_{m}(i,l,j) = R_{i}(i,l,m) \). Therefore,

\[
\sum_{t=i+1}^{m} o_{t}(i,l,j) = \sum_{t=i+1}^{m} o_{t}(i,l,j) - \sum_{t=i+1}^{m} o_{t}(i,l,j) \\
= R_{i}(i,l,m) - R_{i}(i+1,l,m) \\
= R_{i}(i+1,l,m) - R_{i}(i+1,l,m)
\]

This is in contradiction with the fact that \( \tau^{*} < m^{*} < \tau^{*} \) which implies \( \sum_{t=m+1}^{m} o_{t}(i,l,j) = \sum_{t=m+1}^{m} o_{t}(i+1,l,j) \). With a similar argument, this conclusion can also be derived for the case where \( \tau^{*} > t^{*} \) and \( \varphi^{*}(\tau^{*},l,j) < \varphi^{*}(t^{*},l,j) \). The detailed proof is omitted.

In a symmetrical manner, we can prove that it is impossible that \( \tau^{*} < t^{*} \). For the case \( u \in [0,1] \), it can be proved with a similar argument to the above one. Moreover, it can be done more easily since it is not necessary to take period \( l \) into account.

Appendix D. Proof of Theorem 5

Consider an optimal solution in which there is a subplan \( i,u,j,k,v \) with a production period \( j \) such that for any \( l \) and \( n \) we have \( p_{i}(i,l,j) < r_{j}(j,n,k) \). We denote by \( L(i < t < k) \) the outsourcing quantity in an optimal solution of the subplan, which satisfies the link condition (27). At the same time, \( \{V_{i}(j,n,k) \} \) and \( L_{j-1} \) represent the value of \( \sigma_{i} = -d_{i-1} - \sum_{t=i}^{j-1} L_{i} \) and the inventory level at period \( j-1 \) in this optimal solution, respectively. We would have \( \chi_{i} = V_{i}(j,n,k) - (L_{j-1} + J_{j-1}) \geq 0 \) from (27). By the assumption that \( p_{i}(i,l,j) < r_{j}(j,n,k) \), at least one of the relations \( L_{j-1} + L_{j} \neq p_{i}(i,l,j) \), or \( L_{j-1} < p_{i}(i,l,j) \) must be true. Consider the case \( L_{j-1} < p_{i}(i,l,j) \), that is \( L_{j-1} < p_{i}(i,l,j) \), we have

\[
L_{j-1} = \frac{L_{j} + \sigma_{i} - d_{i-1}}{p_{i}(i,l,j)} < \frac{L_{j-1}}{p_{i}(i,l,j)} = \frac{L_{j-1}}{p_{i}(i,l,j)} = \frac{L_{j-1}}{p_{i}(i,l,j)}
\]

Due to the fact that \( L_{j-1} > 0 \) if \( l = j-1 \), and \( p_{i}(i,l,j) < 0 \) if \( l < j-1 \), we have

\[
R_{i}(i,l,j - 1) \leq \frac{L_{j}}{p_{i}(i,j-1,j)} \leq \frac{L_{j}}{p_{i}(i,j,1,j)} \quad \text{if} \quad l = j-1
\]

\[
\sum_{t=i+1}^{m} o_{t}(i,j-1,j) \leq \sum_{t=i+1}^{m} o_{t}(i,j,j) \quad \text{if} \quad l < j-1
\]
From these relations, there must be some \( \tau \) such that \( i < \tau < j \) and \( 0 < L_{\tau} < \alpha_{\tau}(i,l,j) \leq \delta_{\tau} \). In the remainder of the proof, let \( \tau' \) be the largest one of such \( \tau \)'s. Therefore, \( 0 \leq L_{\tau'} < \alpha_{\tau'}(i,l,j) \leq \delta_{\tau'} \). Here, we need to consider the following four situations: (i) \( l = j - 1 \) and \( \omega_{\tau}(i,l,j) < 0 \); (ii) \( j = l + 1 \) and \( \omega_{\tau}(i,l,j) > 0 \); (iii) \( j < l + 1 \) and \( \omega_{\tau}(i,l,j) < 0 \); (iv) \( l < j - 1 \) and \( \omega_{\tau}(i,l,j) > 0 \). Since these cases can be proved in a similar way, we just present the proof of case (i) for the sake of succinctness.

