P. Hansen, ed., Studies on Graphs and Discrete Programming

A SOLVABLE MACHINE MAINTENANCE MODEL WITH APPLICATIONS

A.H.G. RINNOOY KAN and J. TELGEN
Erasmus University, P.O. Box 1738, Rotterdam, The Netherlands

At the end of each time period maintenance work can be performed on a machine at constant cost, reducing operating cost in the next period to zero. If this option is not exercised, the operating cost in the next period increases by a constant. We derive a closed form expression for the maintenance policy that minimizes total cost. Various applications of this model are discussed, one of which is connected with the optimal reinversion policy for linear programming.

1. Introduction

Consider the following *machine maintenance problem.* At the end of each time period, the possibility exists of performing maintenance work on a machine at given constant cost. If this option is not exercised, the operating cost for the next period increases by a constant; if it is, this cost decreases to zero. At some given point in the future, the entire machine will be scrapped. What is the maintenance policy that minimizes total cost?

More explicitly, consider a system with denumerable state space $\{0, 1, 2, 3, \ldots\}$ over a time interval consisting of T periods. At the end of period $t \in \{1, \ldots, T\}$, we can reduce the state of the system to 0 at cost a or increase it from $i-1$ to i at cost bi, where a and b are given nonnegative constants. Initially, at time 0, the system is in state i_0. The problem is to determine a *policy* that minimizes total cost. As will be shown in Section 2, an integer programming formulation of this problem can be solved to yield a closed form solution, characterizing the optimal policy as a function of a, b, i_0 and T. If the value of the latter parameter is not known in advance, one can resort to various devices, depending on the information that is available. For example, the probability distribution of T may be known, we may be able to estimate T or a bound on T from the past behaviour of the system or—in a Bayesian framework—we may know the type of probability distribution of T, together with some prior information on the parameters of the distribution.

In Section 3 we turn to applications of the model. The most interesting one occurs in the context of the *product form of the inverse* variant of the simplex method for linear programming. Here, the state of the system corresponds to the number of elementary matrices that are currently stored in a file, the product of which determines the basis inverse. After each simplex iteration, we can either perform a reinversion or add another elementary matrix to the file. The results of the above model can be immediately applied to this case. As will be illustrated in the same section, the model readily lends itself to various other interpretations as well.

The final section contains concluding remarks and some topics for further research.

2. A closed form solution

In this section we assume that the four parameters of the model, a, b, i_0 and T, are fixed and known in advance.

A maintenance policy corresponds to a partition of the time horizon in *maintenance intervals*. Let $t_1 < t_2 < \cdots < t_{k-1}$ be the points in time at which maintenance occurs, at total cost $(k-1)a$. The operating cost is

$$\sum_{i=i_0}^{t_0+t_1-1} ib + \sum_{j=2}^{k-1} \sum_{i=0}^{t_j-t_{j-1}-1} ib + \sum_{i=0}^{T-t_{k-1}-1} ib.$$

We define the *state* in which the maintenance is done as follows:

$$x_1 = t_1 + i_0,$$
$$x_j = t_j - t_{j-1}, \quad (j = 2, \ldots, k-1)$$
$$x_k = T - t_{k-1},$$

and rewrite the combined maintenance and operating cost C as

$$\begin{aligned} C &= (k-1)a + b\left(\sum_{i=i_0}^{x_1-1} i + \sum_{j=2}^{k} \sum_{i=1}^{x_j-1} i\right) \\ &= (k-1)a + \tfrac{1}{2}b\left(\sum_{j=1}^{k} x_j^2 - \sum_{j=1}^{k} x_j - i_0^2 + i_0\right) \\ &= \tfrac{1}{2}b(-i_0^2 - T) + (k-1)a + \tfrac{1}{2}b \sum_{j=1}^{k} x_j^2. \end{aligned}$$

Ignoring constant terms, we arrive at the following mathematical programming problem:

$$\begin{aligned} &\text{minimize} \quad \alpha k + \sum_{j=1}^{k} x_j^2, \\ &\text{subject to} \quad \sum_{j=1}^{k} x_j = \tau, \\ &\qquad k \in \{1, \ldots, T+1\}, \\ &\qquad x_j \geq 0 \text{ and integer} \quad (j = 1, \ldots, k) \end{aligned}$$

with

$$\alpha = 2a/b, \qquad \tau = T + i_0.$$

First of all, let us note that the problem obtained by relaxation of the integrality constraints on x_j ($j = 1, \ldots, k$) can be solved easily. For given k, we calculate

