Journal of the Operations Research
Society of Japan
Vol. 29, No. 3, September 1986

AN APPROACH FOR THE OPTIMAL PRODUCTION RATE OF A SINGLE PRODUCT SYSTEM WITH DYNAMIC DEMANDS

Chang Sup Sung M. H. Park
Korea Advanced Institute of Science & Technology (KAIST)

(Received October 11, 1983: Final February 3, 1986)

Abstract The optimal production rate (production capacity) for a single product system with some deterministic dynamic demands is considered. Under a linear-additivity assumption on the associated cost information, the existence of the optimal solution for production capacity is verified. For the model, two solving techniques (point-wise computation and L.P. approaches) are developed for both unfilled-demand handling cases of lost-sales and backlogging, with which example problems are tested to show the superiority of the point-wise computation approach to L.P. approach.

1. Introduction

Given a finite time horizon $[0,T]$ partitioned into several time-periods (time-intervals) during which the period-wise demands are variant accordingly, the optimal capacity determination of a production system is practically of great importance in investment view. Furthermore, when the operation of the system at a fixed production rate, say λ, through the time horizon is considered, the associated inventory systems should be accounted for to derive the optimal investment decision giving rise to an optimal value λ^*

Florian and Klein [2] and Lambrecht et al. [5] have studied the problems of production scheduling with the allowance of the period-wise different production quantity, given the known demands. Their works were based on a given system with bounded capacities, so that some of its production schedules may include certain idle periods.

By the way, from the standpoint of the system utilization and cost caused from production changes, the operation of a system at a fixed production rate is sometimes important. For example, consider the problem of selecting a production system for a commodity. Then, the system operation schedules would be studied with respect to its full utilization. This problem can be interpreted as the problem of finding a schedule for the ssytem to be operated at a constant (fixed) production rate and without any idle period.

For this study, both cases of backlogging and lost-sales are to be treated for the optimal production rate λ^* at which the total profit through the time horizon is maximized.

In the following section, a production rate decision model for both cases of lost-sales and backlogging is developed. In fact, a more general problem including this problem as a special case can be formulated as a linear programming problem and hence can be solved efficiently. Therefore, for the model, the optimal solution algorithms are proposed in L.P. and point-wise computation approaches. It is then shown that for the special case the point-wise computation algorithm is more efficient than L.P. algorithm.

2. Total Profit Model

The model dealt with in this section is of determining the optimal capacity of a production system corresponding to a sequence of the deterministic future demands. For its analysis, a total profit through the time horizon is defined as follows;

Total Profit = (quantity sold) x (unit price) + (salvage value)
(*TR*) - (inventory holding cost) - (shortage cost)
- (initial investment) - (manufacturing cost).

Let N be the total number of periods over the time horizon and m_t be the manufacturing cost for a product in period t. Then, the first term of TR is represented as $\sum_{t=1}^{N} U_t S_t$, where S_t is the quantity sold during period t and U_t is the unit price of a product. Salvage value is composed of two elements, one for the initial investment and the other for the products salvaged. The latter is denoted by kI_N^+, where k is the salvage value of a product ($k \leq m_t$) and I_N^+ is the positive inventory. The former is represented by $rC\lambda$ estimated at the end of period N, where r ($|r| < 1$) is the salvage value conversion rate and $C\lambda$ is the initial investment for the system which is assumed to be linearly proportional to λ.

Let h_t be holding cost per unit held from period t to t+1 and π_t be shortage cost. Under the assumption that inventory holding costs are linearly proportional to the net inventory at the end of priod t (denoted by I_t), TR is expressed as follows;

$$(1)\qquad TR(\lambda) = \sum_{t=1}^{N} U_t S_t + rC\lambda + kI_N^{\ +} - \sum_{t=0}^{N-1} h_{t+1} I_t^{\ +} - \sum_{t=1}^{N} \pi_t I_t^{\ -} - C\lambda - \sum_{t=1}^{N} m_t \lambda$$

where $I_t^{\ +} = \max(0,\ I_t)$, and $I_t^{\ -} = \max(0,\ -I_t)$.

