TECHNICAL REPORT NO. CS75026-R

ON COMPUTING A BUY/COPY POLICY USING THE PITT-KRAFT MODEL

by

Harvey J. Greenberg†

and

Donald H. Kraft*

†Department of Computer Science
Virginia Polytechnic Institute and State University
Blacksburg, Virginia 24061

October 1975

Computing Reviews No. 3.7

Keywords: library operations research, photocopy policy, nonlinear programming, optimization

* School of Librarianship, University of California, Berkeley, CA 94720.

ABSTRACT

The Pitt-Kraft model of buying versus photocopying results in a small, but complex, nonlinear program. This paper identifies a Kuhn-Tucker point and demonstrates that for certain parameter values it is not optimal. A policy generation procedure is presented; the purpose is to prevent convergence of a primal algorithm to this inferior policy, which satisfies the Kuhn-Tucker optimality conditions.

INTRODUCTION

Pitt and Kraft [2] presented a resource allocation model for a branch library system to determine how much capital should be allocated to the acquisition of new information items (e.g., monographs, serials, reports) and how much to allocate to the photoduplication of demand items not in inventory. The four decision variables (for a one-year planning horizon) are:

variable	meaning
x_1	number of information items to be acquired by the branch
x_2	number of trips to the main library (to photocopy)
x_3	number of photocopy requests for each trip
x_4	dollar cost to users for photocopy

Their objective is to maximize demand satisfaction, expressed as a function, $f(x)$, subject to a budgetary constraint of the form:

$$g(x) \leqq b,$$

and a demand limit of the form:

$$h(x) \leqq d,$$

where $x \equiv (x_1, x_2, x_3, x_4)$. Further, the first three decision variables must be nonnegative and integer-valued.

Thus, the Pitt-Kraft model is a nonlinear program with four decision variables and two constraints. The policy space consists of vectors in 4-space with the first three components required to be nonnegative and integer-valued. This is denoted:

P: maximize $f(x)$:

(1) $g(x) \leqq b$

(2) $h(x) \leqq d$

(3) $(x_1, x_2, x_3) \geqq 0$

(4) (x_1, x_2, x_3) integer-valued.

The functional forms, f, g and h, were derived by Pitt and Kraft and are of the following form:

$$f(x) = C_1 \ln \{C_2 + x_1\} + p(x)$$

$$g(x) = C_3 x_1 + C_4 x_2 + (C_5 - x_4) p(x)$$

$$h(x) = x_2 x_3 + C_1 \ln (C_2 + x_1)$$

where $p(x)$, appearing in the definitions of f and g, is:

$$p(x)=A_1 x_2 x_3 \exp \{-q(x_1) x_3 - A_2 x_4 + q(x_1)\}$$

$$q(x_1)=A_3/(A_4 - A_5 \ln\{C_2 + x_1\}).$$

The twelve constants $(C_1, C_2, C_3, C_4, C_5, A_1, A_2, A_3, A_4, A_5, b, d)$ depend upon input parameters (e.g., cost of photocopy, annual growth rate of the main library collection, number of days the branch is open); see ref, [2] for details.

To solve this nonlinear program a number of algorithms may be applied [1]. A feasible policy is not difficult to identify; namely, x=0 (the 'do-nothing' policy). This is feasible since

$$g(0) = h(0) = 0$$

and b, d > 0. The objective functional value is:

$$f(0) = C_1 \ln(C_2) .$$

This is the expected number of demands satisfied by the branch with its current inventory (no new acquisitions and no photocopy from the main library).

This suggests the use of a primal algorithm (i.e., one which iteratively improves the policy while remaining feasible), perhaps based on a direct ascent method with a penalty function or projection (see [1]). In our discussion the integer restriction (4) is relaxed. This may be justified by the largeness of noninteger values permitting roundoff (truncation to remain feasible); however, even if the integer problem is to be solved, the continuous approximation may be solved in conjunction with a branch-and-bound scheme.

