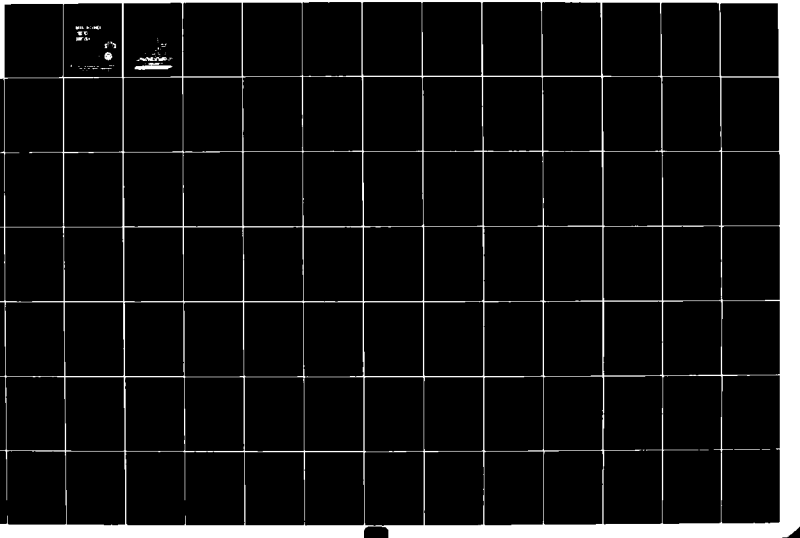
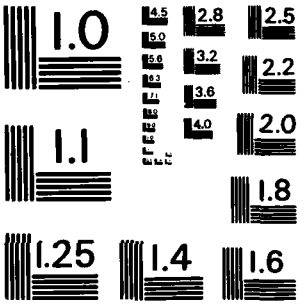
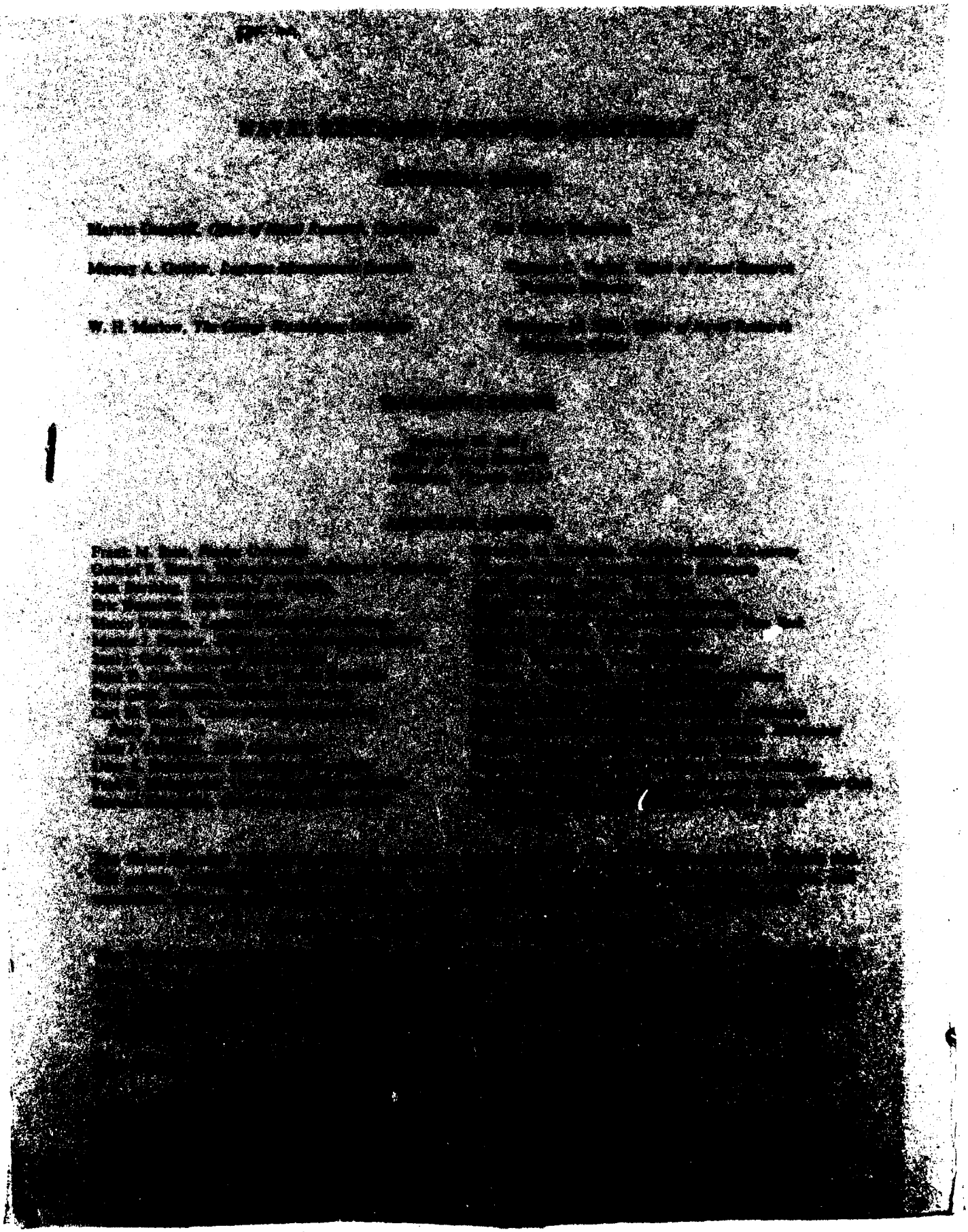
AD A125 206 NAVAL RESEARCH LOGISTICS QUARTERLY VOLUME 29 NUMBER 3 1/2
(U) OFFICE OF NAVAL RESEARCH ARLINGTON VA H SOLOMON
SEP 82

UNCLASSIFIED F/G 15/5 NL





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS - 1963 - A



FUTURE PUBLICATION OF NAVAL RESEARCH LOGISTICS QUARTERLY

The Office of Naval Research will discontinue publication of the Naval Research Logistics Quarterly after distribution of the December 1982 issue.

The journal will continue to be published, however, and will appear as a nongovernmental periodical to be published with the cooperation of the Office of Naval Research.

The new publisher will be John Wiley and Sons, Inc. The Naval Research Logistics Quarterly will become a copyrighted journal in the Wiley-Interscience series, and will be sold on a subscription basis by the publisher. The cost for a one year subscription will be $60.00 to individuals and institutions. Wiley-Interscience publication will be initiated with the March 1983 issue, Vol. 30, No. 1.

A substantial professional discount will be offered to members of the Operations Research Society of America and The Institute for Management Sciences. Members of ORSA or TIMS may subscribe to the journal for $20.00 per year.

If you are currently receiving the NRLQ by paid subscription through the Superintendent of Documents, U.S. Government Printing Office, and your subscription is due to expire after the December 1982 issue, a refund for the unfulfilled portion of your subscription will be issued to you by the Government Printing Office. The new publisher has agreed to honor requests at the GPO rate for unfulfilled portions which were subscribed to previously. The Superintendent of Documents will notify subscribers if they are due refunds because of such termination, and refund checks will be mailed subsequently. Subscribers may then elect to extend their subscriptions at the GPO rate by contacting the publisher and enclosing their endorsed refund checks in payment for the issues due subsequent to termination. An order form to facilitate this extension and/or renewal by the new publisher is enclosed.

Distribution through officially authorized Navy lists and SupDocs dissemination to government repository library centers will be discontinued after the December 1982 issue.

The Office of Naval Research regrets that it cannot provide support incident to such distribution and that those activities requiring copies for official purposes will be expected to bear the necessary costs directly.

Due to restrictions governing the use of GPO subscription lists, the publisher will be unable to follow-up with solicitations mailed with listings provided by the Superintendent of Documents; hence, future promotional efforts may only be through professional and trade sources. Because of this some current recipients may not receive anticipated additional notices from the publisher, and may not maintain the continuity of their collections if they rely upon the receipt of subscription solicitations from the publisher. Thus, all present recipients who are interested in receiving the journal are encouraged to initiate communications with the publisher promptly so that their names will not be purged by default.

It is expected that current editorial policies will be continued in the commercially published version. Professor Herbert Solomon of Stanford University will be the new editor in charge and will be responsible for the review of manuscripts in the future. New manuscripts submitted for consideration for publication may be sent to him immediately (4 copies) at the following address:

Professor Herbert Solomon, Editor
Naval Research Logistics Quarterly
Department of Statistics
Sequoia Hall
Stanford University
Stanford, California 94305

The Office of Naval Research is pleased with the recognition of the contributions to research promulgated through the Naval Research Logistics Quarterly. As ONR's own scientific journal it has reflected Navy interests in diverse areas of mathematics research which we believed would enrich future work in logistics and systems analysis, and has well served its intended purpose of providing a forum for mathematicians and logistics engineers to interact at the highest scientific level. The willingness of the publisher and scientific community to perpetuate the journal on a self-sustaining independent basis is profoundly reassuring that its value has been clearly established. With the forthcoming transition to the private sector, the journal will be able to better serve the entire scientific community in the area of logistics research.

ON SOME STOCHASTIC INEQUALITIES INVOLVING MINIMUM OF RANDOM VARIABLES*

Peter Kubat

The Graduate School of Management
The University of Rochester
Rochester, New York

ABSTRACT

Let X_i be independent IFR random variables and let Y_i be independent exponential random variables such that $E[X_i] = E[Y_i]$ for all $i = 1, 2, \ldots, n$. Then it is well known that $E[\min_{1 \leq i \leq n}(X_i)] \geq E[\min_{1 \leq i \leq n}(Y_i)]$. Nevertheless, for exponentially distributed Y_i's and for a decreasing convex function $\Phi(\cdot)$ it is shown that

$$E[\Phi(\min_{1 \leq i \leq n}(X_i))] \leq E[\Phi(\min_{1 \leq i \leq n}(Y_i))].$$

1. INTRODUCTION

The bounds for the mean lives of series and parallel systems under various assumptions on component life distribution were extensively treated in the well-known book by Barlow and Proschan [3].

In particular, if we denote by $X_i(Y_i)$ the life length of the ith component in the first (second) series system, then the following proposition holds:

THEOREM 1: (Barlow and Proschan [3, p. 122]). Let $X_i(Y_i)$ have continuous distribution $F_i(G_i)$ with the mean μ_i. Let $X_1, \ldots, X_n$ $(Y_1, \ldots, Y_n)$ be independent and $F_i \lesssim_* G_i$, $i = 1, \ldots, n$, i.e., F_i is star-shaped with respect to G_i[$F_i \lesssim_* G_i$ iff $(1/x)G_i^{-1}F_i(x)$ is increasing for $x \geq 0$]. Then the mean life of a series system using components with lives $X_1, \ldots, X_n$ is greater than the corresponding system mean life using components with lives $Y_1, \ldots, Y_n$, that is,

$$E[\min(X_1, \ldots, X_n)] \geq E[\min(Y_1, \ldots, Y_n)]. \tag{1.1}$$

In this note, it will be shown that for F_i having Increasing Failure Rate (IFR) and G_i being exponential we get

$$E[\Phi(\min(X_1, \ldots, X_n))] \leq E[\Phi(\min(Y_1, \ldots, Y_n))], \tag{1.2}$$

*This work was supported in part by the Office of Naval Research under grant No. CNA SUB N00014-76-C-0001.

provided that $\Phi(\cdot)$ is decreasing convex. A special case of the inequality (1.2) was already discussed in [2] where bounds for the discounted mean of the series system life was considered.

2. BOUNDS FOR THE MEAN OF A FUNCTION OF LIFE OF THE SERIES SYSTEM

Before we prove the main result the following two lemmata are needed.

LEMMA 1: Let $X_i \sim F_i$, $i = 1, 2$, be two positive continuous r.v.'s such that $F_1 \lesssim_* F_2$ and $E[X_1] = E[X_2]$. Then,

$$E[\Phi(X_1)] \leqslant E[\Phi(X_2)] \tag{2.1}$$

for any convex and bounded function $\Phi(\cdot)$.

PROOF: For $i = 1, 2$ we have

$$E[\Phi(X_i)] = \int_0^\infty \Phi(x)\,dF_i(x)$$
$$= -\Phi(x)\bar{F}_i(x)\Big|_0^\infty + \int_0^\infty \psi(x)\,\bar{F}_i(x)\,dx$$

where we denoted $\psi(x) = \Phi'(x)$ and $\bar{F}_i(x) = 1 - F_i(x)$. Since $\Phi(\cdot)$ is bounded then $-\Phi(x)\bar{F}_i(x)\Big|_0^\infty = c$ is finite. Moreover, since $\Phi(\cdot)$ is convex then $\psi(x)$ is increasing and thus all the conditions of Lemma 6.4 [3, p. 112] are satisfied and

$$\int_0^\infty \psi(x)\bar{F}_1(x)\,dx \leqslant \int_0^\infty \psi(x)\bar{F}_2(x)\,dx.$$

The rest of the proof follows easily. □

LEMMA 2: Let $Z \sim F_Z(x) = 1 - e^{-\lambda x}$ and $X \sim F_X(x) = 1 - e^{-\mu x}$ with $E[Z] > E[X]$. Then there exists a unique point, say x_0, where the density of X crosses the density of Z and the crossing is from above.

PROOF: $E[Z] > E[X]$ implies $\mu > \lambda$. Solving $\mu e^{-\mu x_0} = \lambda e^{-\lambda x_0}$ yields $x_0 = (\mu - \lambda)^{-1} \ln(\mu/\lambda) > 0$. □

Now we are ready to prove the main theorem.

THEOREM 2: Let $X_i \sim F_i$ IFR and $Y_i \sim G_i$ with G_i being exponential and $E[X_i] = E[Y_i]$ for $i = 1, \ldots, n$. Let $\Phi(\cdot)$ be decreasing convex bounded function on $[0,\infty)$. Then

$$E[\Phi(\min_i X_i)] \leqslant E[\Phi(\min_i Y_i)]. \tag{2.2}$$

PROOF: Denote by $X = \min_i(X_i)$. Let $X \sim F$, then F is IFR and there exists an exponential r.v. Z with cumulative distribution function (c.d.f.) G such that $E[X] = E[Z]$. Furthermore, $F \lesssim_* G$ [2, p. 107].

By the Lemma 1 we have $E[\Phi(X)] \leqslant E[\Phi(Z)]$, and by Theorem 1 we have $E[Z] \geqslant E[Y]$, where $Y = \min(Y_1, \ldots, Y_n)$. Note that Y is also exponential. If $E[Z] = E[Y]$ we are done.

If $E[Z] > E[Y]$, then there exists a unique point x_0 where the density of Y crosses the density of Z and the crossing is from above. Since

$$E[\Phi(Z)] = \int_0^\infty \Phi(x)\lambda e^{-\lambda x}\,dx$$

and

$$E[\Phi(Y)] = \int_0^\infty \Phi(x)\mu e^{-\mu x}\,dx$$

we get

$$E[\Phi(Z)] - E[\Phi(Y)] = \int_0^\infty \Phi(x)\,(\lambda e^{-\lambda x} - \mu e^{-\mu x})\,dx \tag{2.3}$$

which can be written as

$$\int_0^\infty [\Phi(x) - \Phi(x_0)]\,(\lambda e^{-\lambda x} - \mu e^{-\mu x})\,dx + \Phi(x_0)\int_0^\infty (\lambda e^{-\lambda x} - \mu e^{-\mu x})\,dx. \tag{2.4}$$

The second term of (2.4) is, of course, equal to zero and the first term written as,

$$\int_0^{x_0} = [\Phi(x) - \Phi(x_0)]\,(\lambda e^{-\lambda x} - \mu e^{-\mu x})\,dx + \int_{x_0}^\infty [\Phi(x) - \Phi(x_0)]\,(\lambda e^{-\lambda x} - \mu e^{-\mu x})\,dx \leqslant 0,$$

is nonpositive, since for $x \in [0,x_0]$, $\Phi(x) - \Phi(x_0) \geqslant 0$ ($\Phi(\cdot)$ is decreasing) and $\lambda e^{-\lambda x} - \mu e^{-\mu x} \leqslant 0$ (by Lemma 2). Similarly, for $x \in (x_0, \infty)$, $\Phi(x) - \Phi(x_0) \leqslant 0$ and $\lambda e^{-\lambda x} - \mu e^{-\mu x} \geqslant 0$. □

When the function $\Phi(\cdot)$ is exponential we obtain the following corollary:

COROLLARY 1: Let $X_i \sim F_i$ IFR and $Y_i \sim G_i$ exponential, for $i = 1, \ldots, n$, $E[X_i] = E[Y_i] = 1/\mu_i$. Then for $\Phi(x) = e^{-\alpha x}$ we get

$$E[\exp[-\alpha(\min_i X_i)]] \leqslant \frac{\sum_i \mu_i}{\alpha + \sum_i \mu_i}.$$

Finally, let us remark that Theorem 2 will hold even when F_i is Increasing Failure Rate Average (IFRA), since this condition together with G_i being exponential still implies that $F \lesssim_* G$ [3, p. 107].

3. EXAMPLES

Two simple examples will illustrate the theorem.

EXAMPLE 1: Consider an n-unit series system, where the unit lifetime $X_i \sim F_i$. The F_i is IFR and $E[X_i] = 1/\mu_i$. When the system fails it is replaced by a new system. The cost of the new system at the time $t = 0$ is C. The expected discounted cost of replacement at the time of system failure is $CE[e^{-\alpha X}]$, where $X = \min(X_1, \ldots, X_n)$ denotes the life time of the system.

It follows from Corollary 1 that the upper bound for the discounted replacement cost is

$$C \frac{\sum_i \mu_i}{\alpha + \sum_i \mu_i}.$$

Suppose now that F_i is Gamma (λ_i, a_i), $a_i > 1$, for all $i = 1, \ldots, n$, i.e., the probability distribution function (p.d.f.) of X_i is

$$f_i(x) = \frac{\lambda_i(\lambda_i x)^{a_i - 1}}{\Gamma(a_i)} e^{-\lambda_i x}, \quad x > 0, \ \lambda_i > 0.$$

In this case F_i is IFR, $E[X_i] = a_i/\lambda_i$ and thus $\mu_i = \lambda_i/a_i$. Moreover,

$$CE[e^{-\alpha X}] \leqslant C/(\alpha/\Lambda + 1),$$

where $\Lambda = \sum_i (\lambda_i/a_i)$.

EXAMPLE 2: Let $X = \min(X_1, \ldots, X_n)$. All X_i's are independent, have IFR distribution F_i and $E[X_i] = 1/\mu_i$. Assume that $\Phi(x) = (\alpha x + \beta)^{-1}$, $\alpha > 0$, $\beta > 0$. Clearly, $\Phi(x)$ is decreasing convex and bounded on $[0,\infty)$. It follows from Theorem 2 that in this case

$$E[\Phi(X)] = E[(\alpha X + \beta)^{-1}] \leqslant \int_0^\infty \frac{1}{\alpha x + \beta} \Lambda e^{-\Lambda x} \, dx$$

$$= \frac{\Lambda}{\alpha} e^d \int_d^\infty \frac{1}{t} e^{-t} = \frac{\Lambda}{\alpha} e^d E_1(d),$$

where $\Lambda = \sum_i \mu_i$, $d = \Lambda\beta/\alpha$ and $E_1(d)$ denotes the Exponential Integral. The numerical values of $E_1(d)$ are tabulated in [1].

REFERENCES

[1] Abramowitz, M. and I.A. Stegun, *Handbook of Mathematical Functions,* (Dover Publishing, Inc., New York, 1965).

[2] Agnihothri, S., U.S. Karmarkar, and P. Kubat, "Stochastic Allocation Rules," Operations Research, *30*, 545-555 (1982).

[3] Barlow, E.R. and F. Proschan, *Statistical Theory of Reliability and Life Testing: Probability Models* (Holt, Rinehart and Winston, Inc., New York, 1975).

COMPOUND AVAILABILITY MEASURES

Laurence A. Baxter

Department of Applied Mathematics and Statistics
State University of New York at Stony Brook
Stony Brook, New York

ABSTRACT

An availability measure is the probability that a two-state system modeled by an alternating renewal process is available at one or more points or intervals. The concept of availability measures is extended to formulae for the joint prediction of availability and numbers of breakdowns (or repairs) of the system during a fixed interval.

1. INTRODUCTION

Consider a two-state system, i.e., a machine subject to stochastic failure and repair, and suppose that the breakdown/reactivation cycle of the machine can be modeled by means of an alternating renewal process (e.g., Cox [4], Chapter 7). Under this assumption, we can derive a variety of useful formulae for predicting the reliability of the system. In particular, we can determine the distributions of the $N(t)$, the numbers of failures and repairs in a fixed interval [1], and we can obtain *availability measures*, probabilities of the form $p\{I(t) = 1 \ \forall \ t \in T\}$ where $I(t) = 1(0)$ if the system is operating (failed) at t and where T is an index set comprising a (finite) series of points or intervals [2]. The dependence between the $N(t)$ and the $I(t)$ precludes the simultaneous prediction of numbers of failures (or repairs) and availability using existing theory. In this paper, we present a series of formulae which generalize the availability measures, enabling us jointly to predict numbers of breakdowns (or repairs) and availability. We call these formulae *compound availability measures.*

Availability measures are of use in assessing the likelihood that a repairable machine modeled by an alternating renewal process is available at specified times. If the probability falls below a certain threshold, arrangements for securing an adequate back-up can be made. The distributions of the $N(t)$, on the other hand, are of use in deciding on the allocation of sufficient repair facilities for a fixed time interval. The formulae presented in this paper enable us to consider these probabilities simultaneously.

In Section 2 we examine the simplest case and consider probabilities of the form $p\{I(t) = 1, N(t) = j\}$ and $p\{I(u) = 1 \ \forall \ u \in [t, t + x], N(t) = j\}$. In Section 3 we consider probabilities of the form $p\{I(t + x) = 1, N(t) = j\}$, and in Section 4 we consider probabilities of the form $p\{I(t) = 1, N(t + x) - N(t) = j\}$. Some numerical examples for the Weibull/gamma process are given in Section 5. It is first necessary to introduce some notation and briefly to review some relevant results.

Let F and G denote the distribution functions of the failure times and repair times, respectively, and let their Stieltjes convolution be denoted

$$K(t) = F * G(t) = \int_0^t F(t-u)\, dG(u).$$

Further, the n-fold recursive Stieltjes convolution of K is written $K^{(n)}$ for $n = 1, 2, 3, \ldots$, and we define $K^{(0)}(t) = 1(0)$ if $t \geqslant (<) 0$. Let f, g and k be the densities corresponding to F, G and K, respectively. The renewal counting functions of the numbers of failures and repairs in $(0,t]$ are denoted $N_1(t)$ and $N_2(t)$ ($N_2(t)$ and $N_3(t)$), respectively, assuming that there is a repair (failure) at time 0. The corresponding renewal functions and renewal densities are $H_i(t) = E\{N_i(t)\}$ and $h_i(t) = dH_i(t)/dt$ $(i = 1, 2, 3)$.

The distributions of the $N_i(t)$ are as follows:

$$p\{N_1(t) = n\} = \begin{cases} 1 - F(t) & n = 0 \\ F * K^{(n-1)}(t) - F * K^{(n)}(t) & n \geqslant 1 \end{cases} \tag{1.1}$$

$$p\{N_2(t) = n\} = K^{(n)}(t) - K^{(n+1)}(t) \qquad n \geqslant 0 \tag{1.2}$$

$$p\{N_3(t) = n\} = \begin{cases} 1 - G(t) & n = 0 \\ G * K^{(n-1)}(t) - G * K^{(n)}(t) & n \geqslant 1 \end{cases} \tag{1.3}$$

(see Barlow and Hunter [1]). The point availability of the two-state system is defined as $A_k(t) = p\{I_k(t) = 1\}$ where $k = 0(1)$ if there is a failure (repair) at time 0. The interval availability is defined as $R_k(x,t) = p\{I_k(u) = 1 \ \forall u \in [t, t+x]\}$. It can be shown that

$$A_1(t) = \bar{F}(t) + \bar{F} * H_2(t) \tag{1.4}$$

$$A_0(t) = G(t) - \bar{G} * H_2(t) \tag{1.5}$$

$$R_1(x,t) = \bar{F}(t+x) + \int_0^t h_2(u)\, \bar{F}(t+x-u)\, du \tag{1.6}$$

$$R_0(x,t) = \int_0^t h_3(u) \bar{F}(t+x-u)\, du \tag{1.7}$$

where

$$\bar{F}(t) = 1 - F(t) \text{ [2].}$$

2. COMPOUND MEASURES OF $I(t)$ AND $N(t)$

The expressions for the simplest compound availability measures, i.e., $p\{I(t) = 1, N(t) = j\}$, are of particular interest as surprisingly simple formulae can be obtained for the covariances of the $I_k(t)$ and the $N_i(t)$.

$$p\{I_1(t) = 1,\ N_1(t) = j\} = \bar{F} * K^{(j)}(t) \qquad j \geqslant 0 \tag{2.1}$$

$$p\{I_1(t) = 1,\ N_2(t) = j\} = \bar{F} * K^{(j)}(t) \qquad j \geqslant 0 \tag{2.2}$$

$$p\{I_0(t) = 1,\ N_2(t) = j\} = \bar{F} * G * K^{(j)}(t) \qquad j \geqslant 0 \tag{2.3}$$

$$p\{I_0(t) = 1,\ N_3(t) = j\} = \begin{cases} 0 & j = 0 \\ \bar{F} * G * K^{(j-1)}(t) & j \geqslant 1. \end{cases} \tag{2.4}$$

Observe that $p\{I_1(t) = 1,\ N_1(t) = j\} = p\{I_1(t) = 1,\ N_2(t) = j\}$ for all j. This is to be expected in view of the identity $I_1(t) = N_2(t) - N_1(t) + 1$.

The corresponding formulae for $P_k^i(x,t,j) = p\{I_k(u) = 1\ \forall\ u \in [t,t+x],\ N_i(t) = j\}$ are as follows:

$$P_1^1(x,t,j) = \begin{cases} \bar{F}(t+x) & j = 0 \\ \int_0^t k^{(j)}(u)\,\bar{F}(t+x-u)\,du & j \geqslant 1 \end{cases} \tag{2.5}$$

$$P_1^2(x,t,j) = \begin{cases} \bar{F}(t+x) & j = 0 \\ \int_0^t k^{(j)}(u)\,\bar{F}(t+x-u)\,du & j \geqslant 1 \end{cases} \tag{2.6}$$

$$P_0^2(x,t,j) = \int_0^t b_j(u)\bar{F}(t+x-u)\,du \qquad j \geqslant 0 \tag{2.7}$$

$$P_0^3(x,t,j) = \begin{cases} 0 & j = 0 \\ \int_0^t b_{j-1}(u)\,\bar{F}(t+x-u)\,du & j \geqslant 1 \end{cases} \tag{2.8}$$

where $k^{(j)}(t) = dK^{(j)}(t)/dt$ and

$$b_j(t) = \begin{cases} \int_0^t g(t-u)\,k^{(j)}(u)\,du & j \geqslant 1 \\ g(t) & j = 0. \end{cases}$$

It should be noted that the above formulae, and many of those of Sections 3 and 4, do not need to treat the case $j = 0$ separately, but this has been done in the interests of clarity.

The following covariances are readily derived from (2.1)-(2.4) above:

$$\text{cov}\{I_1(t),N_1(t)\} = A_1 * H_2(t) = A_1(t)H_1(t) \tag{2.9}$$

$$\text{cov}\{I_1(t),N_2(t)\} = A_1 * H_2(t) - A_1(t)H_2(t) \tag{2.10}$$

$$\text{cov}\{I_0(t),N_2(t)\} = A_0 * H_2(t) - A_0(t)H_2(t) \tag{2.11}$$

$$\text{cov}\{I_0(t),N_3(t)\} = A_0 * H_2(t) - A_0(t)\,[H_3(t) - 1]. \tag{2.12}$$

These may be obtained directly, in which case the identity

$$\sum_{j=1}^{\infty} jK^{(j)}(t) = H_2 * H_2(t) + H_2(t)$$

will be found helpful, or by exploiting the binary nature of the indicator variable. Thus, for example, expression (2.10) follows from the identity

$$\frac{\text{cov}\{I_1(t),N_2(t)\}}{A_1(t)[1 - A_1(t)]} = E\{N_2(t)|I_1(t) = 1\} - E\{N_2(t)|I_1(t) = 0\}$$

and the other covariances may be derived similarly.

Observe that the expectation of the product of the indicator variable and $N_2(t)$ is the Stieltjes convolution of their expectations.

3. COMPOUND MEASURES OF $I(t+x)$ AND $N(t)$

We now present formulae for probabilities of the form $p\{I_k(t+x)=1,\ N_i(t)=j\}$.

(3.1)
$$\begin{aligned} p\{I_1(t+x)=1,\ N_1(t)=j\} &= \bar{F}(t+x)+\int_0^x f(t+y)\,A_0(x-y)\,dy && j=0\\ &= \int_0^t k^{(j)}(u)\,\bar{F}(t+x-u)\,du\\ &\quad+\int_0^x\int_0^t k^{(j)}(u)\,f(t+y-u)\,A_0(x-y)\,dudy\\ &\quad+\int_0^x\int_0^t a_{j-1}(u)\,g(t+y-u)\,A_1(x-y)\,dudy && j\geqslant 1 \end{aligned}$$
$$\text{where } a_j(t)=\begin{cases}\int_0^t f(t-u)k^{(j)}(u)\,du & j\geqslant 1\\ f(t) & j=0\end{cases}$$

(3.2)
$$\begin{aligned} p\{I_1(t+x)=1,\ N_2(t)=j\} &= \bar{F}(t+x)+\int_0^x f(t+y)\,A_0(x-y)\,dy\\ &\quad+\int_0^x\int_0^t f(u)g(t+y-u)A_1(x-y)\,dudy && j=0\\ &= \int_0^t k^{(j)}(u)\bar{F}(t+x-u)\,du\\ &\quad+\int_0^x\int_0^t k^{(j)}(u)\,f(t+y-u)\,A_0(x-y)\,dudy\\ &\quad+\int_0^x\int_0^t a_j(u)g(t+y-u)\,A_1(x-y)\,dudy && j\geqslant 1 \end{aligned}$$

(3.3)
$$\begin{aligned} p\{I_0(t+x)=1,\ N_2(t)=j\} &= \int_0^t b_j(u)\,\bar{F}(t+x-u)\,dudy\\ &\quad+\int_0^x\int_0^t b_j(u)f(t+y-u)\,A_0(x-y)\,dudy\\ &\quad+\int_0^x\int_0^t k^{(j)}(u)\,g(t+y-u)\,A_1(x-y)\,dudy && j\geqslant 0 \end{aligned}$$

(3.4)
$$\begin{aligned} p\{I_0(t+x)=1,\ N_3(t)=j\} &= \int_0^x g(t+y)\,A_1(x-y)\,dy && j=0\\ &= \int_0^t b_{j-1}(u)\,\bar{F}(t+x-u)\,du\\ &\quad+\int_0^x\int_0^t b_{j-1}(u)\,f(t+y-u)\,A_0(x-y)\,dudy\\ &\quad+\int_0^x\int_0^t k^{(j)}(u)\,g(t+y-u)\,A_1(x-y)\,dudy && j\geqslant 1. \end{aligned}$$

Note that, as would be expected, expressions (3.1)-(3.4) degenerate to (2.1)-(2.4), respectively, at $x=0$ and that application of the key renewal theorem shows that

$$\lim_{x\to\infty} p\{I_k(t+x)=1,\ N_i(t)=j\}=\frac{\mu_1}{\mu_1+\mu_2}\,p\{N_i(t)=j\}$$

where μ_1 and μ_2 are the means of F and G, respectively.

The corresponding formulae for $p\{I_k(u) = 1 \ \forall \ u \in [t + x, t + x + w], \ N_i(t) = j\}$ are obtained from (3.1)-(3.4) above on replacing $\bar{F}(t + x)$ by $\bar{F}(t + x + w)$ and $A_k(x - y)$ by $R_k(w, x - y)$ as appropriate. Further extensions of these formulae to more general compound availability measures of the form

$$p\{I_k(t_1) = I_k(t_2) = \cdots = I_k(t_n) = 1, \ N_i(t) = j\} \text{ for } t < t_1 < t_2 < \cdots < t_n$$

are similarly obtained.

4. COMPOUND MEASURES OF $I(t)$ AND $N(t, t + x)$

In this section we consider a different type of compound availability measure, that of the form $p\{I_k(t) = 1, \ N_i(t, t + x) = j\}$ where $N_i(t, t + x) = N_i(t + x) - N_i(t)$ is the number of renewals in $[t, t + x]$.

(4.1)
$$\begin{aligned} p\{I_1(t) &= 1, \ N_1(t, t + x) = j\} \\ &= R_1(x, t) && j = 0 \\ &= \int_0^x f(t + u)\, p\{N_2(x - u) = j - 1\}\, du \\ &\quad + \int_0^x \int_0^t h_2(s)\, f(t - s + u)\, p\{N_2(x - u) = j - 1\}\, ds du && j \geq 1 \end{aligned}$$

(4.2)
$$\begin{aligned} p\{I_1(t) &= 1, \ N_2(t, t + x) = j\} \\ &= R_1(x, t) + \int_0^x f(t + u)\, \bar{G}(x - u)\, du \\ &\quad + \int_0^x \int_0^t h_2(s)\, f(t - s + u)\, \bar{G}(x - u)\, ds du && j = 0 \\ &= \int_0^x f(t + u)\, p\{N_3(x - u) = j\}\, du \\ &\quad + \int_0^x \int_0^t h_2(s)\, f(t - s + u)\, p\{N_3(x - u) = j\}\, ds du && j \geq 1 \end{aligned}$$

(4.3)
$$\begin{aligned} p\{I_0(t) &= 1, \ N_2(t, t + x) = j\} \\ &= R_0(x, t) && j = 0 \\ &= \int_0^x \int_0^t h_3(s)\, f(t - s + u)\, p\{N_2(x - u) = j - 1)\}\, ds du && j \geq 1 \end{aligned}$$

(4.4)
$$\begin{aligned} p\{I_0(t) &= 1, \ N_3(t, t + x) = j\} \\ &= R_0(x, t) + \int_0^x \int_0^t h_3(s)\, f(t - s + u)\, \bar{G}(x - u)\, ds du && j = 0 \\ &= \int_0^x \int_0^t h_3(s)\, f(t - s + u)\, p\{N_3(x - u) = j\}\, ds du && j \geq 1. \end{aligned}$$

Observe that, as would be expected, (4.1) and (4.2) degenerate to (1.1) and (1.2), respectively, at $t = 0$ whereas both (4.3) and (4.4) vanish. Note further that on applying the key renewal theorem to the above formulae we see that

$$\lim_{t \to \infty} p\{I_1(t) = 1, \ N_1(t, t + x) = j\} = \lim_{t \to \infty} p\{I_0(t) = 1, \ N_2(t, t + x) = j\}$$

$$= \begin{cases} A\, \bar{\Psi}(x) & j = 0 \\ A \int_0^x \psi(u)\, p\{N_2(x - u) = j - 1\}\, du & j \geq 1 \end{cases}$$

and that

$$\lim_{t\to\infty} p\{I_1(t) = 1,\ N_2(t,t+x) = j\} = \lim_{t\to\infty} p\{I_0(t) = 1,\ N_3(t,t+x) = j\}$$
$$= A\int_0^x \psi(u)\, p\{N_3(x-u) = j\}\, du \qquad j \geqslant 0$$

where $A = \mu_1/(\mu_1+\mu_2)$, $\psi(t) = \bar{F}(t)/\mu_1$ and $\Psi(t) = \int_0^t \psi(u)\, du$. Thus, we see that the asymptotic probabilities are equal to the corresponding probabilities for the equilibrium alternating renewal process.