3. Optimal Solution Search: Lost-Sales Case

In this section, a model will be analyzed for which the unsatisfied demand for a period assumes lost. Then, S_t and I_t can be expressed as follows: denoting by d_t the demand during period t,

$$(2)\qquad S_t = \min(d_t,\ \lambda + \max(I_{t-1},\ 0)) = \min(d_t,\ \lambda + I_{t-1}^{\ +})$$

$$(3)\qquad I_t = \max(0,\ I_{t-1}) + \lambda - d_t = I_{t-1}^{\ +} + \lambda - d_t = I_t^{\ +} - I_t^{\ -}.$$

Eqs.(2) and (3) depend piece-wisely upon λ. Such dependency is summarized in Table 1.

Table 1. I_t and S_t Piece-Wisely Depending upon λ.

t	I_t	$I_t^{\ +}$	$I_t^{\ -}$	S_t	range of λ
1	$\lambda-d_1$	$\lambda-d_1$	0	d_1	$\lambda\geq d_1$
		0	$d_1-\lambda$	λ	$\lambda\leq d_1$
2	$2\lambda-d_1-d_2$	$2\lambda-d_1-d_2$	0	d_2	$\lambda\geq d_1$ & $2\lambda\geq d_1+d_2$
		0	$d_1+d_2-2\lambda$	$2\lambda-d_1$	$\lambda\geq d_1$ & $2\lambda\leq d_1+d_2$
	$\lambda-d_2$	$\lambda-d_2$	0	d_2	$\lambda\leq d_1$ & $\lambda\geq d_2$
		0	$d_2-\lambda$	λ	$\lambda\leq d_1$ & $\lambda\leq d_2$
⋮	⋮	⋮	⋮	⋮	⋮
N	$N\lambda-d_1-..-d_N$	$N\lambda-d_1-..-d_N$	0	d_N	
		0	$d_1+...d_N-N\lambda$	$N\lambda-d_1-..-d_{N-1}$	
	⋮	⋮	⋮	⋮	⋮
	$\lambda-d_N$	$\lambda-d_N$	0	d_N	
		0	$d_N-\lambda$	λ	

Now, it will be shown that TR is a piece-wise affine function of λ.

Theorem 1. The function TR is a piece-wise affine function of λ and its slope changes occur at the points, d_1, $d_2, \ldots, d_N$, $(d_1+d_2)/2, \ldots, (d_{N-1}+d_N)/2$, $(d_1+d_2+d_3)/3, \ldots, (d_{N-2}+d_{N-1}+d_N)/3, \ldots, (d_1+\ldots+d_N)/N$.

Proof: From Table 1, it is seen that I_t^+, I_t^- and S_t for $t=1,2,\ldots,N$ are piece-wise linear functions, respectively, depending upon the ranges of λ. This implies that $\sum_{t=0}^{N-1} h_{t+1}I_t^+$, $\sum_{t=1}^{N} \pi_t I_t^-$ and $\sum_{t=1}^{N} U_t S_t$ are piece-wise linear functions. Meanwhile, the rest part of the function TR is also a linear function of λ. Therefore, TR is a piece-wise linear function of λ whose slope changes occur at the boundary points given above.

For example, consider the case of "$N = 1$". Then, the two possible distinct ranges (intervals) of λ are in the set $\{0\le\lambda\le d_1,\ d_1\le\lambda\}$. Over each range in the set, $TR(\lambda)$ is defined as follows:

(a) For the range $d_1\le\lambda$;

$$TR_a(\lambda) = (rC-C-m_1+k)\lambda + (U_1d_1-kd_1),$$

(b) For the range $0\le\lambda\le d_1$;

$$TR_b(\lambda) = (rC-C-m_1+U_1+\pi_1)\lambda - \pi_1 d_1,$$

where $TR_i(\lambda)$ denotes the objective function defined over the range i ($i=a,b$). These two functions, $TR_a(\lambda)$ and $TR_b(\lambda)$, show that each of them is a linear function defined over its associated range (interval), and further that they are equal at the point $\lambda = d_1$.