Generally, primal algorithms converge to a Kuhn-Tucker point [1] - i.e., a policy, $\bar{x}$, which satisfies constraints (1) - (3) (but may not be integer-valued), and for which there exists nonnegative multipliers (u, v) to satisfy:

(1) for j= 1, 2, 3, 4: $$\left. \frac{\partial f}{\partial x_j} - u\frac{\partial g}{\partial x_j} - v\frac{\partial h}{\partial x_j} \right|_{x=\bar{x}} \leqq 0 \qquad (=0 \text{ if } \bar{x}_j > 0)$$

(2) $g(\bar{x}) < b$ implies $u=0$ and $h(\bar{x}) < d$ implies $v=0$.

The point of this paper is first to identify a Kuhn-Tucker point. Then, we shall demonstrate that there are realistic parameter values for which this point is not optimal. Finally, a policy generation procedure is given such that the objective value is greater than that of this Kuhn-Tucker point (if possible). This would prevent convergence to the inferior Kuhn-Tucker policy when starting at the generated policy.

A KUHN-TUCKER POINT

Let us consider the policy which specifies no trips to the main library to photocopy. Specifically, let $x_2^o = x_3^o = x_4^o = 0$. With this part of the policy fixed, let x_1^o be as large as possible without violating the constraints. Then, since $p(x^o) = 0$ (no matter what x_1^o is), we require:

$$C_3\, x_1^o \leqq b$$

and

$$C_1 \ln\,(C_2 + x_1^o) \leqq d.$$

This implies

$$x_1^o = \text{Min}\ \{b/C_3,\ \exp\ \{d/C_1\} - C_2\}.$$

Notice that x_1^o maximizes $f(x_1, x_2^o, x_3^o, x_4^o)$ over the feasible choices of x_1.

Theorem 1: x^o is a Kuhn-Tucker policy with multipliers:

$$u^o = 0 \text{ and } v^o = 1 \qquad \text{if } x_1^o < b/C_3$$

$$u^o = C_1/(b + C_3C_2) \text{ and } v^o = 0 \qquad \text{if } x_1^o = b/C_3$$

The proof immediately follows upon substituting x^o and (u^o, v^o) into the Kuhn-Tucker conditions.

We shall develop a strategy to generate a feasible policy which can be used to initiatiate an exact algorithm, such as those described in [1]. In so doing, we shall also demonstrate that the Kuhn-Tucker policy (x^o) is not optimal for certain values of the input constants. The analysis is followed by an example, to illustrate the policy generation algorithm which we shall define.

POLICY GENERATION

In this section we develop a two-phase method to generate a feasible policy. Starting with the Kuhn-Tucker policy (x^o), we investigate a class of policies that maintain the acquisition level, x_1^o, and examine potential improvement by using photocopy. The result of the algorithm generates either x^o or an improvement, if one exists, in the class of policies considered.

Fixing $x_1 = x_1^o$ removes one decision variable, and at least one of the two constraints holds with equality. For definiteness, we shall suppose the budget is tight, so that $x_1^o = b/C_3$. The remaining demand is

$$D \triangleq d - f(x^o).$$

Now observe that if we consider no photocopy (i.e. $x_2\,x_3 = 0$), the objective functional value cannot be increased. Therefore, we want to consider only those policies for which $x_2\,x_3 > 0$. From the budget constraint (1), this implies the photocopy price (x_4) must exceed the cost to the branch (C_5).

This leads to the following subproblem:

P1. Maximize $p(x_1^o, x_2, x_3, x_4)$:

(1') $C_4\,x_2 - (x_4 - C_5)p(x_1^o, x_2, x_3, x_4) \leqq 0$

(2') $x_2\,x_3 \leqq D$

(3') $(x_2, x_3) > 0$ and $x_4 > C_5$.

The maximand in the above subproblem is the amount of improvement we can obtain in the objective functional value; i.e.

$$f(x_1^o, x_2, x_3, x_4) = f(x^o) + p(x_1^o, x_2, x_3, x_4).$$

We shall now prove that, in subproblem P1, the demand limit (2') must hold with equality in any (sub-)optimal solution. The significance of this fact is that we can eliminate the number of trips (x_2) as a decision variable (as well as remove the demand constraint, itself).