$$g(k) = \min\left\{\sum_{j=1}^{k} x_j^2 \;\middle|\; \sum_{j=1}^{k} x_j = \tau, x_j \geqslant 0 \; (j = 1, \ldots, k)\right\}$$

by forming the Lagrangean function

$$\bar{g}(k; x, \lambda) = \sum_{j=1}^{k} x_j^2 - \lambda\left(\sum_{j=1}^{k} x_j - \tau\right)$$

and setting the derivatives equal to 0:

$$\frac{\partial \bar{g}}{\partial x_j} = 2x_j - \lambda = 0 \quad (j = 1, \ldots, k),$$

$$\frac{\partial \bar{g}}{\partial \lambda} = \sum_{j=1}^{k} x_j - \tau = 0.$$

We obtain the well-known result:

$$x_1^* = x_2^* = \cdots = x_k^* = \tau/k,$$

with $x_j^* \geqslant 0$ ($j = 1, \ldots, k$) as required. These values are easily verified to define a minimum with value τ^2/k, so that the relaxed problem becomes:

$$\begin{aligned} \text{minimize} \quad & G(k) = \alpha k + \tau^2/k, \\ \text{subject to} \quad & k \geqslant 0, \\ & k \in \{1, \ldots, T+1\}. \end{aligned}$$

The objective function G is unimodal and convex so that the global minimum G^* will be attained at the integer round-down or round-up of the continuous solution $k = \tau/\sqrt{\alpha}$:

$$k^* \in \{\lfloor \tau/\sqrt{\alpha} \rfloor, \lceil \tau/\sqrt{\alpha} \rceil\}$$

under the reasonable assumption that at least one of these values is in the interval $[1, T+1]$.

If the integrality constraints on x_j ($j = 1, \ldots, k$) are not relaxed, the problem can be approached in a similar manner. We start by calculating

$$f(k) = \min\left\{\sum_{j=1}^{k} x_j^2 \;\middle|\; \sum_{j=1}^{k} x_j = \tau, x_j \geqslant 0 \text{ and integer } (j = 1, \ldots, k)\right\}.$$

We first verify that $(x_1, \ldots, x_k)$ minimizes $\sum_{j=1}^{k} x_j^2$ subject to the above constraints if and only if $|x_i - x_j| \leq 1$ for all (i, j). In fact, we simply consider a pair of feasible solutions $(x_1, \ldots, x_i, \ldots, x_j, \ldots, x_k)$ and $(x_1, \ldots, x_i - 1, \ldots, x_j + 1, \ldots, x_k)$ and note that the latter one will be superior to the former one if

$$(x_i - 1)^2 + (x_j + 1)^2 < x_i^2 + x_j^2,$$

i.e., if $x_i - x_j > 1$.

Hence, we obtain an optimal solution $x_1^*, \ldots, x_k^*$ determining $f(k)$ by defining

$$\tau(k) \equiv \tau \ (\mathrm{mod}\ k)$$

and setting

$$x_1^* = \cdots = x_{\tau(k)}^* = \frac{\tau - \tau(k)}{k} + 1,$$

$$x_{\tau(k)+1}^* = \cdots = x_k^* = \frac{\tau - \tau(k)}{k}.$$

It follows that

$$f(k) = \tau(k)\left(\frac{\tau - \tau(k)}{k} + 1\right)^2 + (k - \tau(k))\left(\frac{\tau - \tau(k)}{k}\right)^2,$$

and the original problem can be rewritten as follows:

$$\text{minimize} \quad F(k) = \alpha k + \frac{\tau^2}{k} + \frac{1}{k}\,\tau(k)(k - \tau(k)),$$

$$\text{subject to} \quad k \in \{1, \ldots, T+1\}.$$

To solve this problem, note that $\tau(k)$ is a discontinuous, piecewise linear function of k: on each interval

$$I_i = \left\{ k \,\middle|\, \frac{\tau}{i+1} < k \leq \frac{\tau}{i},\ k \text{ integer} \right\} \quad (i = 1, \ldots, \tau)$$

it is the case that

$$(i+1)k > \tau, \qquad ik \leq \tau$$

and hence on I_i

$$\tau(k) \equiv \tau \ (\mathrm{mod}\ k) = \tau - ik.$$

Of course, one or more of the intervals I_i may be empty. This, however, can easily be verified not to affect the subsequent arguments.