Generalizations to formulae of the form $p\{I_k(u) = 1\ \forall\ u \in [t,t+x],\ N_i(t+w,t+w+x) = j\}$ are readily obtained by obvious modifications to (4.1)-(4.4).

5. APPLICATIONS

One important industrial application of the alternating renewal process is to model the sequences of failures and repairs of electricity generating plant, such as boilers and turbo-generators. The author's experience in analyzing such data shows that, typically, failure times are identically distributed, as are repair times, and that these random variables do not appear to be dependent. Knowledge of various availability measures is a means of assessing the likelihood that, for example, peak demand will be met over one or more periods. The distributions of the renewal counting functions, on the other hand, give some indication of the number of breakdowns likely in a fixed interval. These may be used, for example, in deciding the levels of stocks of those spare parts which are most commonly used in repair work. The compound availability measures enable us to make the two sets of predictions simultaneously.

As an example, we examine the alternating renewal process with Weibull failure times and gamma repair times, the scale parameter being unity in each case, i.e.,

$$f(t) = \alpha t^{\alpha-1} e^{-t^\alpha},\quad g(t) = t^{\eta-1} e^{-t}/\Gamma(\eta).$$

Values of

$$p\{I_1(t) = 1,\ N_1(t) = j\} = p\{I_1(t) = 1,\ N_2(t) = j\} = \bar{F} * K^{(j)}(t)$$

were evaluated for $0 \leqslant t \leqslant 10$ and $j = 1,2,3,\ldots$ for a variety of values of α and η. The numerical integration was achieved by means of the Cléroux-McConalogue algorithm [3] for recursively-defined Stieltjes convolutions using FORTRAN subroutines developed by McConalogue [5]. (See McConalogue [6] for further details.) The values of $\bar{F} * K^{(j)}(t)$ obtained are illustrated in Figures 1-4 for the following combinations of shape parameters:

Figure	α	η
1	1	1
2	2.5	2.5
3	4.5	2.5
4	5	1.25.

ACKNOWLEDGMENT

I should like to thank the referee for a number of suggestions which improved the presentation.

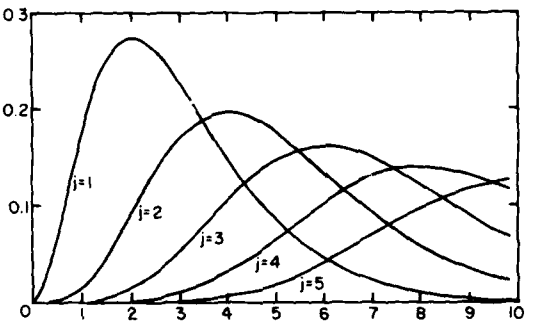


FIGURE 1. Curves of $\bar{F} * K^{(j)}(t)$ for $\eta = 1$, $\alpha = 1$.

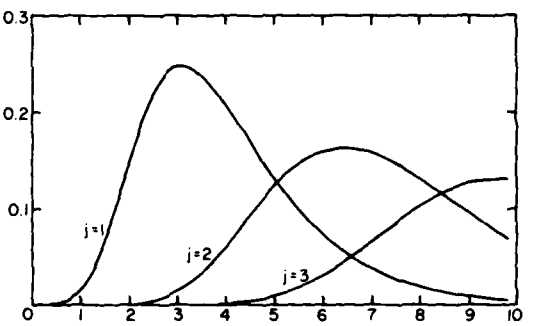


FIGURE 2. Curves of $\bar{F} * K^{(j)}(t)$ for $\eta = 2.5$, $\alpha = 2.5$.

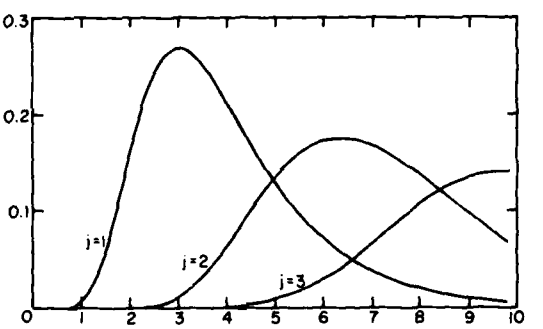


FIGURE 3. Curves of $\bar{F} * K^{(j)}(t)$ for $\eta = 2.5$, $\alpha = 4.5$.

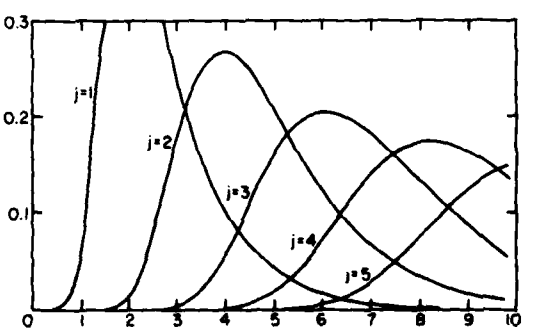


FIGURE 4. Curves of $\bar{F} * K^{(j)}(t)$ for $\eta = 1.25$, $\alpha = 5$.

REFERENCES

[1] Barlow, R.E. and L.C. Hunter, "Reliability Analysis of a One-Unit System," Operations Research 9, 200-208 (1961).
[2] Baxter, L.A., "Availability Measures for a Two-State System," Journal of Applied Probability *18*, 227-235 (1981).
[3] Cléroux, R. and D.J. McConalogue, "A Numerical Algorithm for Recursively-Defined Convolution Integrals Involving Distribution Functions," Management Science, *22*, 1138-1146 (1976).
[4] Cox, D.R., "Renewal Theory" (Methuen, London, 1962).
[5] McConalogue, D.J., "Convolution Integrals Involving Probability Distribution Functions" (Algorithm 102), Computer Journal, *21*, 270-272 (1978).
[6] McConalogue, D.J., "Numerical Treatment of Convolution Integrals Involving Distributions with Densities Having Singularities at the Origin," Communications in Statistics, B *10*, 265-280 (1981).

A COMPARISON OF SEVERAL ESTIMATES OF AVAILABILITY

M. Mazumdar

University of Pittsburgh
Pittsburgh, Pennsylvania

ABSTRACT

We compare several competing estimates of the availability of a system which alternates between two states, "up" and "down," in accordance with an alternating renewal process. Both interval and point estimators are compared under several special but representative situations. The comparison reaffirms the validity and robustness of the log-logistic jackknifed estimates. However, when the point estimates are compared from the intrinsic criterion of probability of concentration, the uniformly minimum variance estimate obtained for the Markov model performs very well.

1. INTRODUCTION

In a system which alternates between two states, "up" or "down," in accordance with an alternating renewal process, the long-term point availability (often simply abbreviated as *availability*) measures the probability that the process is up at a given instant of time far enough in the future. This index measures the long-run expected fraction of time that the system operates satisfactorily. It has been shown in reliability textbooks, (see, for example, Barlow and Proschan [1]), that under not too restrictive conditions, this index equals the ratio of the mean uptime to the sum of the mean uptime and mean downtime. Such two-state models are considered frequently in the literature in problems related to design and maintenance of computer systems, communication systems, power plants, etc. System productivity is directly related to equipment availability, and, therefore, much attention is currently being placed in the industry on the assessment and optimization of plant availability. Any availability improvement program has to begin with the estimation of current availability. In this paper, we examine several contending estimators of availability, and study some of their properties.

In a recent paper, Gaver and Chu [2] have studied the behavior of the jackknife method for estimating availability under several different probability models for the underlying distributions. They have pointed out that frequently in availability studies, it will be difficult to obtain clear and unequivocal probability specifications. Their results obtained by Monte Carlo simulation show that the jackknife method is reasonably robust for providing interval estimates and valid under several different special but representative probability models. In this paper, we carry out an investigation similar in spirit to Gaver and Chu [2]. Our point of departure from this paper is that we consider an additional estimate of availability (Mazumdar [3]), and we also consider the intrinsic properties of the estimates from the point of view of point estimation. Our main result can be stated as follows: from the point of view of providing interval estimates, the jackknife estimate of Gaver and Chu appears to be suitably valid and robust, and it

performs as well or better than the other contending estimates. However, from the point of view of point estimation, when the intrinsic property of probability of concentration (Rao [5]) is considered, the uniformly minimum variance unbiased (UMVU) estimator seems to perform as well or better.

2. THE MODEL AND THE ESTIMATES OF AVAILABILITY

We assume that the times spent by the process in the "up" and "down" state are mutually independent, and we have at hand a sequence of $2n$ observations giving the successive up and downtimes, $U_1, D_1, U_2, D_2, \ldots, U_n, D_n$ of the system. Except for one case, we shall make the usual assumption that the uptimes U_i have the exponential distribution. The downtimes D_i are identically distributed, and in Section 3, we shall assume that their common distribution belongs to the exponential, gamma, lognormal or the Weibull family. The parameter of interest, availability, is given by

$$A = \frac{E[U]}{E[U] + E[D]} \tag{1}$$

where $E[U]$ and $E[D]$ are, respectively, the expectations of U_i and D_i.

We consider the following estimators of availability:

(a) the maximum likelihood estimate:

$$\hat{A}_{\text{mle}} = \frac{\bar{U}}{\bar{U} + \bar{D}} \tag{2}$$

where $\bar{U} = \sum_{i=1}^{n} U_i/n$ and $\bar{D} = \sum_{i=1}^{n} D_i/n$.

(b) the jackknifed maximum likelihood estimate:

$$\hat{A}_{\text{jk,mle}} = \frac{\bar{V}}{\bar{V} + \bar{E}} \tag{3}$$

where

$$\bar{V} = \sum_{i=1}^{n} V_i/n, \quad \bar{E} = \sum_{i=1}^{n} E_i/n, \tag{3a}$$

$$V_i = n\bar{U} - (n-1)\bar{U}_{-i}, \tag{3b}$$

$$\bar{U}_{-i} = \frac{1}{(n-1)} \sum_{j \neq i} U_j, \tag{3c}$$

$$E_i = n\bar{D} - (n-1)\bar{D}_{-i}, \tag{3d}$$

and

$$\bar{D}_{-i} = \frac{1}{(n-1)} \sum_{j \neq i} D_j. \tag{3e}$$

(c) the log-logistic jackknifed estimate (Gaver and Chu [2])

$$A_{\text{jk,ll}} = \frac{e^{\bar{Z}}}{1 + e^{\bar{Z}}} \tag{4}$$

where

$$\bar{Z} = \frac{1}{n}\sum_{j=1}^{n} Z_j \tag{4a}$$

$$Z_j = nZ - (n-1)Z_{-j} \tag{4b}$$

$$Z_{-j} = \ln\left[\sum_{i\neq j} U_i\right] - \ln\left[\sum_{i\neq j} D_i\right] \tag{4c}$$

$$Z = \ln(\bar{U}) - \ln(\bar{D}). \tag{4d}$$

(d) the uniformly minimum variance unbiased estimate (Mazumdar, [3]): This estimator was derived for the case when both U_i and D_i are exponentially distributed with unknown parameters λ and μ, respectively. Denote $\bar{U}/\bar{D}$ by S. Then the estimator is given by

$$\hat{A}_{\text{umvu}}(n) = \begin{cases} 1 - (n-1)\displaystyle\sum_{j=0}^{n-1}\binom{n-1}{j}\frac{S^j(1-S)^{n-1-j}}{n-1+j} & \text{if } S \leqslant 1 \\ 1 - (n-1)\displaystyle\sum_{j=0}^{n-2}\binom{n-2}{j}\frac{S^{-j}(1-S^{-1})^{n-2-j}}{n+j} & \text{if } S > 1. \end{cases} \tag{5a}$$

The above reduces to:

$$\hat{A}_{\text{umvu}}(n) = \frac{\binom{n-1}{1}}{\binom{n}{1}}S - \frac{\binom{n-1}{2}}{\binom{n+1}{2}}S^2 + \frac{\binom{n-1}{3}}{\binom{n+2}{3}}S^3 - \ldots + (-1)^n\frac{\binom{n-1}{n-1}}{\binom{2n-2}{n-1}}S^{n-1} \quad \text{if } S \leqslant 1$$

and

$$\hat{A}_{\text{umvu}}(n) = 1 - \frac{\binom{n-1}{1}}{\binom{n}{1}}S^{-1} + \frac{\binom{n-1}{2}}{\binom{n+1}{2}}S^{-2} - \ldots + (-1)^{n-1}\frac{\binom{n-1}{n-1}}{\binom{2n-2}{n-1}}S^{-(n-1)} \quad \text{if } S > 1. \tag{5b}$$

Some examples of $\hat{A}_{\text{umvu}}$ are given below. When $n = 2$,

$$\hat{A}_{\text{umvu}}(2) = \begin{cases} 1/2S & \text{if } S \leqslant 1 \\ 1 - 1/2S^{-1} & \text{if } S > 1 \end{cases} \tag{5c}$$

When $n = 4$,

$$\hat{A}_{\text{umvu}}(4) = \begin{cases} \frac{3}{4}S - \frac{3}{10}S^2 + \frac{1}{20}S^3 & \text{if } S \leqslant 1 \\ 1 - \frac{3}{4}S^{-1} + \frac{3}{10}S^{-2} - \frac{1}{20}S^{-3} & \text{if } S > 1 \end{cases} \tag{5d}$$

(3) the jackknifed uniformly variance unbiased estimate.

$$\hat{A}_{jk,\text{umvu}}(n) = \frac{1}{n}\sum_{i=1}^{n} W_i(n) \tag{6a}$$

where

$$W_i(n) = n\hat{A}_{\text{umvu}}(n) - (n-1)\,\hat{A}_{-\text{umvu},i}(n-1) \tag{6b}$$

$\hat{A}_{\text{umvu}}(n)$ is given by (5b),

$$\hat{A}_{-\text{umvu},i}(n-1) = \frac{\binom{n-2}{1}}{\binom{n-1}{1}} S_{-i} - \frac{\binom{n-2}{2}}{\binom{n}{2}} S_{-i}^2 + \frac{\binom{n-2}{3}}{\binom{n+1}{3}} S_{-i}^3 - \ldots$$

$$+ (-1)^{n-1} \frac{\binom{n-2}{n-2}}{\binom{2n-4}{n-2}} S^{n-2} \quad \text{if } S_{-i} \leqslant 1,$$

$$\hat{A}_{-\text{umvu},i}(n-1) = 1 - \frac{\binom{n-2}{1}}{\binom{n-1}{1}} S_{-i}^{-1} + \frac{\binom{n-2}{2}}{\binom{n}{2}} S_{-i}^2 - \ldots + (-1)^{n-2} \frac{\binom{n-2}{n-2}}{\binom{2n+4}{n-2}} S_{-i}^{-(n-2)}$$

$$\text{if } S_{-i} > 1 \tag{6c}$$

and

$$S_{-i} = \frac{\bar{U}_{-i}}{\bar{D}_{-i}} \tag{6d}$$

In the terminology related to jackknife estimates, the quantities V_i, E_i, Z_i, W_i are referred to as pseudovalues. Let $\hat{\theta}$ be an estimator of the parameter θ based on a sample of size n. Let $\tilde{\theta}_i$ refer to the set of pseudovalues in this situation, $i = 1, 2, \ldots, n$ and let

$$\tilde{\theta} = \frac{1}{n}\sum_{i=1}^{n} \tilde{\theta}_i. \tag{7}$$

It has been stated (Miller [4]), that under certain general conditions, the statistic

$$\frac{\sqrt{n}\,(\tilde{\theta} - \theta)}{\left\{\frac{1}{n-1}\sum_{i=1}^{n} (\tilde{\theta}_i - \theta)^2\right\}^{1/2}} \tag{8}$$

has an approximate t distribution with $(n-1)$ degrees of freedom. Thus, the approximate two-sided 100 $(1-\alpha)$% confidence interval for θ will be given by (L_α, U_α) where

$$U_\alpha = \tilde{\theta} + t_{1-\frac{\alpha}{2}} \left\{\frac{1}{n-1}\sum_{i=1}^{n} (\tilde{\theta}_i - \theta)^2\right\}^{1/2} \Big/ n^{1/2} \tag{9}$$

$$L_\alpha = \tilde{\theta} - t_{1-\frac{\alpha}{2}} \left\{ \frac{1}{n-1} \sum_{i=1}^{n} (\tilde{\theta}_i - \theta)^2 \right\}^{1/2} \Big/ n^{1/2}$$

where $t_{1-\frac{\alpha}{2}}$ refers to the $\left(1-\frac{\alpha}{2}\right)$ 100% percentile of Student's t with $n-1$ degrees of freedom.

In a recent paper, Rao [5] has commented on the wisdom of using minimum mean square error as the only criterion for obtaining optimum point estimates. He has suggested that because of mathematical convenience and intuitive considerations, the minimum mean squared error could be used as a postulate to derive estimators, but he has recommended that their acceptability should be judged on more intrinsic criteria such as the probability of concentration (PC), where this quantity measures

(10) $$\text{PC} = \Pr\{|\text{Estimate} - \text{True Value}| < a\}$$

for different values of a.

For the purpose of comparing the performance characteristics of the estimates (a)-(e), we choose the following three criteria: (i) estimate probability of coverage of the true availability using the formula (9) corresponding to a nominal value α; ii) the estimated mean and variance of the confidence interval length; and iii) the estimated probability of concentration. (The mean squared error was used in [3] to compare the maximum likelihood and the umvu estimate, and we do not repeat the use of this criterion here.) These estimates are obtained by performing Monte Carlo simulation experiments where 1,000 synthetic system realizations were observed through a cycle of n consecutive up and downtimes.

Following Gaver and Chu [2], we assume the following distribution forms for the up and down times for the purpose of the simulation study. Although these distribution forms represent special cases, they are representative of many different practical situations.

(A) U is exponentially distributed, $E[U] = \lambda^{-1}$; D is exponentially distributed, $E[D] = \mu^{-1}$; $\{U_i\}$, $\{D_i\}$ are mutually independent sequences of independent random variables.

(B) U is exponential with $E[U] = \lambda^{-1}$; D is gamma distributed with $E[D] = (k\mu)^{-1}$; Var $[D] = (\sqrt{k}\,\mu)^{-2}$, where k and μ represent, respectively, the shape and scale parameter of the distribution. The successive up and dowtimes are mutually independent as in (A).

(C) U is exponential with $E(U) = \lambda^{-1}$; D is Weibull distributed with the scale and shape parameters γ and δ, respectively, given by the following density:

(11) $$f_D(d) = \gamma^\delta \delta t^{\delta-1} e^{-(\gamma t)^\delta} \; ; \quad t \geqslant 0.$$

The successive up and downtimes are independently distributed.

(D) U is exponential with $E(U) = \lambda^{-1}$; D is lognormally distributed with parameters μ and σ, given by the following density:

(12) $$f_D(d) = \frac{1}{\sqrt{2\pi}\sigma d} \exp[-1/2(\ln d - \mu)^2/\sigma^2]; \quad d \geqslant 0.$$

The successive up and downtimes are independently distributed.

(E) U is represented by a long-tailed h distribution, see Gaver and Chu [2]. For a fixed value of the tail-stretch parameter h, $(0 < h < 1)$, represent an uptime as

$$U = \frac{(1-h)^2}{\lambda} Xe^{hX},$$

where X is exponentially distributed with $E[X] = 1$. The sequence $\{U_i\}$ is one of independent random variables, themselves independent of $\{D_i\}$. The downtimes are here assumed independent and exponential.

In case (A), the ratio $\overline{U}/\overline{D}$ is proportional to a statistic having the F distribution with $(2n, 2n)$ degrees of freedom. Thus, an exact confidence interval for the availability parameter A can be obtained in this situation with the help of the F-tables. The confidence intervals using the F-statistic were also computed.

3. NUMERICAL COMPARISONS

In this section we compare the different estimators using the three criteria described above for the probability models (A) to (E). 1,000 Monte Carlo realizations were used to obtain the estimates of these performance measures. As far as practicable, the same random numbers were used for the purpose of the comparison. We examined the case where $n = 15$. Table 1 gives the properties of the confidence intervals obtained from (9). Table 2 gives the properties of the point estimates using the PC criterion of equation (10).

TABLE 1 — *Simulation Results* (1000 Runs) Comparing the Two-sided 95% Confidence Intervals of Various Estimators of Availability; $n = 15$, $t = 2.145$ (True Value of Availability = 0.9901).*

Underlying Distributions	Coverage (%)	Average Length	Variance of Length
A. (exponential, exponential) $\lambda = 0.01, \mu = 1$	F:95.4	1.65×10^{-2}	3.65×10^{-5}
	JK.LL:94.5	1.87×10^{-2}	8.00×10^{-5}
	JK.UMVU:92.3	1.54×10^{-2}	4.59×10^{-5}
	JK.MLE:93.2	1.64×10^{-2}	5.02×10^{-5}
B. (exponential, gamma) $\lambda = 0.01, \mu = 3.0$ $k = 1/3$	F:98.4	1.67×10^{-2}	2.52×10^{-5}
	JK.LL:94.7	1.40×10^{-2}	3.21×10^{-5}
	JK.UMVU:93.4	1.29×10^{-2}	2.64×10^{-5}
	JK.MLE:94.8	1.38×10^{-2}	2.89×10^{-5}
C. (exponential, Weibull) $\lambda = 0.01, \gamma = 1.13$ $\delta = 2.0$	F:98.0	1.67×10^{-2}	2.65×10^{-5}
	JK.LL:94.3	1.39×10^{-2}	3.24×10^{-5}
	JK.UMVU:92.5	1.29×10^{-2}	2.71×10^{-5}
	JK.MLE:94.0	1.38×10^{-2}	2.98×10^{-5}
D. (exponential, lognormal) $\lambda = 0.01, \mu = 0.50,$ $\sigma = 1.0$	F:92.7	1.68×10^{-2}	5.30×10^{-5}
	JK.LL:94.9	2.28×10^{-2}	4.37×10^{-4}
	JK.UMVU:92.5	1.66×10^{-2}	8.86×10^{-5}
	JK.MLE:89.8	1.76×10^{-2}	9.53×10^{-5}
E. (long-tailed h, exponential) $\lambda = 0.01, \mu = 1,$ $h = 0.2$	F:88.2	1.77×10^{-2}	6.74×10^{-5}
	JK.LL:93.4	2.41×10^{-2}	1.60×10^{-4}
	JK.UMVU:92.7	1.91×10^{-2}	8.48×10^{-5}
	JK.MLE:94.0	2.01×10^{-2}	9.18×10^{-5}

(*F: F-statistic; JK.LL: "Log-Logistic Jackknife;" JK.UMVU: "Jackknife UMVU;" JK.MLE: "Jackknife Maximum Likelihood Estimate)

TABLE 2 — *Simulation Results (1000 Runs) Comparing the Probabilities of Concentration for Various Estimates of Availability; n = 15.*
[PC = Pr (|*Estimate* - *Availability*| < ϵ)]

Underlying Distribution	Estimate	Probability of Concentration (PC) in %			
		$a = 0.001$	$a = 0.0025$	$a = 0.005$	$a = .01$
A. (exponential, exponential) $\lambda = 0.01$, $\mu = 1$	$\hat{A}_{mle}$	22.9	53.8	83.5	97.4
	$\hat{A}_{jk,mle}$	20.8	52.7	86.1	98.3
	$\hat{A}_{umvu}$	22.5	53.9	86.0	98.3
	$\hat{A}_{jk,umvu}$	21.5	52.6	85.8	98.2
	$\hat{A}_{jk,11}$	22.4	53.4	83.4	97.7
B. (exponential, gamma) $\lambda = 0.01$, $\mu = 3.0$ $k = 1/3$	$\hat{A}_{mle}$	28.1	61.1	89.0	98.6
	$\hat{A}_{jk,mle}$	26.6	61.6	91.5	99.0
	$\hat{A}_{umvu}$	28.1	62.1	91.7	99.1
	$\hat{A}_{jk,umvu}$	26.9	61.9	91.3	99.0
	$\hat{A}_{jk,11}$	28.1	61.8	89.8	98.7
C. (exponential, Weibull) $\lambda = 0.01$, $\gamma = 1.13$ $\delta = 2.0$	$\hat{A}_{mle}$	26.2	59.9	89.1	98.3
	$\hat{A}_{jk,mle}$	25.8	59.1	91.8	98.9
	$\hat{A}_{umvu}$	26.9	60.8	92.4	99.0
	$\hat{A}_{jk,umvu}$	26.1	59.3	91.6	98.9
	$\hat{A}_{jk,11}$	26.0	59.4	90.5	98.4
D. (exponential, lognormal) $\lambda = 0.01$, $\mu = 0.50$ $\sigma = 1.0$	$\hat{A}_{mle}$	20.5	47.1	82.5	95.1
	$\hat{A}_{jk,mle}$	19.9	47.4	83.3	96.6
	$\hat{A}_{umvu}$	19.1	46.9	84.6	96.4
	$\hat{A}_{jk,umvu}$	20.4	47.4	83.6	96.6
	$\hat{A}_{jk,11}$	20.0	47.2	80.6	94.6
E. (long-tailed h, exponential) $\lambda = 0.01$, $\mu = 1$ $h = 0.2$	$\hat{A}_{mle}$	16.5	42.6	73.5	92.8
	$\hat{A}_{jk,mle}$	15.2	39.2	70.8	94.5
	$\hat{A}_{umvu}$	17.0	43.4	74.9	95.1
	$\hat{A}_{jk,umvu}$	15.5	40.0	71.1	94.1
	$\hat{A}_{jk,11}$	16.5	40.7	72.1	93.0

Table 1 shows that the log-logistic jackknife estimate provides approximate nominal coverage of the true parameter value under a wide variety of situations. In comparison, the jackknifed uniformly variance unbiased estimate or the jackknifed maximum likelihood estimate does not fare as well. However, when the probability of concentration of the point estimate is considered, the uniformly minimum variance unbiased estimate performs very well. In many cases, it performs better than the maximum likelihood estimate or the jackknife log-logistic estimate. These findings are not surprising in view of the more pronounced locally linear characteristic of the log-logistic estimate.

4. SUMMARY AND CONCLUSIONS

Several competing estimates of availability of a system which alternates between two states — "up" and "down" have been compared. The data is assumed to consist of n complete cycles of successive up and downtimes. Monte Carlo simulations carried on several special but representative situations indicate the general validity and robustness of the log-logistic jackknife estimate of Gaver and Chu. However, when point estimates are considered, from the point of view of probability of concentration, the uniformly minimum variance unbiased estimate for the Markov case performs very well in these situations.

6. ACKNOWLEDGMENT

The author wishes to thank Ms. Amy Chen for her help with the computations.

REFERENCES

[1] Barlow, R.E., and F. Proschan, *Mathematical Theory of Reliability* (John Wiley and Sons, New York, 1964).
[2] Gaver, D.P. and B.B. Chu, "Jackknife Estimates of Component and Systems Availability," Technometrics, *21*, 443-449 (1979).
[3] Mazumdar, M., "Uniformly Unbiased Estimates of Operational Readiness and Reliability in a Two-State System," Naval Research Logistics Quarterly, *16*, 199-206 (1969).
[4] Miller, R.G., "The Jackknife — a Review," Biometrika, *61*, 1-16 (1974).
[5] Rao, C.R., "Some Comments on the Minimum Mean Square Error as a Criterion of Estimation," Technical Report No. 80-21, Institute for Statistics and Applications, Department of Mathematics and Statistics, University of Pittsburgh, Pittsburgh, PA (1980).

A GOOD SIMPLE PERCENTILE ESTIMATOR OF THE WEIBULL SHAPE PARAMETER FOR USE WHEN ALL THREE PARAMETERS ARE UNKNOWN

Stelios H. Zanakis

Florida International University
College of Business Administration
Department of Decision Sciences
Miami, Florida

Nancy R. Mann*

University of California at Los Angeles
School of Medicine
Department of Biomathematics
Los Angeles, California

ABSTRACT

In this paper we consider a simple three-order-statistic asymptotically unbiased estimator of the Weibull shape parameter c for the case in which all three parameters are unknown. Optimal quantiles that minimize the asymptotic variance of this estimator, $\hat{c}$, are determined and shown to depend *only* on the true (unknown) shape parameter value c and in a rather insensitive way. Monte Carlo studies further verified that, in practice where the true shape parameter c is unknown, using always $\hat{c}$ with the optimal quantities that correspond to $c = 2.0$ produces estimates, $\hat{c}^*$, remarkably close to the theoretical optimal. A second stage estimation procedure, namely recalculating $\hat{c}$ based on the optimal quantiles corresponding to $\hat{c}^*$, was not worth the additional effort. Benchmark simulation comparisons were also made with the best percentile estimator of Zanakis [20] and with a new estimator of Wyckoff, Bain and Engelhardt [18], one that appears to be the best of proposed closed-form estimators but uses *all* sample observations.

The proposed estimator, $\hat{c}^*$, should be of interest to practitioners having limited resources and to researchers as a starting point for more accurate iterative estimation procedures. Its form is independent of all three Weibull parameters and, for not too large sample sizes, it requires the first, last and only one other (early) ordered observation. Practical guidelines are provided for choosing the best anticipated estimator of shape for a three-parameter Weibull distribution under different circumstances.

1. INTRODUCTION

The cumulative distribution function of a three-parameter Weibull variate X is given by:

$$F(x) = 1 - \exp\{-[(x - a)/b]^c\} \quad x \geq a \tag{1}$$

*Research performed by this author was supported by the US Office of Naval Research under Contract No. N00014-76-C-0723.

where "a" is the threshold, or location parameter, "b" is the scale parameter and "c" determines the shape of the distribution. See Johnson and Kotz, [7] and Mann, Schafer, Singpurwalla [11] for comprehensive information about this distribution and its applications.

Maximum likelihood estimates (MLEs) of all three parameters have desirable asymptotic properties. However, since no closed-form expression for the MLEs exists, the use of a numerical iterative procedure is necessary to solve, for a given sample, the corresponding nonlinear optimization problem. This is a computationally difficult problem for which many algorithms fail to provide solutions [19]. Other deficiencies of the maximum-likelihood procedure as it applies to this three-parameter distribution are detailed in [16].

Practitioners having limited time and access to analysts or computing facilities would thus prefer using closed-form (analytic) estimators. However, only rather inefficient estimators of this type for the three-parameter Weibull distribution have been available in the past [11]. Recently, more than seventeen closed-form estimators for the three Weibull parameters were compared by Zanakis, and the most efficient ones identified [20].

Of the three Weibull parameters, the shape parameter, c, is particularly important because it affects the shape of reliability and hazard rate curves (see [7], [12]), and is the primary cause of computational difficulties and estimation errors in MLE problems [10], [21].

Here, we examine an improved simple percentile estimator, $\tilde{c}$, of the shape parameter, with quantiles determined optimally and independently of the other two Weibull parameters.

2. A PERCENTILE ESTIMATOR OF SHAPE $\tilde{c}$ WHICH IS INDEPENDENT OF THE LOCATION AND SCALE PARAMETERS.

Let $t_1 \leqslant t_2 \leqslant \ldots \leqslant t_n$ be the ordered observations from a random sample of size n from (1).

For the *two* parameter Weibull (when the location parameter is known or zero) Dubey [3] proposed a two percentile ($p_i < p_k$) estimator for the shape parameter c:

$$\bar{c} = \ln\{\ln(1 - p_k)/\ln(1 - p_i)\}/\ln\{t_k/t_i\} \tag{2}$$

with $t_i = t_{[np_i]+1}$ and $t_k = t_{[np_k]+1}$, $[np_i]$ being the greatest integer less than or equal to np_i.

He showed (as was shown earlier by Lieblein [9] in investigating the extreme-value distribution) that its asymptotic variance is minimized when

$$p_i = 0.16731 \text{ and } p_k = 0.97366. \tag{3}$$

Murthy and Swartz [13] derived unbiasing factors and hypothesis tests for $\bar{c}$. Recently, Bryson [1] argued that the asymptotic efficiency of $\bar{c}$ is not especially sensitive to small deviations from the optimal percentiles.

In a recent paper involving simple estimators for all three Weibull parameters [20], five estimators of c were compared. These included the following proposed simple percentile estimator,

$$\tilde{c} = 0.5\ln[\ln(1 - p_k)/\ln(1 - p_i)]/\ln[(t_k - t_j)/(t_j - t_i)] \tag{4}$$

where p_i, p_j, p_k are three quantiles satisfying

$$0 < p_i < p_j < p_k \leqslant 1 \tag{5}$$

$$-\ln(1 - p_j) = \{\ln(1 - p_i) \cdot \ln(1 - p_k)\}^{1/2} \tag{6}$$

and t_i, t_j, t_k $(t_i < t_j < t_k)$ are the corresponding three ordered sample observations. (See also Johnson and Kotz [7].)

This estimator requires only three sample observations and is independent of the other two Weibull parameters. Thus, when all three parameters are unknown, estimator $\tilde{c}$ is appealing if the three percentiles can be selected in some optimal way. It was shown before [20] and verified here that using (3) and (4), i.e.,

$$\tilde{c}_2 = 1.494/\ln[(t_k - t_j)]/(t_j - t_i)] \tag{7}$$

produces a rather poor behavior.

The purpose of this paper is to derive optimal (or nearly optimal) quantiles p_i and p_k (p_j then obtained from (6)) for $\tilde{c}$ when all three Weibull parameters are unknown and to compare the estimators based on these optimal quantiles with certain others which also require no complicated iterative procedures. Such simple estimators can be used to obtain final estimates or as initial estimates for iterative procedures.

3. OPTIMAL QUANTILES FOR $\tilde{c}$

With the use of a theorem of Mosteller [12] and a lemma of Rao [14], the following analytic expression for the asymptotic variance of $\tilde{c}$ was obtained:

$$\begin{aligned}\phi(p_i,p_k,c) = {} & \frac{1}{4nc^2}\left\{\ln\left[\frac{Q_k}{Q_i}\right]^c \Bigg/ \left[\ln\left(\frac{Q_k - Q_j}{Q_j - Q_i}\right)\right]^2\right\}^2 \\ & \left\{\frac{R_i}{Q_i^{c-1}(Q_j - Q_i)}\left[\frac{1}{Q_i^{c-1}(Q_j - Q_i)} - \frac{2}{Q_j^{c-1}}\left[\frac{1}{Q_j - Q_i} + \frac{1}{Q_k - Q_j}\right] + \frac{2}{Q_k^{c-1}(Q_k - Q_j)}\right]\right. \\ & - \frac{R_j}{Q_j^{c-1}}\left[\frac{1}{Q_j - Q_i} + \frac{1}{Q_k - Q_j}\right]\left[-\frac{1}{Q_j^{c-1}}\left[\frac{1}{Q_j - Q_i} + \frac{1}{Q_k - Q_j}\right] + \frac{2}{Q_k^{c-1}(Q_k - Q_j)}\right] \\ & \left. + \frac{R_k}{Q_k^{c-1}Q_k^{c-1}} \cdot \frac{1}{(Q_k - Q_j)^2}\right\}\end{aligned} \tag{8}$$

where

$$R_s = p_s/(1 - p_s) \quad s = i,j,k$$

and

$$Q_s = [-\ln(1 - p_s)]^{1/c} \quad s = i,j,k$$

so that from (6)

$$Q_j = (Q_iQ_k)^{1/2}.$$

Thus, for a given sample size n, this asymptotic variance is a function of the true shape parameter c and only two of the three quantiles.

A pattern search procedure [6], extended by Zanakis [21] for bounded-variable nonlinear optimization problems with or without derivatives, was employed to obtain, for selected values of the shape parameter c, the quantiles p_i and p_k that minimize $\phi(p_i, p_k, c)/n$. The results are summarized in Table 1 along with the function values at the point given by (3).

