stichting
mathematisch
centrum

MC

AFDELING MATHEMATISCHE BESLISKUNDE BW 93/79 MEI
(DEPARTMENT OF OPERATIONS RESEARCH)

M. FLORIAN, J.K. LENSTRA, A.H.G. RINNOOY KAN

DETERMINISTIC PRODUCTION PLANNING: ALGORITHMS AND COMPLEXITY

Preprint

2e boerhaavestraat 49 amsterdam

Printed at the Mathematical Centre, 49, 2e Boerhaavestraat, Amsterdam.

The Mathematical Centre, founded the 11-th of February 1946, is a non-profit institution aiming at the promotion of pure mathematics and its applications. It is sponsored by the Netherlands Government through the Netherlands Organization for the Advancement of Pure Research (Z.W.O).

AMS(MOS) subject classification scheme (1970): 90B30, 90B05, 68A20

First printing 1978
Second edition 1979

DETERMINISTIC PRODUCTION PLANNING: ALGORITHMS AND COMPLEXITY

M. FLORIAN

Université de Montréal, Montréal, Canada

J.K. LENSTRA

Mathematisch Centrum, Amsterdam, The Netherlands

A.H.G. RINNOOY KAN

Erasmus University, Rotterdam, The Netherlands

ABSTRACT

A class of production planning problems is considered in which known demands have to be satisfied over a finite interval at minimum total costs. For each period, production and storage cost functions are specified. The production costs may include set-up costs and the production levels may be subject to capacity limits. The computational complexity of the problems in this class is investigated. Several algorithms proposed for their solution are described and analyzed. It is also shown that some special cases are NP-hard and hence unlikely to be solvable in polynomial time.

KEY WORDS & PHRASES: *production planning, demand, production cost, storage cost, set-up cost, capacity limit, dynamic programming, polynomial algorithm, pseudopolynomial algorithm, NP-hardness.*

NOTE: This report is not for review; it will be submitted for publication in a journal.

1. INTRODUCTION

We consider a class of production planning problems, in which a facility manufactures a single product to satisfy known demands over a finite planning interval of n periods. For each period, production and storage cost functions are specified. The production cost functions may include set-up costs and the amount produced in each period may be subject to a capacity limit. The problem is that of determining the amounts to be produced in each period in order to supply each demand on time (no backlogging) and to minimize the total costs of production and storage.

In this paper we investigate the computational complexity of these problems for various types of cost functions, set-up costs and capacity limits. As a first step, we consider in Section 2 the standard dynamic programming approach for the most general problem in the class. Its running time is $O(R_n C_n)$ or $O(nR_n^2)$, where R_n is the total demand and C_n is the total capacity over the entire interval. This algorithm could be called "pseudo-polynomial" in the sense that its running time is exponential in the size of a problem instance under a binary representation of the numerical data, but polynomial in the data themselves. We also establish NP-hardness for the problem, even for the special case in which all demands are equal, all storage costs are zero, and the production cost functions can be interpreted as being either concave with arbitrary capacity limits or convex with additional unit set-up costs. Hence, it is very unlikely that these restricted versions of the problem allow solution in truly polynomial time.

In Section 3 we consider problems with concave cost functions. We recall results of Wagner and Whitin [24] and Florian and Klein [5], who characterized the structure of optimal production plans and presented polynomial algorithms for the special cases of infinite and equal capacities. These algorithms can be implemented to run in $O(n^2)$ and $O(n^4)$ time, respectively. Further, we discuss some enumerative methods for the case of arbitrary capacities, as possible alternatives to the general dynamic programming approach.

In Section 4 we turn to problems with convex cost functions. For the case that no additional set-up costs are specified, results of Johnson [10] and Veinott [23] yield a simple pseudopolynomial algorithm, which runs in

$O(nR_n)$ time. It remains an open question whether a strictly polynomial algorithm exists.

Finally, all results are summarized in Section 5.