<u>Theorem 2</u>: <u>Every optimal solution to subproblem</u> P1 <u>must satisfy</u> (2') <u>with equality</u>.

To prove Theorem 2, first note that satisfaction of the budget constraint is independent of the number of trips (x_2). That is, we can divide (1') by x_2, and, using the definition of $P(\cdot)$, we obtain the equivalent inequality:

$$C_4 - (x_4 - C_5)A_1 x_3 \exp[-q(x_1^o)x_3 - A_2 x_4 + q(x_1)] \leqq 0.$$

Therefore, if the demand limit does not hold with equality for a policy, then we can increase the number of trips (x_2). This increase results in a proportional increase in the objective functional value, so the policy could not be optimal.

In other words, if we charge enough to photocopy and meet only some of the demand, then we can increase demand satisfaction by making more trips.

It should be emphasized that the subproblem may not be feasible; indeed P1 cannot be feasible if x^o is optimal. Part of the analysis below considers this, and a test condition is derived which compares the relative values of the input constants to determine feasibility. In particular, notice that P1 is infeasible if D=0 (i.e. both the demand and budget constraints are simultaneously tight at x^o). That is, constraint (2') forces $x_2x_3=0$, which contradicts (3'). In other words, if there is no remaining demand to be satisfied, then the objective functional value cannot be increased, by photoduplication. In view of this fact, we can, and shall, assume $D>0$ for the remaining analysis.

Using Theorem 2 let us eliminate the number of trips as a decision variable by using the equation:

$$x_2 = D/x_3.$$

Then, rearranging (1'), our subproblem is reduced to the following:

P2: maximize $P(x_3,x_4)$:

$$x_3(x_4 - C_5)P(x_3, x_4) \geq C_4 D$$

and

$$x_3 > 0, \; x_4 > C_5,$$

where

$$P(x_3, x_4) = p(x_1^o, D/x_3, x_3, x_4).$$

Let us recapitulate the significance of subproblem P2. If it has a feasible solution, then we can improve the objective functional value (in P) by the amount, $P(x_3, x_4)$, the maximand in P2. If we solve P2, we shall have an optimal policy in the class for which the acquisition level is kept at x_1^o. If, on the other hand, subproblem P2 is not feasible, then x^o is optimal over the class of policies for which $x_1 = x_1^o$.

Thus, in any case we shall be able to generate a feasible policy which is optimal over the class that maintains the acquisition level at x_1^o. For certain parameter values, this will be the Kuhn-Tucker policy, x^o, and for other values we shall generate an improvement over x^o.

We shall now derive a necessary and sufficient condition for the subproblem, P2, to have a feasible solution. This condition will compare the relative values of the input constants.

First, note that the maximand, in P2, satisfies:

$$P\ (x_3,x_4)=A_1\ D\ \exp\ \{-Qx_3 - A_2\ x_4 + Q\},$$

where

$$Q=q\,(x_1^o).$$

Therefore, the constraint to be satisfied is:

$$x_3\ \exp\ \{-Qx_3\}\ x_4'\exp\{-A_2x_4'\} \geqq B,$$

where

$$B \equiv (C_4/A_1)\ \exp\ \{-Q + A_2\ C_5\}$$

and

$$x_4' = x_4 - C_5.$$

This must hold for x_3, $x_4' > 0$.

<u>Theorem 3</u>: <u>Subproblem</u> P2 <u>is feasible if, and only if,</u>

$$B\ Q\ A_2\ e^2 \leqq 1.$$

The proof uses the fact that the constraint function is maximal when $x_3=1/Q$ and $x_4' = 1/A_2$. Therefore, substitution of these values yields the largest value, and the result immediately follows.

An implication of Theorem 2 is that the Kuhn–Tucker policy (x^o) can be improved by allowing photoduplication if enough revenue can be generated by charging users to cover costs, while the budget is spent totally on new acquisition. If the test condition in Theorem 3 fails, then subproblem P2 has no feasible solution. This means we cannot improve x^o while maintaining $x_1 = x_1^o$ and $x_2 = D/x_3$.