TABLE 1 — *Optimal Results for* $\tilde{c}$

True Shape Parameter c	Optimal Values			Non-Optimal* $n\cdot\phi(p_i',p_k',c)$	Asymptotic Inefficiency $\phi(p_i',p_k',c)/\phi(p_i,p_k,c)$
	p_i	p_k	$n\cdot\phi(p_i,p_k,c)$		
0.5	0.0086	0.9746	0.230	0.484	2.10
1.0	0.0048	0.9816	1.028	3.194	3.11
1.5	0.0028	0.9887	3.155	12.314	3.90
2.0	0.0033	0.9920	9.096	34.976	3.85
2.5	0.0051	0.9932	23.215	81.286	3.50
3.0	0.0072	0.9939	51.070	164.374	3.22
3.5	0.0092	0.9944	99.545	300.142	3.02
4.0	0.0109	0.9947	176.936	507.770	2.87
4.5	0.0124	0.9949	292.965	809.210	2.76
5.0	0.0137	0.9951	458.762	1229.437	2.68
7.5	0.0179	0.9957	2522.945	6194.795	2.46
10.0	0.0202	0.9960	8314.425	19586.645	2.36

*$p_i' = 0.16731$ and $p_k' = 0.97366$

These results reveal that the asymptotically optimal quantiles (p_i,p_k) depend on the true value of the shape parameter in a rather insensitive way, especially for p_k.

It should be noted that using instead (p_i,p_k) from (3), which is optimal for the two-parameter Weibull distribution, essentially triples the asymptotic variance of estimator $\tilde{c}$ if all three Weibull parameters are unknown. This is more pronounced when the true $c = 1.5$ to 2.0, as the last column of Table 1 indicates. Note also, from Dubey [3] and Kimball [8], that the asymptotic variance of $\bar{c}$ (given by our Equation (2)) when the location parameter, a, is known to be or can be set equal to zero, is $0.92\ c^2/n$. Thus, incorrectly assuming that "a" is nonzero makes the estimation of c much more difficult than necessary.

4. MONTE CARLO INVESTIGATIONS

In order to gain further insight into the behavior of the percentile estimator $\tilde{c}$, two Monte Carlo investigations were made. The first investigated the sensitivity of $\tilde{c}$ to small deviations from the optimal quantiles and whether a simple iterative scheme could improve the performance of $\tilde{c}$. The second investigation compared $\tilde{c}$ to a very recently proposed efficient percentile estimator, $\hat{c}$, based on *all* order statistics [18] and the best percentile estimator, $\tilde{c}'$, found in our previous study [20].

Sensitivity of $\tilde{c}$

Since the asymptotically optimal quantiles depend upon the true (unknown) shape parameter c in such an insensitive way, a Monte Carlo study with 1,000 replications was made using $\tilde{c}$, defined by (4), (5), and (6), with p_i and p_k corresponding to optimal quantile values for

$c = 2.0$ and 3.0. Over the range of investigation, i.e., $c = 0.5\ (0.5)3.5$, somewhat better results were obtained with $\tilde{c}^*$ corresponding to $p_i = 0.0033$, $p_k = 0.9920$ than with one corresponding to $p_i = 0.0072$, $p_k = 0.9939$. We define

$$\tilde{c}^* = 3.643/\ln[(t_k - t_i)/(t_j - t_i)] \tag{9}$$

where $t_i = t_{[0.0033n]+1}$, $t_k = t_{[0.9920n]+1}$, and $t_j = t_{[0.1187n]+1}$. Note that for most sample sizes of interest, $t_i = t_1$ and $t_k = t_n$.

In this investigation, we also examined the following two estimators:

$\tilde{c}_0$: defined by (4), (5), (6) and based on the *optimal* combination of p_i and p_k from Table 1, although this would not ordinarily be possible since the true c value is not known. For this reason, we also examined the following:

$\tilde{c}^{**}$: a single iteration modified estimator

$$\tilde{c}^{**} = \tilde{c}(p_i^*,\ p_k^*) \tag{10}$$

with p_i^* and p_k^* selected in a manner prescribed by the value of $\tilde{c}^*$ as given in Table 2.

Table 2 — *Rule for Selection of p_i and p_k in Calculation of $\tilde{c}^{**} = \tilde{c}\ (p_i^*, p_k^*)$*

$\tilde{c}^*$	p_i^*	p_k^*
0.0 - 0.5	0.0086	0.9746
0.5 - 1.0	0.0048	0.9816
1.0 - 1.5	0.0028	0.9887
1.5 - 2.0	0.0033	0.9920
2.0 - 2.5	0.0033	0.9920
2.5 - 3.0	0.0033	0.9920
3.0 - 3.5	0.0033	0.9920

The schedule shown in Table 2 for selection of combinations of values of p_i^* and p_k^* was determined from results involving $\tilde{c}_0$ and $\tilde{c}^*$. Because of the negative bias of $\tilde{c}^*$ for values of $c \geqslant 1.5$, a correcting term (ranging from 0.1 to 0.7) was also investigated for calculating $\tilde{c}^{**}$ as a function of $\tilde{c}^*$. This tended to decrease the bias of $\tilde{c}^{**}$ at the expense of higher mean square errors. Early simulation results also revealed that for $n \leqslant 100$, perfect choice of percentiles (those corresponding to the true but unknown c) reduces the bias but not necessarily the MSE, particularly when the true c is large. This theoretical estimator, $\tilde{c}_0$, has optimal asymptotic properties and is not necessarily good for small n and large c values.

Comparison with Two Previous Percentile Estimators $\tilde{c}'$ and $\hat{c}$

As a benchmark comparison, the proposed estimators $\tilde{c}^*$ and $\tilde{c}^{**}$ were also compared via a similar Monte Carlo study of 5,000 replications with the two best percentile estimators, $\tilde{c}'$ and $\hat{c}$, suggested in earlier studies [20 and 18].

The best of the simple estimators for the shape of a three-parameter Weibull distribution examined in [20] was found to be:

(11) $$\tilde{c}' = 2.989/\ln[(t_k - \tilde{a})/(t_i - \tilde{a})]$$

where

(12) $$\tilde{a} = (t_1 t_n - t_2^2)/(t_1 + t_n - 2t_2)$$

was the best estimate of the Weibull location parameter and $2.989 = \ln[\ln(1 - 0.97366/\ln(1 - 0.16731)]$.

Wyckoff, Bain and Engelhardt [18] recently extended their earlier estimator $\hat{c}$ [4] via the following three step procedure:

First determine the shape parameter initial estimator (C12.20 in [20])

(13) $$\hat{c}_0 = 2.989/\ln[(t_{\lfloor np_k \rfloor + 1} - t_1)/(t_{\lfloor np_i \rfloor + 1} - t_1)]$$

with p_i, p_k given by (3), and the location parameter estimator

(14) $$\hat{a} = [t_1 - \bar{t}/n^{1/\hat{c}_0}]/[1 - 1/n^{1/\hat{c}_0}]$$

where $\bar{t}$ is the sample mean. Note the similarity of estimators (11) and (13). Then the Wyckoff, Bain, Engelhardt estimator is given by:

(15) $$\hat{c} = nk_n/\left[-\sum_{r=1}^{s} \ln(t_r - \hat{a}) + \frac{s}{n-s}\sum_{r=s+1}^{n} \ln(t_r - \hat{a})\right]$$

where $s = [0.84n]$ and the constant k_n is tabulated in [4].

This estimator uses *all* order statistics and consequently it is not surprising that it was found in [18] to have smaller bias and MSE than the much simpler estimator $\tilde{c}'$, which is based only on three quantiles. However, $\tilde{c}'$ performed better in [20] than the earlier version of $\hat{c}$ proposed by Engelhardt and Bain [4].

Monte Carlo results comparing the proposed new estimators $\tilde{c}^*$ and $\tilde{c}^{**}$ with the previous estimators $\tilde{c}'$ and $\hat{c}$ are shown in Table 3. The nonoptimal estimator $\tilde{c}_2$ given by (7) was also computed in the same simulation, but MSE values were so large and unpredictable for large values of c, that no comparisons were necessary.

TABLE 3 — *Comparison of Average Values and Mean Squared Errors of* $\tilde{c}', \hat{c}, \tilde{c}^*$, and $\tilde{c}^{**}$

		Average Value, True c							MSE, True c						
		0.50	1.00	1.50	2.00	2.50	3.00	3.50	0.50	1.00	1.50	2.00	2.50	3.00	3.50
$n = 30$	$\tilde{c}'$	0.52	0.99	1.38	1.70	1.95	2.18	2.36	0.008	0.031	0.087	0.227	0.510	1.031	1.551
	$\hat{c}$	0.53	1.04	1.52	1.91	2.28	2.60	2.87	0.008	0.035	0.097	0.191	0.391	0.705	1.133
	$\tilde{c}^*$	0.54	1.01	1.43	1.81	2.13	2.45	2.80	0.011	0.038	0.120	0.313	0.673	1.528	2.258
	$\tilde{c}^{**}$	0.50	0.99	1.44	1.81	2.13	2.45	2.80	0.008	0.053	0.120	0.310	0.668	1.523	2.271
$n = 50$	$\tilde{c}'$	0.52	1.01	1.42	1.77	2.05	2.30	2.48	0.005	0.020	0.050	0.143	0.343	0.714	1.144
	$\hat{c}$	0.53	1.06	1.55	2.01	2.36	2.75	3.09	0.005	0.021	0.052	0.131	0.236	0.392	0.804
	$\tilde{c}^*$	0.51	0.98	1.41	1.80	2.16	2.52	2.83	0.005	0.021	0.067	0.187	0.417	0.851	1.644
	$\tilde{c}^{**}$	0.50	0.99	1.42	1.80	2.16	2.52	2.83	0.004	0.025	0.063	0.183	0.414	0.848	1.642
$n = 100$	$\tilde{c}'$	0.50	0.99	1.42	1.79	2.09	2.36	2.58	0.002	0.009	0.027	0.087	0.232	0.518	0.991
	$\hat{c}$	0.52	1.02	1.52	1.99	2.37	2.78	3.13	0.002	0.008	0.022	0.059	0.129	0.240	0.395
	$\tilde{c}^*$	0.50	0.97	1.43	1.87	2.27	2.70	3.11	0.002	0.011	0.033	0.093	0.217	0.434	0.823
	$\tilde{c}^{**}$	0.51	1.00	1.47	1.87	2.27	2.70	3.11	0.002	0.013	0.026	0.088	0.215	0.434	0.823
		Number of replications = 5000							Number of replications = 5000						

Estimator's $\tilde{c}^{**}$ slightly better performance over $\tilde{c}^{*}$ is not considered worthy of the additional effort required. Therefore, the latter is instead recommended in all cases—especially when the true c is large, but with caution if also $n \leqslant 30$

All four estimators behave similarly for small values of c, particularly as n increases. Comparing the proposed estimator $\tilde{c}^{*}$ with our earlier $\tilde{c}'$ we see that the former has a smaller bias in almost all cases. The MSE of $\tilde{c}^{*}$ is clearly smaller than that of $\tilde{c}'$ when $n = 100$ and $c \geqslant 2.50$. In problems with large sample sizes n, $\tilde{c}'$ tends to have almost no variance around its expectation; thus, for large n, the MSE of $\tilde{c}'$ consists mostly of the bias squared, with almost no contribution from its variance.

Finally, the bias of $\tilde{c}^{*}$ is about equal to that of $\hat{c}$, but the MSE of $\hat{c}$ is smaller than that of $\tilde{c}^{*}$ when $c \geqslant 2.00$. Our results about $\hat{c}$ and $\tilde{c}'$ are consistent with those in [18].

5. SOME LITERATURE EXAMPLES

The estimators examined earlier in this paper, along with some other percentile estimators used in [20] were applied to seven literature test problems, with known shape parameter. Practitioners may find the results summarized in Table 4 particularly useful. However, these results should not be generalized since they represent a static picture to a few examples; something like setting your watch at 9:00 and keeping it there: it will be an excellent estimate of time twice a day.

TABLE 4 — *Literature Test Problem Results*

Problem / Source			1 [5]	2 [5]	3 [17]	4 [17]	5 [2]	6 [15]	7 [15]
Sample		Size	40	40	100	100	100	100	100
True		a	10	20			0	1.975	0
		b	100	100			1	47.072	47.072
		c	2	3	2	2	1.2	1.328	1.328
$p_l = 0.16731$ $p_k = 0.97366$	Zanakis	$\tilde{c}_2 = C8$‡	7.697	2.027	1.360	1.577	0.986	1.546	1.344*
	Zanakis	$C12.21$‡	2.950	3.633	2.001*	2.139	1.305	1.436	1.600
	Zanakis	$\tilde{c}' = C12.23$‡	1.801	2.226	1.714	1.747	1.080	1.350	1.377
	Hassanein	$C25.23$‡	1.502	1.933	1.755	1.154	1.170	1.321*	1.041
	Engelhardt & Bain	$C27.23$‡	1.657	1.962	1.770	1.225	1.175*	1.262	1.139
	Wyckoff, Bain & Englehardt	$\hat{c}$	2.185	2.571*	1.970	1.870*	1.280	1.380	1.444
Zanakis-Mann $\tilde{c}^{*}$ ($p_l = 0.0033$, $p_k = 0.9920$)			2.026*	1.939	1.712	1.575	1.137	1.231	1.384
MLE for c			2.48	2.330	1.783	1.857	1.201†	1.330†	1.767

*Best percentile estimate for problem.
†MLE better than best percentile estimate for problem.
‡Notation used in [20].

It is interesting to note that in five out of the seven problems, one or more simple percentile estimator was more accurate than the MLEs (iteratively obtained on a computer). As it was shown earlier [19], [21], this occurs mostly in problems with small shape parameter values, particularly when the sample size is small—a fact of particular interest to practitioners having limited resources.

6. CONCLUSIONS

A simple, three-order-statistic estimator $\tilde{c}^*$ is developed for estimation of the Weibull shape parameter c, when all three parameters are unknown. It was found to compare very well, in terms of mean squared error, with an estimator of the same form based on an optimal selection of order statistics, dependent only upon the unknown true value of c. Compared to our previous best simple estimator $\tilde{c}'$ [20], $\tilde{c}^*$ has a smaller bias in almost all cases and a smaller MSE in problems with large values of n and c. For small values of c, $\tilde{c}^*$ and $\tilde{c}'$ are as efficient as the new estimator $\hat{c}$ of Wyckoff, Bain, and Engelhardt [18] which is determined from *all* ordered observations after a single iteration.

We feel that all three percentile estimators $\tilde{c}^*$, $\tilde{c}'$ and $\hat{c}$ are useful to practioners and researchers, depending on the particular circumstances as suggested in Figure 1.

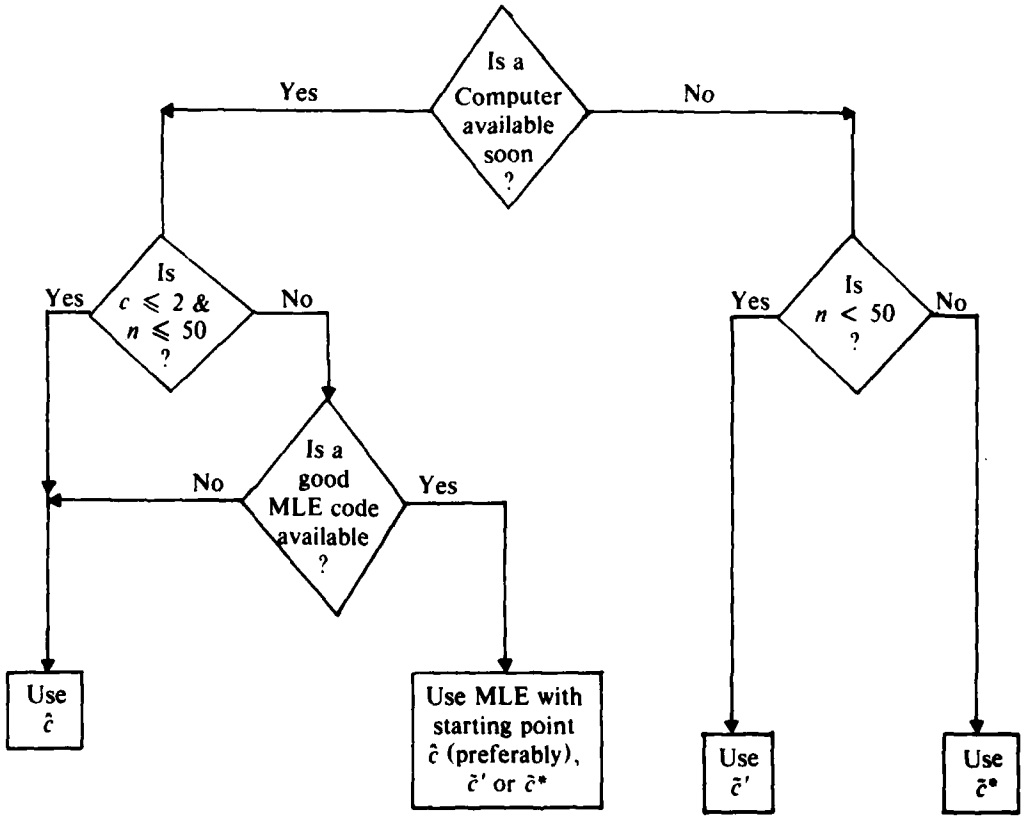


FIGURE 1. Practical guidelines for choosing an estimator for the shape of a three-parameter Weibull distribution.

The following comments will further clarify and support our suggestions:

If a computer code is available, one is naturally tempted to use Maximum Likelihood Estimation iterative procedures. However, research has shown that:

(i) MLEs sometimes do not exist or represent a local optimum [16];

(ii) The shape parameter is the least accurate of the three Weibull parameters to estimate, causing computational difficulties for MLEs as c gets larger [19];

(iii) In nonregular cases ($c \leqslant 2$), the simple percentile estimator $\tilde{c}'$ was found to be more accurate than the corresponding MLE, particularly when n is small [21];

(iv) the choice of a percentile estimator like $\tilde{c}'$, $\tilde{c}^*$ or $\hat{c}$ as a starting point may affect the speed of convergence, but not so much the final shape parameter estimate of a good iterative MLE procedure [21]; and

(v) If the sample size is too large, determination of MLE or even $\hat{c}$ is very time consuming—a real concern in repetitive situations.

So, even if a computer is available, one may use a percentile estimator initially, additionally or instead.

It should be noted that practitioners and students (usually nonstatisticians) often need a good estimator of the Weibull shape parameter and do not have the time or access to a computer facility, but only a pocket calculator. In such cases, estimator $\hat{c}$—which uses all order statistics—will be computationally prohibitive for any but extremely small sample sizes. Instead we recommend using $\tilde{c}^*$ if $n \geqslant 50$, $\tilde{c}'$ otherwise. The unusual advantage of $\tilde{c}^*$ is that it is independent of the other Weibull parameters.

These quick estimators are also well suited for obtaining point and interval estimates for the otherwise unattainable optimum solution to large scale integer, combinatorial or nonconvex mathematical programming problems [22]. A sample of heuristic minimum solutions fits naturally a three-parameter Weibull distribution.

In this paper we also demonstrated how the use of optimization methods can change an erratic estimator ($\tilde{c}_2$) into a good estimator ($\tilde{c}^*$), by specifying the appropriate three order statistics that minimize the asymptotic variance of this simple estimator.

BIBLIOGRAPHY

[1] Bryson, M.S., "On the Efficiency of Weibull Parameter Estimation from Two Order Statistics," Los Alamos Scientific Laboratory Report (1978).

[2] Dubey, S.D., "Comparative Performance of Several Estimators of Weibull Parameters," 20th Annual Technical Conference Transactions, American Society for Quality Control (1966).

[3] Dubey, S.D., "Some Percentile Estimators for Weibull Parameters," Technometrics, *9*, 119-129 (1967).

[4] Engelhardt, M.E. and L.J. Bain, "Simplified Statistical Procedures for the Weibull or Extreme-Value Distributions," Technometrics, *19*, 323-331 (1974).

[5] Harter, H.L. and A.H. Moore, "Maximum Likelihood Estimation of the Parameters of Gamma and Weibull Populations from Complete and from Censored Samples," Technometrics, *7*, 639-643 (1965). Errata, Technometrics, *9* (1967).

[6] Hooke, R. and T. Jeeves, "Direct Search Solution of Numerical and Statistical Problems," Journal of the ACM, *8*, 212-229 (1961).

[7] Johnson, N.L. and S. Kotz, *Distributions in Statistics: Continuous Univariate Distributions—1* (John Wiley and Sons, New York, 1970).
[8] Kimball, B.F., "An Approximation to the Sampling Variance of Given Frequency Based on Fit of Doubly Exponential Distribution of Maximum Values," Annals of Mathematical Statistics, *17*, 299-309 (1949).
[9] Lieblein, J., "A New Method for Analyzing Extreme-Value Data," Technical Note 3053, National Advisory Committee for Aeronautics (1954).
[10] Mann, N.R., "Estimators and Exact Confidence Bounds for Weibull Parameters Based on a Few Ordered Observations," Technometrics, *12*, 345-361 (1970).
[11] Mann, N.R., R.E. Schafer, and N.D. Singpurwalla, *Methods for Statistical Analysis of Reliability and Life Data* (John Wiley and Sons, New York, 1974).
[12] Mosteller, F., "On Some Useful Inefficient Statistics," Annals of Mathematical Statistics, *17* (1946).
[13] Murthy, V.K. and G.B. Swartz, "Estimation of Weibull Parameters from Two-order Statistics," Royal Statistical Society Journal, Series B, *37*, 96-102 (1975).
[14] Rao, C.R., *Advanced Statistical Methods in Biometric Research* (John Wiley and Sons, New York, 1952).
[15] Ravenis, J.V.J., II, "Estimating Weibull-Distribution Parameters," Electrotechnology, 46-54 (1964).
[16] Rockette, H., C. Antle, and L.A. Klimko, "Maximum Likelihood Estimation with the Weibull Model," Journal of the American Statistical Association, *69*, 246-249 (1974).
[17] Smith, H. and S.D. Dubey, "Some Reliability Problems in the Chemical Industry," Industrial Quality Control, 64-70 (1964).
[18] Wyckoff, J., L.J. Bain, and M.E. Engelhardt, "Some Complete and Censored Sampling Results for the Three-Parameter Weibull Distribution," Journal of Statistical Computation and Simulation, *11*, 139-151 (1980).
[19] Zanakis, S.H., "Computational Experience with Some Nonlinear Optimization Algorithms in Deriving Maximum Likelihood Estimates for the Three-Parameter Weibull Distribution," in *Algorithmic Methods in Probability*, M.F. Neuts, Editor (North Holland/TIMS Studies in Management Sciences, *7*, 63-77, 1977).
[20] Zanakis, S.H., "A Simulation Study of Some Simple Estimators for the Three-Parameter Weibull Distribution," Journal of Statistical Computation and Simulation, *9*, 101-116 (1979).
[21] Zanakis, S.H., "Extended Pattern Search with Transformations for the Three-Parameter Weibull MLE Problem," Management Science, *25*, 1149-1161 (1979).
[22] Zanakis, S.H. and J.R. Evans, "Heuristic Optimization: Why, When and How to Use It," Interfaces, *11*, 84-91 (1981).

LINEARLY CONSTRAINED PSEUDO-NEWTON METHOD

Jerrold H. May

Graduate School of Business
University of Pittsburgh
Pittsburgh, Pennsylvania

ABSTRACT

Must Newton-type methods for linearly constrained optimization be either of the modified Newton or quasi-Newton variety? The contention of this paper is that explicity recomputing part of the projected Hessian may be superior to both approaches. A computational comparison with MINOS is presented.

1. INTRODUCTION

In a previous paper [10], we presented a new algorithm for solving the linearly constrained nonlinear programming problem:

$$\text{Minimize } f(x)$$
$$\text{subject to } Ax \geqq b.$$

That algorithm is a modified Newton one, i.e., for a problem involving n decision variables, an entire matrix of projected second partial derivatives, which may be as large as $n \times n$, must be computed at each iteration. Such an approach, which requires order n^2 function evaluations and order n^3 arithmetic operations (multiplications and divisions) at each iteration, may not be practical when dealing with larger problems, say $n \geqq 20$. This paper proposes a closely related method, which requires substantially less computational work per iteration, while retaining the desirable features of our previous algorithm.

The chief novelty of our approach in this paper is the idea of explicitly updating part of the second derivative matrix at each iteration. To illustrate the idea, the user chooses a "pipe width," π, of rows to update each iteration, and the "pipe" moves from upper left to lower right from iteration to iteration. For example, if the second derivative matrix is 5×5 and $\pi = 2$, at the first iteration the updated elements (indicated by asterisks) would be

$$\begin{bmatrix} * & * & 0 & 0 & 0 \\ * & * & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

at the second iteration,

$$\begin{bmatrix} 0 & 0 & * & * & 0 \\ 0 & 0 & * & * & 0 \\ * & * & * & * & 0 \\ * & * & * & * & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

and at the third iteration

$$\begin{bmatrix} * & 0 & 0 & 0 & * \\ 0 & 0 & 0 & 0 & * \\ 0 & 0 & 0 & 0 & * \\ 0 & 0 & 0 & 0 & * \\ * & * & * & * & * \end{bmatrix}.$$

Broadly speaking, Newton-like methods fall into two categories—modified Newton ones and quasi-Newton ones. As indicated above, modified Newton methods update an entire second derivative matrix at each iteration, while quasi-Newton algorithms iteratively update an approximated matrix by adding a matrix of low rank (typically rank 1 or rank 2) to the last such approximation. While modified Newton methods generally perform well, they are impractical for larger problems. Quasi Newton methods require less work (usually order n function evaluations and order n^2 arithmetic), but are potentially subject to a number of numerical shortcomings, chiefly due to the second derivative approximation process, related to scaling the elements, restarts, and inability to deal with regions of nonpositive curvature. Our idea is to take a middle ground between those two approaches, by letting the user decide how much of the second derivative information to recompute each time. Our mathematical analysis shows that such an algorithm is a viable idea, and some computational evidence indicates that it may yield solutions more rapidly than either equivalent modified or quasi-Newton methods.

The present method inherits four major strengths from our modified Newton approach.

First, the theoretical convergence results carry over to the present algorithm. Thus, under relatively mild conditions, the algorithm can be shown to converge to a point satisfying first order necessary optimality conditions. Under somewhat stronger conditions, the methods find a point also satisfying second order necessary conditions. The rate of convergence is also shown to be superlinear, that is $\{||x^{k+1} - \bar{x}||/||x^k - \bar{x}||\} \to 0$ as $k \to \infty$, or of order two ($||x^{k+1} - \bar{x}|| \leq c||x^{k+1} - \bar{x}||^2$ for some c), depending on the assumptions made, where $\{x^k\}$ is the sequence of points generated by the method and $\bar{x}$ is the optimal solution.

Second, the method is reasonably easy to use. It does not require the user to provide analytical derivatives. However, while we have had some success in using this approach on problems whose derivatives are not smooth (May, Shocker and Sudharshan [11]), the convergence proofs do assume such properties. Our computational experience indicates that the performance of the method is sensitive to only a few parameters, but will converge dependably even with highly nonoptimal parameter values.

Third, the algorithm can deal directly with areas of nonpositive curvature. When minimizing a function, and using a Newton-like method, one would like the second derivative matrix

(or its projection, if appropriate) to be positive definite, since then the usual search direction generation approach yields a descent (improvement) direction. When the second derivative matrix is not positive definite, though, a modified approach is required if we want to be sure to obtain a useful search direction. Now the second derivative matrix is stored in a factorized form, so it can be efficiently used in the solution of the set of linear equations yielding the search direction. In our modified Newton method, we utilized the modified LDL^T factorization of Gill and Murray [6], where L is a unit diagonal lower triangular matrix, and D is a diagonal matrix with all positive entries. That factorization, when applied to a nonpositive definite matrix A, will actually find the factors of $A + E$, where E is a diagonal matrix with positive entries, instead of breaking down, as the usual LDL^T factorization would. In our present algorithm, the Gill and Murray LDL^T factorization could not be used, since we could not find a way of updating it in order n^2 arithmetic when the second derivative matrix is modified by the addition of a rank-1 matrix. Instead, we use the factorization of Bunch and Parlett [2], which they denote Q^TMDM^TQ, where Q is a permutation matrix, M unit diagonal lower triangular, and D block diagonal, i.e., it may have entries in the first sub- and super-diagonal, with only 1 × 1 and 2 × 2 blocks allowed. (We will suppress the permutation matrix Q to simplify the notation.) Sorenson [17] has shown how to modify the MDM^T factorization when a rank 1 matrix is added to the matrix under consideration. While modified Newton methods often have provisions for dealing with the case of nonpositive definiteness, using special line search techniques, our method appears to be the first one which extends these methodologies to a quasi-Newton environment.

Finally, the algorithm is a "least-constrained" method, in the sense of Lenard [8], which means it may find the optimal constraint set, and thus converge, quickly. Since our approach uses an ϵ-active constraint strategy, it is desirable to identify those constraints met with equality at the optimal solution as quickly as possible. A least constrained method allows for the dropping or adding of several constraints from the active set at each iteration, as opposed to the usual simplex-technique related strategy of dropping or adding only one at a time.

2. THE COORDINATE SYSTEM

A basic idea in our approach is the representation of the locally feasible region in the neighborhood of a point x by a matrix factorization of the active constraint matrix N. That is, the columns of N are the normals to the constraint considered active at x. Since we always assume that N is full rank, for an $n \times u$ matrix N, we can find an $n \times n$ orthogonal matrix Q and a $u \times u$ upper triangular nonsingular matrix R such that

$$QN = \begin{bmatrix} R \\ 0 \end{bmatrix}$$

where 0 is a matrix of zeros.

Now the last $(n - u)$ rows of Q are all orthogonal to the columns of N, which means that they define a set of vectors spanning the linear manifold defined by the active constraints. Moving away from x along any such row keeps all the active constraint active. Also, the generalized inverse of N, N^+, is given by $N^+ = [R^{-1}0]Q$. The rows of N^+ have the property that movement away from x along the first row of N^+, say, corresponds to dropping the first active constraint while retaining all the others.

The rows of N^+, plus the last $(n - u)$ rows of Q, constitute our coordinate system for performing an iteration of the method. Since N changes with any change in the active set, so does the coordinate system. The method is thus adaptive, adjusting itself to the local structure of the feasible region. This adaptability, though, implies a need for proper second derivative matrix maintenance, as outlined in Section 5.

Note that by the orthogonality of Q, the two sets of rows are orthogonal, but that the rows of N^+ are not necessarily mutually orthogonal.

3. THE ALGORITHM

The algorithm requires positive real numbers $\alpha, \beta, \gamma, \epsilon, \tau$, and ρ, with $\rho < 1$ and $\beta^2 < \rho/2\mu$. α and β are used for move point acceptability, and ρ is used for a second-order Armijo type test for Newton point acceptability. An ϵ-active constraint strategy is incorporated by letting any column A_j of A satisfying $(A_j)^T x - b_j < \epsilon s$ be considered active. γ is of the order of machine precision, and is introduced to avoid numerical difficulties. τ is required in the theoretical development as a lower bound on the negative curvature curve stepsize but could be set arbitrarily small. Also required are an initial feasible point, x^0, an initial stepsize $s^0 > 0$, an initial n-vector σ^0 of $+1$'s and -1's, a symmetric matrix G^0, and π, the number of Hessian rows / columns to be updated each iteration.

Let N^+ and Q be determined by QR factorization of N. Denote the ith column of $[N^+]^T$ by n_i and ith column of Q^T by q_i, that is,

$$[N^+]^T = [n_1\, n_2 \ldots n_u] \text{ and } Q^T = [q_1\, q_2 \ldots q_n].$$

Initialize the current solution point $x = x^0$, the current stepsize $s = s^0$, the current direction sign indicator $\sigma = \sigma^0$, the current matrix of QR approximate second partial derivatives $G = G^0$, the sequence index $k = 0$, the last updated column $l = n$, and the local variations failure indicator $r = 0$.

General Iteration

Step 1: (Determine active constraints.) Set the square of the gradient norm $\eta^2 = 0$. Determine N for the current x and s. If the active set has changed since the last execution of this step, update Q, N^+, and G.

Step 2: (Computer second order multiplier approximations.) For each constraint i, $i = 1, 2, \ldots, u$, compute Δf_i, an approximate Karush-Kuhn-Tucker multiplier, by evaluating f at two feasible points, a stepsize and a half a stepsize along n_i, and using a three point forward difference approximation. If, for any 0, $\Delta f_i > 0$ and neither of the two points evaluated along n_i yields an improvement, dropping constraint i would be undesirable, so set $d_i = 0$. For each i such that $\Delta f_i \leqslant 0$ or one of those two points does yield an improvement, compute an approximate second partial derivative $\Delta^2 f_i$, set $d_i = |\Delta f_i|$, and let $\eta^2 = \eta^2 + (\Delta f_i)^2$.

Let x_B denote the point with lowest objective function value amongst all those evaluated at the step.

Step 3: (First and second derivatives for manifold directions explicitly updated). Let q_i denote the row of Q corresponding to the j th row of G, which is to be explicitly updated this iteration. Then evaluate f at two points, one a feasible stepsize along q_i away from x and the other a feasible stepsize along $-q_i$ away from x. Using a central difference formula, compute g_j and G_{jj}, approximate first and second partial derivatives along q_i (the discrepancy in subscripts is due to the fact that we are interested in only the last $(n-u)$ rows of Q). Set $\eta^2 = n^2 + g_j^2$, adding g_j to the norm of the (projected) gradient.

Update x_B to represent the better of x_B from the last step and the lowest function value point evaluated at this step.

Step 4: (First derivatives for manifold directions not updated.) Let q_i denote the row of Q corresponding to the j th row of G, which is not to be explicitly updated this iteration. Then evaluate f at one point, a feasible stepsize along $\sigma_i q_i$ away from x. Using the value of G_{jj} from the last iteration, and the new function value, compute a second order approximation to g_j. Add g_j^2 to η^2.

Update x_B if appropriate.

Step 5: (Compute mixed second partial derivatives.) Consider each row q_i as in Step 3. Then for every other q_k corresponding to the l th row of G, $l < j$, evaluate f at a feasible point appropriately chosen along $q_i + q_k$, and compute the mixed second partial derivative G_{jl} (this fills in the "pipe" as in Section 1).

Update x_B if appropriate.

Incorporate all the new second derivatives into the matrix factors of G.

Note that $(n-u)/\pi$ passes through this step, with the same N, will explicitly update all of G, using $\frac{1}{2}(n-u)^2 - (n-u)$ evaluations. In the worst case $(u=0)$, then, $\frac{1}{2}\pi(n-1)$ evaluations are used, on the average.

The factors of G may be updated at this point using any rank-one or rank-two formula which does not force positive definiteness on G on an infinite number of iterations.

Updating row/column l of G, for $l \neq 1$, is equivalent to adding a rank-two update of the form $we_l^T + e_l w^T$, where $w_{l+1} = \ldots = w_n = 0$, is the l th column of the identity matrix, so that if a scheme utilizing this special structure was devised, at most π updating passes using Sorenson's method [17] would have to be performed. We have not yet devised a procedure exploiting this structure. Our code uses a strategy that is reasonably efficient when $\pi < \frac{1}{2}(n-u)$, and does two rank-one modifications for each row/column updated—the second of which wipes out the undesired results of the first. Using the operation count in [18], between $2\pi(n-u)^2 + 0(n)$ and $(11/3)\pi(n-u)^2 + 0(n)$ operations are required for this updating. Full details on updating the Bunch-Parlett factorization are given in [17].