2. GENERAL CASE

We start by introducing some notation. For period i ($i = 1,\ldots,n$), let r_i be the demand, b_i the production set-up cost, c_i the production capacity limit and x_i the production amount, and let $R_i = \sum_{j=1}^{i} r_j$, $C_i = \sum_{j=1}^{i} c_j$ and $X_i = \sum_{j=1}^{i} x_j$. The cost of producing an amount x_i in period i is given by $p_i(x_i)$, with $p_i(0) = 0$ and $p_i(x) = b_i + p_i'(x)$ for $x > 0$, where p_i' is a continuous and nondecreasing function with $p_i'(0) = 0$. The cost of storing an inventory $I_i = X_i - R_i$ from period i to period i+1 is given by $h_i(I_i)$, where h_i is a continuous and nondecreasing function with $h_i(0) \geq 0$.

The production planning problem is that of determining amounts $x_1,\ldots,x_n$ that minimize the total costs of production and storage:

(1) $\sum_{i=1}^{n} (p_i(x_i) + h_i(I_i))$,

subject to the conditions of satisfying each demand on time and observing the capacity limits:

(2) $I_i \geq 0 \qquad (i = 1,\ldots,n-1)$,

(3) $I_n = 0$,

(4) $0 \leq x_i \leq c_i \quad (i = 1,\ldots,n)$.

Note that (2) and (3) correspond to flow conservation equations on an appropriately defined network. Feasibility is assured by the assumption that $R_i \leq C_i$ ($i = 1,\ldots,n$). A positive initial inventory I_0 could be handled by appending a period 0 in which $x_0 = I_0$ is produced at suitably small costs. Zangwill [25] has shown that the final inventory I_n can be assumed to be equal to zero without loss of generality. Under the assumptions that all r_i and c_i are integers and that all p_i' and h_i are linear functions between successive integer values of the argument, we may restrict our attention to integer valued $x_1,\ldots,x_n$. In analyzing the complexity of problems and algorithms, we assume that any $p_i(x)$ and any $h_i(I)$ can be evaluated in unit time.

The standard way to solve problems of this type is by means of *dynamic programming*. Let $D_i(X)$ be the cost of an optimal production plan over periods $1,\ldots,i$ subject to $X_i = X$, let $\underline{\underline{X}}_i$ be the set of feasible cumulative production levels in period i, and let $\underline{\underline{x}}_i(X)$ be the set of feasible production amounts in period i subject to $X_i = X$. It is clear that

$$
(5)\qquad
\begin{aligned}
D_0(X) &= \begin{cases} 0 & (X = 0), \\ \infty & (X \neq 0), \end{cases} \\
D_i(X) &= \left.\begin{cases} \min_{x \in \underline{\underline{x}}_i(X)} \{D_{i-1}(X-x) + p_i(x)\} + h_i(X-R_i) & (X \in \underline{\underline{X}}_i) \\ \infty & (X \notin \underline{\underline{X}}_i) \end{cases}\right\} (i \geq 1).
\end{aligned}
$$

The cost of an optimal production plan over the entire interval is equal to $D_n(R_n)$, which is calculated according to the above forward recursion; the corresponding values of $x_1,\ldots,x_n$ are obtained by standard backtracing techniques. A backward recursion could be formulated just as easily.

To estimate the running time of the dynamic programming algorithm, we note that $\underline{\underline{X}}_i \subseteq \{R_i, R_i+1, \ldots, R_n\}$ and $\underline{\underline{x}}_i(X) \subseteq \{0,1,\ldots,c_i\}$. Hence, for fixed i, all $D_i(X)$ are determined in $O(R_n c_i)$ time. It follows that the complete recursion requires $O(R_n C_n)$ time. If no capacity limits are specified, we have $\underline{\underline{x}}_i(X) \subseteq \{0,1,\ldots,R_n - R_{i-1}\}$, and an $O(nR_n^2)$ running time results.

Thus, dynamic programming provides an *exponential* algorithm: its running time is an exponential function of the size of a problem instance, as long as the numerical data are represented in a reasonable way, e.g., in a binary or decimal encoding. However, the algorithm could be called *pseudo-polynomial* in the sense that its running time is bounded by a polynomial function of the data themselves [6;18]. We shall now present strong circumstantial evidence that a truly *polynomial* algorithm for the production planning problem will probably never be found.