Also, consider the case of "$N = 2$". Then, all the demands, d_1 and d_2, can be ranked as $d_{(1)}\le d_{(2)}$ in their nondecreasing order, where $d_{(i)}$ denotes the i^{th} smallest value. Thereupon, the associated range set can be defined as $\{0\le\lambda\le d_{(1)},\ d_{(1)}\le\lambda\le(d_{(1)}+d_{(2)})/2,\ (d_{(1)}+d_{(2)})/2\le\lambda\}$. Now, assume without loss of generality that $d_1\le d_2$. Then over each element in the range set $\{0\le\lambda\le d_1$, $d_1\le\lambda\le(d_1+d_2)/2,\ (d_1+d_2)/2\le\lambda\}$, $TR(\lambda)$ is defined as follows:

(a) For the range $0\le\lambda\le d_1$;

$$TR_a(\lambda) = (rC-C-m_1-m_2+U_1+U_2+\pi_1+\pi_2)\lambda - (\pi_1d_1+\pi_2d_2),$$

(b) For the range $d_1\le\lambda\le(d_1+d_2)/2$;

$$TR_b(\lambda) = (rC-C-m_1-m_2+2U_2-h_2+2\pi_2)\lambda + (u_1d_1-u_2d_1+h_2d_1-\pi_2d_1-\pi_2d_2),$$

(c) For the range $(d_1+d_2)/2\le\lambda$;

$$TR_c(\lambda) = (rC-C-m_1-m_2+2k-h_2)\lambda + (U_1d_1+U_2d_2-kd_1-kd_2+h_2d_1).$$

This also shows that each of the three functions is a linear function, and further that $TR_a(\lambda)$ and $TR_b(\lambda)$ are equal at $\lambda = d_1$, and $TR_b(\lambda)$ and $TR_c(\lambda)$ are equal at $(d_1+d_2)/2$. These imply that TR is a piece-wise affine function defined over the whole domain of λ, $0\leq\lambda$.

Likewise, in general, N demands $d_1, d_2, \ldots,$ and d_N can be ranked as $d_{(1)}\leq d_{(2)}\leq\ldots\leq d_{(N)}$ in their nondecreasing order. It follows that the whole domain of λ ($0\leq\lambda$) can be decomposed into the distinct intervals of whose boundary points are in the set $\{0,d_1,\ldots,d_N,(d_1+d_2)/2,\ldots,(d_{N-1}+d_N)/2,\ldots,(d_1+d_2+d_3)/3,\ldots,(d_1+d_2+\ldots+d_N)/N\}$. Then, it can be seen that over each of these intervals, TR is defined as a flinear function of λ, and further defined over the whole domain of λ as a piece-wise affine function. This completes the proof.

The results of Theorem 1 lead to Theorem 2 which shows the existence of the optimal solution.

Theorem 2. The function TR has the maximum value at either 0 or one of the boundary points resulted from Theorem 1.

Proof: From the piece-wise affinity, $TR(\lambda)$ may have local maximums at the boundary points. Therefore, it is sufficient to check the values of λ in the ranges outside the smallest and the greatest boundary points.

If λ is in the range beyond the greatest boundary points, $S_t=d_t$, $I_t^+=t\lambda-d_1-\ldots-d_t$, $I_t^-=0$ for $t=1,2,\ldots,N$. Thence, $TR(\lambda)=\lambda(rC-C-\sum_{t-1}^{N} m_t-\sum_{t=1}^{N-1} t\ h_{t+1} + NK)$ + constant, which decreases as λ increases, because $(rC-C-\sum_{t=1}^{N} m_t-\sum_{t=0}^{N-1} t\ h_{t+1} + NK) < 0$. So, the local maximum of $TR(\lambda)$ over the range is attained at the greatest boundary point.