Step 6: (Check accuracy of derivative approximations.) If G is positive semidefinite and the gradient norm η is sufficiently small, stop; x is an acceptable solution.

If G is positive definite but the stepsize is too big relative to the gradient norm, i.e., $s > (\eta/\alpha)$, set the failure indicator $r = 1$ and go to Step 12, where the stepsize will be reduced with no movement made. Otherwise, scale the Newton search direction d relative to the N^+ directions. If G is positive definite, go to Step 7, otherwise to Step 8.

Step 7: (Eigenanalysis when Hessian approximation is not positive definite.) Determine Z and Λ such that Z is orthogonal, Λ is diagonal, and $D = Z\Lambda Z^T$. Let Λ_{qq} denote the most negative eigenvalue of D. If $\Lambda_{qq} = 0$, D is positive semidefinite; go to Step 8. Otherwise, a direction of strictly negative curvature exists. Solve $M^T y = Z_q$ for it. If the gradient norm is too small, go to Step 9.

Step 8: (Compute direction of line search.) Solve $M(D + ZEZ^T)\, M^T\delta = -g$ for δ, where E is a diagonal matrix given by $E_{jj} = [\max\{|\Lambda_{jj}|,\ \gamma(n-u)\lambda, \gamma\} - \Lambda_{jj}]$ for $j = 1, 2, \ldots, n-u$, and $\lambda = \max_{1 \leqslant i \leqslant n-u} |\Lambda_{ii}|$. E is thus the "smallest" diagonal additions matrix necessary to force positive definiteness on G. If D is positive definite, go to Step 10 and search only along the usual Newton direction. If D is not positive definite, consider either the "fixed-up Hessian" direction $[N^+]^T d + Q_0^T\delta$ or the quadratic curve $\{x_t | x_t = x + t[(N^+)^T d - Q_0^T g] + t^2[(N^+)^T d + Q_0^T y]\}$, where Q_0 is the submatrix consisting of the last $(n-u)$ rows of Q. Note that $(N^+)^T d - Q_0^T g$ is just the negative gradient in QR coordinates. The second vector, $(N^+)^T d + Q_0^T y$, is guaranteed to be a descent direction for f if $\Lambda_{qq} < 0$ and the derivative approximations are sufficiently accurate. Our curve search is thus similar to that of McCormick [12] and More and Sorenson [14], except that we reverse the t and t^2 terms. Our motivation for doing this is that, unlike them, we allow a stepsize t to exceed one. Since the gradient direction is most useful only close to the current point, we wanted it to be multiplied by t rather than t^2.

If the curve is to be searched, go to Step 9; if the "fixed-up" Hessian direction, go to Step 10.

Step 9: (Search along a curve.) Use a one variable search method to find a $t^* \geqslant \tau$ such that the point x_{t^*} satisfies the two term Armijo-type acceptability condition $f(x_{t^*}) - f(x) \leqslant \rho[t^*\eta^2 + (t^*)^2 y^T g + \frac{1}{2}(t^*)^4 y^T G y]$, trying $t = 1$ first. If successful, let $x_B = x_{t^*}$. In either case, go to Step 11.

Step 10: (Regular line search.) Search for a $t > 0$ such that $f(x + t\{(N^+)^T d + Q_0^T g\}) - f(x) \leqslant \rho[t(d^T\Delta f + \delta^T g) + \frac{1}{2}\, t^2\delta^T G]$, trying $t = 1$ first. If successful, redefine x_B.

Step 11: (If the move point is good enough, reduce the stepsize.) If $f(x_B) - f(x) > -\alpha^2\beta^2 s^2$, there has not been sufficient function value decrease; go to Step 12. If $f(x_B) - f(x) \leqslant -\alpha^2\beta^2 s^2$, set the local variations failure indicator $r = 0$. If $f(x_b) - f(x) \leqslant \beta^2\eta^2$, there has been, in addition, sufficient decrease relative to the gradient norm, so choose a new smaller stepsize s' and go to Step 13. Otherwise, $x \leftarrow x_B$; go to Step 1, having moved to x_B, but using the same stepsize on the next iteration.

Step 12:(Failure; try reversing the local variations directions.) Reverse the σ direction indicators. If $r = 0$, this is the first time this step has been encountered with the current x and s; set $r = 1$ and go to Step 1. Otherwise, if $r = 1$, let the new stepsize $s' = s/2$, set $r = 0$, $x_B \leftarrow x$, and go to Step 13.

Step 13:(Define a new sequence point.) If $x \neq x^k$, $k \leftarrow k + 1$, $x^k \leftarrow x$, $s^k \leftarrow s$, $x \leftarrow x_B$, and $s \leftarrow s'$. Go to Step 1.

4. CONVERGENCE

The asymptotic convergence behavior of our method is similar to that of our modified Newton method [10] from which it inherits its coordinate system and searches, and our unconstrained method [9] from which it obtains its second derivative matrix approximation approach. Proofs for the theorems below are thus obtained by extending results from [9] in the manner of [10], so that we omit the complete proofs and indicate only the key points used.

THEOREM 1: (First order convergence.) If f is continuously differentiable on an open convex set containing the bounded set $\{x | A^T x \geqslant b,\ f(x) \leqslant f(x^0)\}$, then the method converges to a point satisfying the Karush-Kuhn-Tucker first order necessary optimality conditions.

PROOF: Since $f(x)$ is bounded from below, we must have that $\{s^k\}$ is infinite, unless the sequence $\{x^k\}$ is finite. If so, we can show that the last point found satisfies the optimality conditions. Now assume $\{s^k\}$ is infinite. Then $\{s^k\} \rightarrow 0$, and, by the acceptability tests at Steps 6, 10, and 11, we have that the gradient norm is bounded above by a function of the stepsize, so that the projected gradient norm also goes to zero.

The next two results all require that f be twice continuously differentiable, $\{x^k\}$ converge to a unique point $\bar{x}$, at which strictly complementary slackness holds, and that there is some finite bound on the norm of the largest element of the true projected Hessian at $\bar{x}$, which we denote $\bar{H}$.

THEOREM 2: (Second order convergence). If, in addition to the above, $\nabla^2 f(\cdot)$ satisfies a Lipshitz condition in a neighborhood of $\bar{x}$, and τ is chosen small enough, $\bar{x}$ satisfies the second order necessary optimality condition that $\bar{H}$ be positive semidefinite.

PROOF: This result follows from the explicit updating we do of G. If $\bar{H}$ is not positive semidefinite, then it has at least one strictly negative eigenvalue. Since $\{x^k\} \rightarrow \bar{x}$, and, by our updating, $\{G^k\} \rightarrow \bar{H}$, eventually our curve search direction will be sufficiently close to the eigenvector direction associated with that negative eigenvalue, and, if τ is small enough, the search at Step 9 will be able to be successful infinitely often. Since τ is bounded away from zero, this would imply that $\{f(x^k)\} \rightarrow -\infty$, so that $\bar{H}$ must be positive semidefinite. The need for a "sufficiently small τ" is due to the fact that the upper limit of the interval over which t^* may be chosen to satisfy the acceptability test at Step 9 is a function of the magnitude of the most negative eigenvalue of $\bar{H}$, and τ must be chosen smaller than that upper limit for the proof to follow.

THEOREM 3: If, in addition to the above, $\bar{H}$ is sufficiently positive definite in a feasible neighborhood of $\bar{x}$, the convergence rate of the algorithm is superlinear. If a Lipshitz condition also holds, order 2 convergence holds.

PROOF: These two results follow from making our approximated Newton line search direction computed at Step 8 be sufficiently close to the true direction which would be computed using exact derivatives. The stepsize $t = 1$ will then be feasible and acceptable, and, by McCormick and Ritter [13, Theorem 2], the rates claimed follow.

5. MODIFYING THE SECOND DERIVATIVE MATRIX

The algorithm maintains an approximation, G, to the second derivative matrix projected onto the coordinates defined by the last $(n - u)$ rows of Q. M and D factors can be maintained in the lower triangle of G, and, since G is symmetric, it is stored explicitly in the upper triangle of the same array. G can thus be used to aid in adjusting the matrix when the active set changes. Two cases must be considered—dropping a constraint from the active set, and adding one.

Dropping a constraint from the active set removes a column from R and a row from N^+, and adds a row and column to G. An identity column is added to G, requiring no arithmetic. (While this may tend to imbalance G, since it will intermix a direction for which no information is available with those for which much better approximations are known, we compensate by making the new column the first candidate for explicit updating.) The amount of work involved in updating Q and R will depend on the number of constraints deleted and their relative positions in the active set. Removing a single constraint would require from 0 (last constraint dropped) to $3un + \frac{3}{2}u^2 + 0(n)$ multiplications, $3un + \frac{3}{2}u^2 + 0(n)$ additions, and $(u - 1)$ square roots (first constraint dropped) to update R, and an additional $3[(u - 2)(u - 1)/2] + (u - 1)0(1)$ multiplications/additions for updating Q, using the arrangement in [5].

Adding a constraint to the active set adds a column to R and a row to N^+, deflates G from $u \times u$ to $(u - 1) \times (u - 1)$, and modifies Q. n multiplications and n additions are required to determine the new column of R, and then an $(n - u)$ vector must be reduced using orthogonal matrices. Using straightforward multiplication, the $(n - u - 1)2 \times 2$ Givens' matrices each require one square root, 4 multiplications and one addition. Premultiplication of Q involves $3n(n - u) - 3n + 0(n - u)$ multiplications and additions. Updating of G is simplified by using those transformations that will zero out all but the last entry in the vector, as in Gill and Murray [7]. We would like to determine triangular factors, LL^T, for G, since then premultiplication by the successive Givens' transformations would change L to lower Hessenberg form, and postmultiplication would restore triangularity.

Now G is recurred as MDM^T, with D consisting of 1×1 and 2×2 blocks, and difficulties arise when a 1×1 block is nonpositive or whenever a 2×2 block exists, that is, whenever there is a nonpositive eigenvalue. Fortunately, a fix up of G to be positive definite, so as to have LL^T factors, is easily done, since whenever G is not positive definite at this step, the complete eigensystem of D has already been determined during the previous execution of Step 7. Note that this modification of G would have to be performed once, before each set of constraints is added, and it will tend to modify the negative curvature information we try to maintain. Again, a least-constrained approach should tend to minimize such modifications.

The method used follows ideas in Sorenson [17], and is an attempt to replace G with the "nearest" positive definite matrix, i.e., $G = MDM^T$ is replaced with $\bar{G} = M(D + ZEZ^T)M^T$. The exact effect on the second derivative information in G is not clear, but, since we were not

able to derive any scheme for preventing the indefiniteness of G which would be more efficient than an entire refactorization, we opted for the fix-up approach. Note that this fix up would be exactly what we would have obtained if we had used the Bunch-Parlett factorization and forced the projected Hessian approximation to be positive definite.

The fix-up of G and determination of the LL^T factor requires between $\frac{1}{2}(n-u)(n-u-1)$ multiplications, no additions, and $(n-u)$ square roots in the case of all 1×1 blocks, to $(n-u)^2 + 4(n-u)$ multiplications, $(n-u)^2 + (n-u-6)$ additions, and $(n-u)$ square roots in the case of all 2×2 blocks. The subsequent straightforward multiplication of the lower triangular factor and its postmultiplication to regain triangularity involves $4(n-u)^2 - (n-u+2)$ multiplications and $2(n-u)^2 - (n-u)$ additions. It should be noted that the reduction of the $(n-u)$ vector

$$v = \begin{bmatrix} v_1 \\ v_2 \\ . \\ . \\ . \\ v_{n-u} \end{bmatrix}$$

to $||v||e_{n-u}$ by successive premultiplication by $(n-u)$ 2×2 Givens' matrices of the form

$$p_i^{i+1} = \begin{bmatrix} c_i & s_i \\ s_i & c_i \end{bmatrix}$$

could be replaced with a single multiplication by the matrix

$$P_{n-u-1}^{n-u}\, P_{n-u-2}^{n-u-1} \cdots P_2^3 P_1^2 = \begin{vmatrix} c_1 & s_1 & 0 & 0 & 0 & \cdots \\ s_1c_2 & -c_1c_2 & s_2 & 0 & 0 & \cdots \\ s_1s_2c_3 & -c_1s_2c_3 & -c_2c_3 & s_3 & 0 & \cdots \\ s_1s_2s_3c_4 & -c_1s_2s_3c_4 & -c_2s_3c_4 & -c_3c_4 & s_4 & \cdots \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ . & . & . & . & . & . \end{vmatrix}$$

which is a special lower Hessenberg matrix in the sense of [5]. Thus, savings of approximately 25% of the work involved in premultiplying the lower triangular factor of G could be saved by using Lemma IV of Section 2 of [5].

If constraints have been added to the active set, N^+ increases. Due to the complexity of updating N^+, especially since several constraints are often added at the same time, N^+ is recomputed at such times. Recomputation of N^+ requires $\frac{1}{2}u^2n$ multiplications and $\frac{1}{2}u(u-1)n$ additions.

6. COMPUTATIONAL RESULTS

The essential motivating factor in this research was our feeling that a partial explicit update of the second derivative matrix might prove to be superior to the usual quasi-Newton strategy. We coded our method to see whether numerical experimentation would be consistent with that conjecture. (A copy of that code may be obtained from the author.)

There are actually two different major hypotheses to test here. First, we wanted to see if varying π, the number of rows/columns in the update "pipe," could improve convergence behavior. We tested this using different values of π in our new code, and by comparison with our older modified Newton method, thus holding the algorithmic structure constant. Second, we needed to see whether our new method is competitive with other current software. For this comparison, we used a state-of-the-art quasi-Newton method, Murtagh and Saunder's MINOS [15] algorithm. On a secondary level, we were curious as to how the inclusion of a low rank update might affect the results, so we also ran the test problems using Broyden's rank-1 update and the complementary DFP rank-2 update. Note that the latter forces positive definiteness on G.

As with similar numerical experimentation with optimization codes, our results cannot be considered an absolute conclusion for either hypothesis. The test results using our methods are affected by the parameter settings, particularly the starting stepsize. While we tried to find good values for each instance, other parameter settings might have been optimal. The path our algorithm took to the solution may differ with π, so that the results may be an artifact of the starting point. Finally, we used the default parameters in MINOS for runs through it. A more experienced MINOS user might have been able to speed its convergence by setting its various parameters differently. Note that our stopping criterion, since it is a second order one, is more stringent than that of MINOS.

These caveats aside, our experiments do tend to support our conjectures. Table 1 summarizes the number of function evaluations used until convergence (i.e., agreement with the optimal f value to 10 decimal places) for the *best* configuration of our new method, our modified Newton method, and MINOS for six popular test problems. The results are

TABLE 1 — *Comparative Function Evaluation Counts*

Problem and Source	Best Form of the New Method			Our Modified Newton Method	MINOS
	π	Update	Evaluations		
Gauthier [3] (Colville #7)	3	Rank 2	111	187	496
Hydrazine Equilibrium [4]	7	None	213	223	561
Rosenbrock [16]	2	None	109 [9]	97	192
Shell Primal [3] (Colville #1)	0	None	44	50	60
Water Quality [10]	2	None	271	480	2,292
Wood [3] (Colville #4)	3	None	245[9]	447	585

particularly striking for the Water Quality problem, which is a nonconvex 11 variable model derived from a model of the Willmette River. The new method performed about the same as our modified Newton one on Hydrazine Equilibrium, Rosenbrock's banana function, and Shell Primal, and did much better on the other three.

To further see the effect of modifying π, Figure 1 shows, graphically, the number of function evaluations required for convergence on the four linearly constrained problems. Shell Primal is a relatively easy cubic problem, and Figure 1A shows that not much benefit results from either increasing π or adding a low rank update. The best performance occurred for $\pi = 0$ and no update; since we start with an identity matrix for G *this is nothing more than steepest descent.*

Gauthier's problem is a 16 variable quartic. In Figure 1B the convex pattern with respect to π that was so marked for unconstrained problems [9] appears for the "no update" and "rank-2 update" curves. That is, an intermediate value of π is superior to either the modified Newton ($\pi = n$) or quasi-Newton ($\pi = 0$) strategy. The pattern for the rank-1 update is different, confounded in part by local variations moves. Since the method moves to the better of the points found by line (or curve) search or points used for derivative approximations, fortuitous values of the stepsize and coordinate system sometimes lead to fast convergence.

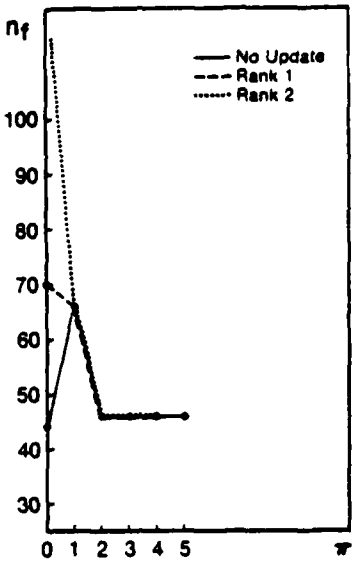
Hydrazine equilibrium is a tightly constrained 10 variable problem. The feasible region is small, and the method spends much time changing constraint sets (the optimum is strictly interior to the inequality constraints). Again, particularly lucky searches on local variation moves had an impact on the convergence. The pattern of Figure 1C is a bit confusing, although the modified Newton-like strategy of larger π seems to be desirable.

Finally, the performance of the Water Quality problem is illustrated in Figure 1D. Since a rank-1 or rank-2 update cannot be performed unless the same constraint set is used for two consecutive iterations, the constant changes of constraint sets meant that such updates were never performed. Here the "U" shaped patterns appear again, lending support to the idea that some explicit updating is better than none, but that a complete update is usually not worthwhile. This is also a rather hard problem; note that MINOS required 2,292 function evaluations to converge.

Table 2 shows the relative percentage of time used by each part of our algorithm. Our code is an experimental one, written by the author, so that a more skillful implementation might redistribute these somewhat. Test runs were performed on the University of Pittsburgh's DEC 1099 in the usual batch environment. In this situation, times tend to vary by $\pm 15\%$. The standardized times reported in the last two columns were computed relative to the 28.48 seconds required for Colville's [3] timing routine (average of 3 runs).

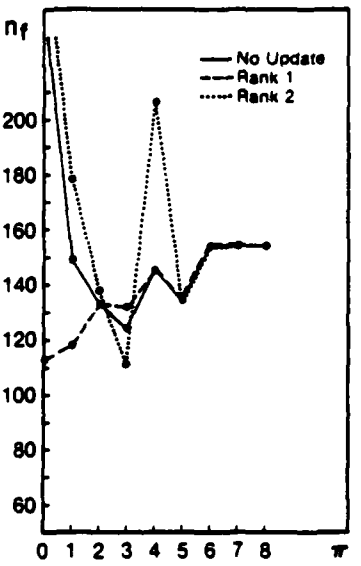
The bulk of the time required by the new method was for the numerical derivative approximations. This was reassuring, since a basic idea in using a nonderivative method, such as ours, *is to trade off increased computer time for the human time necessary to compute the derivatives analytically.*

Finally, the standardized time ratio for our new method compares favorably with that of MINOS. The percentage reduction in time for our method, relative to MINOS, ranges from 37-70% for the constrained problems and is about 90% for the unconstrained ones. Considered together with the additional time necessary to derive and program the derivatives for a code such as MINOS, we can conservatively conclude that our new method is at least competitive with a representative, state-of-the-art, quasi-Newton algorithm.



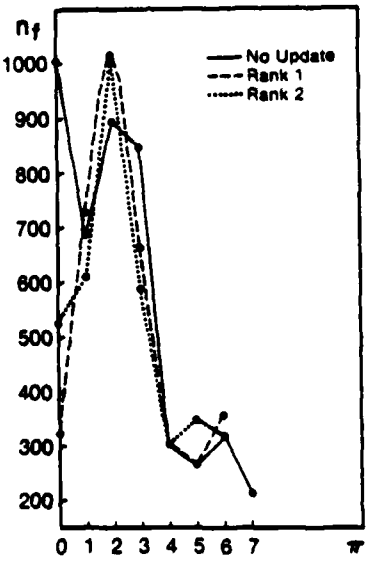
Shell Primal

FIGURE 1A.



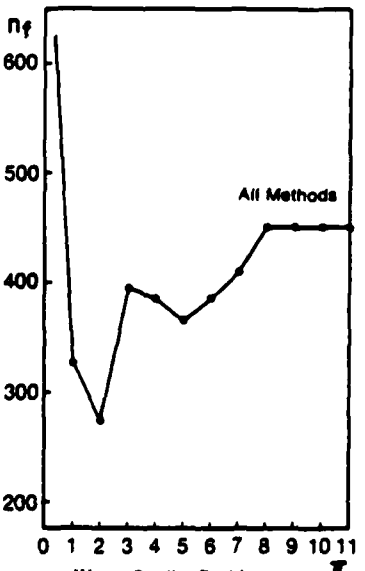
Gauthier's Problem

FIGURE 1B.



Hydrazine Equilibrium

FIGURE 1C.



Water Quality Problem

FIGURE 1D.

TABLE 2 — *Timing Results*
Time (sec.)

Problem	New Method					MINOS
	Setup; Finding Initial Feasible Point	Evaluations and Derivatives	Constraint Handling and Matrix Updating	Searches	Standardized Time Ratio (total)	Standardized Time Ratio (total)
Gauthier	28%	38%	17%	17%	0.0557	0.0885
Hydrazine	16%	75%	2%	6%	0.0331	0.0572
Rosenbrock	—	—	—	—	0.0015[19]	0.0330
Shell Primal	56%	28%	16%	<1%	0.0109	0.0372
Water Quality	14%	61%	14%	12%	0.1131	0.1921
Wood	—	—	—	—	0.0066[19]	0.0509

7. CONCLUSIONS

Our thesis, in this paper, is that explicitly updating a few columns of the approximated (projected) matrix of second partial derivatives at each iteration may yield convergence behavior superior to that of traditional quasi-Newton algorithms. We presented a method for linearly constrained optimization, incorporating that idea into our existing algorithmic framework. The user specifies the number of rows/column to be updated at each iteration. Theoretical convergence proofs show that such a method retains the properties of our modified Newton algorithm—convergence to a second-order Kuhn-Tucker point, and superlinear rate. Numerical experiments indicate that this approach is competitive with a state-of-the-art quasi-Newton code, and that a small number of explicitly updated rows/columns (usually 2 or 3) can significantly improve the speed of convergence.

REFERENCES

[1] Armijo, L., "Minimization of Functionals Having Lipschitz Continuous First Partial Derivatives," Pacific Journal of Mathematics, *16*, 1-3 (1966).
[2] Bunch, J.R. and B.N. Parlett, "Direct Methods for Solving Symmetric Indefinite Systems of Linear Equations," SIAM Journal on Numerical Analysis, *8*, 639-655 (1971).
[3] Colville, A.R., "A Comparative Study on Nonlinear Programming Codes," IBM New York Scientific Center Technical Report No. 320-2949 (June 1968).
[4] Dantzig, G., S. Johnson and W. White, "A Linear Programming Approach to the Chemical Equilibrium Problem," Management Science, *5*, 38-43 (1958).
[5] Gill, P.E., G.H. Golub, W. Murray, and M.A. Saunders, "Methods for Modifying Matrix Factorizations," Mathematics of Computation, *28*, 503-535 (1974).
[6] Gill, P.E. and W. Murray, "Newton Type Methods for Linearly Constrained Optimization," in *Numerical Methods for Constrained Optimization*, W. Murray and P.E. Gill, Editors (Academic Press, New York, 1974).
[7] Gill, P.E. and W. Murray, "Quasi-Newton Methods for Linearly Constrained Optimization," in *Numerical Methods for Constrained Optimization*, W. Murray and P.E. Gill, Editors (Academic Press, New York, 1974).
[8] Lenard, M., "A Computational Study of Active Set Strategies in Nonlinear Programming with Linear Constraints," Technical Report No. 1564, Mathematics Research Center, University of Wisconsin, Madison, WI, November (1975).
[9] May, J.H., "Pseudo-Newton Method for Minimization," University of Pittsburgh, Graduate School of Business, Pittsburg, PA, Working Paper No. 306 (August 1978).

[10] May, J.H., "Solving Nonlinear Programs Without Using Analytic Derivatives," Operations Research, *27*, 457-484 (1979).
[11] May, J.H., A.D. Shocker and D. Sudharshan, "Optimal Product Location: A Simulation Comparison of Methods," University of Pittsburgh, Graduate School of Business, Pittsburgh, PA, Working Paper No. 466, October 1981.
[12] McCormick, G.P., "A Modification of Armijo's Step-Size Rule for Negative Curvature," Mathematical Programming, *13*, 111-115 (1977).
[13] McCormick, G.P. and K. Ritter, "Methods of Conjugate Directions Versus Quasi-Newton Methods," Mathematical Programming, *3*, 101-116 (1972).
[14] Moré, J.J. and D.C. Sorenson, "On the Use of Direction of Negative Curvature in a Modified Newton Method," Mathematical Programming, *16*, 1-20 (1979).
[15] Murtagh, B.A. and M.A. Saunders, "Large-scale Linearly Constrained Optimization," Mathematical Programming, *14*, 41-72 (1978).
[16] Rosenbrock, H.H., "An Automatic Method for Finding the Greatest or Least Value of a Function," Computer Journal, *3*, 175-184 (1960).
[17] Sorenson, D.C., "Updating the Symmetric Indefinite Factorization with Applications in a Modified Newton's Method," Argonne National Laboratory, Report ANL-77-49 (June 1977).

INTERACTIVE SOLUTION OF BI-CRITERIA MATHEMATICAL PROGRAMS*

S. Sadagopan

Assistant Professor of Industrial Engineering
Indian Institute of Technology
Kanpur, India

A. Ravindran

Director, School of Industrial Engineering
University of Oklahoma
Norman, Oklahoma

ABSTRACT

In this paper, we develop efficient interactive methods for the solution of bicriteria nonlinear programming problems. The methods do not require trade-off information from the decision maker, pose less cognitive burden and converge to the "best compromise solution" fast. Two methods, called the paired comparison method and comparative trade-off method, are presented with examples. A real application of the interactive method to a bicriteria problem that arose in the planning of the cardiovascular disease control program in the U.S. Air Force is also presented.

INTRODUCTION

Bicriteria mathematical programming (BCMP) problems as a first generalization of single criterion nonlinear programs have been studied by many researchers [1, 2, 3, 4, 7, 13, 17]. The small dimensionality of the problem permits a visual presentation of the "payoff set," which partly explains such widespread interest. Further, when we restrict ourselves to "efficient solutions," the complementary roles of the two criteria yields additional computational power.

The BCMP problem may be stated as follows:

$$\text{(1)} \qquad \begin{array}{ll} \text{VMAX} & \{f_1(x), f_2(x)\} \\ \text{Subject to} & g_i(x) \leqslant 0 \quad i = 1, \ldots, m \end{array}$$

where x is an n dimensional vector of *decision variables*, f_1 and f_2 are real valued *criterion* functions and g_i's represent a set of nonlinear constraints. Let $S = \{x \mid g_i(x) \leqslant 0\}$ denote the feasible region. An optimal solution of BCMP will be taken to be a "best compromise solution" that maximizes the preferences of a Decision Maker (DM); that is, a feasible solution that solves the following program.

*Research funded under Contract F33615-78-D-0617 for the U.S. Air Force, School of Aerospace Medicine.

$$\text{(2)} \qquad \begin{array}{ll} \text{Max} & U[f_1(x), f_2(x)] \\ \text{Subject to} & x \in S \end{array}$$

where U is the DM's preference function or value function defined over the criterion values (f_1, f_2) such that for any two alternatives x_1 and x_2, $U[f_1(x_1), f_2(x_1)] > U[f_1(x_2), f_2(x_2)]$, if the DM prefers x_1 to x_2.

In one of the earlier papers addressed to the bicriteria problem, Geoffrion [7] considered the "scalar maximization approach," i.e., maximizing a convex combination of the two criteria. He showed how such problems can be numerically solved using parametric programming algorithms. Bacopoulas and Singer [2] on the other hand, used the "constrained criteria approach," i.e., maximizing one criterion keeping the other criterion as a constraint. They showed how all the efficient solutions can be generated by parametrically varying the level of the constrained criterion over a particular interval. Some further refinements and alternate proofs appear in Gearhart [6] and Benson [3]. Pasternak and Passy [13], addressed the special case when the variables are zero-one and devised a special enumerative algorithm for generating the efficient solutions. Choo and Atkins [4] address the case when the criteria are linear fractional functions. Adulbhan [1] considers the special case when the criteria are linear. In all these methods, the emphasis is on generating *all the efficient solutions* of BCMP. But Walker [17] considers the "interactive" approach and uses the Generalized Lagrange Multiplier method of Everett [5]. Walker's method is slow in convergence and seeks difficult interaction from the DM.

In this paper we will consider *interactive approaches* to the solution of BCMP where the preference function U is only *implicitly* known. Such an approach in the more general context of multiple criteria problems have been addressed by Geoffrion, Dyer and Feinberg [8] and Zionts [19]. However, Wallenius [18] has observed that such methods pose considerable cognitive burden on the part of the DM; also the procedures converge to the optimal solution rather slowly. In this paper, we develop interactive procedures that pose less cognitive burden to the DM and converge to the optimal solution fast.

Section 2 presents the principal results that form the basis of the interactive methods developed for BCMP. Section 3 discusses one of the interactive approaches called the *Paired Comparison Method.* Section 4 discusses another approach called the *Comparative Trade off Method.* Section 5 discusses a real application of the interactive method to a bicriteria problem that arose in the control of cardiovascular disease in the U.S. Air Force.

2. PRINCIPAL RESULTS

In this section, we will establish a number of results that form the basis of the new interactive procedures which are develolped in the later sections. We will assume throughout that the feasible region S is a compact convex set; the criterion functions f_1 and f_2 are differentiable concave functions and the preference function U is differentiable, increasing and strongly quasiconcave defined in the following manner.

DEFINITION 1: $f: S \Longrightarrow R^1$, where S is a nonempty convex set in R^n is said to be strongly quasiconcave on S, if for each $x_1, x_2 \in S$ with $x_1 \neq x_2$ and for every $\lambda \in (0, 1)$, we have

$$\text{(3)} \qquad f[\lambda x_1 + (1 - \lambda)x_2] > \text{Min } [f(x_1), f(x_2)].$$

The convexity assumptions on criteria and feasible region are made only to allow the methods of convex programming to be applicable. The quasiconcavity assumption of U is well supported by preference theory (see Intriligator [10]).

Use of the concavity assumptions guarantee the attainment of maxima and quasiconcavity ensures us that local optima are also global optima. In addition it is sufficient to restrict ourselves to efficient solutions defined below as candidates for the optima.

DEFINITION 2: A solution $x^0 \in S$ is said to be efficient to BCMP, if $f_1(x) > f_1(x^0)$ for some $x \in S \Longrightarrow f_2(x^0) > f_2(x)$.

We will now develop procedures for generating the efficient solutions and then a method of "efficiently" searching among the efficient solutions for the "optimal" solution.

(4) Define the payoff set $Y = \{y | f(x) = y \text{ for } x \in S\}$.

(5) Let $A = \{y | f(x) \geqslant y \text{ for } x \in S\}$. Obviously, $Y \subset A$.

LEMMA 1: If S is a convex set and f is concave, then A will be a convex set [12].

Consider the following single objective nonlinear programs:

$$\begin{array}{ll} \text{P1: Max } f_1(x) & \text{P2: Max } f_2(x) \\ \text{Subject to: } x \in S & \text{Subject to: } x \in S \end{array}$$

Let the optimal values of (P1) and (P2) be v^* and w^* respectively. Now consider the following mathematical programs:

$$\begin{array}{ll} P_v\text{: Max } f_2(x) & Q_w\text{: Max } f_1(x) \\ \text{Subject to: } x \in S & \text{Subject to: } x \in S \\ \quad f_1(x) \geqslant v & \quad f_2(x) \geqslant w \end{array}$$

Let x_2^* solve Q_w for $w = w^*$. Then $f_1(x_2^*)$ is the minimum achievable value for f_1 without sacrificing any achievement on f_2, while v^* is the maximum value of f_1 at the expense of f_2. Hence, the range of achievable values for f_1, denoted by $\bar{v}$, is given by $[f_1(x_2^*), v^*]$ and the best compromise value for f_1 lies in this range.

THEOREM 1: In the optimal solution to the program $P_{\bar{v}}$ where $\bar{v} \in [f_1(x_2^*), v^*]$, the constraint $f_1(x) \geqslant \bar{v}$ will be a binding constraint, i.e., if $\bar{x}$ solves $P_{\bar{v}}$ then $f_1(\bar{x}) = \bar{v}$.

PROOF: Assume the contrary, i.e.,

(6) $$f_1(\bar{x}) > \bar{v} \geqslant f_1(x_2^*)$$

w^* being the absolute maximum of f_2 over S,

(7) $$w^* = f_2(x_2^*) \geqslant f_2(\bar{x})$$

we will consider the two cases

(8) (i) Let $f_2(x_2^*) = f_2(\bar{x})$

(6) and (7) taken together contradict the fact that x_2^* solves Q_w^*.

(9) (ii) Let $f_2(x_2^*) > f_2(\bar{x})$

consider the line segment $[\bar{x}, x_2^*]$ which lies in the convex set S. Over this line segment, the concave functions f_1 and f_2 are unimodal. Because of "strict" inequality in (6) and (9), there exists a feasible solution x' on the line segment $[\bar{x}, x_2^*]$ such that

$$f_1(\bar{x}) > f_1(x') > \bar{v} \tag{10}$$
$$f_2(\bar{x}) < f_2(x') < f_2(x_2^*).$$

In equality (10) indicates that x' is a feasible solution to $P_{\bar{v}}$ with $f_2(x') > f_2(\bar{x})$ contradicting the optimality of $\bar{x}$.

Hence, $f_1(\bar{x}) = \bar{v}$.

THEOREM 2: $\bar{x} \in S$ is efficient if and only if $\bar{x}$ solves $P_{\bar{v}}$ where $\bar{v} \in [f_1(x_2^*), v^*]$.

PROOF:

Sufficiency: Let $\bar{x}$ solve $P_{\bar{v}}$ where $\bar{v} \in [f_1(x_2^*), v^*]$. We will show that $\bar{x}$ is efficient. Consider an $x \in S$ such that

$$f_1(x) > f_1(\bar{x}) = \bar{v}.$$

It is clear that x is feasible to $P_{\bar{v}}$. But x cannot be optimal to $P_{\bar{v}}$; as per theorem 1 every optimal solution x^0 satifies $f_1(x^0) = \bar{v}$. Hence, $f_2(x) < f_2(\bar{x})$. Consider an $x \in S$ such that

$$f_2(x) > f_2(\bar{x}).$$

Since $\bar{x}$ is optimal to $P_{\bar{v}}$, x cannot be feasible to $P_{\bar{v}}$. Hence, $f_1(x) < f_1(\bar{x})$. Consequently, $\bar{x}$ must be efficient.

Necessity: Let E be the set of all efficient solutions. Suppose $\bar{x} \in E$. Obviously $f_1(\bar{x}) \geq f_1(x_2^*)$ as, otherwise, x_2^* will dominate $\bar{x}$. Let $f_1(\bar{x}) = \bar{v}$. Assume that $\bar{x}$ does not solve $P_{\bar{v}}$ but some other $x' \in S$ solves $P_{\bar{v}}$. By theorem 1, $f_1(x') = \bar{v} = f_1(\bar{x})$ since x' is optimal for $P_{\bar{v}}$, $f_2(x') \geq f_2(\bar{x})$; but $\bar{x} \in E \Longrightarrow f_2(x') \leq f_2(\bar{x})$ and hence $f_2(x') = f_2(\bar{x})$. Consequently, $\bar{x}$ solves $P_{\bar{v}}$.