Let \( m' = \max \{ m | i < m < j, \sum_{t=i+1}^{m} \alpha_{t}(i,l,m') = R_{\alpha}(i,l,m') \} \). Note that \( m' \) necessarily exists, since \( \sum_{t=i+1}^{m} \alpha_{t}(i,l,m') \leq R_{\alpha}(i,l,m') \). Since \( \sum_{t=i+1}^{m} \omega_{t}(i,l,j) \leq R_{\alpha}(i,l,m') \), while \( \sum_{t=i+1}^{m} \omega_{t}(i,l,j) \geq \sum_{t=i+1}^{m} \alpha_{t}(i,l,j) \), we have \( \sum_{t=i+1}^{m'} \omega_{t}(i,l,j) < 0 \). There must be a \( \tau' \) such that \( m' < \tau' < j \) and \( L_{\tau'} < \alpha_{\tau'}(i,l,j) \). Therefore, \( \tau' > m' \).

The facts that \( \omega_{\tau}(i,l,j) < 0, \tau' > m' \) and \( L_{\tau'} < \delta_{\tau'} \) imply that the cost of the subplan can be strictly reduced by increasing the value of \( L_{\tau'} \) while satisfying all the constraints, which is in contradiction with the optimality of the subproblem.

Similarly, when \( j = l - 1 \) and \( \omega_{\tau}(i,l,j) > 0 \), we can deduce that the cost of subproblem \( P_{\alpha}(i,j,l) \) can be reduced by decreasing \( \alpha_{\tau}(i,l,j) \), which is in contradiction with the assumption that \( \alpha_{\tau}(i,l,j) \geq \delta_{\tau} \). Consequently, this theorem always is true.

**Appendix E. Proof of Theorem 6**

From Theorem 3, if \( \pi_{i} > j_{i} \) then we have \( L_{i}(i,l,j) = \max(-\rho_{i}, i, j, 0 \rangle + d_{i} \) and \( L_{i}(i + 1, l,j) = \max(-\rho_{i}, i + 1, l, j, 0 \rangle + d_{i} \). Moreover, if \( \rho_{i} \leq d_{i} \) we have \( L_{i}(i,l,j) = L_{i}(i + 1, l,j) = 0 \). Obviously, when \( \rho_{i} > d_{i} \) this theorem is true according to Lemma 1. So we merely need to consider the case with \( \pi_{i} > j_{i} \).

If \( \rho_{i}(i,l,j) > 0 \), then we also have \( \rho_{i}(i + 1, l,j) > 0 \) from Lemma 1, thereby

\[
\rho_{i}(i,l,j) + L_{i}(i,l,j) = \rho_{i}(i,l,j) + d_{i} \leq \rho_{i}(i + 1, l,j) + d_{i} = \rho_{i}(i + 1, l,j) + L_{i}(i + 1, l,j)
\]

If \( \rho_{i}(i,l,j) < 0 \) and \( \rho_{i}(i + 1, l,j) > 0 \), we have

\[
\rho_{i}(i,l,j) + L_{i}(i,l,j) = d_{i} \leq \rho_{i}(i + 1, l,j) + d_{i} = \rho_{i}(i + 1, l,j) + L_{i}(i + 1, l,j)
\]

If \( \rho_{i}(i,l,j) < 0 \) and \( \rho_{i}(i + 1, l,j) < 0 \), we can also obtain

\[
\rho_{i}(i,l,j) + L_{i}(i,l,j) = d_{i} \leq \rho_{i}(i + 1, l,j) + L_{i}(i + 1, l,j)
\]

Consequently, this theorem always is true.

**References**


Federgruen, A., & Tzur, M. (1991). A simple forward algorithm to solve general dynamic lot sizing models with n periods in O(nlog n) or O(n) time. Management Science, 37(8), 909–925.