More specifically, we will show that the problem is *NP-hard,* even in the simple case of equal demands and zero storage costs. This results signifies that a polynomial algorithm for the problem could be used to construct similar algorithms for all *NP-complete* problems [11;12]. NP-complete problems are characterized by the following properties:

(a) none of them is known to be solvable in polynomial time;

(b) if any of them is solvable in polynomial time, then all of them are.

Many notorious combinatorial problems have been shown to be NP-complete and the existence of a polynomial algorithm for one (and thus for all) of them is generally considered to be extremely unlikely.

The following problem has been shown to be NP-complete [11]:

KNAPSACK: Given positive integers $a_1,\dots,a_t,A$, does there exist a subset $S \subset T = \{1,\dots,t\}$ such that $\sum_{i \in S} a_i = A$?

NP-hardness of our problem will be established by proving that KNAPSACK is *reducible* to it, i.e., that for any instance of KNAPSACK an instance of the production planning problem can be constructed in polynomial time such that finding an optimal production plan also resolves KNAPSACK.

Given any instance of KNAPSACK, we define the following instance of the simplified production planning problem, where, for notational convenience, the periods are numbered from 0 to n:

$$n = t;$$

$$r_i = A \qquad (i = 0,\dots,t);$$

$$b_0 = 1,\ c_0 = tA,\ p_0(0) = 0,\ p_0(x) = 1 \qquad (0 < x \le c_0);$$

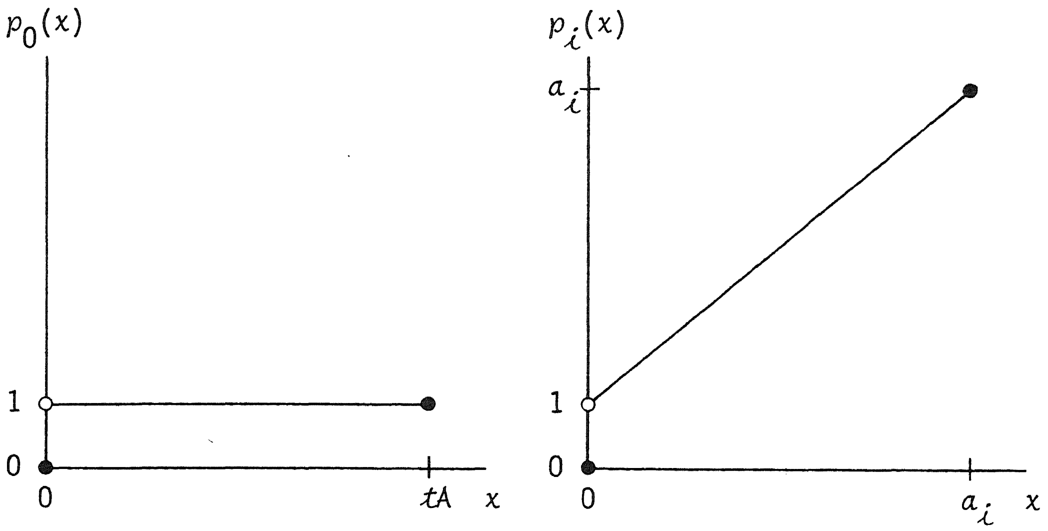
$$b_i = 1,\ c_i = a_i,\ p_i(0) = 0,\ p_i(x) = 1 + \frac{a_i - 1}{a_i}x \quad (0 < x \le c_i) \quad (i = 1,\dots,t);$$

$$h_i(I) = 0 \qquad (I \ge 0) \qquad (i = 0,\dots,t).$$

The production cost functions are illustrated in Figure 1. We claim that KNAPSACK has a solution if and only if there exists a feasible production plan with total costs at most equal to A+1.