Theorem 2 enables us to generate the entire set of efficient solutions by parametrically solving $P_{\bar{v}}$. By theorem 1, the generated solutions will have specific levels of attainment of f_1, namely $\bar{v}$. The next theorem would give some unimodality property enabling us to make an efficient parametrization.

Consider the problem where we maximize U over the pay-off set Y, keeping f_1 fixed at v; i.e.,

$$\text{P3:} \quad \max_{(v,f_2) \in Y} \quad U(v,f_2)$$

where $(v,f_2) \in Y$ is a subset of Y defined by $\{(v,f_2) | f_2 = f_2(x)$ for $x \in S$ and $f_1(x) = v\}$

By Theorem 1 and the fact that U is increasing in f_2, it is easy to see that an optimal solution to (P3) is also an optimal solution to program (P_v) where $v \in [f_1(x_2^*), v^*]$. We now establish the unimodality of $g(v)$, defined below:

THEOREM 3: Let Y' be a convex subset of R^2 and $U(f_1, f_2)$ is a strongly quasiconcave increasing function of f_1 and f_2 defined over Y', then

$$g(v) = \underset{(v, f_2) \in Y'}{\text{Max}} U(v, f_2)$$

is strongly quasiconcave, i.e., unimodal in v.

PROOF: Let

$$g(v_3) = \underset{(v_3, f_2) \in Y'}{\text{Max}} U(v_3, f_2) = U(v_3, w_3) \text{ (say)}$$

$$g(v_4) = \underset{(v_4, f_2) \in Y'}{\text{Max}} U(v_4, f_2) = U(v_4, w_4) \text{ (say)}.$$

Assuming boundedness of functions the maxima are attained; by convexity of Y' and strong quasiconcavity of U they are attained uniquely.

Note that $(v_3, w_3) \in Y'$ and $(v_4, w_4) \in Y'$ by convexity of Y',

$$[\lambda v_3 + (1 - \lambda)v_4,\ \lambda w_3 + (1 - \lambda)w_4] \text{ also } \in Y, \text{ for } 0 \leqslant \lambda \leqslant 1.$$

$$g[v_3 + (1 - \lambda)v_4] = \underset{(\lambda v_3 + (1-\lambda)v_4, f_2) \in Y'}{\text{Max}} U[\lambda v_3 + (1 - \lambda)v_4, f_2]$$

$$\text{(By convexity of } Y') \geqslant U[\lambda v_3 + (1 - \lambda)v_4, \lambda w_3 + (1 - \lambda)w_4]$$

$$\text{(By strong quasi-} \quad > \text{Min } [U(v_3, w_3),\ U(v_4, w_4)]$$

$$\text{concavity of } U) \quad = \text{Min } [g(v_3),\ g(v_4)]$$

Hence, $g(v)$ is unimodal in v.

In the proof of the above theorem, we have heavily exploited the convexity of Y'. When we want to apply the result to Y, the pay-off set, we find that the assumption of convexity of pay-off set may be overly restrictive. For example our assumptions of convexity of S and concavity of f_i's are *not* sufficient to guarantee convexity of Y, as the following simple example would illustrate.

EXAMPLE: $S = x > 0$; $f = (f_1, f_2) = (x, -\log x)$ Obviously, S is a convex set and f_i's are concave. But the pay-off set Y is the graph of $-\log x$ which obviously is not a convex set [12].

However, we will presently establish the fact that we can extend the pay-off set Y to the set A defined in Equation (5) without affecting the results. Lemma 1 established the convexity of the set A. Hence identifying A with Y', the result of theorem 3 holds even when the pay-off set is nonconvex.

We will establish in the next theorem, the equivalence of solutions to the following two problems.

$$\text{P4: } \max_{(\alpha,f_2)\in Y} U(\alpha,f_2) \qquad \text{P5: } \max_{(\alpha,f_2)\in A} U(\alpha,f_2)$$

THEOREM 4: The optimal solutions to (P4) and (P5) are identical, i.e., if Y_m be the set of optimal solutions to (P4) and A_m be the set of optimal solutions to (P5), then $Y_m = A_m$.

PROOF:

(a) We will first prove that every element of A_m is an element of Y_m i.e., $A_m \supset Y_m$. Let $y^* = (y_1^*, y_2^*)$ maximize P5 for $\alpha = y_1^*$, i.e., $y^* \in A_m$. We have to show that $y^* \in Y$ and hence $y^* \in Y_m$. Since $A_m \subset A$, by definition $\exists$ and x^*, such that $f(x^*) = (f_1^*, f_2^*) \geqslant y^*$. Note that $y^* \in Y$ is $f(x^*) = y^*$. Assume that $f(x^*) > y^*$, i.e.,

(11) $$f_1^* \geqslant y_1^*,\ f_2^* > y_2^* \text{ (say)}.$$

Since $U(f_1,f_2)$ is increasing in f_1 and f_2, (11) gives

(12) $$U(y_1^*, y_2^*) < U(y_1^*, f_2^*).$$

But from the fact that y^* maximizes $U(y_1^*, f_2)$ over $(y_1^*, f_2) \in A$ we get

(13) $$U(y_1^*, y_2^*) \geqslant U(y_1^*, f_2^*).$$

(13) contradicts (12); hence our assumption (11) must be invalid and $f(x^*) = y^*$. By definition of the set Y, $y^* \in Y$. Being a maximizer of (P5) for $\alpha = y_1^*$, $y^* \in Y_m$ and hence $A_m \supset Y_m$.

(b) Assume $y^* \in Y_m$. We will show that $y^* \in A_m$ as well or $Y_m \subset A_m$. Since $Y_m \subset Y \subset A$, we get $y^* \in A$. Assume y^* is not optimal to (P5). Hence $\exists$ another $y' = (y_1^*, y_2')$ such that

$$U(y_1^*, y_2') = \max_{(y_1^*, f_2)\in A} U[y_1^*, f_2]$$

(14) $$U(y_1^*, y_2') \geqslant U(y_1^*, f_2) \text{ for any } (y_1^*, f_2) \in A$$

and in particular $\geqslant U(y_1^*, y_2^*)$.

Now $(y_1^*, y_2') \in A_m$; by part (a) of the proof we will have $(y_1^*, Y_2') \in Y$. By optimality of (y_1^*, y_2^*) for (P4)

(15) $$U(y_1^*, y_2') \leqslant U(y_1^*, y_2^*)$$

(14) and (15) imply that

$$U(y_1^*, y_2') = U(y_1^*, y_2^*)$$

or in other words (y_1^*, y_2^*) optimizes (P5) as well. Hence

$$y^* \in A_m \text{ or } Y_m \subset A_m.$$

THEOREM 5: With the assumptions mentioned earlier the following relationship is true:

$$\operatorname*{Max}_{(v,f_2)\in\Gamma} U(v,f_2) = g(v) = \operatorname{Max} U(v,f_2)$$

$$\text{Subject to: } f_1(x) \geqslant v$$

$$x \in S$$

PROOF: Follows directly from Theorems 3 and 4.

3. PAIRED COMPARISON METHOD

Using the results of Section 2 we will develop a procedure needing only a *paired comparison* from the DM, i.e., given any two feasible solutions and their outcomes, say $y^{(1)}$ and $y^{(2)}$ the DM only has to specify whether

(16) $\qquad y^{(1)} \succ y^{(2)}$ or $y^{(2)} \succ y^{(1)}$ or both

where "$\succ$" denotes "preferred to."

Interaction of this form is presumably much less demanding than asking the DM to specify his local tradeoffs. We will presently indicate how (16) can be used to eliminate a portion of the efficient set and progressively converge to the "best compromise solution."

Based on the theorems given in Section 2, the BCMP problem has been reduced to determining the maximum of $g(v)$ where v belongs to the interval $[v_l, v^*]$ where $v_l = f_1(x_2^*)$. However, $g(v)$ is not known explicitly since U is not known. But, using a search technique which requires only *functional comparison* and not function values, we can still solve the BCMP problem, using the following region elimination concept:

For a unimodal function $g(v)$ defined over a finite interval (v_l, v^*), let v_A and v_B be two points in the interval such that $v_A < v_B$. Then, $g(v_A) < g(v_B)$ implies that the maximum of $g(v)$ will not lie in the interval (v_l, v_A). On the other hand, $g(v_A) > g(v_B)$ implies that the maximum will not lie in the interval (v_B, v^*).

General Steps of the Algorithm

Step 1: Solve P1: Max $f_1(x)$, subject to $x \in S$.
Set max $f_1(x) = v^*$

Solve P2: Max $f_2(x)$, subject to $x \in S$.
Set max $f_2(x) = w^*$

Step 2: Solve Q_u^*: Max $f_1(x)$, subject to $x \in S$ and $f_2(x) \geqslant w^*$.
Set the maximum of f_1 v_l. Now the optimal value of f_1 to the BCMP lies between v_l and v^*.

Step 3: Choose two values, v_A and v_B, such that $v_l < v_A < v_B < v^*$. Solve the problem P_v: Maximize $f_2(x)$, subject to $x \in S$ and $f_1(x) \geqslant v$ for $v = v_A$ and $v = v_B$; let $g(v) = \operatorname{Max} f_2(x)$ for P_v.

Step 4: Let $y^{(1)} = [v_A, g(v_A)]$, $y^{(2)} = [v_B, g(v_B)]$. Given $y^{(1)}$ and $y^{(2)}$, the DM is asked to specify whether $y^{(1)} \succ$ (preferred to) $y^{(2)}$, or $y^{(2)} \succ y^{(1)}$, or indifferent. Using the DM's response, a portion of the efficient set can be eliminated.

If $y^{(1)} \succ y^{(2)}$, then set $v_u = v_B$; go to step 5.

If $y^{(2)} \succ y^{(1)}$, then set $v_l = v_A$; go to step 5.

Step 5: If $|v_l - v_u| < \epsilon$ (chosen small number) stop; otherwise return to step 3.

Figure 1 illustrates the paired comparison method. Here, $y^{(2)} \succ y^{(1)}$ since $U(y^{(2)}) > U(y^{(1)})$. Hence, the maximum of U cannot lie in the interval $[v_l, v_A]$. Eliminating the region (v_l, v_A), the interval of uncertainty where the optimum lies reduces to (v_A, v^*). If Golden Section Section Search is used to generate the two new points between v_A and v^* one of the new points will turn out to be v_B. With Golden Section Search, we will be able to bracket the best compromise solution y^0 to less than 10% of the original interval on v with just 5 paired comparisons from the DM; in 10 paired comparisons, the optimal solution will be within 1% of the original interval of uncertainty $[v_l, v^*]$. Thus, for a specified level of uncertainty we have a finite, rapidly convergent procedure using only paired comparison of two efficient solutions.

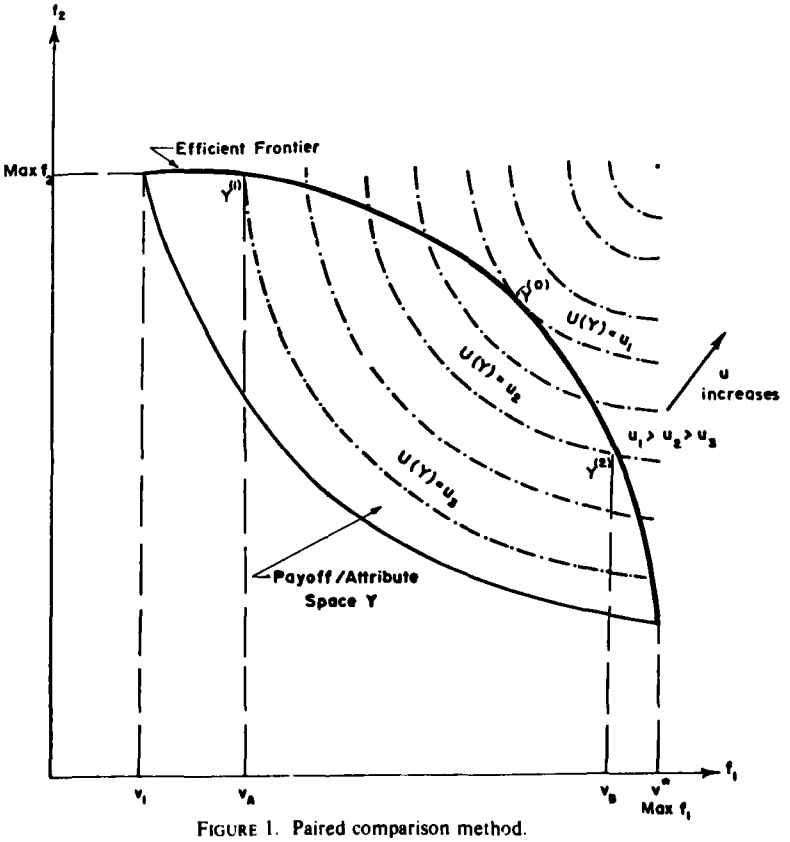


FIGURE 1. Paired comparison method.

Numerical Example

Consider the following bicriteria program [7] with

$$f_1(x) = 32 \qquad -40\,x_2 + 23\,x_3 - 7\,x_4$$
$$f_2(x) = 32 - 10\,x_1 \qquad -4\,x_2 + 7\,x_3 - 7\,x_4$$

and the feasible region S is specified by the following linear equality constraints in detached coefficient form in addition to the usual nonnegativity restrictions:

x_1	x_2	x_3	x_4	x_5	x_6	b
1	0	−23/50	27/50	0	0	132/50
0	1	−65/100	35/100	0	0	60/100
0	0	1/25	1/25	1	0	16/25
0	0	15/50	−35/50	0	1	40/50

In order to simulate the interaction process, we will assume that the DM's implicit preference function (which guides him in the selection process) is the following quasiconcave function

$$U(f_1,f_2) = f_1^{2/3} f_2.$$

The true optimal solution is $x^* = (x_1,x_2,x_3,x_4,x_5,x_6)$

$= (1.2857,\ 0,\ 0.7895,\ 3.1805,\ 0.4812,\ 2.7895)$ with $f^* = (f_1^*,\ f_2^*)$

$= (27.895,\ 2.406)$ and $U(f_1^*,\ f_2^*) = 22.13.$

Solution Using Paired Comparison Method

Step 1: Solving (P1) and (P2) we get $v^* = 60$ and $w^* = 3.2$

Hence $v_u = 60$

Step 2: Solving (Q_w^*) yields $v_l = 8$

Iteration 1

Step 3: Using Golden Section ratios 0.618 and 0.382, $v_A = 27.864$ and $v_B = 40.136$. Solve the problem P_v for $v = v_A$ and v_B. $g(v_A) = 2.407$ and $g(v_B) = 1.707$

Step 4: Interact with the DM and seek his paired comparison between $y^{(1)} = (27.864,\ 2.407)$ and $y^{(2)} = (40.136,\ 1.707)$. Assuming that DM is guided by the implicit preference function $U(f_1,f_2) = f_1^{2/3} f_2$, $U[y^{(1)}] = 22.12$ and $U[y^{(2)}] = 20.01$. By the property of the preference function, $y^{(1)} \succ y^{(2)}$ if $U[y^{(1)}] > U[y^{(2)}]$. Hence, $y^{(1)}$ would be preferred to $y^{(2)}$. Consequently, v_u will be updated to $v_B = 40.136$, thereby eliminating a portion of the efficient frontier.

Step 5: Assuming that the interval of uncertainty is to be reduced to 10%, the search has to be continued further on the reduced efficient frontier. For brevity of presentation the results of further calculations are summarized in Table 1.

TABLE 1 — *Solution of Bicriteria Problem Using Paired Comparison Method*

Iteration #	v_l	v_u	v_A	v_B	$y^{(1)}$	$y^{(2)}$	$U[y^{(1)}]$	$U[y^{(2)}]$	Remaining Interval (% of Original)
0	8	60	–	–	–	–	–	–	100
1	8	40.136	27.864	40.136	(27.864, 2.407)	(40.136, 1.707)	22.12	20.01	61.8
2	20.276	40.136	20.276	27.864	(20.276, 2.841)	(27.864, 2.407)	21.12	22.12	38.2
3	20.276	32.549	27.864	32.549	(27.864, 2.407)	(32.549, 2.140)	22.12	21.82	23.6
4	24.964	32.549	24.964	27.864	(24.964, 2.573)	(27.864, 2.407)	21.97	22.12	14.6
5	24.964	29.652	27.864	29.652	(27.864, 2.407)	(29.652, 2.306)	22.12	22.09	9.0

4. COMPARATIVE TRADEOFF METHOD

The local trade-off at a feasible point $x \in S$, between criteria 2 and 1 is defined to be

$$T_{21} = \left(\frac{\partial U}{\partial f_1} \Big/ \frac{\partial U}{\partial f_2} \right) \tag{17}$$

evaluated at x.

T_{21} measures the loss in criterion f_2 from the current level the DM is prepared to trade-off for unit gain in criterion f_1.

If the DM can provide local trade-offs then the problem can be solved interactively using Geoffrion-Dyer-Feinberg [8] type approaches. However, it has been repeatedly observed in practice that precise tradeoffs are difficult to provide. However, the DM can provide "imprecise" tradeoffs much more easily, e.g., "interval estimates" in place of "point estimates." For the bicriteria case, an estimate of the local tradeoff that is easier than the interval estimate is the "comparative estimate." The DM is provided with a number and will be asked to respond whether or not he would be prepared to tradeoff more than, less than, or equal to the specified number in criterion f_2 for a unit gain in criterion f_1. Based on the DM's response, a portion of the efficient frontier can be eliminated. We will presently indicate how the results of Section 2 can be used to provide the basis for the "comparative trade-off method."

At any feasible point $x \in S$, let λ_{21} be the perturbation in f_2 from its current level per unit perturbation in f_1. Limiting ourselves only to efficient solutions, we find that Theorem 2 provides us a method of generating efficient solutions; also Theorem 1 specifies that the constraint $f_1(x) \geqslant \bar{v}$ in program $P_{\bar{v}}$ will be binding in every optimal solution. Hence, λ_{21} is obtainable as the negative of the Lagrange multiplier associated with the constraint $f_1(x) \geqslant \bar{v}$. Note that λ_{21} is part of the solution output of most mathematical programming algorithms.

λ_{21} has the following interpretation: λ_{21} measures the loss in f_2 from its current level that must be suffered per unit gain in f_1. On the other hand T_{21} measures the loss in f_2 the DM is prepared to trade-off per unit gain in f_1. Naturally, these two quantities must be related to the optimality of BCMP and the precise relation is the subject of our next theorem.

THEOREM 6:

Let x^* be the optimal solution to BCMP with $f(x^*) = (f_1^*, f_2^*)$. Let x be any efficient solution. Then

(a) x^* is optimal to BCMP if and only if $\lambda_{21} = T_{21}$, at x^*.

(b) At x if $\lambda_{21} < T_{21}$ then $f_1 < f_1^*$; similarly, if $\lambda_{21} > T_{21}$, then $f_1 > f_1^*$.

PROOF: Part (a) *Sufficiency:* At x^* let $\lambda_{21} = T_{21}$. By definition of T_{21}, at x^* the DM is indifferent to unit gain in f_1 and Δf_2 units of loss in f_2 where $\Delta f_2 = \left(\frac{\partial U}{\partial f_1}\right) \Big/ \left(\frac{\partial U}{\partial f_2}\right)$. In other words

$$[f_1(x^*), f_2(x^*)] \sim [f_1(x^*) + \delta_1, f_2(x^*) - \Delta f_2 \cdot \delta_1] \tag{18}$$

where '$\sim$' denotes the indifference of the DM and δ_1 is the differential change in f_1. When $\lambda_{21} = T_{21}$, in the neighborhood of x^*, a differential change δ_1 in f_1 is accompanied by a differential change of $-(\Delta f_2) \cdot \delta_1$ in f_2, as per the definition of λ_{21}. Hence, points in the neighborhood of x^* are equally preferred to x^*; hence, x^* is the local optimum. By quasiconcavity of U, x^* must be the global optimum too.

Necessity: Let x^* be optimal to BCMP but $\lambda_{21} \neq T_{21}$ say $\lambda_{21} < T_{21}$. Around a neighborhood of x^*, $\exists$ a point $[f_1(x^*) + \delta_1, f_2(x^*) - \delta_1\lambda_{21}]$ where δ_1 is a differential change in f_1 as per the definition of λ_{21}. At $\lambda_{21} < T_{21}$,

$$(f_2(x^*) - \delta_1\lambda_{21}) > (f_2(x^*) - \delta_1 T_{21}) \text{ for } \delta_1 > 0. \tag{19}$$

Since U is strictly increasing in both f_1 and f_2, inequalities (18) and (19) taken together imply that

$$\begin{aligned}[f_1(x^*) + \delta_1, f_2(x^*) - \delta_1\lambda_{21}] &\succ [f_1(x^*) + \delta_1, f_2(x^*) - \delta_1 T_{21}] \\ &\sim [f_1(x^*), f_2(x^*)].\end{aligned} \tag{20}$$

By transitivity of preferences

$$[f_1(x^*) + \delta_1,\ f_2(x^*) - \delta_1\lambda_{21}] \succ [f_1(x^*), f_2(x^*)]. \tag{21}$$

Obviously, (21) contradicts the optimality of x^*. Hence, $\lambda_{21} \not< T_{21}$. Similarly we can show that $\lambda_{21} \not> T_{21}$. Hence the proof.

Part (b) Let $\lambda_{21} < T_{21}$. By (21) an increase in f_1 from its current value leads to a preferred solution. Hence, an increase in f_1 is a locally increasing direction for U along the efficient frontier. Also x being an efficient solution is a solution to program (P_v) by Theorem 2 and $f_1(x) = v$ by Theorem 1. We noted earlier that x is a solution to program (P3) also, i.e., $\underset{(v,f_2)\in A}{\text{Max}}\ U(v, f_2)$. Theorem 5 established the unimodality of $g(v)$ which is the "value" of

optimal solution to program (P3); as $f_1(x) = v$, the value of optimal solution to program (P3) is unimodal in $f_1(x)$. By the property of unimodality, U increases along a locally improving direction continuously up to the optimal solution and continuously decreases in the other direction. Hence, $f_1(x^*) > f_1(x)$ or $f_1^* > f_1$. Similarly, we can prove the other case also.

Note that Theorem 6 provides a convenient way of ruling out a portion of the efficient frontier. All that is needed is an interaction with the DM whether or not $\lambda_{21} \lesseqgtr T_{21}$. Depending on his response we can narrow down the search of efficient points to only those with f_1 value greater than or less than the current f_1 value. We wish to point out here that the *Surrogate Worth Tradeoff* method due to Haimes, Hall and Freedman [9] is essentially based on a result similar to part (a) of Theorem 6. However, they do not have a result comparable to part (b) of the Theorem. Thus no "region elimination" is possible in their method. Also the DM has to specify how far T_{21} is greater than or less than λ_{21} on a subjective scale of $+10$ to -10.

We now formally state the procedure as follows:

Step 1: Solve (P1) and (P2) and determine v^* and w^*.

Step 2: Solve (Q_{w^*}) and determine $f_1(x_2^*)$. Set $v_l = f_1(x_2^*)$.

Step 3: Solve P_v for $v = v_A$ where $v_l < v_A < v_u$. Determine λ_{21}, i.e., the Lagrange multiplier corresponding to the constraint $f_1(x) \geqslant v_A$ in the program P_v.

Step 4: Interact with the DM and present him with λ_{21} and seek his comparative trade-off T_{21}, i.e., whether $T_{21} \lesseqgtr \lambda_{21}$. Depending on the outcome of interaction eliminate a portion of the efficient frontier using the following rule:

If $\lambda_{21} < T_{21}$, update $v_l = v_A$; go to step 5.

If $\lambda_{21} > T_{21}$, update $v_u = v_A$; go to step 5.

If $\lambda_{21} = T_{21}$, stop; the current solution to P_v is optimal to BCMP.

Step 5: If $|v_l - v_u| < \xi$ (chosen small number) stop; else go to step 3.

Note that selection of v_A values can be done efficiently using the midpoint of the interval. For example, with 5 stages of interaction, we will be able to bracket the optimal solution to less than 4% of the original interval. In 7 stages, we will be within 1%, etc. Thus we have a finite, rapidly converging procedure using only "comparative tradeoffs." We shall illustrate the method using the same example given in Section 3.

Numerical Example

Solution Using Comparative Trade-Off Method:

Step 1: As before $v^* = 60$; $w^* = 3.2$. Hence $v_u = 60$.

Step 2: As before $v_l = 8$.

Iteration 1

Step 3: Using the 'Bisection' method (i.e., midpoint values) to choose v_A values, $v_A = 34$. x_A which solves P_v for $v = v_A$ has f values of (34,2.057); also the Lagrange multiplier $\lambda_{21} = 0.0571$.

Step 4: Interact with the DM and seek his comparative estimate of local trade-off T_{21}, i.e., whether $T_{21} > \lambda_{21}$, $T_{21} < \lambda_{21}$ or $T_{21} = \lambda_{21}$. Assuming that the DM is guided by the implicit preference function

$U(f_1,f_2) = f_1^{2/3} f_2$; we get

$\frac{\partial U}{\partial f_1} = 2/3\ f_1^{-1/3} f_2$ and $\frac{\partial U}{\partial f_2} = f_1^{2/3}$. Evaluating at x_A,

$\frac{\partial U}{\partial f_1} = \frac{2}{3}\ (34)^{-1/3}\ (2.057) = 0.4233$ and

$\frac{\partial U}{\partial f_2} = 34^{2/3} = 10.4951$. Hence, $T_{21} = \frac{\partial U}{\partial f_1} \Big/ \frac{\partial U}{\partial f_2} = \frac{0.4233}{10.4951} = 0.0403.$

Thus, $\lambda_{21} = 0.0571 > T_{21} = 0.0403$. Hence, update v_u to $v_A = 34$

Step 5: As the limits v_l and v_u are not close enough, the search has to be continued further on the reduced efficient frontier. The details of the calculation for further stages are summarized in Table 2.

TABLE 2 — *Solution of Bicriteria Problem Using Comparative Tradeoff Method*

Iteration #	u_l	v_u	v_A	$(v_A, g(v_A))$	λ_{21}	T_{21}	Remain Interval (% of original interval)
0	8	60	–	–	–	–	100
1	8	34	34	(34,2.057)	0.0571	0.0403	50
2	21	34	21	(21,2.800)	0.0571	0.0889	25
3	27.5	34	27.5	(27.5,2.429)	0.0571	0.0589	12.5
4	27.5	30.75	30.75	(30.74,2.243)	0.0571	0.0486	6.25

5. BICRITERIA OPTIMIZATION APPLIED TO THE U.S. AIR FORCE CARDIOVASCULAR DISEASE CONTROL PROGRAM

In this section we will demonstrate an application of the paired comparison method to a bicriteria optimization problem of designing a cardiovascular disease control program for the U.S. Air Force personnel. this part of a study currently undertaken by Purdue's School of Industrial Engineering, under contract with the USAF School of Aerospace Medicine.

The United States Air Force (USAF) is planning a comprehensive program for reducing the incidence of cardiovascular disease (CVD) among its active duty personnel. This program, known as the *Health Evaluation and Risk Tabulation* (HEART) program, would consist of

(i) an extensive educational program about CVD, its causes and prevention,

(ii) a risk screening program to identify those AF personnel who have a high susceptibility for developing CVD, and

(iii) a risk intervention program to treat such high risk personnel and reduce their CVD risk by medications and/or behavior modifications.

In order for such a program to be viable, the expected benefits resulting from reduced incidence of heart disease must outweigh the cost of the HEART program.

Risk Identification: A general cardiovascular risk profile, based on the Framingham Study [11], is presently being considered as the means of identifying high-risk personnel for treatment. With this approach a logistic regression model would be applied to each individual and would yield his CVD risk; namely, the probability of manifestation of cardiovascular disease in the individual in an eight-year time period.

Risk Intervention: By rank ordering the CVD risk scores of USAF personnel, a certain percentage of the highest risk individuals are selected for specialized face to face therapy for modifying their elevated values of risk factors. At present, those high-risk individuals with elevated systolic blood pressures will be treated with medication, while those with elevated cholesterol levels will be treated with diet modification only. Both individual and group therapies will be used to persuade smokers to quit smoking.

The recurring cost of the HEART program, excluding the initial start-up costs, is estimated between 7 and 9 million dollars a year depending on the total size of the therapy group. Hence, before implementing the HEART program on all airforce bases, the Air Force was interested in answers to the following question:

> Given the total size of the therapy group, determine the *optimal* number of Air Force personnel to be selected in each age group of flyers and nonflyers such that the total "cost-effectiveness" of the HEART program is maximized.

The therapy selection problem has indeed two competing objectives. One objective is to minimize the USAF CVD risk and the other objective is to minimize the HEART program cost. By increasing the size of the therapy program, the CVD incidence in the Air Force will be progressively reduced but this can be achieved only at the cost of increased HEART program expenditure. If we follow a policy of selecting the high risk personnel from each age group for risk intervention, the graph of CVD incidence versus therapy size will be a monotonically decreasing function. Hence, there is an optimal expenditure level for the HEART program, an optimal level of CVD incidence, an optimal therapy size and an optimal selection strategy for that therapy size. Determination of all these quantities is a nontrivial problem for which we developed the following bicriteria mathematical programming model which was solved using the paired comparison method developed in Section 3.

BCMP Model

$$\text{Minimize} \quad f_1(x)$$

$$\text{Minimize} \quad f_2(x)$$

$$\text{Subject to} \quad \sum_{j=1}^{17} x_j \leqslant S$$

$$a_j \leqslant x_j \leqslant b_j \quad j = 1, \ldots, 17.$$

Where:

x = selection strategy $(x_1, x_2, \ldots, x_{17})$ for each age group of flyers and nonflyers.

$f_1(x)$ = USAF CVD risk for a given selection strategy x

$f_2(x)$ = HEART program cost for a given selection strategy x

S = Maximum size of the therapy group (fixed)

a_j = Minimum number to be selected from age group j (flyers and nonflyers); $j = 1, \ldots, 8$ correspond to flyers in age groups 20-24,...,55-59, respecitvely; $j = 11, \ldots, 17$ correspond to nonflyers in age groups 15-19,...,55-59, respectively.

b_j = Maximum number that can be selected from age group j.

Even though $f_2(x)$ could be determined analytically, $f_1(x)$ is indeed a random variable because of the unpredictability of the effectiveness of the risk intervention programs on modifying the risk factor levels. Hence, a comprehensive simulation model known as P-HEART (Purdue Health Evaluation and Risk Tabulation) was developed by the authors [14]. P-HEART simulates the USAF population and performs risk identification, therapy selection and risk intervention. The description of the simulation model, its validation and findings are given in Reference [14]. Through extensive simulation runs, the expected value of $f_1(x)$ for different selection strategies were estimated and a polynomial equation was fitted to get analytical forms for $f_1(x)$ by age group and flying status.

The paired comparison method was then programmed within the frame work of interactive solution of BCMP for minimizing CVD risk and HEART program cost. The Generalized Reduced Gradient Algorithm was used to solve the nonlinear programs in Steps 1, 2, and 3 of the interactive algorithm. The interactive program was extensively tested at Purdue using Air Force Officers as decision makers. The program has been well received by the policy makers in the USAF HEART program office and is being used as part of the long term project planning currently underway in the Air Force. For more details on the interactive computer program and its output, the reader is referred to Sadagopan [16].

6. CONCLUSIONS

In this paper we have developed interactive procedures for the solution of bicriteria programs. Even though the entire set of efficient solutions can be generated and pictured, such iteractive procedures do have merit due to the following reasons: in many real life problems

complete generation of all efficient solutions may be computationally expensive; even granting such generation, the DM is unaided in the selection process. Hence, the procedures developed in this paper may be thought of as "decision aids" [15].

The primary power behind our procedures is the unimodality property (Theorem 3). A similar result is available in Geoffrion [7]. However, our proof of unimodality is entirely new and allows extension to more general cases unlike Geoffrion's result. Also our relationship between Lagrange Multipliers and local tradeoffs is new. For more details on the bicriteria results, U.S. Air Force application and extensions to the general multi-criteria problem, the reader is referred to Sadagopan [16].

REFERENCES

[1] Adulbhan, P. and M.T. Tabucanon, "Bicriterion Linear Programming," Computers and Operations Research, *4*, 147-153 (1977).
[2] Bacopoulas, A. and I. Singer, "On Convex Vectorial Optimization in Linear Spaces," Journal of Optimization Theory and Applications, *21*, 175-188 (1977).
[3] Benson, H.P., "Vector Maximization with Two Objective Functions," Journal of Optimization Theory and Applications, *28*, 253-258 (1979).
[4] Choo, E.U. and D.R. Atkins, "Bicriteria Linear Fractional Programming," Paper presented at the Joint ORSA/TIMS Conference, Los Angeles, CA (1978).
[5] Everett, H. "Generalized Lagrange Multiplier Method for Solving Problems of Optimal Allocation of Resources," Operations Research, *11*, 399-417 (1969).
[6] Gearhart, W.B., "On the Characterization of Pareto Optimal Solutions to Bicriteria Optimization," Journal of Optimization Theory and Applications, *27*, 301-308 (1979).
[7] Geoffrion, A.M., "Solving Bicriterion Mathematical Programs," Operations Research, *15*, 39-54 (1967).
[8] Geoffrion, A.M., J.S. Dyer and A. Feinberg, "An Interactive Approach to Multi-Criterion Optimization with an Application to the Operation of an Academic Department," Management Science, *19*, 357-368 (1972).
[9] Haimes, Y.Y., W.A. Hall and H.T. Freedman, *Multi-Objective Optimization in Water Resource Systems* (American El Sevier, New York, 1975).
[10] Intriligator, M.D., *Mathematical Optimization and Economic Theory* (Prentice-Hall, Englewood Cliffs, NJ, 1971).
[11] Kannel, W.B., D. McGee, and T. Gordon, "A General Cardiovascular Risk Profile: The Framingham Study," American Journal of Cardiology, *38*, 46-51 (1976).
[12] Moeseke, P.V., "Towards A Theory of Efficiency," in *Readings in Quantitative Economics*, J.P. Quirk and A.M. Zarley, Editors (University Press of Kansas, Lawrence, KS, 1-30, 1965).
[13] Pasternak, H., and U. Passy, "Bicriterion Mathematical Program with Boolean Variables," in *Multiple Criteria Decision Making*, T.L. Cochrane and M. Zeleny, Editors (University of South Carolina Press, Columbia, S.C., 327-348, 1973).
[14] Ravindran, A., S. Sadagopan, C.C. Petersen and C.P. Hatsell, "Using Simulation in Cardiovascular Disease Control," Simulation, *32* (July 1980).
[15] Roy, B., "Problems and Methods with Multiple Objective Functions," Mathematical Programming, *1*, 239-266 (1972).
[16] Sadagopan, S., "Multiple Criteria Mathematical Programming—A Unified Interactive Approach," unpublished Ph.D. dissertation, School of Industrial Engineering, Purdue University, W. Lafayette, IN (December 1979).
[17] Walker, J., "An Interactive Method as an Aid in Solving Bicriteria Mathematical Programming Problems," Journal of Operations Research Society, *29*, 915-922 (1978).

[18] Wallenius, J., "Comparative Evaluation of Some Interactive Approaches to Multicriteria Optimization," Management Science, *21*, 1387-1396 (1975).
[19] Zionts, S., Editor, *Multiple Criteria Problem Solving* (Springer-Verlag, NY, 1978).