Meanwhile, if λ is in the range below the smallest boundary point, $S_t=\lambda$, $I_t^+=0$ and $I_t^-=d_t-\lambda$ for $t=1,2,\ldots,N$. Thence,

$$TR(\lambda) = (\sum_{t=1}^{N} U_t + rC - C - \sum_{t=1}^{N} m_t + \sum_{t=1}^{N} \pi_t)\lambda + \text{constant}.$$

This implies that, as λ decreases, $TR(\lambda)$ decreases or increases depending upon the coefficient of λ. Hence, the local maximum over the range is attained at either the smallest point or 0.

Thus, the proof is completed.

A solution algorithm, $PWA1$(abbreviation of point-wise computation approach 1), is then suggested based on Theorems 1 and 2.

Algorithm *PWA*1

Step 1 Compute values of $(d_1+d_2)/2, \ldots, (d_{N-1}+d_N)/2, \ldots, (d_1+d_2+d_3)/3, \ldots, (d_{N-2}+d_{N-1}+d_N)/3, \ldots, (d_1+\ldots+d_N)/N$ (Let them be P_i $(i=1, 2, \ldots, N(N+1)/2+1)$ including $d_1, \ldots, d_N$ and 0)

Step 2 Compute $TR(P_i)$ for each P_i, where
$I_t = I_{t-1}^+ + P_i - d_t$
$S_t = \min(d_t, P_i + I_{t-1}^+)$

Step 3 Compare the values of $TR(P_i)$ and find the optimal value λ^* maximizing TR such as
$\lambda^* = \{P_k;\ TR(P_k) \geq TR(P_i),\ \forall i\}$

Now, consider the computational complexity of Algorithm *PWA*1. When the order of computation rather than the detail number of computations is taken into account as the measure of complexity, the algorithm is of order N^3, which is proved in Lemma 1.

Lemma 1. Point-wise approach is $O(N^3)$ for the lost-sales case.

Proof: The number of points to be computed is the number of boundary points for the intervals of λ and zero point, and so $N+(N-1)+\ldots+1+1=N(N+1)/2+1$.

Given a boundary point of λ, values of $I_t^+(I_t^-)$ and S_t are obtained from iterative computation at each partitioning point of time horizon $[0, T]$, and so the number of iterations is N.

Thus, the number of computations for optimal λ^* is $[N(N+1)/2+1]N=N^2(N+1)/2+N$. This completes the proof.

The algorithm is now illustrated with a numerical example with $N=4$.

Example 1

Assume that the required informatoin is given as follows;
$d_1=3$, $d_2=1$, $d_3=4$, $d_4=2$, $N=4$, $\pi_t=0.5$, $h_t=0.2$, $m_t=2$, $U_t=3.3$ for $t=1,\ldots,4$, $C=4$, $r=0.1$, $k=2.5$.
Then,
$P_1=3, P_2=1, P_3=4, P_4=2, P_5=2, P_6=2.5, P_7=3, P_8=8/3, P_9=7/3, P_{10}=2.5, P_{11}=0$.
So,

$TR(P_1) = TR(P_7) = 2.6$	$TR(P_2) = -1.4$
$TR(P_3) = -0.2$	$TR(P_4) = TR(P_5) = 2.0$
$TR(P_6) = TR(P_{10}) = 3.05$	$TR(P_8) = 2.9$
$TR(P_9) = 2.7$	$TR(P_{11}) = -5$