Since $x_0 > 0$ in any feasible plan and $p_0(x)$ is constant for $0 < x \le tA$, we may assume that the production in period 0 is at capacity and supplies the demands in periods $0,\dots,t-1$. The production in periods $1,\dots,t$ has to supply the demand in period t only. Therefore, we may restrict our attention to production plans defined by

$$x_0 = tA,\quad \sum_{i \in T} x_i = A,\quad 0 \le x_i \le a_i \quad (i \in T).$$



$(i = 0)$ $(i = 1,\dots,t)$

Figure 1 The production cost functions in the reduction.

Since for all $i \in T$

$$p_i(x) = x \quad \text{for } x = 0 \text{ and } x = a_i,$$

$$p_i(x) > x \quad \text{for } 0 < x < a_i,$$

the total costs of such a plan are at least equal to A+1:

$$\sum_{i=0}^{t} p_i(x_i) = 1 + \sum_{i\in T} p_i(x_i) \geq 1 + \sum_{i\in T} x_i = A+1.$$

Moreover, they are exactly equal to A+1 if and only if $x_i \in \{0,a_i\}$ for all $i \in T$, i.e., if and only if there exists a subset $S \subset T$ such that $\sum_{i\in S} a_i = A$. This establishes the desired result.

We have already noted that our NP-hardness proof applies to the case of equal demands and zero storage costs. Moreover, the production cost functions involved are both concave and convex, with additional unit set-up costs and arbitrary capacity limits. It is easily seen that concave cost functions can be adapted to incorporate set-up costs, in such a way that they still are linear between successive integer points. Similarly, convex cost functions can be adapted to effectuate capacity limits. Hence, the following restricted versions of the production planning problem are NP-hard:

- arbitrary cost functions, no set-up costs, no capacity limits;

- concave cost functions, no set-up costs, arbitrary capacity limits;
- convex cost functions, unit set-up costs, no capacity limits.

This leaves only a few possibilities for truly polynomial algorithms. The will be considered in Sections 3 and 4.

3. CONCAVE COSTS

Let us assume that all p_i and h_i are concave functions, possibly includin set-up costs. Wagner and Whitin [24] studied the case in which no capacit limits are specified; their characterization of the structure of optimal production plans immediately yielded an $O(n^2)$ algorithm. Florian and Klei [5] characterized the structure of optimal production plans for the case arbitrary capacities and obtained an $O(n^4)$ algorithm for the special case of equal capacities. These results are summarized below. It should also b mentioned that for the related problem in which upper bounds on inventory rather than on production are specified, Love [19] developed an $O(n^3)$ alg rithm.

Recall the formulation of the problem in Section 2. The constraints (2)-(4) define a closed bounded convex set. The objective function (1) is concave and hence its minimum value is achieved at one of the extreme poi of this set. The special structure of the set allows a simple characteriz tion of the production plans corresponding to its extreme points. It was shown in [5] that such plans consist of a sequence of subplans in which

(a) the inventory is strictly positive in every period, except the last, where it is zero, and
(b) the production is either zero or at full capacity in every period, except for at most one period, which will be called the *fractional period*.

Thus, we are led to consider $\frac{1}{2}n(n+1)$ subproblems $P_{\ell m}$ $(0 \leq \ell < m \leq n)$ in which our objective is to minimize

$$\sum_{i=\ell+1}^{m} (p_i(x_i)+h_i(I_i))$$

subject to

$$I_\ell = I_m = 0,$$

$$I_i > 0 \qquad (i = \ell+1,\ldots,m-1),$$

$$0 \le x_i \le c_i \quad (i = \ell+1,\ldots,m),$$

$$0 < x_f < c_f \quad \text{for at most one } f \ (\ell+1 \le f \le m).$$

Let $E_{\ell m}$ be the optimal solution value to problem $P_{\ell m}$ and D_m^* the cost of an optimal production plan over periods $1,\ldots,m$. Given $\frac{1}{2}n(n+1)$ values $E_{\ell m}$ $(0 \le \ell < m \le n)$, we solve the original problem by calculating D_n^* as follows:

$$D_0^* = 0,$$

$$D_m^* = \min_{0\le\ell<m}\{D_\ell^*+E_{\ell m}\} \quad (m = 1,\ldots,n).$$

This recursion can be carried out in $O(n^2)$ time.