A NEW METHOD TO SOLVE STEADY STATE QUEUEING EQUATIONS

W. K. Grassmann

University of Saskatchewan
Saskatoon, Saskatchewan, Canada

M. L. Chaudhry

Royal Military College of Canada
Kingston, Ontario, Canada

ABSTRACT

There are a great number of queueing systems, including the $M^X/M^Y/c$, the $GI^X/M/c$ and the discrete $GI/G/1$ queue in which the state probabilities are determined by repeated queue equations. This paper gives a simple, efficient and numerically stable algorithm to calculate the state probabilities and measure of performance for such systems. The method avoids all complex arithmetic and matrix manipulations.

1. INTRODUCTION

This paper gives a new method to find the equilibrium probabilities of continuous-time Markov processes in which the rate of going from state m to n is equal to d_{n-m} for any n exceeding a certain limit, say r. The largest and smallest possible jumps are h and $-g$ respectively, that is, the subscript of d_k runs from $-g$ to h. These concepts will be further expanded in the next section. Here, we demonstrate it shortly, using the $M^X/M^Y/1$ queue as an example. In this queue, arrivals of size j, $1 \leqslant j \leqslant h$ occur at rate λ_j, and departures of groups of size i occur at a rate of μ_i, $1 \leqslant i \leqslant g$. Hence,

$$d_j = \lambda_j, \quad 1 \leqslant j \leqslant h$$

$$d_{-i} = \mu_i, \quad i \leqslant n, \ 1 \leqslant i \leqslant g.$$

The paper also considers discrete Markov processes with a similar structure. In this case, there is a probability of p_k to go from state n to $n + k$. Queues with such structure include the system size of the $GI^X/M/c$ queue [4] and the waiting time of the discrete $GI/G/1$ queue [5].

At present, there are several approaches available to solve such queueing problems. In the first approach, one sets up generating functions, and uses partial fraction expansions to invert these generating functions. These expansions require one to find the roots of an equation of degree $g + h$. The same is true if one uses operators on the queueing equations and determines the characteristic roots. For practical applications, these approaches have some disadvantages. The equations are often difficult to solve, especially when they are of a high

degree. Their roots are usually complex and difficult to work with. Moreover, there may be double roots which lead to further complications. Neuts [3] therefore suggested solving these problems using matrix equations.

The method in this paper combines the approach of Neuts with the classical method. In a sense, it can be considered as a modification of the method of Neuts. The main difference between our method and the Method of Neuts is the following. In Neuts' approach, a matrix R of dimension $h \times h$ has to be calculated. In our approach, only the first column of R is calculated, which reduces the number of operations approximately by a factor of h. Even though the method is similar to the one of Neuts, its derivation is done, using the classical approach. In this way, our method bridges the gap between the classical approach and the approach of Neuts.

2. PROBLEM DEFINITION

In this section, a precise problem definition is given and demonstrated, using a number of examples. First, we discuss the discrete case. Let there be a Markov chain with the transition probabilities, $p_{m,n}$, $n,m \geq 0$. The equilibrium probabilities of this chain are denoted by π_n, that is, π_n is probability to be in state n. As is well known, the π_n are determined by the following equilibrium equations, provided the system has an equilibrium

$$\pi_n = \sum_{m=0}^{\infty} \pi_m p_{m,n}, \quad n \geq 0. \tag{1}$$

We assume that there is a certain limit r such that for $n \geq r$, $p_{m,n} = d_{n-m}$ only depends on the difference between m and n. In other words, if Q represents the state, d_k is the probability to increase Q by k from n to $n + k$. If π_m is defined to be zero for $m \leq 0$, equation (1) becomes therefore for $n \geq r$

$$\pi_n = \sum_{m=-\infty}^{\infty} \pi_m d_{n-m} = \sum_{k=-\infty}^{\infty} \pi_{n-k} d_k, \quad n \geq r.$$

We furthermore assume that $d_k = 0$ for $k < -g$ or $k > h$. Thus, one has

$$\pi_n = \sum_{k=-g}^{h} \pi_{n-k} d_k, \quad n \geq r.$$

If one solves this equation for π_n, one gets

$$\pi_n = \sum_{k=-g}^{h} \pi_{n-k} e_k, \quad n \geq r. \tag{2}$$

The e_k in this expression are defined as

$$\left.\begin{aligned} e_k &= d_k/(1 - d_0), \quad k \neq 0 \\ e_0 &= 0. \end{aligned}\right\} \tag{3}$$

For $n < r$, one uses equation (1), that is

$$\pi_n = \sum_{m=0}^{\infty} \pi_m p_{m,n}, \quad n < r. \tag{4}$$

Equation (2) will be referred to as the *queueing equation*, whereas equation (4) gives the *initial conditions*. There are r initial conditions. However, it can be shown that one of them is redundant. In its place, one has the normalizing condition

$$\sum_{n=0}^{\infty} \pi_n = 1. \tag{5}$$

If there exists a unique equilibrium vector $[\pi_n]$, this vector is fully determined by the queueing equation, the $r-1$ initial condition and the normalizing condition. Of course, if $r = 1$, there are no initial conditions. As is well known, an equilibrium solution will exist if

$$\rho = \sum_{i=1}^{h} i\, d_i / \sum_{i=1}^{g} i\, d_{-i} < 1.$$

This paper will show how to find this equilibrium solution in an efficient way.

In continuous-time Markov chains, the equilibrium probabilities are determined by

$$0 = \sum_{m=0}^{\infty} \pi_m q_{m,n}, \quad n \geqslant 0. \tag{6}$$

Here, $q_{m,n}$, $m \neq n$, are the transition rates, and

$$q_{m,m} = -\sum_{n \neq m} q_{m,n}.$$

We now consider problems where

$$q_{m,n} = d_{n-m}, \quad n \geqslant r.$$

Here, d_0 is defined as

$$d_0 = -\sum_{k \neq 0} d_k.$$

As in the discrete case, it is assumed that $d_k = 0$ outside the range $-g \leqslant k \leqslant h$. If $\pi_n = 0$ for $n < 0$, one finds thus from equation (6)

$$0 = \sum_{k=-g}^{h} \pi_{n-k} d_k, \quad n \geqslant r.$$

This equation can again be solved for π_n, which gives

$$\pi_n = \sum_{k=-g}^{h} \pi_{n-k}\, e_k, \quad n \geqslant r. \tag{7}$$

Here,

$$\left.\begin{aligned} e_0 &= 0 \\ e_k &= d_k/(-d_0), \quad k \neq 0. \end{aligned}\right\} \tag{8}$$

Note that equation (2) and equation (7) have exactly the same structure. In particular, $e_k \geqslant 0$ and the sum of the e_k is one in either case. The normalizing condition is given by (5) both in the discrete and the continuous case. The initial conditions of the continuous case are easily found from (6) by letting $0 \leqslant n < r$.

We now give 3 examples which will be used later for the purpose of demonstration.

EXAMPLE 1: First, consider the waiting time in queue of a discrete $GI/G/1$ queue. Following Ponstein [5], a Markov chain is set up, in which the state space are the waiting times rather than the number of elements in the system. To set up this chain, it is assumed that the difference between the service time (S) and the interarrival time (A) is (a) always integer and

(b) always between $-g$ and h. It is well known (see e.g. Ponstein [5]), that the waiting time of customer j exceed the one of customer $j-1$ by an amount $S-A$, provided, of course, that customer $j-1$ had a waiting time above $-(S-A)$. Consequently, if W represents the waiting time of a customer in equilibrium, W increases by k with probability $P(S-A=k)$. This means

$$d_k = P(S-A=k),\ -g \leqslant k \leqslant h$$

and

$$\left.\begin{aligned} e_k &= P(S-A=k)/[1-P(S=A)],\ k \neq 0 \\ e_0 &= 0 \end{aligned}\right\} \tag{9}$$

In the problem considered, there is only one initial condition, and this condition can be dropped as redundant. The queueing equation, together with normalizing condition is thus sufficient to determine all π_n. Here, π_n represents the probability of having a waiting time of n.

EXAMPLE 2: In the $GI^h/M/1$ queue, arrivals occur in groups of size h. Let α_j be the probability of servicing j elements between two consecutive arrivals and let π_n the probability of having n elements immediately before an arrival. It is easily verified that the queueing equation is given by (2) with $r=1$ and

$$\left.\begin{aligned} e_k &= \alpha_{h-k}/(1-\alpha_h),\ -\infty < k \leqslant h,\ k \neq 0 \\ e_0 &= 0. \end{aligned}\right\} \tag{10}$$

Here, α_k is the probability of serving k customers between two successive arrivals. The initial conditions consist only of one equation which can be omitted. Also, in the case considered, g is infinite. However, for practical calculations, it is sufficient to carry only a finite number of e_k.

EXAMPLE 3: Consider the $M^Y/M^Y/1$ queue. λ_j, $1 \leqslant j \leqslant h$, is the rate at which groups of size j arrive. No service is done unless at least g elements are in the system. If the number in the system is g or more, there is a rate of μ_i, $1 \leqslant i \leqslant g$, that i elements leave the system after having received service together. For this problem, one has the following transition rates

$$q_{m,n} = \lambda_{n-m},\ 0 < n-m \leqslant h$$
$$q_{m,n} = \mu_{m-n},\ 0 < m-n \leqslant g,\ m \geqslant g.$$

All other $q_{m,n}$, $m \neq n$ are zero. In particular, all $q_{m,n}$, $m < g$, $n < m$ are zero because nobody leaves if there are less than g elements in the system. The $q_{m,m}$ become

$$q_{m,m} = -\sum_{j=1}^{h} \lambda_j \qquad ,\ m < q$$

$$q_{m,m} = -\left[\sum_{j=1}^{h} \lambda_j + \sum_{i=1}^{g} \mu_i\right],\ m \geqslant g.$$

A little reflection shows that $r = g$, and the queueing equation is given by (5) with

$$\left.\begin{aligned} e_0 &= 0 \\ e_k &= \lambda_k \Big/ \left[\sum_{j=1}^{h} \lambda_j + \sum_{i=1}^{g} \mu_i\right], \quad k > 0 \\ e_k &= \mu_k \Big/ \left[\sum_{j=1}^{h} \lambda_j + \sum_{i=1}^{g} \mu_i\right], \quad k < 0 \end{aligned}\right\} \tag{11}$$

The initial conditions are given by (7), that is

$$0 = \sum_{k=0}^{\infty} \pi_k q_{k,0} = -\pi_0 \sum_{j=1}^{h} \lambda_j + \pi_g \mu_g$$

$$0 = \sum_{k=0}^{\infty} \pi_k q_{k,n} = \sum_{j=1}^{h} \pi_{n-j} \lambda_j - \pi_n \sum_{j=1}^{h} \lambda_j + \sum_{i=g-n}^{g} \pi_{n+i} \mu_i, \quad n < g.$$

Again, the convention was used that $\pi_m = 0$ if $m < 0$. From the initial conditions, one can be omitted, and it turns out to be convenient to omit the one corresponding to $n = g - 1$.

Above, we discussed a few examples which are amenable to our method. Additional examples can easily be generated. We now show how to solve these examples.

3. THE MAIN RESULT

The fundamental result of this paper is the following. If $\rho < 1$, and if the Markov chain is ergodic, there are h values $a_j \geqslant 0$, $1 \leqslant j \leqslant h$ such that

$$\pi_n = \sum_{j=1}^{h} a_j \pi_{n-j}, \quad n \geqslant r. \tag{12}$$

The a_j, $1 \leqslant j \leqslant h$ are determined in a unique way by the e_k, $-g \leqslant k \leqslant h$. Furthermore,

$$\sum_{j=1}^{h} a_j < 1.$$

This paper will present several methods to find the a_j. Once the a_j are found, the problem is actually solved. Together with the $r - 1$ initial conditions and the normalizing equation (12) gives exactly $r + 1$ equation to determine $\pi_0, \pi_1, \ldots, \pi_r$. Moreover, the generating function of the π_n can be expressed in terms of a_j and $\pi_0, \pi_1, \ldots, \pi_{r-1}$. To see this it is convenient to define a_0 as -1, and set

$$A(z) = \sum_{j=0}^{h} a_j z^j$$

$$P(z) = \sum_{n=0}^{\infty} \pi_n z^n.$$

In this case, one finds, as will be shown in Section 4:

$$A(z) P(z) = \sum_{j=0}^{h} a_j \sum_{n=j}^{r-1} \pi_{n-j} z^n. \tag{13}$$

When setting $z = 1$, $P(z)$ becomes 1, and

$$A(1) = \sum_{j=0}^{h} a_j \sum_{n=j}^{r-1} \pi_{n-j} = \sum_{n=0}^{r-1} \pi_n \sum_{j=0}^{e} a_j, \quad e = \min(r - n - 1, h).$$

This means

$$\sum_{j=0}^{h} a_j = -1 + a_1 + a_2 + \ldots + a_h = \sum_{n=0}^{r-1} \pi_n \sum_{j=0}^{e} a_j, \quad e = \min(r - n - 1, h). \tag{14}$$

This equation provides a convenient normalizing condition and can be used in place of (5). To find L, the expected number in the system, one can differentiate (13) and find

$$A'(z)\,P(z) + A(z)\,P'(z) = \sum_{j=0}^{h} a_j \sum_{n=j}^{r-1} n\,\pi_{n-j}\,z^{n-1}.$$

If $z = 1$, $P(z) = 1$ and $P'(z) = L$. Using these values, the above equation gives after some calculation

$$L = \sum_{j=0}^{h} a_j \left[\sum_{n=j}^{r-1} n\,\pi_{n-j} - j\right] \Big/ \sum_{j=0}^{h} a_j. \tag{15}$$

Equations (14) and (15) simplify if $r = 1$. Then

$$A(z)\,P(z) = a_0\pi_0 = -\pi_0.$$

If one sets $z = 1$, this gives,

$$\pi_0 = -A(1) = 1 - \sum_{j=1}^{h} a_j. \tag{16}$$

This means that

$$\Sigma\, a_j < 1.$$

If Q is a random variable representing the state of the system, $E(Q)$ can be found as:

$$E(Q) = \sum_{n=0}^{\infty} n\pi_n = -A'(1)/A(1) = \sum_{j=1}^{h} ja_j/\pi_0. \tag{17}$$

The ideas just presented shall now be demonstrated, using the examples discussed above. First, consider the discrete $GI/G/1$ queue. Specifically, suppose that the distribution of interarrival time and the distribution of the service time are as given in Table 1.

TABLE 1 — *Distribution of the Interarrival Time and the Service Time.*

k	1	2	3
$P(S = k)$	0.4	0.3	0.3
$P(A = k)$	0.1	0.3	0.6

Using these values, one can find the e_k, $-g \leqslant k \leqslant h$, from equation (9), and the e_k in turn, determine the a_j, $1 \leqslant j \leqslant h$ (see Sections 4 and 5). For the problem described by Table 1, one finds,

$$a_1 = 0.23156$$
$$a_2 = 0.05039.$$

Equations (16) and (17) give now

$$\pi_0 = P(W = 0) = 1 - a_1 - a_2 = 0.71804$$

$$E(W) = (a_1 + 2a_2)/P(W = 0) = 0.46287.$$

Since π_{-1} is assumed to be zero, one can use (12) to calculate π_1, π_2, π_3 and so on.

$$\pi_1 = \pi_0 a_1 + \pi_{-1} a_2 = 0.1663$$

$$\pi_2 = \pi_1 a_1 + \pi_0 a_2 = 0.0747$$

$$\vdots$$

Since all $a_i \geqslant 0$, this recursion is numerically very stable.

Next, consider the $GI^h/M/1$ queue. Specifically, let the arrival time distribution be deterministic, let $h = 2$, $\lambda_2 = 0.4$ and $\mu = 1$. The α_k of equation (10) are in this case Poisson-distributed with parameter 2.5. One finds in this case

$$a_1 = 0.41045$$

$$a_2 = 0.13711.$$

Again, all entities of interest can be calculated. L, the expected number in the system preceeding an arrival becomes for instance

$$L = (a_1 + 2a_2)/(1 - a_1 - a_2) = 1.5132.$$

Finally, consider the $M^X/M^Y/1$ queue of Example 3. Specifically let

$$\lambda_1 = \lambda_2 = \mu_1 = \mu_2 = \mu_3 = 1.$$

In this case, the a_j turn out to be

$$a_1 = 0.34960$$

$$a_2 = 0.24582.$$

The initial conditions are now

$$0 = -2\pi_0 + \pi_3$$

$$0 = \pi_0 - 2\pi_1 + \pi_3 + \pi_4.$$

π_3 and π_4 are given by (12) as

$$\pi_3 = a_1\pi_2 + a_2\pi_1 = 0.34960\pi_2 + 0.24582\pi_1$$

$$\pi_4 = a_1\pi_3 + a_2\pi_2 = (a_1^2 + a_2)\pi_2 + a_1 a_2 \pi_1 = 0.36804\pi_2 + 0.08594\pi_1.$$

Using these values, the initial conditions become

$$0 = -2\pi_0 + 0.24582\pi_1 + 0.34960\pi_2$$

$$0 = \pi_0 - 1.66824\pi_1 + 0.71764\pi_2.$$

As a third equation, one uses the normalizing condition as given by (14).

$$-1 + a_1 + a_2 = \pi_0(-1 + a_1 + a_2) + \pi_1(-1 + a_1) - \pi_2$$

or

$$0.40458 = 0.40458\pi_0 + 0.65040\pi_1 + \pi_2.$$

These 3 equations can be solved for π_0, π_1 and π_2.

The result is

$$\pi_0 = 0.06741,\ \pi_1 = 0.15840,\ \pi_2 = 0.27428.$$

It is now simple to find L from Equation (15). One has

$$L = [-(\pi_1 + 2\pi_2) + a_1(\pi_0 + 2\pi_1 - 1) + a_2(2\pi_0 - 2)]/(-1 + a_1 + a_2)$$
$$= 3.4128.$$

Thus, once the a_i are found, the problem is solved easily. The next two sections show several methods to find the a_i.

4. THE SIGNIFICANCE OF a_i AND THEIR CALCULATION

In this section, it is shown that equation (12) holds, and how one can find the a_i in the general case. From the initial conditions and from the queueing equation, one finds the following expression:

$$P(z) = U(z)/V(z).$$

Here $U(z)$ is a polynomial of degree $r + g - 1$ which is of no further interest, and $V(z)$ is equal to

$$V(z) = \sum_{k=0}^{g+h} e_{k-g} z^k - z^g.$$

It is well known (see e.g. [5]) that $V(z)$ has a zero at $z = 1$, $g - 1$ zeros inside the unit circle and h zeros outside. It follows that $V(z)$ can be written as

$$V(z) = A(z)B(z).$$

Here

$$A(z) = -1 + a_1 z + a_2 z^2 + \ldots + a_h z^h$$

contains all the h zeros outside the unit circle.

Because of the convergence of $P(z)$, $U(z)$ must be divisible by $B(z)$. Since $U(z)$ has a degree of $r + g - 1$, and $B(z)$ has a degree of g, $C(z)$ has a degree of $r - 1$. Consequently,

$$U(z)/B(z) = C(z) = \sum_{i=0}^{r-1} c_i z^i,$$

or

$$P(z) = C(z)/A(z).$$

This equation can be written as

$$P(z)A(z) = C(z). \tag{18}$$

The coefficient of z^n of the product $P(z)A(z)$ equals, as is well known

$$\sum_j a_j \pi_{n-j}.$$

For $n \leqslant r-1$, this means

$$\sum_{j=0}^{h} a_j \pi_{n-j} = c_n. \tag{19}$$

For $n \geqslant r$, one finds

$$\sum_{j=0}^{h} a_j \pi_{n-j} = 0. \tag{20}$$

Equation (20) is identical with Equation (12), and Equation (18) and (19) can be combined to give Equation (13). This proves Equations (12) and (13).

From the above discussion, the significance of the a_j becomes clear. $P(z)$ can be factorized into two factors $A(z)$ and $B(z)$, and the a_j are merely the coefficients of $A(z)$, provided a_0 is defined as -1. The problem is thus reduced to finding $A(z)$. The most convenient method to do this seems to be the following one. One starts with the identity, which will be proven in Section 6.

$$a_j = e_j + \sum_{i=1}^{q} e_{-i}\, a_{i,j}, \quad 1 \leqslant j \leqslant h, \tag{21}$$

with

$$\left.\begin{aligned} a_{0,j} &= a_j, \quad 1 \leqslant j \leqslant h \\ a_{i+1,j} &= a_{i,1} a_j + a_{i,j+1}, \quad 1 \leqslant i \leqslant g-1, \ 1 \leqslant j \leqslant h \end{aligned}\right\} \tag{22}$$

Furthermore, $a_{i,j}$ is to be taken as zero for $j > h$. Equations (21) and (22) can be used to find a_j by successive approximation. As starting values, one can use $a_j^0 = e_j$, $j = 1, 2, \ldots, h$. These values can be used to replace the a_j on the right of (22), and a new approximation for a_j can be found from Equation (21). In this way, one continues until a suitable stopping criterion is satisfied.

We tried two different stopping criteria. First, we required that the π_j generated by Equation (12) satisfy the queueing equation with a specified precision ϵ. It was found that for high values of h and/or g, this criterion performed poorly. The reason for this is simple. If h is say 100, a small change in d_{100} will have a dramatic impact on ρ and, consequently, on the performance measures of the system. As an alternate stopping criterion, we used the change of the calculated $A'(1) = \Sigma\, ja_j$ to decide when the precision of the a_j is adequate. Since $A'(1)$ is closely related to the expected number in the system (see Equation (17)) this seems to be a good stopping criterion.

If there are q iterations, one needs $4qgh$ operations to find all a_j. The reader may want to verify this. If one adds all $e_{-i}a_{i,j}$ to the sum at the right of Equation (21) as soon as the $a_{i,j}$ becomes available one only needs to store the $a_{i,j}$ for the current value of i. If this is done, the algorithm requires only an array area of $4h + g$ as the reader may verify. The algorithm is thus efficient and does not require huge arrays.

The algorithm described above was programmed in FORTRAN and run on the DEC 2060. The execution times were negligible. A problem with $h = 100$ and $g = 100$ and $\rho = 0.9$

only required 8.4 seconds and 53 iterations until all a_j were obtained. At this point there was a relative change of $A(1)$ of less than 0.0001 between two iterations. The computer time to find the a_j for $g = 100$ and $h = 5$ is 0.2307 seconds (19 iterations). The method performs thus very well.

5. ALTERNATE METHODS

To gain a check for our results, and also to get further insight into the problem, we decided to try other methods as well. These methods are based on $\hat{V}(z) = V(z)/(z-1)$, that is $V(z)$ was deflated by $z-1$. $\hat{V}(z)$ can be written as

$$\hat{V}(z) = A(z)\hat{B}(z),$$

where $\hat{B}(z)$ is equal to $B(z)/(z-1)$. This deflation decreases the degree of the polynomials one works with by one, and it also increases the difference between the zeros of $A(z)$ and $B(z)$, improving thus the efficiency of the algorithms to be discussed. A further deflation by the only positive zero of $A(z)$ is possible.

To factorize $\hat{V}(z)$, two methods were used. The first one is given in [1, page 158]. The second one used the fact that $\hat{V}(z)$ can be interpreted as the characteristic polynomial of a difference equation. If these difference equations are used to calculate values x_n, x_{n+1}, x_{n+2} recursively, the zeros greater than one will dominate, and eventually, the effect of $\hat{B}(z)$ will become negligible. The x_n are thus almost identical to a series generated by a difference equation that has $A(z)$ as characteristic polynomial. In other words, the x_n will almost satisfy the following relationship

$$x_n = \sum_{j=1}^{h} a_j x_{n+j}. \tag{23}$$

If x_n is known for $n = k, k+1, \ldots, k+2h-1$, this gives h equations for the h unknown a_j.

The x_n will satisfy Equation (23) precisely if the initial conditions $x_0, x_1, \ldots, x_{h-1}$ satisfy (23). Consequently, one can repeat the above algorithm several times to gain a higher precision for the a_j. In each repetition, one uses the a_j obtained in the previous iteration in order to calculate the starting values needed. In the first iteration, one can use $a_j^0 = d_j$. This algorithm gave good results for most cases we tried.

The disadvantage of the two algorithms just mentioned is that they require the solution of h equations in h unknowns during each iteration, and this requires h^3 operations. This means that they are inefficient as compared to the algorithm suggested in the previous section, at least if h is high.

Of course, $A(z)$ can also be found, provided one knows all the zeros of $V(z)$ outside the unit circle. Let $z_1, z_2, z_3, \ldots, z_h$ be these zeros. $A(z)$ becomes, given one uses the fact that $a_0 = -1$

$$\begin{aligned} A(z) &= -(z-z_1)(z-z_2)\ldots(z-z_h)/[(-z_1)(-z_2)\ldots-(-z_h)] \\ &= -(1-z/z_1)(1-z/z_2)\ldots(1-z/z_h). \end{aligned}$$

The factors can now be multiplied in the usual way, (see e.g. [4]) giving

$$A(z) = -1 + a_1 z + a_2 z^2 + \ldots + a_h z^h.$$

Note that this procedure even works for multiple roots, provided, of course, all z_i are known.

A final comment shall now be made. Suppose one has the $h + g$ probabilities π_{k+1}, π_{k+2}, ..., π_{k+h+g}. Then, one can theoretically calculate π_m, $m > k + h + g$, recursively by using Equation (2), that is

$$\pi_{n+g} = \frac{1}{e_{-g}} \left[\pi_n - \sum_{k=-g+1}^{h} e_k \pi_{n-k} \right].$$

This recursion is numerically unstable, that is, the round-off errors will increase with each iteration. The characteristic equation of this difference equation is $V(1/x) = A(1/x)\ B(1/x)$, and any recursion based on it will eventually be dominated by $B(1/x)$. (Indeed, this very effect was used earlier to find $A(z)$.) Instead of doing such a recursion, it is better to use 2h subsequent π_n, and use (12) to obtain h equations for the a_j, $j = 1, 2, \ldots, h$.

6. THE SIGNIFICANCE OF THE $a_{i,j}$

This section proves Equations (21) and (22) and establishes the relationship of our method with the method of Neuts. We start to prove the following equation, in which the $a_{i,j}$ are calculated as given by (22)

$$\pi_{n+i} = \sum_{j=1}^{h} a_{i,j} \pi_{n-j} \cdot i \geqslant 0,\ n \geqslant r \tag{24}$$

Since $a_{0,j} = a_j$, this equation is certainly correct for $i = 0$. Moreover, if it is true for i, it is also true for $i + 1$. One has, if $a_{i,h+1} = 0$

$$\begin{aligned} \pi_{n+i+1} &= \sum_{j=1}^{h+1} a_{i,j} \pi_{n+1-j} = a_{i,1} \pi_n + \sum_{j=2}^{h+1} a_{i,j} \pi_{n+1-j} \\ &= a_{i,1} \sum_{j=1}^{h} a_j \pi_{n-j} + \sum_{j=1}^{h} a_{i,j+1} \pi_{n-j} \\ &= \sum_{j=1}^{h} [a_{i,1} a_j + a_{i,j+1}] \pi_{n-j}. \end{aligned}$$

Because of (22), this gives

$$\pi_{n+i+1} = \sum_{j=1}^{h} a_{i+1,j} \pi_{n-j}.$$

This proves that Equation (24) is correct for $i + 1$, given it holds for i. To prove Equation (21), one rewrites the queueing Equation (2) as

$$\pi_n = \sum_{i=1}^{q} e_{-i} \pi_{n+i} + \sum_{j=1}^{h} e_j \pi_{n-j}.$$

Using Equations (12) and (24) this gives

$$\begin{aligned} \sum_{j=1}^{h} a_j \pi_{n-j} &= \sum_{j=1}^{h} e_j \pi_{n-j} + \sum_{i=1}^{g} e_{-i} \sum_{j=1}^{h} a_{i,j} \pi_{n-j} \\ &= \sum_{j=1}^{h} [e_j + \sum_{i=1}^{g} e_{-i} a_{i,j}] \pi_{n-j}. \end{aligned}$$

Consequently,

$$a_j = e_j + \sum_{i=1}^{g} e_{-i} a_{i,j},$$

and Equation (21) is proven.

Equation (24) can also be used to simplify the initial conditions. Consider for instance Equation (4) which can be written as

$$\pi_n = \sum_{m=0}^{\infty} \pi_m p_{m,n} = \sum_{m=0}^{r-1} \pi_m p_{m,n} + \sum_{m=0}^{\infty} p_{r+m,n} \pi_{r+m}, \quad n < r.$$

If π_{r+m} is replaced by (24), one obtains after some calculation

$$\pi_n = \sum_{m=0}^{r-h-1} \pi_m p_{m,n} + \sum_{m=r-h}^{r-1} \pi_m [p_{m,n} + \sum_{k=0}^{\infty} p_{r+k,n} a_{k,r-m}], \quad 0 \leqslant n \leqslant r.$$

Together with Equation (14), this provides a set of r independent equations in the r unknowns $\pi_0, \pi_1, \ldots, \pi_{r-1}$.

We now compare our results with the ones obtained by Neuts. [3,4] Neuts sets up the following matrix equation for the unknown matrix R.

$$R = \sum_{n=0}^{\infty} R^n A_n. \tag{25}$$

The A_n are problem-dependent matrices. In our case, they become

$$A_n = [A_{i,j}^n], \quad n = 0, 1, 2, \ldots, [g/h] + 1$$

Here, $[g/h]$ is the lowest integer above g/h, and

$$A_{i,j}^n = e_{h-i+j-nh}.$$

All other A_n are zero. As before, all e_k outside the range $-g \leqslant k \leqslant h$ are assumed to be zero. If one defines

$$\pi^k = [\pi_{kh+1}, \pi_{kh+2}, \ldots, \pi_{kh+h}],$$

one has according to Neuts

$$\pi^{k+1} = \pi^k R.$$

If $R = [r_{i,j}]$, this gives

$$\pi_{(k+1)h+i} = \sum_{j=1}^{h} \pi_{kh+j} r_{j,i}$$

or

$$\pi_{n+i} = \sum_{j=1}^{h} \pi_{n+1-j} r_{h+1-j,i}.$$

When this equation is compared with (24), one finds

$$a_{i,j} = r_{h+1-j,i+1}.$$

In particular,

$$a_{0,j} = a_j = r_{h+1-j,1}.$$

The a_j are thus contained in the first column of R. We also note that Equation (21) follows from Equation (25), as the reader may verify.

To calculate R according to (25) one has to find the terms $R^n A_n$, $n = 2, 3, \ldots, [g/h] + 1$ and this can be done in $2[g/h]$ matrix multiplications. However, each matrix multiplication requires $2h^3$ operations, which means that there are $4h^3[g/h]$, or approximately $4h^2g$ operations per iteration to find R, not counting the matrix additions. Thus, by taking advantage of the special structure of the problem one obtains a more efficient algorithm, namely the one described in this paper.

The equivalence of the a_j with $r_{h+1-j,1}$ is of great importance. In particular, Neuts [2] proved that all $r_{i,j}$ are nonnegative. Consequently, all a_j are nonnegative. Neuts also finds that the $r_{i,j}$ have a probabilistic interpretation. Indeed, the $r_{i,j}$ are the expected number of visits to state $n + H + j$ starting at $n + 1$ without hitting $\{n + 1, \ldots, n + i, \ldots, n + h\}$. This means that a_j gives the expected number of visits to state $n + j$ starting at n without hitting $\{n + j - h, \ldots, n + j - 1\}$.

7. CONCLUSIONS

This paper derives an extremely efficient algorithm which can be used to numerically determine the state probabilities of many queueing problems efficiently and precisely. This algorithm was obtained a by combining the classical approach with the approach of Neuts. Indeed it bridges the gap between these two approaches and opens new horizons for further research.

ACKNOWLEDGMENT:

The authors wish to express thanks to the referee of this paper for his careful review. This paper was partially supported by NSERC Grant A8112, and by the Defence Board of Canada, Grant 3610-603.

REFERENCES

[1] Carnahan, B., H.A. Luther, and J.O. Wilkes, *Applied Numerical Methods* (John Wiley and sons, chic., New York 1969).

[2] Neuts, M.F., "The Probabilistic Significance of the Rate Matrix in Matrix-Geometric Invariant Vectors," Technical Report No. 47 B, Institute for Mathematical Sciences, University of Delaware, Newark, Delaware (1978).

[3] Neuts, M.F., "Markov Chains with Applications in Queueing Theory which Have a Matrix-Geometric Invariant Vector," Advances in Applied Probability, *10*, 185-212 (1978).

[4] Neuts, M.F., "An Algorithmic Solution to the *GI/M/c* Queue with Group Arrivals," Technical Report No. 78/2, Department of Statistics and Computer Science, University of Delaware, Newark, Delaware (1978).

[5] Ponstein, J., "Theory and Numerical Solution of a Discrete Queueing Problem," Statistica Neerlandica, *20*, 139-152 (1974).

INDIVIDUAL HEADSTART STRATEGIES FOR COMBATING CONGESTION

Attahiru Sule Alfa

Department of Civil Engineering
Ahmadu Bello University
Samaru-Zaria, Nigeria

ABSTRACT

This paper examines the process by which a user of a queueing system selects his arrival time to the system to compensate for unpredictable delays in the system if he wishes to complete service at a particular time. Considering the case in which all the system users have already decided on their arrival times to the system and will not change these times, this paper investigates how a new user of this system develops his strategy for selecting his arrival time. The distribution of this customer's arrival time is then obtained for a special case.

1. INTRODUCTION

If a user of a queueing system can choose when to enter the system, then he would probably choose to enter when the delays he experiences will be minimum. However, if he wishes to complete receiving service at a particular time, called the "target time," then he would have to arrive at the system early enough to allow for his service time and unpredictable delays in the system. This allowance was termed "headstart" by Gaver [3] and the same terminology will be adopted here. The strategy by which a user selects his headstart is studied in this paper.

Consider the case in which all the system users already have fixed headstart strategies and will not change these strategies under any condition. This paper investigates the case in which a new user who wishes to use the system develops a headstart strategy. This will be done by assuming that this user, whom we name U (cf. Alfa and Minh [2]), attaches some perceived costs to delays and to early and late completion of service, and that he selects his headstart in order to minimize his total cost. Gaver [3] obtained the "best" headstart strategy for this user, in this type of situation, on the assumption that the user wishes to minimize his expected total cost. This paper proposes the headstart strategy usually adopted by such a user, on the assumption that he selects his headstart each day in order to minimize the total cost he incurs. It is assumed that on the first day he uses the system he does not know what the delay distribution is and therefore selects his headstart arbitrarily, and hence probably incurs high total cost. However, for the following day, he uses his knowledge of the previous day's outcome to try and reduce his total cost; and this continues until a stage is reached in the long-run when he settles for a particular headstart and does not change any further. Let us call this stage "steady-state." This headstart reflects his arrival time at the system and our interest is in the distribution of his arrival time at the system at steady-state.

This paper does not restrict its analysis to nonrush hour traffic or linear cost parameters as done by Gaver [3].

2. PROBLEM FORMULATION

2.1 Assumptions and Definitions

Consider a queueing system used by a finite number I, $(I > 0)$, of customers who behave identically and independently. For each day, let us observe the system at equally spaced epochs sequentially numbered 0,1,2,3,... and assume that all the customers arrive at and depart from the system only at instants immediately prior to these epochs.

It will be assumed that once a customer is served he does not return to the pool of potential customers. This assumption and that about the number of customers being finite are not necessary for the analysis but they allow us to use the results from Alfa [1] and Minh [4].

Assume that each customer, considered separately, has the time-dependent probability λ_n of arriving at the system prior to epoch $n + 1$, given that he has not arrived at the system by epoch n $(n = 0,1,2,3,...)$. Assume that the service times of the successive customers are independent, identically distributed random variable $\mathbf{S}$ in the set $\{1, 2, \ldots, M\}$; $M < \infty$.