$\lambda^* = 2.5$

Our problem can also be handled in linear programming (LP) approach. From Eqs. (1), (2) and (3), the problem can be formulated in LP as follows:

$$\text{maximize} \quad \sum_{t=1}^{N} U_t S_t + (rC-C-\sum_{t=1}^{N} m_t)\lambda - \sum_{t=0}^{N-1} h_{t+1} I_t^{+} + kI_N^{-} - \sum_{t=1}^{N} \pi_t I_t^{-}$$

$$\begin{array}{lllll}
\text{s.t.} & \lambda - I_1^{+} & & + I_1^{-} & = d_1 \\
& 2S_1 - \lambda + I_1^{+} & & + I_1^{-} & = d_1 \\
& \lambda & + I_{t-1}^{+} - I_t^{+} & + I_t^{-} & = d_t \\
& 2S_t - \lambda & - I_{t-1}^{+} + I_t^{+} & + I_t^{-} & = d_t \\
& S_t \geq 0,\ I_t^{+} \geq 0,\ I_t^{-} \geq 0,\ \forall t & & & \\
& \lambda \geq 0. & & &
\end{array}$$

Observe that the vector of constraint coefficients for I_t^{+} is linearly dependent with that of I_t^{-}. This means that I_t^{+} and I_t^{-} cannot both be in a basic solution, and hence that the constraint $I_t^{+} \cdot I_t^{-} = 0$, implied by the definition of on-hand inventory and backorders, is automatically satisfied and does not have to be represented explicitly in the model.

4. Optimal Solution Search: Backlogging Case

Another model shall be analyzed for which unsatisfied demands can be satisfied later. In this case, S_t and I_t are expressed as follows:

$$(4) \qquad S_t = \min\,[\max(I_{t-1}, 0) + \lambda,\ d_t - \min(I_{t-1}, 0)]$$
$$= \min\,[I_{t-1}^{+} + \lambda,\ d_t + I_{t-1}^{-}]$$

$$(5) \qquad I_t = I_{t-1} + \lambda - d_t = I_t^{+} - I_t^{-}$$

Eqs.(4) and (5) also depend piece-wisely upon λ. Such dependency is summarized in Table 2.

The function TR, even if it is newly defined with S_t and I_t given in Eqs.(4) and (5), can be proved to have similar solution properties to those in the lost-sales case. The properties are stated in Theorems 3 and 4.

Table 2. I_t and S_t Piece-Wisely Depending upon λ

t	I_t	I_t^+	I_t^-	S_t	range of λ
1	$\lambda-d_1$	$\lambda-d_1$	0	d_1	$\lambda\geq d_1$
		0	$d_1-\lambda$	λ	$\lambda\leq d_1$
2	$2\lambda-d_1-d_2$	$2\lambda-d_1-d_2$	0	d_2	$\lambda\geq d_1$ & $2\lambda-d_1\geq d_2$
				$d_1+d_2-\lambda$	$\lambda\leq d_1$ & $\lambda\geq d_1+d_2-\lambda$
		0	$d_1+d_2-2\lambda$	$2\lambda-d_1$	$\lambda\geq d_1$ & $2\lambda-d_1\leq d_2$
				λ	$\lambda\leq d_1$ & $\lambda\leq d_1+d_2-\lambda$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
N	$N\lambda-d_1-\ldots-d_N$	$N\lambda-d_1-\ldots-d_N$	0	d_N	$\vdots$
				$d_1+\ldots+d_{N-1}-(N-1)\lambda$	$\vdots$
		0	$d_1+\ldots+d_N-N\lambda$	$N\lambda-d_1-\ldots-d_{N-1}$	$\vdots$
				λ	$\vdots$

Theorem 3. The function TR is a piece-wise affine function of λ whose slope changes occur at the points d_1, $(d_1+d_2)/2$, $(d_1+d_2+d_3)/3$, ..., $(d_1+\ldots+d_N)/N$.

Proof: The proof is straightforward when the proof steps of Theorem 1 are followed by use of Table 2.

Theorem 4. The function TR has the maximum value at either 0 or one of the boundary points resulted from Theorem 3.

Proof: It is easily proved by use of the results of Theorem 3 when the proof steps of Theorem 2 are followed.

Based on the results of Theorems 3 and 4, a solution algorithm, *PWA*2 (point-wise computation approach 2), for the backlogging case is then developed.