In the case that $c_i = \infty$ $(i = 1,\ldots,n)$, an optimal solution to problem $P_{\ell m}$ is clearly given by $x_{\ell+1} = R_m - R_\ell$, $x_i = 0$ $(i = \ell+2,\ldots,m)$, so that

$$E_{\ell m} = p_{\ell+1}(R_m-R_\ell) + \sum_{i=\ell+1}^{m} h_i(R_m-R_i)$$

(cf. [24]). All $E_{\ell m}$ can be determined in $O(n^2)$ time. It follows that the original problem is solved in $O(n^2)$ time.

In the case that $c_i = c$ $(i = 1,\ldots,n)$, problem $P_{\ell m}$ can be solved in the following way (cf. [5]). For notational convenience, we assume that $\ell = 0$. Dividing total demand by the capacity, we find $R_m = kc+\varepsilon$, where k is the number of periods in which the production will be at capacity and ε $(0 \le \varepsilon < c)$ is the amount to be produced in the fractional period. In order to apply the dynamic programming recursion (5), we observe that the sets $\underline{\underline{X}}_i$ of feasible cumulative production levels in period i are given by

$$\underline{\underline{X}}_i = \{X \mid X \in \{0,\varepsilon,c,c+\varepsilon,\ldots,kc,kc+\varepsilon\},\ R_i < X \le ic\} \quad (i = 1,\ldots,m-1),$$

$$\underline{\underline{X}}_m = \{R_m\},$$

and the sets $\underline{\underline{x}}_i(X)$ of feasible production amounts in period i subject to $X_i = X$ by

$$\underline{\underline{x}}_i(jc) = \{0,c\}, \quad \underline{\underline{x}}_i(jc+\varepsilon) = \{0,\varepsilon,c\} \quad (j = 0,\ldots,k;\ i = 1,\ldots,m).$$

We solve P_{0m} by calculating $E_{0m} = D_m(R_m)$.

With respect to the running time of the algorithm, we note that $|\underline{\underline{X}}_i| = O(i)$ and $|\underline{\underline{x}}_i(X)| \leq 3$ $(X \in \underline{\underline{X}}_i)$. Hence, for fixed i, all $D_i(X)$ are determined in $O(i)$ time. It follows that E_{0m} is found in $O(m^2)$ time and that the original problem is solved in $O(n^4)$ time.

In the case that the c_i need not be equal, the problem is NP-hard. The general dynamic programming approach solves the problem in $O(R_n C_n)$ time. The question arises, however, if the available information on the structure of optimal production plans is useful in deriving an efficient algorithm. We can solve problem P_{0m} by successively fixing the period f that is allowed to be fractional. Let $P_{0m}^{(f)}$ denote the subproblem under the additional restriction that $x_i \in \{0,c_i\}$ $(i \neq f)$, and let $E_{0m}^{(f)}$ be the optimal solution value to this problem. The sets of feasible cumulative production levels for $P_{0m}^{(f)}$ satisfy the following restrictions:

$$\underline{\underline{X}}_0 = \{0\},$$

$$\underline{\underline{X}}_i = \{X \mid X \in \{X', X'+c_i \mid X' \in \underline{\underline{X}}_{i-1}\},$$
$$\max\{R_i+1, R_m-(C_m-C_{i+1})\} \leq X \leq R_m\} \qquad (i = 1,\ldots,f-1),$$

$$\underline{\underline{X}}_i = \{X \mid X \in \{X', X'-c_i \mid X' \in \underline{\underline{X}}_{i+1}\},$$
$$R_i+1 \leq X \leq C_i\} \qquad (i = f,\ldots,m-1),$$