Suppose the new customer, U who also wishes to use this system, chooses to arrive at the system at epoch $\mathbf{a} = n$. Let the delay he experiences by so doing be $\mathbf{W}^n$. The distribution of $\mathbf{W}^n$ can be obtained as shown in Alfa [1]. Note that the addition of this customer U to the system has raised the total number of customers to $I + 1$. Although the distribution of U's arrival time is not necessarily the same as that of the other I customers, the results in Alfa [1] can still be applied keeping in mind that $\mathbf{W}^n$ is a conditional random variable. The distribution of U's arrival time shall be developed in this paper.

Let $\mathbf{V}^n$ be the time spent in the system by U, given that he entered the system at epoch n, then $\mathbf{V}^n = \mathbf{W}^n + \mathbf{S}$. Let $V_i^n \triangleq \Pr\{\mathbf{V}^n = i\}$.

For practical purposes it will be assumed that U shall arrive at the system only at an epoch between 1 and N; where $1 < N < \infty$.

Suppose U wishes to complete service at epoch τ, where for convenience $1 < \tau \leqslant N$. If he arrived at the system at epoch $\mathbf{a} = n$, got delayed $\mathbf{W}^n = i$ units of time and took $\mathbf{S} = j$ units of time for service then he would complete service at epoch $n + i + j$. One of his objectives is for $n + i + j$ to equal τ. However, if $n + i + j < \tau$ or $n + i + j > \tau$ then he is early by an amount $\tau - n - i - j$ or he is late by an amount $n + i + j - \tau$, respectively. He attaches a cost to either of the outcomes and these costs are not necessarily the same. In addition, he attaches a cost to the time spent in the system $i + j$. He would, therefore, incur a total cost $C(n, i + j)$, where $C(n, i + j)$ is the sum of the cost of time spent in the system and the cost for either earliness or lateness, depending on which is the outcome, keeping in mind that if $n + i + j = \tau$ then he only incurs the cost for his time in the system.

Let $\mathbf{C}_n$ be the total cost incurred by U given that he arrived at the system at epoch n where $\mathbf{C}_n$ can only assume the values in the set $\{C(n,i) | i = 1, 2, \ldots, (I + 1) \times M\}$. For brevity, $\mathbf{C}_n$ will be termed the cost associated with epoch n.

Let us define the following sets that will be used in the next subsection.

(i) $G \triangleq \{i | i = 1, 2, \ldots, (I + 1) \times M\}$

(ii) $L \triangleq \{i | i = 1, 2, 3, \ldots, N\}$

(iii) $L_n \triangleq \{i | i = 1, 2, 3, \ldots, n-1, n+1, \ldots, N\}$

(iv) $L_m^n \triangleq L_n \cap L_m$

If there are v epochs, $(1 \leqslant v \leqslant N-2)$, $(N > 2)$, labelled $m_1, m_2, \ldots, m_v$ for which $\mathbf{C}_m = \mathbf{C}_{m_1} = \mathbf{C}_{m_2} = \ldots = \mathbf{C}_{m_v}$ where $\forall m_r \in L_m^n$, $r = 1, 2, \ldots, v$, then let

(v) $L_{m,v}^n \triangleq \{i | i = m_1, m_2, \ldots, m_v; \forall m_r \in L_m^n, r = 1, 2, \ldots, v\}$.

Let

(vi) $\bar{L}_{m,v}^n \triangleq L_m - L_{m,v}^n$; i.e. $\bar{L}_{m,v}^n$ is the set difference of L_m and $L_{m,v}^n$.

2.2 The Model

2.2.1 The General Model

Suppose on the d^{th} day, $(d = 1, 2, \ldots)$, U arrived at the system at epoch $\mathbf{a}^d = n$. Let $a_n^d \triangleq \Pr\{\mathbf{a}^d = n\}$, $n \in L$. U would incur a total cost $\mathbf{C}_n$, and by definition

(1) $$\Pr\{\mathbf{C}_n = C(n, i)\} = V_i^n.$$

We assume that U's choice of arrival time epoch for one day depends entirely on the outcome of the previous day's choice of arrival time i.e., on the total cost incurred. U wishes to minimize his total cost. Hence, $\mathbf{a}^{d+1}$ is a decision variable with an action space L, and $\mathbf{a}^{d+1}$ depends on the cost incurred on day d. Given the action on the d^{th} day the action for the $(d+1)^{\text{th}}$ day, $\mathbf{a}^{d+1}$, is thus a Markov Chain.

Let $t_{n,m} \triangleq \Pr\{\mathbf{a}^{d+1} = m | \mathbf{a}^d = n\}$ be the transition probabilities of the chain. These transition probabilities define U's strategy.

U's intention is to choose an arrival time epoch that minimizes his total cost each day he uses the system, hence for the $(d+1)^{\text{th}}$ day, his strategy for choosing his arrival time epoch will be

(2) $$t_{n,n} = \Pr\{\mathbf{C}_n \leqslant \mathbf{C}_u; \forall u \in L_n\};$$

(3) $$t_{n,m} = \Pr\{\mathbf{C}_m < \mathbf{C}_w; \forall w \in L_m\} + \sum_{v=1}^{N-2} \frac{1}{v+1} \Pr\{\mathbf{C}_m = \mathbf{C}_{m_r} < \mathbf{C}_w; \forall m_r \in L_{m,v}^n; \forall w \in \bar{L}_{m,v}^n\};$$
$$\forall m \neq n.$$

The arguments leading to (2) and (3) above are as follows:

(a) If U arrived at the system at epoch n on the d^{th} day he will not change his arrival time to any other epoch on the $(d+1)^{\text{th}}$ day if he can not reduce his total cost by so doing. This leads to (2).

(b) If U arrived at the system at epoch n on the d^{th} day he will change his arrival time to epoch m on the $(d+1)^{\text{th}}$ day if this could reduce his total cost, provided there is no other epoch which has equal potential reduction in total cost as epoch m—this leads to the first term on the R.H.S. of (3). In addition, however, if there are v other epochs, other than n and m, at which U could choose to arrive at the system and achieve the same reduction in total cost as arriving at epoch m, then U would arrive at any of these v epochs or epoch m with the same probability, provided the total costs to be incurred at each of these epochs, including epoch m, is less than at any other epoch. This leads to the second term on the R.H.S. of (3). (2) and (3) above constitute the transition probabilities that define U's strategy.

Define an $N \times N$ matrix $\mathbf{T} = (t_{n,m})$ and an N vector $\mathbf{A}^d = [a_1^d, a_2^d, \ldots, a_N^d]$. Then U selects his $(d+1)^{\text{th}}$ day's headstart according to

$$\mathbf{A}^{d+1} = \mathbf{A}^d \times \mathbf{T}. \tag{4}$$

THEOREM 1: $\sum_{m=1}^{N} t_{n,m} = 1.$

PROOF: Consider one epoch n, $(n \in L)$, and the associated cost $\mathbf{C}_n$. If we compare this cost $\mathbf{C}_n$ to the costs associated with other epochs in the set L_n then $\mathbf{C}_n$ is either the minimum cost, one of the equally minimum costs or neither of the two. This can be stated as

$$\Pr\{\mathbf{C}_n \leqslant \mathbf{C}_u ; \forall u \in L_n\} + \Pr\{\mathbf{C}_n \not\leqslant \mathbf{C}_u ; \forall u \in L_n\} = 1. \tag{5}$$

Substituting $t_{n,n}$ for the first term on the L.H.S. of (5) gives

$$t_{n,n} + \Pr\{\mathbf{C}_n \not\leqslant \mathbf{C}_u ; \forall u \in L_n\} = 1. \tag{6}$$

The second term on the L.H.S. of (6) is the probability of having at least one epoch, in the set L_n, whose associated cost is less than $\mathbf{C}_n$. This implies that there is at least one epoch, other than n, whose associated cost is minimum. If there is only one such epoch then the probability of such event is given by

$$\sum_{\forall m \in L_n} \Pr\{\mathbf{C}_m < \mathbf{C}_w ; \forall w \in L_m\}.$$

If, however, there are v other epochs $(1 \leqslant \mathrm{v} \leqslant N-2)$, labelled $m_1, m_2, \ldots, m_{\mathrm{v}}$, $\forall m_r \in L_m^n$, $1 \leqslant r \leqslant \mathrm{v}$, such that $\mathbf{C}_m = \mathbf{C}_{m_1} = \mathbf{C}_{m_2} = \ldots = \mathbf{C}_{m_{\mathrm{v}}}$, then the probability of such event is given by

$$\sum_{\forall m \in L_n} \sum_{\mathrm{v}=1}^{N-2} \frac{1}{\mathrm{v}+1} \Pr\{\mathbf{C}_m = \mathbf{C}_{m_r} < \mathbf{C}_w ; \forall m_r \in L_{m,\mathrm{v}}^n, \forall w \in \bar{L}_{m,\mathrm{v}}^n\}.$$

The second term on the L.H.S. of (6) can thus be written as;

$$\Pr\{\mathbf{C}_n \nleq \mathbf{C}_u;\ \forall u \in L_n\} = \sum_{\forall m \in L_n} \Pr\{\mathbf{C}_m \prec \mathbf{C}_w;\ \forall w \in L_m\}$$

$$+ \sum_{\forall m \in L_n} \sum_{v=1}^{N-2} \frac{1}{v+1} \Pr\{\mathbf{C}_m = \mathbf{C}_{m_r} < \mathbf{C}_w;\ \forall m_r \in L^n_{m,v},\ \forall w \in \bar{L}^n_{m,v}\}$$

$$= \sum_{\forall m \in L_n} t_{n,m}. \tag{7}$$

Substituting this into (6) gives

$$t_{n,n} + \sum_{\forall m \in L_n} t_{n,m} = 1$$

i.e., $\sum_{m=1}^{N} t_{n,m} = 1$; and this proves Theorem 1.

REMARK: If $t_{n,n} = 1$ for any $n \in L$ then n is an absorbing state and the solution to Equation (4) is trivial. However, the occurrence of such a situation in real life would be quite rare—we therefore assume for our present problem that $0 \leqslant t_{n,m} < 1,\ \forall (n,m) \in L$.

LEMMA 1: If $t_{n_1,n_2} = 0,\ (n_1,n_2) \in L$, for any $n_1 \neq n_2$ and $t_{n_1,n_1} \neq 1$, then:

(a) either $t_{n_3,n_2} = 0,\ \forall n_3 \in L$, and n_2 is a transient state which can be deleted from the chain, or

(b) there exists at least one $n_4 \neq n_1$ such that $t_{n_4,n_2} > 0$ and therefore state n_2 can be reached from state n_1 via state n_4, hence, all $n \in L$ form an irreducible markov chain (Note that $n_4 \in L$).

PROOF: Lemma 1(a) is a compliment of Remark above and does not require a proof. For Lemma 1(b), if there exists $n_4 \neq n_1$, such that $t_{n_4,n_2} > 0$, then there also exists at least one N tuple $(i_1,i_2,\ \ldots,\ i_N)$ such that $C(n_4,i_4) > C(n_2,i_2) \leqslant C(n_3,i_3) \leqslant C(n_1,i_1) \leqslant C(n_k,i_k);\ \forall k > 4,\ \forall i_j \in G,\ 1 \leqslant j \leqslant N$.

Further let

(vii) $L_0 \triangleq \{i | t_{n,i} = 0;\ \forall n \in L\}$

(viii) $\bar{L}_0 \triangleq L - L_0$

(ix) $\bar{L}_{0,n} \triangleq \bar{L}_0 \cap L_n$ and let N_0 be the number of elements in $\bar{L}_0$.

Define an $N_0 \times N_0$ matrix $\mathbf{T}_0 = (t_{n,m}),\ \forall (n,m) \in \bar{L}_0$, and also define an N_0 vector $\mathbf{A}_0^d = (a_n^d);\ \forall n \in \bar{L}_0$. Equation (4) can now be stated in a modified form, for only the epochs contained in $\bar{L}_0$, as;

$$\mathbf{A}_0^{d+1} = \mathbf{A}_0^d \times \mathbf{T}_0. \tag{8}$$

LEMMA 2: $t_{n,n} > 0,\ \forall n \in \bar{L}_0$, hence all $n \in \bar{L}_0$ are aperiodic states.

PROOF: From the cost structure, it is apparent that if $t_{n,n} = 0$ then $t_{m,n} = 0$, $\forall m \in L$, and hence $n \notin \bar{L}_0$ i.e., state n would have been deleted from the chain considered in equation (8) (i.e., in $\bar{L}_0$). Therefore, $t_{n,n} > 0$, $\forall n \in \bar{L}_0$, and hence Lemma 2 follows.

THEOREM 2: There exists a limiting probability vector $\mathbf{A}_0 = \lim_{d \to \infty} \mathbf{A}_0^d$ such that

$$\mathbf{A}_0 = \mathbf{A}_0 \times \mathbf{T}_0 \tag{9}$$

has a positive solution which is unique.

PROOF: The necessary and sufficient conditions for the existence of a positive and unique solution $\mathbf{A}_0$ is that the chain should be irreducible and aperiodic. Both conditions were established by Lemma 1(b) and Lemma 2 respectively.

The interest of this paper is in U's choice of arrival time in the long run, $\mathbf{A}_0$, which can now be solved for in Equation (9).

2.2.2. A Special Case

In the general model it was assumed that if U arrived at epoch n on the d^{th} day, he would not change his arrival time to any other epoch if $\mathbf{C}_n \leqslant \mathbf{C}_m$, $\forall m \in L_n$. Suppose we modify this assumption and now assume that when $\mathbf{C}_n = \mathbf{C}_m$ for any $m \neq n$ we would not rule out the possibility of U changing his arrival time to epoch m. In that case we shall attach equal chances of U changing his arrival time to epoch m and to him not changing his arrival time from epoch n. Thus, U's strategy will now be modified such that the transition probabilities that define his strategy will be given by:

$$t_{n,m} = \Pr\{\mathbf{C}_m < \mathbf{C}_u;\ \forall u \in L_m\} + \sum_{v=1}^{N-1} \frac{1}{v+1} \Pr\{\mathbf{C}_m = \mathbf{C}_{m_r} < \mathbf{C}_w;\ \forall m_r \in L_{m,v};\ \forall w \in \bar{L}_{m,v}\};\ \forall n,m \in L, \tag{3a}$$

where

(x) $L_{m,v} \triangleq \{i | i = m_1, m_2, \ldots, m_v\}$ such that $\mathbf{C}_m = \mathbf{C}_{m_1} = \mathbf{C}_{m_2} = \ldots = \mathbf{C}_{m_v}$; $1 \leqslant v \leqslant N-1$, and $L_{m,v} \subseteq L$; and

(xi) $\bar{L}_{m,v} \triangleq L_m - L_{m,v}$.

It is immediately apparent that the right hand side of Equation (3a) is independent of n, hence $t_{n,m} = t_{v,m}$; $\forall (n,m,v) \in L$. Let $t_m \triangleq t_{n,m}$; $\forall (n,m) \in L$. For this special case, therefore, all the rows of $\mathbf{T}$ are identical, and the steady state solution is given by

$$a_n = \begin{cases} t_n\ ;\ \forall n \in \bar{L}_0 \\ 0\ \forall n \in L_0. \end{cases} \tag{10}$$

2.2.3. Illustrative Example

For illustrative purpose let,

$$I = 2,\ N = 3,\ \tau = 3,\ M = 1,$$
$$\lambda_0 = 0.3,\ \lambda_1 = 0.5,\ \lambda_2 = 1.0.$$

Let;

$$C(n,i) = C_\omega \times (i) + \begin{cases} C_e \times (\tau - n - i), & n + i \leqslant \tau \\ C_l \times (n + i - \tau), & n + i \geqslant \tau \end{cases} \tag{11}$$

and let $C_e = 1.0,\ C_l = 1.0,\ C_\omega = 2.0$.

By using results from Minh [4] we obtain the distribution of queue lengths and from Alfa [1], the distribution of waiting is obtained as

W_i^n

i	n		
	1	2	3
0	0.73	0.60	0.57
1	0.24	0.36	0.39
2	0.03	0.04	0.04

Hence, $V_i^n = W_{i-1}^n,\ i = 1, 2, 3$.

The total cost $C(n,i)$ are given as

$C(n,i)$

i	n		
	1	2	3
1	3	2	3
2	4	5	6
3	7	8	9

The transition probabilities are thus given as

$$t_{1,1} = V_1^1[V_2^2 + V_3^2] + V_2^1[V_2^2 + V_3^2][V_2^3 + V_3^3] + V_3^1V_3^2V_3^3, \tag{12}$$

$$t_{1,2} = V_1^2 + V_2^2V_3^1[V_2^3 + V_3^3], \tag{13}$$

$$t_{1,3} = V_1^3[V_2^1 + V_3^1][V_2^2 + V_3^2] + V_2^3V_3^1V_3^2, \tag{14}$$

$$t_{2,1} = V_1^1[V_2^2 + V_3^2][V_2^3 + V_3^3] + V_2^1[V_2^2 + V_3^2][V_2^3 + V_3^3] + V_3^1V_3^2V_3^3 + V_1^1[V_2^2 + V_3^2]V_1^3/2, \tag{15}$$

$$t_{2,2} = V_1^2 + V_2^2V_3^1[V_2^3 + V_3^3], \tag{16}$$

$$t_{2,3} = V_1^3V_2^1[V_2^2 + V_3^2] + [V_1^3 + V_2^3]V_3^2V_3^1 + V_1^3V_1^1[V_2^2 + V_3^2]/2, \tag{17}$$

$$t_{3,1} = [V_1^1 + V_2^1][V_2^2 + V_3^2]V_2^3 + [V_1^1 + V_2^1][V_2^2 + V_3^2]V_3^3 + V_3^1 V_3^2 V_3^3, \tag{18}$$

$$t_{3,2} = V_1^2 + V_2^2 V_3^1 [V_2^3 + V_3^3], \tag{19}$$

$$t_{3,3} = V_1^3 [V_2^2 + V_3^2] + V_3^2] + V_2^3 V_3^2 V_3^1. \tag{20}$$

Thus, this leads to the transition matrix,

$$\mathbf{T} = \begin{bmatrix} 0.333 & 0.605 & 0.062 \\ 0.256 & 0.605 & 0.139 \\ 0.167 & 0.605 & 0.228 \end{bmatrix} \tag{21}$$

from which we obtain

$$\mathbf{A} = [0.265,\ 0.605,\ 0.130]. \tag{22}$$

Note that for this example $L_0 = L$, hence $\mathbf{T}_0 = \mathbf{T}$ and $\mathbf{A}_0 = \mathbf{A}$.

3. CONCLUSION

This model, based on its assumptions, can only be used to predict the customer's arrival time at the system when all other customers previously using the system have settled for a particular arrival time and will not change it under any circumstances. However, when all the other customers change their arrival times as a result of the new customer's "interference" then there will be a slight modification to the problem.

If other customers change their arrival times everyday, then the distribution of $\mathbf{W}^n$ will change from day to day and, hence, $\mathbf{T}$ and $\mathbf{T}_0$ will also change from day to day. If we let $\mathbf{T}^d$ or $\mathbf{T}_0^d$ represent the transition matrix for the strategy for the $(d+1)^{\text{th}}$ day then U's choice of arrival time can be reformulated as

$$\mathbf{A}^{d+1} = \mathbf{A}^d \times \mathbf{T}^d. \tag{23}$$

However, with (23) existence of a steady state solution cannot be assured. If we further let there be more than one server in parallel such that a customer develops a strategy not just for selecting his arrival time only but for joint selection of arrival time and of the server to serve him, then the problem becomes similar to that in Alfa [2].

REFERENCES

[1] Alfa, A.S., "A Numerical Method for Evaluating Delay to a Customer in a Time-Inhomogeneous, Single Server Queue with Batch Arrivals," Journal of the Operational Research Society, *30*, 7, 665-667 (1979).

[2] Alfa, A.S., "A Strategy for Combined Selection of Travel Route and Departure Times from Home During the Peak Period—A Theoretical Model," in Proceedings of the International Conference on Transportation, New Delhi, India, November 26-28, 1980. N.K. Jaiswal, Editor, 356-362 (1980).

[3] Gaver, D.P. Jr., "Headstart Strategies for Combating Congestion," Transportation Science *2*, 2. 172-181 (1968).

[4] Minh, D.L., "Discrete Time, Single Server Queue from a Finite Population," Management Science, *23*, 7, 756-767 (1977).

PLANNING HORIZON PROCEDURES FOR MACHINE REPLACEMENT MODELS WITH SEVERAL POSSIBLE REPLACEMENT ALTERNATIVES

Suresh Chand

Purdue University
West Lafayette, Indiana

Suresh Sethi

Faculty of Management Studies
University of Toronto
Toronto, Canada

ABSTRACT

This paper develops a forward algorithm and planning horizon procedures for an important machine replacement model where it is assumed that the technological environment is improving over time and that the machine-in-use can be replaced by any of the several different kinds of machines available at that time. The set of replacement alternatives may include (i) new machines with different types of technologies such as labor- and capital- intensive, (ii) used machines, (iii) repairs and/or improvements which affect the performance characteristics of the existing machine, and so forth.

The forward dynamic programming algorithm in the paper can be used to solve a finite horizon problem. The planning horizon results give a procedure to identify the forecast horizon T such that the optimal replacement decision for the first machine based on the forecast of machine technology until period T remains optimal for any problem with horizon longer than T and, for that matter, for the infinite horizon problem. A flow chart and a numerical example have been included to illustrate the algorithm.

1. INTRODUCTION

In our previous paper [5], we developed forward algorithm and planning horizon procedures for a machine replacement model under the assumption that only one kind of machine is available for replacement in any given period. We showed that there exists a forecast horizon T, such that the optimal replacement decision for the first period (based on the forecast of machine technology until period T remains optimal for any longer (than T) horizon and, for that matter, the infinite horizon.

In this paper, we relax the assumption of a single possible replacement alternative by multiple alternatives. That is, we develop a model in which the machine-in-use can be replaced by any of the several different kinds of machines available at that time. This model can deal with

many realistic situations. For example, the set of replacement alternatives may include (i) new machines with different types of technologies such as labor- and capital- intensive, (ii) used machines, (iii) repairs and/or improvements which affect the performance characteristics of the existing machine, and so forth. It should be noted that the set of replacement alternatives can change from period to period.

This multiple alternative replacement model represents a major generalization of the various single alternative replacement models developed by Terborgh [7], Thompson [8], Sethi and Morton [6], Gordon [1]. For a survey of the extensive literature on the machine replacement models, the reader is directed to Rapp [4]; see also Pierskalla and Voelker [3].

In the next section, we formulate the multiple alternative replacement model under improving technologies. In section 3, we develop an efficient forward algorithm for solving finite horizon problems. Section 4 develops a variant of the regeneration- monotonicity property [2, 5] and planning horizon procedures for the model. A flow chart summarizes the complete solution procedure. A numerical example is solved in Section 5 to illustrate the steps of the forward algorithm and the application of the planning horizon procedure. We weaken the assumption of improving technologies in Section 6 and show that planning horizon procedures can be adapted to this case. Section 7 concludes the paper with some important remarks.

2. MODEL FORMULATION

Consider the situation of a production shop which must keep a single machine of a particular capacity at all times. To run this machine, the shop incurs operating expenses which may include labor cost, electricity cost, maintenance cost, depreciation, and so forth. Usually, the performance of the machine deteriorates, i.e., operating expenses increase over time, so that the shop might consider selling the existing machine for its salvage value and buying a new one. This new machine is selected from various different alternative machines available. If a major repair is decided on the existing machine, then this alternative can be considered as if a new machine of another technology is purchased at the cost of repair plus the salvage value of the existing machine. Once the new machine is purchased, the same situation repeats with the new machine. Consequently, for any finite or infinite horizon, the shop will make a chain of replacement decisions [4], [5]. The problem of the shop, of course, is to find simultaneously the optimal times of these replacements and the types of machines selected at these times. Obviously, these decisions will depend on the prices of the future machines and their per period operating expenses over time. More precisely, we are considering the following model which the shop must solve:

Let A_t^h denote the machine of technology h available at time t. Let there be machines of N alternative technologies available to choose from in any period. It is noted that the extension to the case, when the set of alternative technologies is changing over time, is straightforward and will be described in Section 7.

Let $O_{t,k}^h$, $k \geq t$, be the operating cost of the machine A_t^h in period k. With reference to our previous paper [5], we note that

$$O_{t,t}^h = \pi_t^h - S_{t,t}^h + M_{t,t}^h$$

and

$$O_{t,k}^h = S_{t,k-1}^h - S_{t,k}^h + M_{t,k}^h$$

where π_t^h is the purchase price (it is assumed that the new machine A_t^h is purchased at the beginning of period t) and $S_{t,k}^h$ is the salvage value at the end of period k of the machine A_t^h. The term $M_{t,k}^h$ denotes all the costs, which are specific to the technology h, of operating the machine A_t^h in period k excluding the cost of depreciation (or loss in salvage value from period $k-1$ to k).

To obtain a forward algorithm, we must solve the multiple alternative replacement model for any given finite horizon T. Suppose the shop uses n machines during the interval $<1,T>$. Let $t_1, t_2, \ldots, t_{n-1}, t_n$ be their salvage times and let $h_1, h_2, \ldots, h_n$ be their technologies. Note that $t_i \geq 1$ for $i = 1, 2, \ldots, (n-1)$ and $t_n = T$, since we assume that the shop goes out of business at the end of the given finite horizon T. Define $t_0 = 0$. Also note that $<p,q> = \{p, p+1, \ldots, q\}$ with nonnegative integers p and q with $q \geq p$. We can now state the finite horizon problem as

$$(1) \qquad \text{Minimize} \quad \sum_{i=0}^{n-1} \sum_{k=t_i+1}^{t_{i+1}} O_{t_i+1,k}^{h_{i+1}}$$

$$n, t_1, t_2, \ldots, t_{n-1}$$

$$h_1, h_2, \ldots, h_n.$$

In the next section, we develop a forward dynamic programming algorithm to solve (1).

3. FORWARD ALGORITHM

We start with defining the following terms:

Purchase Point. A period t is defined to be a purchase point (or P-point) if a machine is purchased at the beginning of period t.

Regeneration Point. A period t is a regeneration point (or R-point) if a machine is salvaged at the end of period t.

It is easy to see that any P-point is immediately preceded by an R-point. This property allows us to develop an efficient forward algorithm [2].

We need to introduce the following notation:

$C(T)$ = minimum cost of the T-period problem (1),

$C^h(j,T)$ = the total operating cost for the machine

(2) A_{j+1}^h in the interval $<j+1,T>$

$$= \sum_{k=j+1}^{T} O_{j+1,k}^h$$

$C_j^h(T)$ = minimum total cost of the T-period problem when $(j+1)$ is the last P-point and h is the technology of the last machine

(3) $$= C(j) + C^h(j,T)$$

$C_j(T)$ = minimum total cost of the T-period problem with $(j+1)$ as the last purchase period.

Clearly,

$$C_j(T) = \min_h C_j^h(T) \tag{4}$$

and

$$C(T) = \min_{j \in \langle 0, T-1 \rangle} C_j(T) \tag{5}$$

Equations (2)-(5) complete the statement of the forward algorithm. It is possible to simplify the computation of $C_j^h(T)$; using (2) in (3) gives the following sequential procedure

$$C_j^h(T) = \begin{cases} C_j^h(T-1) + O_{j+1,T}^h & \text{for } j < T-1 \\ C(T-1) + O_{T,T}^h & j = T-1 \end{cases} \tag{6}$$

4. PLANNING HORIZON RESULTS

To obtain planning horizon results, we need to assume that every technology is an improving technology. Thus, for technology h, we assume that

$$O_{j-1,k}^h \geq O_{j,k}^h \text{ for } k \geq j+1. \tag{7}$$

In other words, we assume that the operating cost in a period for a machine of technology h is lower than the operating cost for an older machine of the same technology, except perhaps in the first period of operation of the new machine when the amount of depreciation on the new machine is usually high.

Lower Bound Monotonicity Property:

Let $j^*(T)$ denote the latest next to the last R-point in the optimal solution of the T-period subproblem; note that period T is the last R-point. By the *lower bound monotonicity property* we mean that the lower bound of $j^*(T)$ increases monotonically with T. To prove this property, we must also define $j^{*h}(T)$ to be the latest next to the last R-point for the optimal solution of the $C^h(T)$-problem, where the $C^h(T)$-problem is the T-period problem subject to the constraint that the last machine be machine of technology h.

THEOREM 1: Under the assumption (7) of improving technology over time, $j^{*h}(T)$ satisfies the regeneration monotonicity property, i.e.,

$$j^{*h}(T+1) \geq j^{*h}(T). \tag{8}$$

PROOF: Let $j^{*h}(T+1) = a$ and $j^{*h}(T) = b$. It is easy to see that $C_a^h(T) \geq C_b^h(T)$. Furthermore, if $a < b$ then $O_{a+1,T+1}^h \geq O_{b+1,T+1}^h$.

If $a < b$, then using (6) we can write

$$\begin{aligned} C_a^h(T+1) &= C_a^h(T) + O_{a+1,T+1}^h \\ &\geq C_b^h(T) + O_{b+1,T+1}^h \\ &= C_b^h(T+1). \end{aligned}$$

This contradicts the optimality of the $C_a^h(T+1)$-solution, implying that $a \geqslant b$. This completes the proof.

THEOREM 2: If we define

$$\lambda(T) = \min_h \{j^{*h}(T)\}, \tag{9}$$

then $\lambda(T)$ represents a lower bound of $j^*(T+1)$ and $\lambda(T)$ is monotonically increasing in T, i.e.,

$$j^*(T+1) \geqslant \lambda(T) \geqslant \lambda(T-1). \tag{10}$$

PROOF: The proof easily follows from (8) and (9).

Regeneration Set:

The regeneration set of the $C(T)$-problem can now be defined as

$$R(T) - set = \{\lambda(T), \lambda(T)+1, \ldots, T-1\}. \tag{11}$$

We note that a set is known as an $R(T)$-set if it contains at least one R-point of an optimal solution to any problem with horizon T or longer.

In many cases, it is possible to reduce the size of the $R(T)$-set. This reduction makes it easier to obtain planning horizons as can be seen in the planning horizon theorem below. The reduction procedure requires computations of the $S^h(T)$-sets for all h, where an $S^h(T)$-set is defined to be a set of regeneration points such that $j^{*h}(T+l) \in S^h(T)$-set for all periods $(T+l)$ with $j^{*h}(T+l) < T$ and $l \geqslant 0$. It is convenient to express this procedure by the flow chart on the next page.

The reduced $R(T)$-set can be obtained as follows:

$$R(T)\text{-set} = \bigcup_h S^h(T)\text{-set}. \tag{12}$$

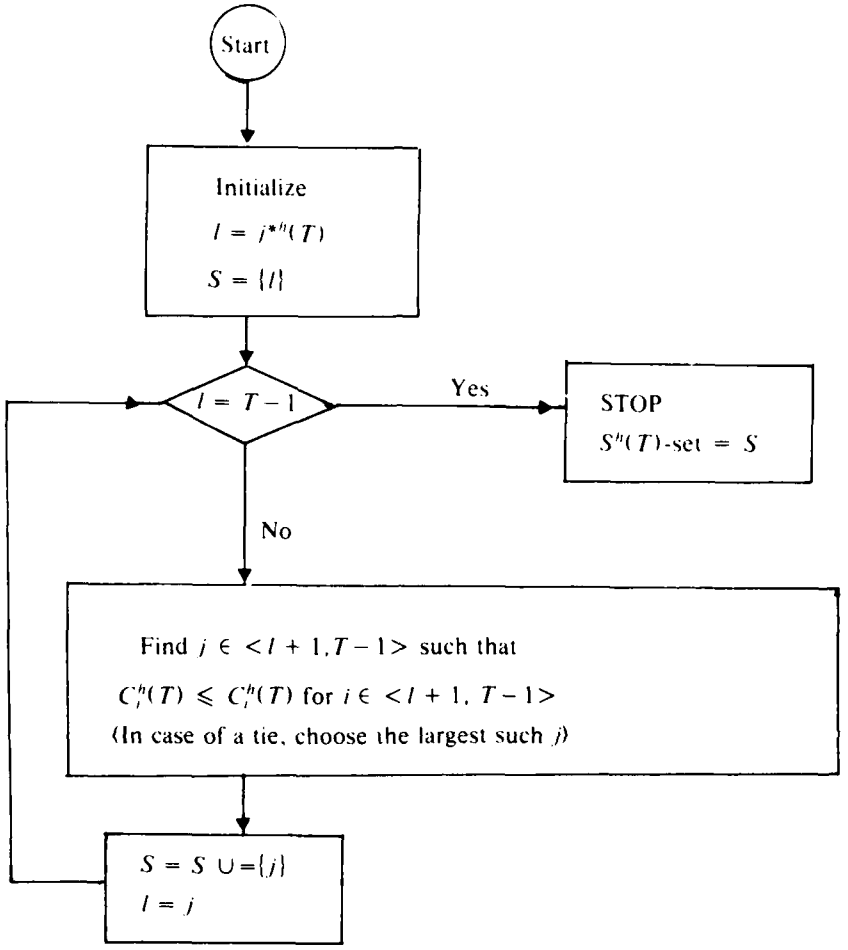
It should be noted that the $R(T)$-set in (12) cannot be bigger than the $R(T)$-set in (11). We now prove the following important result for the set S found in the flow chart.

THEOREM 3: If a period $t \notin$ the S-set found in the flow chart for technology h, then $j^{*h}(T+l) \neq t$ for any value of $l \geqslant 0$.

PROOF: If $t \notin$ the S-set found in the above flow chart, then there is a period u, $T-1 \geqslant u > t$, such that $C_u^h(T) \leqslant C_t^h(T)$. From the improving technology assumption, we also have $O_{u+1,k}^h \leqslant O_{t+1,k}^h$ for $k \geqslant T+1$. We can now write for any period $T+l$:

$$\begin{aligned} C_u^h(T+l) &= C_u^h(T) + \sum_{k=T+1}^{T+l} O_{u+1,k}^h \\ &\leqslant C_t^h(T) + \sum_{k=T+1}^{T+l} O_{t+1,k}^h \\ &= C_t^h(T+l), \end{aligned}$$

implying that $j^{*h}(T+l) \neq t$. This completes the proof.



Flow Chart for Computing the $S^h(T)$-set.

Planning Horizon Theorem:

Let $f^*(r)$ denote the period when the first machine is salvaged in an optimal $C(r)$-solution. We now state the planning horizon theorem which gives us a stopping rule for the forward algorithm to find the optimal replacement time in an infinite horizon problem.