Algorithm *PWA*2.

Step 1 Compute the values of d_1, $(d_1+d_2)/2$, $(d_1+d_2+d_3)/3$, ..., $(d_1+\ldots+d_N)/N$ (Let them be represented by P_i $(i=1,2,\ldots,N+1)$, respectively, with $P_{N+1}=0$)

Step 2 Compute $TR(P_i)$ for each P_i, where

$I_t = I_{t-1} + P_i - d_t$

$S_t = \min(I_{t-1}^+ + P_i, d_t + I_{t-1}^-)$

Step 3 Same as step 3 of *PWA*1.

As done for Algorithm *PWA*1, when the order of computation is considered as the measure of computational complexity, Lemma 2 shows that Algorithm *PWA*2 is of order N^2, which can be proved in the same way as for Lemma 1.

Lemma 2. For the backlogging case, point-wise approach is $O(N^2)$.

Example 2 is presented for illustrating Algorithm *PWA*2.

Example 2

The required data are the same as those of Example 1 except $\pi_t=0.3$ for $t=1, 2, 3, 4$.

Then,

$P_1=3$, $P_2=2$, $P_3=8/3$, $P_4=2.5$, $P_5=0$ and so, $TR(P_1) = 2.6$, $TR(P_2) = -3.3$, $TR(P_3) = 3.36$, $TR(P_5) = 3.5$, $TR(P_5) = -7.5$, $\lambda^* = 2.5$

From Eqs. (1), (4) and (5), the backlogging problem is also formulated into LP as follows:

$$\text{maximize} \quad \sum_{t=1}^{N} U_t S_t + (rC - C - \sum_{t=1}^{N} m_t)\lambda - \sum_{N=0}^{N-1} h_{t+1} I_t^+ + kI_N^+ - \sum_{t=1}^{N} \pi_t I_t^-$$

$$\begin{array}{lrcl}
\text{s.t.} & \lambda - I_1^+ + I_1^- & = & d_1 \\
& 2S_1 - \lambda + I_1^+ + I_1^- & = & d_1 \\
& \lambda + I_{t-1}^+ - I_t^+ - I_{t-1}^- + I_t^- & = & d_t \\
& 2S_t - \lambda - I_{t-1}^+ + I_t^+ - I_{t-1}^- + I_t^- & = & d_t \\
& t=2,\dots,N & & \\
& S_t \ge 0,\ I_t^+ \ge 0,\ I_t^- \ge 0,\ \forall t. \quad \lambda \ge 0 & &
\end{array}$$

5. Comments on Computational Complexity (Point-wise Computation vs. LP)

Bazaraa [1] illustrated the number of operations required during an iteration in solving LP problem by the revised simplex method. Bazaraa's result with the number of operations required for the point-wise computation approach is listed in Table 3, where LP is based on one iteration but point-wise approach is based on the whole computation, and d is the density of non-zero elements in the constant matrix.

Table 3. LP vs. Point-Wise Computation Approach (N periods)

Method	Operation		
	Multiplication	Addition	Comparison
Revised Simplex (one iteration)	$(2N+1)^2$ $+ d\ 2N(2N+1)$	$2N(2N+1)$ $+ d\ 2N(2N+1)$	$2N$ $+ 3N+1$
$PWA1$	$\frac{3}{2}N^2(N+1) + \frac{3}{2}N(N+1)$ $+N(N-1)/2$	$\frac{3}{2}N^2(N+1)+N(N+1)$ $+N+N(N+1)(N+2)/6$	$\frac{3}{2}N^2(N+1) + \frac{1}{2}N(N+1)$ $+N(N+1)/2$
$PWA2$	$2N^2+3N$ $+(N-1)$	$5N^2+3N+N$ $+N(N+1)/2-1$	$2N^2+2N$ $+N$

Garey and Johnson [3] have stated that with reference to Klee and Minty [4], the simplex algorithm for LP has exponential time complexity. However, in practice, the simplex method has an impressive record of running quickly. Meanwhile, there is no general conclusion for the number of iterations required for finding the optimal solution in the simplex method. Therefore, it is difficult to conclude which approach is the more efficient one, in general, of two.