It follows that on I_i the function $F(k)$ can be written as

$$\begin{aligned} F(k) &= \alpha k + \frac{\tau^2}{k} + \frac{1}{k}(\tau - ik)(k - \tau + ik) \\ &= (\alpha - i(i+1))k + (2i+1)\tau. \end{aligned}$$

Thus, on each interval I_i the function F is linear in k, and its minimum F_i^* is assumed at one of the endpoints, depending on the sign of $\alpha - i(i+1)$. The only positive root of $\alpha - i(i+1) = 0$ is equal to

$$i^* = -\tfrac{1}{2} + \tfrac{1}{2}\sqrt{1+4\alpha}.$$

It follows that:

(i) if $i \geqslant i^*$, the minimum value F_i^* is attained at $k = \lfloor \tau/i \rfloor$;

(ii) if $i \leqslant i^*$, the minimum value F_i^* is attained at $k = \lfloor \tau/(i+1) \rfloor + 1$.

We shall now show that in case (i) $F_i^* \leqslant F_{i+1}^*$, and that in case (ii) $F_i^* \leqslant F_{i-1}^*$.

(i)
$$\begin{aligned} F_i^* - F_{i+1}^* &= (\alpha - i(i+1))\left\lfloor \frac{\tau}{i} \right\rfloor - (\alpha - (i+1)(i+2))\left\lfloor \frac{\tau}{i+1} \right\rfloor - 2\tau \\ &= (\alpha - i(i+1))\left(\left\lfloor \frac{\tau}{i} \right\rfloor - \left\lfloor \frac{\tau}{i+1} \right\rfloor\right) + \left((2i+2)\left\lfloor \frac{\tau}{i+1} \right\rfloor - 2\tau\right). \end{aligned}$$

The first term is nonpositive because $i \geqslant i^*$ implies that $\alpha \leqslant i(i+1)$, the last term is nonpositive because

$$\left\lfloor \frac{\tau}{i+1} \right\rfloor \leqslant \frac{\tau}{i+1} = \frac{2\tau}{2i+2}.$$

(ii)
$$\begin{aligned} F_i^* - F_{i-1}^* &= (\alpha - i(i+1))\left(\left\lfloor \frac{\tau}{i+1} \right\rfloor - \left\lfloor \frac{\tau}{i} \right\rfloor\right) \\ &\quad + \left((i(i-1) - i(i+1))\left(\left\lfloor \frac{\tau}{i} \right\rfloor + 1\right) + 2\tau\right). \end{aligned}$$

This expression can be proved to be nonpositive by means of an argument similar to the one above.

It follows that the values F_i^* form a nonincreasing sequence for $i \leqslant i^*$ and a nondecreasing sequence for $i \geqslant i^*$. The global minimum F^* will either be realized by the right endpoint of $I_{\lceil i^* \rceil}$ for

$$k = \lfloor \tau / \lceil i^* \rceil \rfloor$$

or by the left endpoint of $I_{\lfloor i^* \rfloor}$ for

$$k = \left\lfloor \frac{\tau}{\lfloor i^* \rfloor + 1} \right\rfloor + 1 = \left\lceil \frac{\tau}{\lceil i^* \rceil} \right\rceil,$$

where the last equality holds only if i^* and $\tau/(\lfloor i^* \rfloor + 1)$ are not integers.

Assuming that this is the case, we have that k^* is the integer round-down or round-up of $\tau/\lceil i^* \rceil$:

$$k^* \in \{\lfloor \tau/\lceil i^* \rceil \rfloor, \lceil \tau/\lceil i^* \rceil \rceil\}.$$

It is instructive to compare this result to the outcome of the continuous approximation

$$k^* \in \{\lfloor \tau/\sqrt{\alpha} \rfloor, \lceil \tau/\sqrt{\alpha} \rceil\}.$$

The two denominators, $\lceil i^* \rceil = \lceil -\frac{1}{2}+\frac{1}{2}\sqrt{1+4\alpha} \rceil$ and $\sqrt{\alpha}$ are close to each other. In particular,

$$\sqrt{\alpha}-\tfrac{1}{2} < -\tfrac{1}{2}+\tfrac{1}{2}\sqrt{1+4\alpha} \leq \sqrt{\alpha}$$

so that

$$|\sqrt{\alpha}-\lceil -\tfrac{1}{2}+\tfrac{1}{2}\sqrt{1+4\alpha} \rceil| \leq 1.$$

Of course, depending on the value of τ the continuous approximation can be arbitrarily far away from the real optimum. In particular, if the cost of maintenance and operation in the first period is less than the operating cost in the second period (i.e. $a+b<2b$ or $a<b$ and hence $\lceil i^* \rceil = 1$), maintenance will be performed in every time period. But in the continuous approximation maintenance will be performed in every time period only if $2a \leq b$.

Our conclusion is that we have obtained a closed form solution to an integer programming problem. It can be calculated just as simply as the continuous approximation and, depending on the particular values of the parameters a, b, i_0 and T it can be of significantly better quality.