$$\underline{\underline{X}}_m = \{R_m\}.$$

Thus, we generate the sets $\underline{\underline{X}}_0, \underline{\underline{X}}_1, \ldots, \underline{\underline{X}}_{f-1}$ according to a forward recursion and the sets $\underline{\underline{X}}_m, \underline{\underline{X}}_{m-1}, \ldots, \underline{\underline{X}}_f$ according to a backward recursion. Considering $\underline{\underline{X}}_{f-1} \times \underline{\underline{X}}_f$ we need only retain those pairs (X_{f-1}, X_f) for which $0 \leq X_f - X_{f-1} \leq c_f$. Therefore, we may subsequently carry out a backward and a forward recursion to reduce the size of $\underline{\underline{X}}_i$ for $i < f-1$ and $i > f$ respectively even further. Defining the sets of feasible production amounts by

$$\underline{\underline{x}}_i(X) = \{0, c_i\} \quad (i = 1,\ldots,f-1,f+1,\ldots,m), \quad \underline{\underline{x}}_f(X) = \{0,1,\ldots,c_f\},$$

we solve $P_{0m}^{(f)}$ by calculating $E_{0m}^{(f)} = D_m(R_m)$ according to the recursion (5).

With respect to the running time, we note that $|\underline{\underline{X}}_i| \leq R_m$, $|\underline{\underline{x}}_i(X)| = 2$ $(i \neq f)$ and $|\underline{\underline{x}}_f| = c_f+1$. Hence, $E_{0m}^{(f)}$ is determined in $O(mR_m + R_m c_f)$ time. It follows that $E_{0m} = \min_{1 \leq f \leq m}\{E_{0m}^{(f)}\}$ is found in $O(m^2 R_m + R_m C_m)$ time and that the original problem is solved in $O(n^4R_n+n^2R_nC_n)$ time. In terms of worst-case running time, this approach is inferior to standard dynamic programming. Some limited computational experience suggests that it is not likely to be a practical alternative, unless R_n and C_n are particularly large and the reduction achieved in the size of the sets $\underline{\underline{X}}_i$ becomes really significant.

Although practical experience has confirmed that polynomial algorithms are properly referred to as "good" ones, a similar statement would not be correct for pseudopolynomial algorithms. In particular, tree search methods with an exponential worst-case running time may be competitive under certain circumstances. For example, the KNAPSACK problem of Section 2 can be solved by dynamic programming in $O(tA)$ time [2], but for large values of A branch-and-bound tends to be more efficient [20]. Methods of the latter type are usually evaluated on an empirical basis by comparing their average performance on a "reasonable" set of test problems. The formal analysis of expected behavior of tree search algorithms requires the specification of a probability distribution over the class of all problem instances and appears to be technically complicated; see [13] for some results.

For the production planning problem with concave costs and arbitrary capacities, several tree search methods have been suggested. We mention the work of Chen [4] and Lambrecht and Vander Eecken [15] and, in particular, the algorithm recently proposed by Baker, Dixon, Magazine and Silver [1]. The latter authors considered the special case in which $p_i'(x) = \bar{p}x$, $h_i(I) = \bar{h}I$ $(i = 1,\ldots,n)$. (This problem is NP-hard, even for $\bar{p} = \bar{h} = 0$, as can be proved by a slight modification of the reduction in Section 2.) They found that for each subproblem $P_{\ell m}$ only the first period can be fractional, so that $E_{\ell m} = E_{\ell m}^{(\ell+1)}$. It follows that in this case the above dynamic programming approach requires only $O(n^3R_n+nR_nC_n)$ time.

Returning to the case of arbitrary concave costs, we have in fact considered the possibility of solving a typical subproblem P_{0m} by branch-and-bound. A search tree can be defined in an obvious manner. A node at the ℓ-th level $(1 \le \ell \le m)$ which represents a subset of solutions with fixed $x_i \in \{0,c_i\}$ $(i = 1,\ldots,\ell-1)$, has three immediate descendants, corresponding to the choices $x_\ell = 0$, $0 < x_\ell < c_\ell$, and $x_\ell = c_\ell$. If period ℓ is chosen to be fractional, an optimal production plan for periods $\ell,\ldots,m$ can be quickly determined by dynamic programming. Otherwise, a lower bound has to be calculated for the case that a period i ($\ell < i \leq m$) might yet be fractional.