Several numerical examples are solved on Cyber 174 to make a computational efficiency comparison in CPU time between PWA's and LP approaches, with the variations of cost coefficients of λ and of the planning horizon N, the results of which are listed in Table 4 and 5. In Table 5, C_λ represents the function of C defined as $C_\lambda = C - rC + \sum_{t=1}^{N} m_t$. Table 4 shows CPU time requirements for finding optimal solutions to each of the problems generated by varying the information of U_t, π_t^+, π_t^-, h_t and N, for the fixed data of C=4.0, r=0.1, k=1.5 and m_t=2.0. And Table 5 shows such CPU time requirements for the problems generated for the fixed data of U_t=3.3, π_t^+=0.3, π_t^-=0.5, h_t=0.2, k=1.5 and m_t=2.0.

Table 4. Computation Time Comparison in CPU on Cyber 174 between PWA's and LP with C=4.0, r=0.1, k=1.5 and m_t=2.0.

Data				lost-sales case (sec.)		backlogging case (sec.)	
U_t	π_t	h_t	periods	$PWA1$	LP	$PWA2$	LP
3.3	0.5 (0.3)	0.2	4	0.058	0.929	0.056	0.933
			12	0.089	1.156	0.082	1.268
			24	0.215	1.518	0.204	1.710
	0.8 (0.6)	0.2	4	0.056	0.902	0.055	0.917
			12	0.086	1.164	0.083	1.263
			24	0.220	1.509	0.205	1.713
		0.4	4	0.058	0.903	0.056	0.911
			12	0.088	1.146	0.085	1.261
			24	0.221	1.509	0.205	1.710
	1.1 (0.9)	0.4	4	0.058	0.905	0.058	0.907
			12	0.084	1.128	0.083	1.244
			24	0.216	1.523	0.207	1.743
3.3	1.1 (0.9)	0.6	4	0.058	0.910	0.056	0.912
			12	0.083	1.142	0.083	1.254
			24	0.221	1.533	0.213	1.712
3.0	0.5 (0.3)	0.2	4	0.056	0.916	0.055	0.917
			12	0.083	1.144	0.086	1.293
			24	0.208	1.512	0.212	1.699
	0.8 (0.6)	0.2	4	0.058	0.920	0.057	0.913
			12	0.085	1.131	0.084	1.271
			24	0.213	1.549	0.210	1.717
		0.4	4	0.055	0.931	0.056	0.911
			12	0.085	1.120	0.083	1.274
			24	0.213	1.529	0.202	1.706
	1.1 (0.9)	0.4	4	0.057	0.913	0.056	0.911
			12	0.086	1.128	0.089	1.248
			24	0.214	1.539	0.213	1.711
		0.6	4	0.057	0.898	0.056	0.907
			12	0.082	1.121	0.082	1.274
			24	0.220	1.572	0.208	1.715

(Note: Values in parentheses indicate π_t^+ values).

Table 5. Computation Time Comparison in CPU on Cyber 174 between *PWA*'s and LP with U_t=3.3, π_t^+=0.3, π_t^-=0.5, r=0.1, h_t=0.2, k=1.5 and m_t=2.0.