What happens if some of these parameters are not known *a priori*? The obvious candidate to consider for this role is T, since it is reasonable to suspect that the life time of the machine may not be perfectly known in advance. In a few simple cases, the previous results are directly applicable. For instance, if the expected value μ_T of T is known, then the expected value of the objective function over the interval I_i is given by

$$(\alpha - i(i+1))k + (2i+1)(\mu_T + i_0)$$

and the same method can be applied to minimize expected cost. If upper and lower bounds on T are known, we can postulate a distribution for T over the interval between these values and use the corresponding expected value in the above calculations.

It seems more realistic to assume that information about T becomes gradually available as time goes on. This information should then be used to update the current estimate of T in a Bayesian fashion and to adjust the maintenance policy accordingly. The application dealt with in the next section provides an example of such a situation.

It is simple to extend the analysis to the case of an infinite time horizon ($T \to \infty$). Clearly, in this case the maintenance points will occur at regular intervals of length $x+1$. The cost per period is given by

$$\frac{1}{x}\left(a+\sum_{i=0}^{x} bi\right)=\frac{a}{x}+\frac{1}{2}bx+\frac{1}{2}b$$

which is minimized by choosing x equal to the integer round-up or round-down of $\sqrt{2a/b}=\sqrt{\alpha}$. These can be viewed as limiting cases of the continuous approximation values derived above.

A less trivial extension, that seems worthy of further exploration, is to describe the state transitions as a stochastic rather than as a deterministic process and to invoke techniques from Markov programming. We shall not investigate this possibility any further in this paper.

3. Applications

The model described and solved in the preceding sections can be applied in the context of linear programming routines. In the *product form of the inverse* variation of the simplex method, the inverse of the current basis is not explicitly available, but only implicitly as the product of a sequence of elementary matrices stored in the so called *ETA-file* [8]. In each simplex iteration, a new elementary matrix describing the corresponding pivot transformation is added to the file. At certain moments a complete basis *reinversion* is performed, as a result of which the file length is reduced to a constant. The decision when to reinvert can be based on considerations of numerical accuracy, but also on computational arguments: the amount of work in each iteration is proportional to the current file length and at some point a time-consuming reinversion will become preferable.

The problem of choosing the proper reinversion point on other than numerical arguments has been recognized since the early days of linear programming. Hoffman wrote in 1955: 'Dantzig has informed us that the use of the product form leads to exciting moments for the operator. If a large number of vectors have accumulated in the product, one is tempted to clean up in the manner previously described. This takes substantial time, however, and one is also tempted to hang on a few iterations longer in the hope that the problem will be solved'. [6]

Nowadays, all major linear programming codes [3; 7] contain *triggering mechanisms* that decide upon the moment of reinversion (*e.g.* [2, p. 225]). A few examples of some triggering criteria are the following:

(i) a fixed number of iterations since the last reinversion, e.g.: reinversion after every 100 or 150 iterations;

(ii) a problem dependent number of iterations, e.g.: reinversion after every $\frac{1}{2}m$ iterations, where m is the number of rows;

(iii) an absolute amount of time, e.g.: reinversion after every 10 seconds of CPU time;

(iv) a problem dependent amount of time, e.g.: reinversion after every $\frac{1}{25}m$ seconds of CPU time;

(v) reinversion whenever the total time required for the iterations since the last reinversion plus reinversion time, divided by the number of these iterations, starts to increase [1, p. 15].

Quite often, one finds a mixture of these triggers; for instance, in the IBM MPSX/370 package, reinversion takes place after min{max{20, $m/10$}, 150} iterations [7, p. 91].

Our maintenance model from the previous section can be used to determine an *optimal reinversion point*. The parameter a represents the cost of reinversion, b represents the cost increase due to the appearance of an extra elementary matrix in the file, i_0 can be taken equal to 0 or to the current file length and T is the number of simplex iterations required to arrive at the final solution. Interestingly enough, the outcome of our infinite time horizon analysis

$$x_1^* = \cdots = x_{k^*}^* \in \{\lfloor\sqrt{2a/b}\rfloor, \lceil\sqrt{2a/b}\rceil\}$$

supports the triggering mechanism described under (v) above. This mechanism calls for a reinversion whenever

$$\frac{1}{x}\left(a + \sum_{i=0}^{x} bi\right) = \frac{a}{x} + \frac{1}{2}bx + \frac{1}{2}b$$

starts to increase, and this is easily seen to correspond to the above rule.