We have not found a satisfactory solution to this bounding problem, but it may be instructive to indicate one approach that will be ineffective in general. Suppose that production and storage costs are underestimated by linear functions. The optimal solution value to the resulting problem can be obtained by a linear cost network flow computation. Alternatively, one can associate Lagrangean multipliers with the inequalities $I_i > 0$ ($i = \ell+1,\ldots,m-1$) and append these to the objective function. For fixed multiplier values, the resulting Lagrangean problem can be written as that of determining values $y_i = x_i/c_i$ that minimize a linear function subject to one linear equality, $0 \leq y_i \leq 1$ ($i = \ell+1,\ldots,m$), and the constraint that all but at most one y_i are integer. This is a continuous knapsack problem and its solution belongs to folklore; see [17] for an $O(m-\ell)$ implementation. One can then try to obtain a strong lower bound by searching for feasible multiplier values that maximize the solution value [7]. However, the very superfluity of the integrality constraints in the Lagrangean problem implies that this approach will do no better than the standard linear programming relaxation obtained by ignoring these constraints in P_{0m} [7]. The latter bound proved to be too weak to generate strong lower bounds and was thus disconsidered.

4. CONVEX COSTS

Let us now assume that all p_i and h_i are convex functions. Thus, they do not include set-up costs, but they can easily be adapted to enforce capacity limits.

Veinott [23] has shown that an optimal production plan can be obtained by satisfying each unit of demand in turn as cheaply as possible. An algorithm based on this rule has $O(nR_n)$ running time. It generalizes an algorithm due to Johnson [10] for the case of linear storage costs; Johnson's method essentially yields an initial solution to a linear transportation problem, which also happens to be optimal. As pointed out earlier, the problem can be formulated as a minimum convex cost network flow problem, and Veinott's work can also be interpreted as the use of the "incremental" algorithm [3;8] for solving such problems.

Examples can be constructed to show that the above rule cannot be stretched to allocate more than one unit of demand at a time. In any case, however, the problem is not harder than linear programming [21], and we conjecture that it is solvable in strictly polynomial time. Indeed, in the case of linear storage costs, generalized sorting techniques (cf. [9]) can be applied to yield such an algorithm [14]. In the case of arbitrary convex storage costs, it seems likely that the out-of-kilter method combined with scaling techniques [16] solves the problem in polynomial time.

In the case that additional set-up costs b_i are specified, the problem is NP-hard, even if $b_i = 1$ $(i = 1,\dots,n)$. As pointed out in Section 3, a branch-and-bound approach may offer a practical alternative to the general dynamic programming recursion. Such an approach could be based on a fixed charge network flow formulation (cf. [22]).

5. SUMMARY

We have analyzed the computational complexity of a class of deterministic production planning problem for various types of cost functions, set-up costs and capacity limits. Table 1 below indicates for each problem type whether it is solvable in polynomial time, NP-hard, or currently open, and also gives the running time of the best available algorithm for its solution (under the assumption that $nR_n \geq C_n$).

TABLE I. SUMMARY OF COMPLEXITY RESULTS

p_i', h_i arbitrary		c_i			p_i', h_i convex
		infinite	equal	arbitrary	
b_i	zero	! $O(nR_n^2)$	! $O(nR_n c)$	! $O(R_n C_n)$	? $O(nR_n)$
	equal	! $O(nR_n^2)$	! $O(nR_n c)$	! $O(R_n C_n)$	! $O(R_n C_n)$
	arbitrary	! $O(nR_n^2)$	! $O(nR_n c)$	! $O(R_n C_n)$	! $O(R_n C_n)$
p_i', h_i concave		* $O(n^2)$	* $O(n^4)$	! $O(R_n C_n)$	

* solvable in polynomial time

? open

! NP-hard, even for equal demands and zero storage costs

We hope that this overview has once again demonstrated the usefulness of arguments from complexity theory, when potential algorithmic improvements for a combinatorial optimization problem are being considered.

ACKNOWLEDGMENTS

The authors are grateful to M.L. Fisher and B.J. Lageweg for their useful comments. This research was partially supported by grants from the Netherlands Organization for the Advancement of Pure Research (Z.W.O.) and by the National Research Council of Canada.