If the test condition in P2 succeeds, then we can improve the policy. In particular, we could define $x_3 = 1/Q$ and $x_4 = C_5 + 1/A_2$, since this must be feasible in P2. However, we can do better by actually solving P2 (globally).

<u>Theorem 4</u>: <u>If</u> P2 <u>is feasible, then an optimal solution is given by</u>:

$$x_3 = y/Q \text{ and } x_4 = y/A_2 ,$$

<u>where</u> y <u>is the least solution to</u>:

$$y^2 \exp\{-2y\} = B\ Q\ A_2.$$

The proof follows from classical Lagrange Multiplier Theory [1]. The parameter, y, must exist when $B\ Q\ A_2 \leqq e^{-2}$, and the smallest root will occur in the interval, (0, 1].

To compute y, any interval reduction method may be employed, such as bisection. Putting these facts together we have the

Policy Generation Algorithm

This algorithm generates a feasible policy whose objective functional value is at least as great as $f(x^o)$.

step 1 (initialize). Set $x_2 = x_3 = x_4 = 0$ and

compute $x_1 = b/C_3$, $f = C_1 \ln (C_2 + x_1)$.

step 2 (test assumption). Compute $D = d-f$.

If $D = 0$, EXIT; if $D<0$, transfer to analogous procedure (with $x_1 = \mathrm{EXP}[d/c_1] - C_2$).

step 3 (compute constants). Compute

$$Q = A_3/(A_4 - A_5 f/C_1),$$

$$B = (C_4/A_1)\ \mathrm{EXP}\ [-Q + A_2 C_5].$$

$$C = B\ Q\ A_2.$$

step 4 (test subproblem). If $C\ e^2 > 1$, EXIT.

step 5 (compute y parameter). EXECUTE procedure to compute

$$y \text{ in } (0, 1]:\ y^2\ \mathrm{EXP}\ (-2y) = C.$$

step 6 (update policy). Compute

$$x_3 = y/Q$$

$$x_4 = C_5 + y/A_2$$

$$x_2 = D/x_3$$

$$f = f + P(x_3, x_4).$$

EXAMPLE

We shall illustrate the Policy Generation Algorithm with an example. The following input values were derived from [2].

PARAMETER	VALUE
C_1	1967
C_2	30001
C_3	10
C_4	20
C_5	0.4
A_1	0.61
A_2	0.2
A_3	0.5
A_4	200
A_5	15.7
b	35000
d	25000

The acquisition level is computed:

$$x_1^o = b/c_3 = 3500.$$

This implies $f(x^o) = 20494$, so the remaining demand is:

$$D = d - f(x^o) = 4506.$$

Next, the constants in step 3 are computed to be:

$Q = 0.0137$ $\qquad B = 33.24$ $\qquad C = 0.0913$

We note that $Ce^2 = 0.674 < 1$, so problem P2 is feasible. We next proceed to compute the parameter, y, from the equation:

$$y^2 \exp\{-2y\} = 0.0913$$

A (near) value for y is 0.50 The policy is then updated by:

$$x_3 = y/Q = 36.42$$

$$x_4 = C_5 + y/A_2 = 2.90$$

$$x_2 = D/x_3 = 123.72$$

The amount of improvement is:

$$P(x_3, x_4) = 853.23.$$

Using truncation the generated policy is:

number of acquisitions (x_1)	= 3500
number of trips (x_2)	= 123
average amount of photocopy per trip (x_3)	= 36
photocopy price (x_4)	= 2.90

In closing, we point out that the above example was solved (including the multiplier search) with a pocket calculator in about 30 minutes.

REFERENCES

[1] D. Luenberger, Introduction to Linear and Nonlinear Programming, Addison-Wesley, Reading, Mass., 1973.

[2] W. B. Pitt and D. H. Kraft, "Buy or Copy? A Library Operations Research Model", Information Storage and Retrieval, 10 (1974) 331-341.