However, as stated before, this approximation can be arbitrarily far away from the optimal solution, indicating that the theoretical quality of triggering mechanisms used in practice is relatively poor. This is caused mainly by the fact that the decision when to reinvert is based on information from the past only. Thus a reinversion might be called for even in situations in which the process will terminate after one more iteration.

Of course, in linear programming the number of iterations T will generally not be known in advance. This provides a typical example of a situation in which more and more reliable estimates of its value can be made as the objective function approaches its optimum value. Some empirical studies suggest that this happens according to the regular pattern illustrated in Fig. 1, taken from [5].

In that case, an estimate of T could be based on appropriately fitting an exponential or linear curve to the available data. However, some preliminary

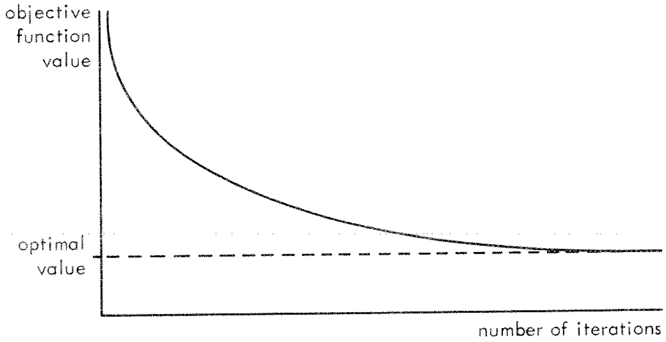

Fig. 1. A typical pattern of objective function value against number of iterations.

experiments that we have carried out indicate that this pattern need not always occur, and additional work is required to investigate the applicability of a more sophisticated reinversion criterion than those currently in use.

The basic ingredient of the model, *i.e.* a confrontation with the increasing cost of postponing a costly decision, can be recognized in many other situations as well. Consider for instance the *transportation of natural gas* through a *pipe line*. As the gas flows through the pipe, its *pressure* decreases linearly with the distance travelled, and at certain points it becomes attractive to install a *repressurization point* where the original pressure is restored. Apart from a certain fixed factor, the costs associated with such a point are quadratically proportional to the pressure loss. It is easy to verify that our closed form formula can be used to yield the optimal number of repressurization points.

We leave it to the reader to concoct other similar situations. (Consider, for instance, the problem of the organization which has to decide when to switch to strong and expensive arguments in order to extract a regular payment from its steady customers.) We do note that if, in addition to the fixed maintenance costs, maintenance charges are incurred that are linearly proportional to the length of the maintenance interval, the model can still be applied since this only adds a constant term to the objective function. For example, by way of one more algorithmic application, the model can be used to determine with which frequency an increasingly large set of linear constraints (*e.g.*, cutting planes) should be checked for possible redundancies [9].

4. Concluding remarks

We have found a closed form solution to the maintenance problem described in the Introduction by means of a linearization technique that may be applicable to other periodic optimization problems. An obvious question is under what extension the model will lose its property of *polynomial-time solvability*. The model involves only four parameters, but in spite of that it should be possible to identify certain generalizations that can be proved to be *NP-complete* [4] and thus are unlikely to admit of a polynomial-bounded algorithm.

Certain stochastic extensions mentioned at the end of Section 2 are worthy of further investigation and should facilitate the application of the model within linear programming routines or in other practical situations.

Acknowledgements

We are grateful to J.K. Lenstra and R.S. Garfinkel for many useful comments. The research of the second author was partially supported by a NATO Science Fellowship awarded by the Netherlands Organization for the Advancement of Pure Research (Z.W.O.).

References

[1] E.M.L. Beale, Sparseness in linear programming, in: J.K. Reid, ed., Large Sparse Sets of Linear Equations (Academic Press, New York, 1971).
[2] S.P. Bradley, A.C. Hax and T.L. Magnanti, Applied Mathematical Programming (Addison-Wesley, Reading, MA, 1977).
[3] C.D.C. APEX III, Reference Manual (1977).
[4] M.R. Garey and D.S. Johnson, Computers and Intractability: A Guide to the Theory of NP-completeness (Freeman, San Francisco, 1979).
[5] P.M.J. Harris, Pivot selection methods of the Devex LP code, Mathematical Programming (1) 1–28 (1973).
[6] A.J. Hoffman, How to solve a linear programming problem, in: H.A. Antosiewicz, ed., Proceedings of the Second Symposium on Linear Programming, Washington D.C. (1955).
[7] I.B.M. MPSX/370, Program Reference Manual (1976).
[8] W. Orchard-Hays, Advanced Linear Programming Computer Techniques (McGraw-Hill, New York, 1968).
[9] J. Telgen, Redundancy and Linear Programs, Ph.D. Thesis, Erasmus University Rotterdam (1979).