REFERENCES

1. K.R. BAKER, P. DIXON, M.J. MAGAZINE, E.A. SILVER (1978) An algorithm for the dynamic lot-size problem with time-varying production capacity constraints. Working Paper 117, Department of Management Sciences, University of Waterloo.
2. R.E. BELLMAN, S.E. DREYFUS (1962) *Applied Dynamic Programming*. Princeton University Press, Princeton, N.J.
3. R.G. BUSACKER, P.J. GOWEN (1961) A procedure for determining a family of minimal-cost network flow patterns. Technical Paper 15, Operations Research Office, John Hopkins University.
4. K.-Y. CHEN (1976) A network approach to the capacitated lot-size problem. Unpublished Manuscript, The Wharton School, University of Pennsylvania, Philadelphia.
5. M. FLORIAN, M. KLEIN (1971) Deterministic production planning with concave costs and capacity constraints. *Management Sci.* 18,12-20.
6. M.R. GAREY, D.S. JOHNSON (1978) "Strong" NP-completeness results: motivation, examples and implications. *J. Assoc. Comput. Mach.* 25,499-508.
7. A.M. GEOFFRION (1974) Lagrangean relaxation for integer programming. *Math. Programming Study* 2,82-114.
8. T.C. HU (1969) Minimum-cost flows in convex-cost networks. *Naval Res. Logist. Quart.* 13,1-9.
9. D.B. JOHNSON, T. MIZOGUCHI (1978) Selecting the Kth element in $X+Y$ and $X_1+X_2+\ldots+X_m$. *SIAM J. Comput.* 7,147-153.

10. S.M. JOHNSON (1957) Sequential production planning over time at minimum cost. *Management Sci.* 3,435-437.
11. R.M. KARP (1972) Reducibility among combinatorial problems. In: R.E. MILLER, J.W. THATCHER (eds.) (1972) *Complexity of Computer Computations*. Plenum Press, New York, 85-103.
12. R.M. KARP (1975) On the computational complexity of combinatorial problems. *Networks* 5,45-68.
13. R.M. KARP (1976) The probabilistic analysis of some combinatorial search algorithms. In: J.F. TRAUB (ed.) (1976) *Algorithms and Complexity: New Directions and Recent Results*. Academic Press, New York, 1-19.
14. R.M. KARP (1978) Personal communication.
15. M. LAMBRECHT, J. VANDER EECKEN (1978) A capacity constrained single-facility dynamic lot-size model. *Eur. J. Operational Res.* 2,132-136.
16. E.L. LAWLER (1976) *Combinatorial Optimization: Networks and Matroids.* Holt, Rinehart and Winston, New York.
17. E.L. LAWLER (1978) Fast approximation algorithms for knapsack problems. *Math. Operations Res.,* to appear.
18. J.K. LENSTRA, A.H.G. RINNOOY KAN (1978) Computational complexity of discrete optimization problems. *Ann. Discrete Math.* 4, to appear.
19. S.F. LOVE (1973) Bounded production and inventory models with piece-wise concave costs. *Management Sci.* 20,313-319.
20. S. MARTELLO, P. TOTH (1977) An upper bound for the zero-one knapsack problem and a branch and bound algorithm. *Eur. J. Operational Res.* 1,169-175.
21. R.R. MEYER (1977) A class of nonlinear integer programs solvable by a single linear program. *SIAM J. Control Optimization* 15,935-946.
22. P. RECH, L.G. BARTON (1970) A non-convex transportation algorithm. In: E.M.L. BEALE (ed.) (1970) *Applications of Mathematical Programming Techniques*. American Elsevier, New York, 250-260.
23. A.F. VEINOTT, JR. (1964) Production planning with convex costs: a parametric study. *Management Sci.* 10,441-460.
24. H.M. WAGNER, T.M. WHITIN (1958) Dynamic version of the economic lot size model. *Management Sci.* 5,89-96.
25. W.I. ZANGWILL (1966) A deterministic multiperiod production scheduling model with backlogging. *Management Sci.* 13,105-119.