Data			lost-sales case (sec.)		backlogging case (sec.)	
C	C_λ	periods	*PWA*1	LP	*PWA*2	LP
4.0	11.6	4	0.058	0.929	0.056	0.933
	27.6	12	0.089	1.156	0.082	1.268
	51.6	24	0.215	1.518	0.204	1.710
100.0	98.0	4	0.052	0.942	0.051	1.013
	114.0	12	0.080	1.555	0.080	1.641
	138.0	24	0.211	2.529	0.204	2.611
200.0	188.0	4	0.051	0.948	0.052	1.051
	204.0	12	0.080	1.558	0.077	1.644
	228.0	24	0.216	2.726	0.206	2.749

Tables 4 and 5 show that given three N values 4, 12 and 24, in our *PWA* approaches CPU time does not vary with the cost coefficients of λ, while it greatly depends on the variations of planning horizon N. However, LP approach spent ten times the CPU time in searching each associated optimal solution, even if it is less sensitive to the variations of N values than *PWA* approaches. Furthermore, Table 5 shows that as C increases, the CPU time requirement in LP approach also increases.

In conclusion, for problems with N not large, our *PWA* approach is efficient more than ten times in comparison with LP approach. In particular, when C is greater than 200 and N is less than 12, the superiority of *PWA* to LP approach in computational efficiency becomes more conspicuous. We further claim that *PWA* approach could be practically appreciated in its application to fairly large planning problems such as monthly production planning over five years, weekly production planning over season, etc. In fact, under a variety of uncertain industrial environments, forecasted demands will be meaningful in product management only over a certain limited time interval, so that the problem size parameter, the number of time-periods N, of our problem won't be large. Therewith, it can be argued that with reference to the computational complexity comparisons shown in Tables 3, 4 and 5 the point-wise computation approach is superior from the practical application view, because the study objective of determining the optimal product rate λ^* is based on the forecasted future demands.

Remarks:

It is noted that the above piece-wise computation approach can also be directly applied for searching the optimal solution of λ to the piece-wise total profit function which consists of convex functions defined over each interval of λ. This occurs when each cost function with respect to inventory holding and shortage and initial investiment is assumed concave in I_t^+, I_t^- and λ, respectively. In fact, the maximum value of a convex function for an interval is attained at one of the boundary points of the interval.

As shown in Algorithms 1 and 2, for the specific problem the computational complexities in *PWA*'s depend on the number of horizon periods rather than the coefficient of λ. If we consider a situation in which the conditions $k \leq m_t$ for all t are removed and rather a production capacity restriction is placed as an upper bound of λ, then a maximum value of *TR* would also be guaranteed, since the capacity bound should be included as a boundary point.

6. Conclusion

As shown through this work, two searching approaches are suggested for finding the optimal production rate for a system, the total profit function of which is a piece-wise affine (linear or convex) function, given a sequence of deterministic demands.

The importance of this study is placed on the case in which the production management is highly dependent upon the optimal utilization of facilities. It may also be of practical interest to some production systems, of which the change of production rate is too costly.

From the efficiency comparison between the two approaches, the point-wise computation approach seems to be advantageous in its applying to real problems.

Certain production systems with stochastic demand processes are currently under investigation as an extension of this work.

Acknowledgment

The authors are grateful to Mr. W.S. Lee for his help with computer calculations and also wish to thank the referees for their numerous valuable suggestions and comments contributed to the improvement of the paper.

References

[1] Bazaraa, M. S., and Jarvis, J. J.: *Linear Programming and Network Flows, Wiley*, New York, 1977.

[2] Florian, M., and Klein, M.: Deterministic Production Planning with Concave Costs and Capacity Constraints. *Management Science*, Vol.18 (1971), 12-20.

[3] Garey, M. R., and Johnson, D. S.: *Computers and Interactibility — A Guide to the Theory of NP-Completeness, W. H. Feeman and Company*, San Francisco, 1979.

[4] Klee, V., and Minty, G. J.: *How good is the simplex algorithm in Inequalities III, edited by O. Shisha, Academic Press*, New York, 1972.

[5] Lambrecht, M., and Vander Eecken, J.: A Capacity Constrained Single Facility Dynamic Lot-Size Model, *European Journal of Operational Research*, Vol.2 (1978), 132-136.

C. S. SUNG: Department of Industrial Engineering, Korea Advanced Institute of Science & Technology, P.O.Box 150, Cheongryang, Seoul, Korea.