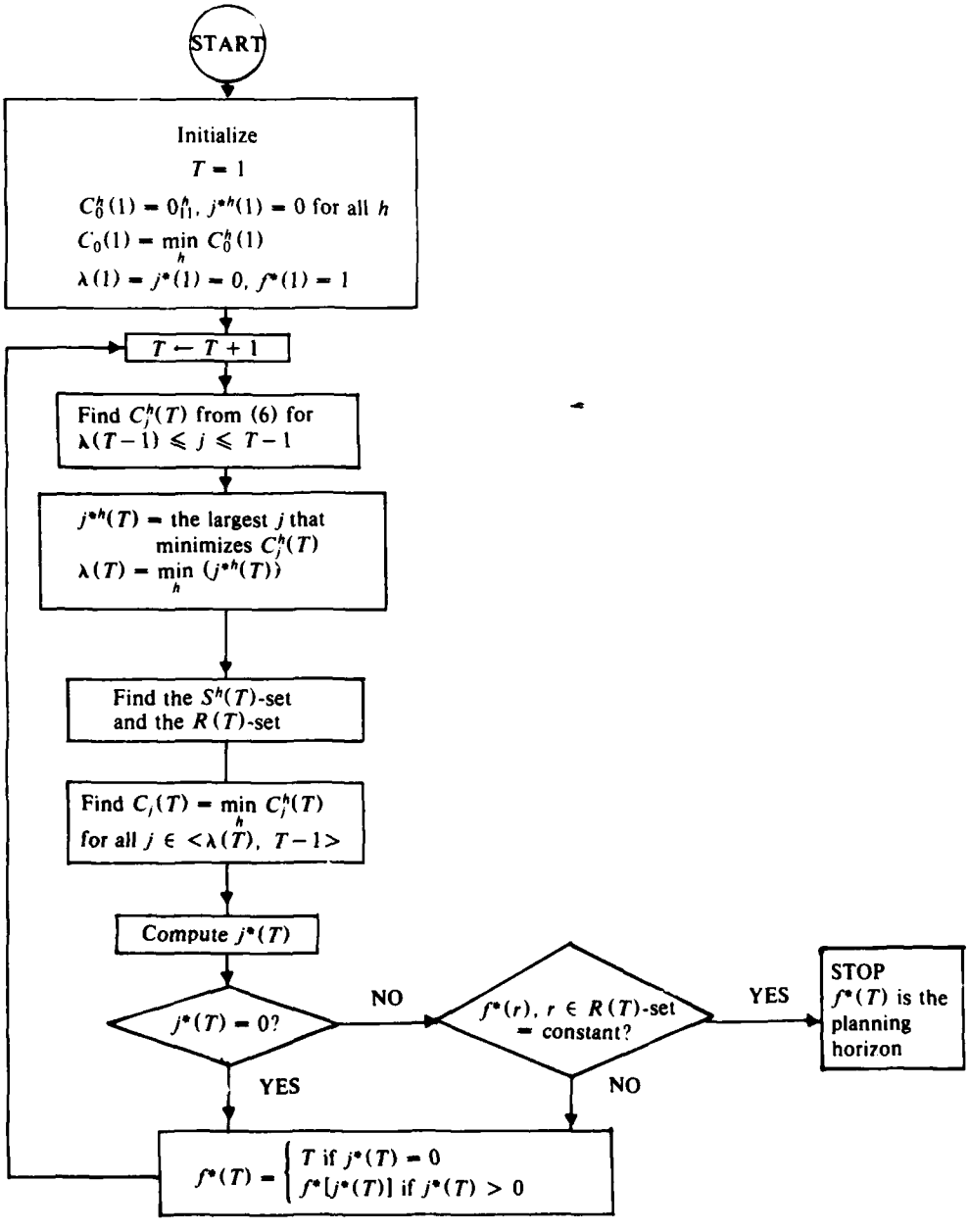
THEOREM 4 (Planning Horizon Theorem): If $f^*(r) = t_1^* =$ constant for all $r \in R(T)$-set (12) for some T such that $j^*(T) \neq 0$, then the salvage time of the first machine in an optimal infinite horizon solution is t_1^*. Furthermore, the optimal technology h_1^* is determined by the condition

$$C_0^{h_1^*}(t_1^*) = C_0(t_1^*). \tag{13}$$

The first replacement time t_1^* is called the planning horizon and T is called the forecast horizon. We need only to forecast the relevant technology up to period T to find the optimal

technology h_1^* and the optimal replacement time t_1^* for the first machine. The reader is referred to [2] for a discussion of such theorems.

We now present a flow chart for the complete forward algorithm along with the planning horizon procedure. We will use this flow chart to solve a numerical example in the next section.



Flow Chart for Implementing the Planning Horizon Procedure

5. NUMERICAL EXAMPLE

We now illustrate the planning horizon procedure by solving a numerical example. This example assumes that machines of two different technologies are available for replacement at any time. The operating costs O_{ij}^1 and O_{ij}^2 are as shown below:

O_{ij}^1

i \ j						
	160	120	130	140	150	160
		160	120	130	140	150
			140	105	115	125
				140	105	115
					120	80
						120

O_{ij}^2

i \ j						
	300	60	70	80	90	100
		300	60	70	80	90
			200	50	60	70
				400	30	40
					400	30
						200

Note that technology 1 can be interpreted as a labor intensive technology and technology 2 can be interpreted as a capital intensive technology. The calculations of $C_j^1(T)$, $C_j^2(T)$, and $C_j(T)$ are shown below:

$C_j^1(T)$

T	1	2	3	4	5	6
$j^{*1}(T)$	0	0	0	2	4	5
0	160	280	410	550	700	
1		320	440	570	710	
2			420	525	640	765
3				550	655	770
4					630	710
5						710

(j)

$C_j^2(T)$

T	1	2	3	4	5	6
$j^{*2}(T)$	0	0	0	0	2	2
0	300	360	430	510	600	
1		460	520	590	670	
2			480	530	590	660
3				810	840	870
4					910	940
5						790

(j)

$C_j(T)$

$\lambda(T)^T$	1	2	3	4	5	6
	0	0	0	0	2	2
$j^*(T)$	0	0	0	0	2	2
$f^*(T)$	1	2	3	4	2	2
0	160	280	410	510	600	
1		320	440	570	670	
2			420	525	590	660
3				550	655	770
4					630	710
5						710

(j)

Sample calculations for $T = 3$ are shown below:

$$C_0^1(3) = O_{11}^1 + O_{12}^1 + O_{13}^1 = 160 + 120 + 130 = 410,$$

$$C_1^1(3) = C(1) + O_{22}^1 + O_{23}^1 = 160 + 160 + 120 = 440,$$

$$C_2^1(3) = C(2) + O_{33}^1 = 280 + 140 = 420,$$

$$j^{*1}(3) = 0.$$

$$C_0^2(3) = O_{11}^2 + O_{12}^2 + O_{13}^2 = 300 + 60 + 70 = 430,$$

$$C_1^2(3) = C(1) + O_{22}^2 + O_{23}^2 = 160 + 300 + 60 = 520,$$

$$C_2^2(3) = C(2) + O_{33}^2 = 280 + 200 = 480,$$

$$j^{*2}(3) = 0,\ \lambda(3) = \min\{j^{*1}(3),\ j^{*2}(3)\} = 0,$$

$$C_0(3) = \min(C_0^1(3),\ C_0^2(3)) = 410,$$

$$C_1(3) = \min(C_1^1(3),\ C_1^2(3)) = 440,$$

$$C_2(3) = \min(C_2^1(3),\ C_2^2(3)) = 420,$$

$$j^*(3) = 0,\ f^*(3) = 3.$$

The planning horizon theorem is not satisfied for $T \leqslant 5$. For example, for $T = 5$, we have $S^1(5) = \{4\}$, $S^2(5) = \{2,3,4\}$, and $R(5) = \{2,3,4\}$. Since $f^*(2) \neq f^*(3) \neq f^*(5)$, the planning horizon theorem is not satisfied. For $T = 6$, we have $s^1(6) = \{5\}$, $S^2(6) = \{2,5\}$, and $R(5) = \{2,5\}$. $f^*(2) = f^*(5) = 2$, so the planning horizon is 2 periods and the forecast horizon is 6 periods.

6. RELAXING THE IMPROVING TECHNOLOGY ASSUMPTION

Let $t(h) \geqslant 1$ be a number associated with technology h such that

$$O_{i-1,j+k}^h \geqslant O_{j,j+k}^h \quad \text{for } k \geqslant t(h).$$

This condition is likely to be satisfied by most improving technologies. The following result holds for this assumption:

THEOREM 5: Let $i^{*h}(T) \leqslant T - t(h)$ (0 if $T \leqslant t(h)$) be the latest period such that $C_i{}^{*h_{(T)}} \leqslant C_j(T)$ for all $j \in <0,\ T - t(h)>$, then $j^{*h}(T+1) \geqslant i^{*h}(T)$.

PROOF: Assume to the contrary that $j^{*h}(T+1) = a < i^{*h}(T)$. It is easy to see that $O_{a+1,T+1}^h \geqslant O_i^h{}^{*h}(T) + 1,\ T+1$ and $C_a^h(T) \geqslant C_i^h{}^{*h_{(T)}}(T)$. We can now write

$$\begin{aligned} C_i^h{}^{*h_{(T)}}(T+1) &= C_i^h{}^{*h_{(T)}}(T) + O_i^h{}^{*h_{(T)+1,T+1}} \\ &\leqslant C_a^h(T) + O_{a+1,T+1}^h \\ &= C_a^h(T+1), \end{aligned}$$

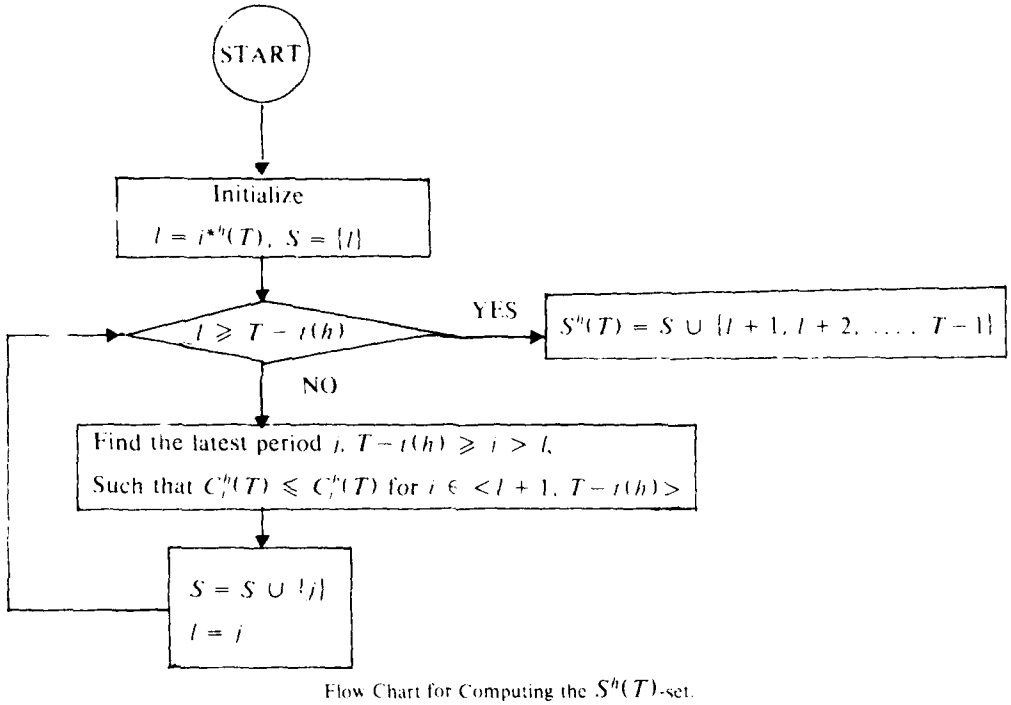
implying that $j^{*h}(T+1) \geqslant i^{*h}(T)$.

Regeneration Set.

Let $\lambda(T) = \min_h [i^{*h}(T)]$, then the regeneration set of the $C(T)$-problem can be defined as

$$R(T) - \text{set} = \{\lambda(T),\ \lambda(T)+1,\ \ldots,\ T-1\}.$$

As in Section 4, it is possible to find a reduced regeneration set. For this we first find the $S^h(T)$-set as below. The reduced regeneration set can be found by using (13).



Flow Chart for Computing the $S^h(T)$-set.

7. EXTENSIONS AND CONCLUDING REMARKS

In this paper we have solved the infinite horizon replacement model with multiple alternatives. We have obtained planning and forecast horizons for this model under reasonably general conditions. Furthermore, the model can be easily adapted to situations where the set of alternative technologies is changing over time. This is done by setting $O_{t,t}^h = \infty$ for the imaginary A_t^h machine for all $j < t$ for a technology which appears at time t for the first time. If a technology h disappears in period t, then we let $O_{t,t}^h = \infty$ for fictitious machines $A_{t,t}^h$ for all $j \geq t$. With these definitions, the model developed in the paper is applicable to the situation of changing sets of technologies.

Another situation to which the model can be easily adapted is the situation in which a switch over cost k^{ab} occurs whenever a machine of technology a is replaced with a machine of technology b. For this, we do require the assumption that k^{ab} is separable, i.e.,

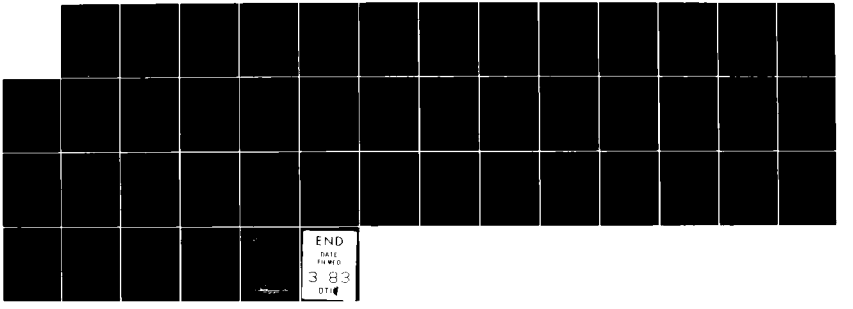
$$k^{ab} = k^a + k^b.$$

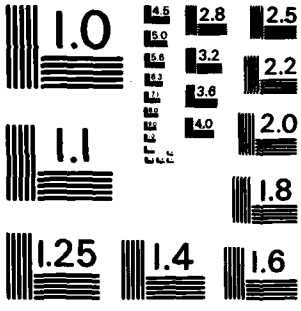
In this case, we can adjust the salvage value of the existing machine downward by amount k^a and adjust the price of the new machine upward by amount k^b.

Finally, we must state that the extensions of the machine replacement model in our previous paper [5] can also be solved in the multiple alternative case in a fairly straightforward manner.

AD A125 206 NAVAL RESEARCH LOGISTICS QUARTERLY VOLUME 29 NUMBER 3 (U) OFFICE OF NAVAL RESEARCH ARLINGTON VA H SOLOMON SEP 82 2/2

UNCLASSIFIED F/G 15/5 NL





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS - 1963 - A

REFERENCES

[1] Gordon, M.J., "Optimal Timing of Captital Expenditures," *Financial Decision Making under Uncertainty*, H. Levy and M. Sarnat (Academic Press, New York, 1977).
[2] Lundin, R.A. and T.E. Morton, "Planning Horizons for the Dynamic Lot Size Model, Zabel vs. Protective Procedures and Computational Results," Operations Research, *23*, 4, 711-734 (1975).
[3] Pierskala, W.P. and J.A. Voelker, "A Survey of Maintenance Models: The Control and Surveillance of Deteriorating Systems," Naval Research Logistics Quarterly, *23*, 3, 353-388 (1976).
[4] Rapp, B., *Models for Optimal Investment and Maintenance Decisions* (Halsted Press, Stockholm, 1974).
[5] Sethi, S. and S. Chand, "Planning Horizon Procedures for Machine Replacement Models," Management Science, *25*, 2, February 1979.
[6] Sethi, S.P. and T.E. Morton, "A Mixed Optimization Technique for the Generalized Machine Replacement Problem," Naval Research Logistics Quarterly, *19*, 3, 471-481 (1972).
[7] Terborgh, G., *Dynamic Equipment Policy* (McGraw-Hill, New York, 1949).
[8] Thompson, G.L., "Optimal Maintenance Policy and Sale Date of a Machine," Management Science, *14*, 9, May 1978.

FLOWSHOP/NO-IDLE OR NO-WAIT SCHEDULING TO MINIMIZE THE SUM OF COMPLETION TIMES

I. Adiri and D. Pohoryles

TECHNION-Israel Institute of Technology
Haifa, Israel

ABSTRACT

This paper deals with flowshop/sum of completion times scheduling problems, working under a "no-idle" or a "no-wait" constraint, the former prescribes for the machines to work continuously without idle intervals and the latter for the jobs to be processed continuously without waiting times between consecutive machines. Under either of the constraints the problem is unary NP-Complete for two machines. We prove some properties of the optimal schedule for $n/2/F$, no-idle/ΣC_i. For $n/m/P$, no-idle/ΣC_i and $n/m/P$, no-wait/ΣC_i with an increasing or decreasing series of dominating machines, we prove theorems that are the basis for polynomial bounded algorithms. All theorems are demonstrated numerically.

INTRODUCTION

The nonpreemptive, flowshop, sum of completion-times scheduling problem is: n jobs $(J_1, J_2, \ldots, J_n)$ have to be processed by m machines $(M_1, M_2, \ldots, M_m)$. Job J_i, $i = 1, 2, \ldots, n$, consists of, at most, m operations $(0_{i1}, 0_{i2}, \ldots, 0_{im})$. Operation 0_{ij} which precedes $0_{i,j+1}$, has to be processed uninterrupted for t_{ij} time units, on M_j, $j = 1, 2, \ldots, m$. t_{ij} is a nonnegative integer – $t_{ij} = 0$ if 0_{ij} is missing and positive if it exists.* Two operations of the same job cannot be processed simultaneously and a machine may process at most one job at a time. Find the operation sequence on each machine, that obeys the problem constraints and minimizes the sum of completeion times. The problem is designated $n/m/F/\Sigma C_i$,† where F stands for flowshop discipline and C_i is the completion time of job J_i.

The *no-idle* constraint: machine M_k, $k = 1, 2, \ldots, m$, works continuously without idle intervals.

The *no-wait* constraint: job J_i, $i = 1, 2, \ldots, n$ is processed continuously without waiting times between consecutive machines.

Both constraints arise in real life situations. Examples of such scheduling are: (i) under a no-idle constraint – use of very expensive equipment (.e.g., a computer and its peripheral devices) with the fee determined by the actual time consumption; (ii) under a no-wait constraint – in metal-processing industries (e.g., hot rolling) where delays between operations interfere with the technological process (e.g., cooling in the above case).

*For further elaboration of the meanings and influence of zero processing times see Hefetz and Adiri [4].
†For notation and classification of scheduling problems we follow Lenstra [5] and Rinnooy-Kan [6].

While the no-wait case has been widely discussed in the professional literature, [1] [6], the no-idle case is defined here for the first time. This paper deals with flowshop disciplines with a no-idle or a no-wait constraint where the objective function being minimum sum of completion times.

Garey, Johnson and Sethi [3] proved that $n/2/F/\Sigma C_i$ is unary NP-Complete by constructing an instance that is no-idle and no-wait, and thus proved at the same time that of $n/2/F$, no-idle/ΣC_i and $n/2/F$, no-wait/ΣC_i. Section 2 deals with some properties of the optimal schedule for $n/2/F$, no-idle/ΣC_i. Sections 3 and 4 are devoted to proofs of theorems that underlie polynomial bounded algorithms for $n/m/P$, no-idle/ΣC_i and $n/m/P$, no-wait/ΣC_i with an increasing or decreasing series of dominating machines, respectively.

1. COMPLEXITY OF FLOWSHOP/NO-IDLE OR NO-WAIT/SUM OF COMPLETION TIMES PROBLEMS

Garey, Johnson and Sethi [3] proved that 3-partition is reducible to $n/2/F/\Sigma C_i$. However, as the constructed instance of $n/2/F/\Sigma C_i$ happens to be no-idle and no-wait, thus at the same time proved the unary NP-Completeness of $n/2/F/\Sigma C_i$, $n/2/F$, no-idle/ΣC_i and $n/2/F$, no-wait/ΣC_i. Moreover, only minor modification of the proof is needed for proving the NP-Completeness of the first two problems where missing operations are prohibited. Specifically, replacement of zero processing times on M_1 (missing operations) by $\epsilon > 0$ (existing operations with infinitesimal processing times) and shifting of all ϵ to the beginning of the schedule on M_1, proves the unary NP-Completeness of $n/2/F$, $t_{ij} > 0/\Sigma C_i$ and $n/2/F$, $t_{ij} > 0$, no-idle/ΣC_i. The complexity of $n/m/F$, $t_{ij} > 0$, no-wait/ΣC_i for fixed $m \geqslant 2$ is an open problem.

2. CONSTRUCTION OF A NO-IDLE SCHEDULE

We distinguish two ways of constructing a no-idle schedule. Both are implemented consecutively on the machines, starting with the second, proceeding to the third and so on until the last machine.

(i) Right shifts. We shift to the right every operation that precedes an idle interval (maintaining the constraints of the problem on M_k, $k > 2$, further right shifts might be needed), until all idle intervals have been eliminated.

(ii) Left shifts. At first we schedule the operations on M_k, $k \geqslant 2$ in a single stretch (a block) without idle intervals, starting when the last job on M_{k-1} has been completed. Afterwards we apply a maximum left shift (without violating the constraints) to the whole block.

Let us define a *blocking job* on M_k, $k \geqslant 2$, in a no-idle schedule as the first job on M_k that prohibits shifting to the left.

3. PROPERTIIES OF THE OPTIMAL SCHEDULE FOR $n/2/F$, NO-IDLE/ΣC_i

Conway, Maxwell and Miller [2] fundamental theorem for flowshop scheduling that states that an optimal schedule exists for a $n/m/F/\delta$ problem (δ—any regular measure of performance) with the same processing order on the first two machines [2, p. 81], holds for the case under discussion. Thus, the set of permutation schedules for $n/2/F$, no-idle/ΣC_i is a dominant one.

THEOREM 1: If in the optimal schedule for $n/2/F$, no-idle/ΣC_i, the blocking job, is the last one, then the optimal schedule is according to SPT (Shortest Processing Time) on M_2, except for the blocking job that is the one with the minimum processing time on M_2.

PROOF: Let us denote, $a_i = t_{i1}$, $b_i = t_{i2}$; S_{ik} starting time of J_i on M_k; and the square brackets indicating the place of a job in the sequence, for example, $S_{[1]2}$—starting time of the first job on M_2. Since the blocking job is the last one, we have (see Figure 1).

$$S_{[1]2} = \sum_{i=1}^{n} a_{[i]} - \sum_{i=1}^{n-1} b_{[i]} = K + b_{[n]}, \tag{1}$$

where

$$K = \sum_{i=1}^{n} a_i - \sum_{i=1}^{n} b_i = \text{Constant}.$$

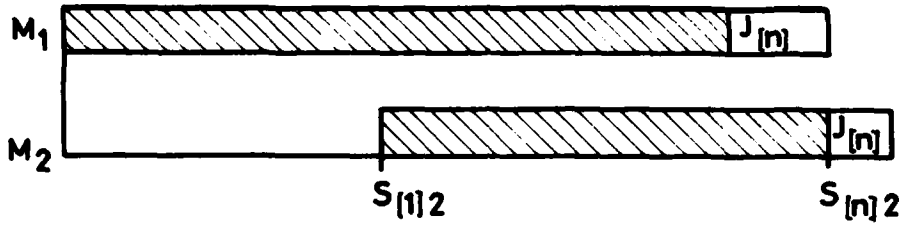


FIGURE 1. Two machines no-idle schedule where the blocking job is the last.

The sum of completion times takes the form

$$\sum_{i=1}^{n} C_i = S_{[1]2} + b_{[1]} + S_{[1]2} + b_{[1]} + b_{[2]} + \ldots + S_{[1]2} + \sum_{i=1}^{n} b_{[i]} \tag{2}$$

$$= nS_{[1]2} + \sum_{i=1}^{n} (n - i + 1) b_{[i]}.$$

Substitution of (1) yields

$$\sum_{i=1}^{n} C_i = nK + nb_{[n]} + \sum_{i=1}^{n} (n - i + 1) b_{[i]} \tag{3}$$

$$= nK + (n + 1) b_{[n]} + \sum_{i=1}^{n-1} (n - i + 1) b_{[i]}.$$

To minimize (3) $b_{[n]}$ should be the smallest and $b_{[i]}$, $i = 1, 2, \ldots, n - 1$, a nondecreasing sequence, thus the optimal schedule is given by the sequence

$$b_{[n]} \leqslant b_{[1]} \leqslant b_{[2]} \leqslant \ldots \leqslant b_{[n-1]}. \qquad \square$$

THEOREM 2: The jobs that (i) precede (ii) succeed the blocking job in the optimal schedule for $n/2/F$, no-idle/ΣC_i are ordered according to SPT on M_2, provided the no-idle constraint is not violated.

PROOF: Let us assume that the blocking job in the optimal schedule is the π-th in the sequence.

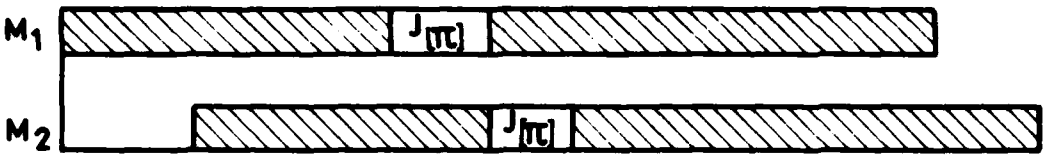


FIGURE 2. Optimal schedule for $n/2/F$, no-idle/ΣC_i where the blocking job is $J_{[\pi]}$.

(i) $S_{[1]2} = \sum_{i=1}^{\pi} a_{[i]} - \sum_{i=1}^{\pi-1} b_{[i]}$.

The situation here is rather similar to that of Theorem 1 with π replacing n. Thus, in the optimal schedule we have $b_{[1]} \leqslant b_{[2]} \leqslant \ldots \leqslant b_{[\pi-1]}$.

(ii) For $i = \pi + 1, \pi + 2, \ldots, n$, we have

$$C_{[i]} = C_{[\pi]} + \sum_{j=\pi+1}^{i} b_{[j]}.$$

Thus,

$$\sum_{i=\pi+1}^{n} C_{[i]} = (n - \pi)C_{[\pi]} + \sum_{i=\pi+1}^{n} (n - i + 1)b_{[i]}.$$

We have that in the optimal schedule the jobs that succeed the blocking job obey the SPT rule on M_2, provided the no-idle constraint is not violated, $b_{[\pi+1]} \leqslant b_{[\pi+2]} \leqslant \ldots \leqslant b_{[n]}$. □

4. $n/m/P$, NO-IDLE/ΣC_i, $m \geqslant 2$, WITH A SERIES OF DOMINATING MACHINES

Machine M_k dominates M_r if

$$\min_i t_{ik} \geqslant \max_i t_{ir}. \tag{4}$$

In abbreviated notation

$$M_k \succ M_r.$$

A scheduling problem with a series of increasing [decreasing] dominating machines is one where $M_1 \prec M_2 \prec \ldots \prec M_m [M_1 \succ M_2 \succ \ldots \succ M_m]$.

For $n/m/F$, no-idle/ΣC_i, $m > 2$, the set of all permutation schedules is not a dominate set. However, for the sake of solvability (development of polynomial bounded algorithms for special cases) we confine our search for optimality to permutation schedules, and the problem is designated $n/m/P$, no-idle/ΣC_i.

Note that for $m = 2$ the set of permutation schedules is a dominate set.

Let γ_{ik} be the processing time of the i-th job on M_k where the order is according to SPT on the last machine, M_m.

THEOREM 3: The optimal permutation schedule for $n/m/P$, no-idle/ΣC_i with an increasing series of dominating machines* is according to SPT on the last machine, except for the first job that satisfies

$$\min_j \left\{ V_j = n \sum_{k=1}^{m-1} \gamma_{jk} + (j-1)\gamma_{jm} - \sum_{i=1}^{j-1} \gamma_{im} \right\}.$$

PROOF: For this case we have

$$\sum_{i=1}^{n} C_i = nS_{[1]m} + \sum_{i=1}^{n} (n-i+1)t_{[i]m}, \tag{5}$$

where $S_{[1]m}$ is the starting time of the first job on the last machine, and

$$S_{[1]m} = \sum_{i=1}^{m-1} t_{[1]i}. \tag{6}$$

Thus, for a given first job the optimal permutation schedule is SPT on M_m,

$$t_{[2]m} \leqslant t_{[3]m} \leqslant \ldots \leqslant t_{[n]m}.$$

Selecting the j-th job in the SPT sequence on M_m to be the first, we have

$$\sum_{i=1}^{n} C_i = n \sum_{i=1}^{m-1} \gamma_{ji} + (j-1)\gamma_{jm} - \sum_{i=1}^{j-1} \gamma_{im} + \sum_{i=1}^{n} (n-i+1)\gamma_{im}. \tag{7}$$

Since the last term in (7) is not affected by the choice of the first job, the latter is taken so as to satisfy

$$\min_j \left\{ V_j = n \sum_{k=1}^{m-1} \gamma_{jk} + (j-1)\gamma_{jm} - \sum_{i=1}^{j-1} \gamma_{im} \right\}. \qquad \square$$

EXAMPLE 1: A 5/4/P, no-idle/ΣC_i with an increasing series of dominating machines with processing times as per Table 1.

TABLE 1 — *Processing Times*

Machines \ Jobs	J_1	J_2	J_3	J_4	J_5
M_1	3	1	3	2	2
M_2	4	5	6	4	6
M_3	7	9	8	8	9
M_4	11	13	14	10	12

$M_4 \geqslant M_3 \geqslant M_2 \geqslant M_1$, thus we have an increasing series of dominating machines.

SPT sequence on M_4 is 4-1-5-2-3.

*Since $M_{k+1} \geqslant M_k$, $k = 1, 2, \ldots, m-1$, the no-idle constraint is not effective and for the case under discussion, the two problems $n/m/P$, no-idle/ΣC_i and $n/m/P/\Sigma C_i$ are equivalent.

The first job, say J_i, satisfies $V_i = \min_j V_j$, is the i-th in the SPT sequence on M_4.

$$\begin{aligned} V_1 &= 5(2+4+8) &&= 70 \\ V_2 &= 5(3+4+7) + 11\cdot10 &&= 71 \\ V_3 &= 5(2+6+9) + 2\cdot12 - (11+12) &&= 88 \\ V_4 &= 5(1+5+9) + 3\cdot13 - (10+11+12) &&= 81 \\ V_5 &= 5(3+8+8) + 4\cdot14 - (10+11+12+13) &&= 95 \end{aligned}$$

The job chosen to be the first is the first in SPT sequence on M_4 ($V_1 = \min_j V_j$), namely J_4. The other jobs are scheduled according to SPT on M_4. Thus, the optimal permutations schedule is 4-1-5-2-3 with $\sum_{i=1}^{5} C_i = 240$, and takes the form as per Figure 3.

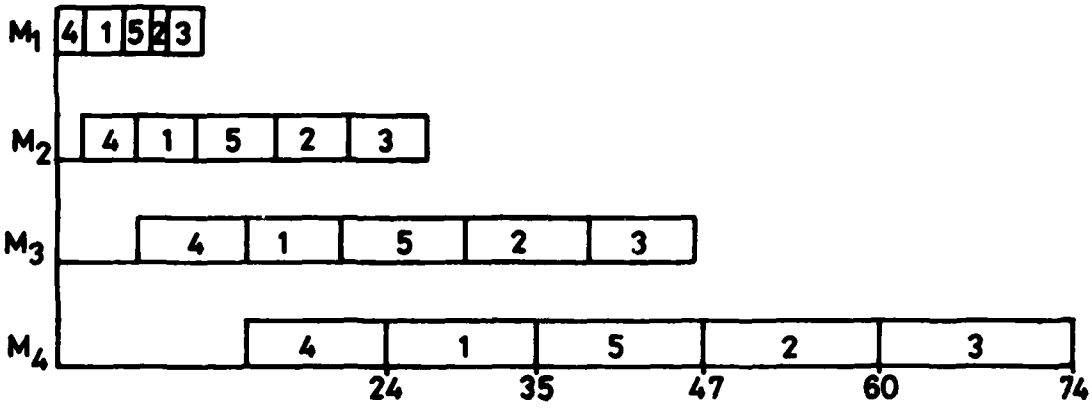


FIGURE 3. Optimal schedule for Example 1.

THEOREM 4: The optimal permutation schedule for $n/m/P$, no-idle/ΣC_i with a decreasing series of dominating machines is according to SPT on the last machine except for the last job that satisifies

$$\min_j \left\{ V_j = n \sum_{k=2}^{m} \gamma_{jk} - (n-j)\gamma_{jm} + \sum_{i=j+1}^{n} \gamma_{im} \right\}.$$

PROOF: Since $M_1 \geq M_2 \geq \ldots \geq M_m$ the blocking job on M_k, $k = 2, 3, \ldots, m$, is the last one, we have

$$\begin{aligned} S_{[1]m} &= \sum_{k=1}^{m-1} (S_{[1](k+1)} - S_{[1]k}) \\ &= \sum_{k=1}^{m-1} \left(\sum_{i=1}^{n} t_{[i]k} - \sum_{i=1}^{n-1} t_{[i](k+1)} \right) \\ &= \sum_{k=1}^{m-1} (T_k - T_{k+1} + t_{[n](k+1)}) = K + \sum_{k=1}^{m-1} t_{[n](k+1)}, \end{aligned} \tag{8}$$

where T_k is the total processing time demanded on M_k, T_k = constant $= \sum_{i=1}^{n} t_{ik}$ and $K = T_1 - T_m$.

Substitution of (8) in (5) yields

$$\sum_{i=1}^{n} C_i = nK + n\sum_{k=1}^{m-1} t_{[n](k+1)} + \sum_{i=1}^{n}(n-i+1)t_{[i]m}. \tag{9}$$

Given the job in the last (n-th) place, we have

$$\sum_{i=1}^{n} C_i = K_1 + \sum_{i=1}^{n-1}(n-i+1)t_{[i]m}, \tag{10}$$

where $K_1 = \text{constant} = nK + n\sum_{k=1}^{m-1} t_{[n](k+1)} + t_{[n]m}$.

Minimum of (10) is obtained by an SPT sequence on M_m,

$$t_{[1]m} \leqslant t_{[2]m} \leqslant \ldots \leqslant t_{[n-1]m}.$$

Choosing the j-th job in the SPT sequence on M_m for the last place, we have

$$\sum_{i=1}^{n} C_i = nK + n\sum_{k=1}^{m-1} \gamma_{j(k+1)} - (n-j)\gamma_{jm} + \sum_{i=j+1}^{n}\gamma_{im} + \sum_{i=1}^{n}(n-i+1)\gamma_{im}. \tag{11}$$

The last term on the right hand side of (11) is independent of the choice for the last place, thus the job is taken so as to minimize

$$\left\{V_j = n\sum_{k=1}^{m-1}\gamma_{j(k+1)} - (n-j)\gamma_{jm} + \sum_{i=j+1}^{n}\gamma_{im}\right\}. \qquad \square$$

COROLLARY 1: The optimal schedule for $n/2/F$, no-idle/ΣC_i where $M_1 > M_2$ is according to SPT on M_2 except for the last job that is the one with minimum processing time on M_2.

PROOF: As was pointed out the two problems $n/2/P$, no-idle/ΣC_i and $n/2/F$, no-idle/ΣC_i, are equivalent. Substitution of $m = 2$ in Theorem 4 yields

$$V_1 = \min_j \left\{V_j = j\gamma_{j2} + \sum_{i=j+1}^{n}\gamma_{i2}\right\}.$$

Thus, the last job is the first in SPT sequence on M_2. This result is in agreement with Theorem 1—since $M_1 > M_2$ the blocking job is the last one and should be the smallest. $\square$

EXAMPLE 2: A $5/4/P$, no-idle/ΣC_i with a decreasing series of dominating machines with processing times as per Table 2.

TABLE 2 — *Processing Times*

Machines \ Jobs	J_1	J_2	J_3	J_4	J_5
M_1	11	13	14	10	12
M_2	7	9	8	8	9
M_3	4	5	6	4	6
M_4	3	1	3	2	2

$M_1 \geq M_2 \geq M_3 \geq M_4$, thus we have a decreasing series of dominating machines.

SPT sequence on M_4 is 2-4-5-1-3.

The last job, say J_i, satisfies $V_i = \min_j V_j$, is the i-th in the SPT sequence on M_4.

$$\begin{aligned} V_1 &= 5(9+5+1) - 4\cdot 1 + 2+2+3+3 = 81 \\ V_2 &= 5(8+4+2) - 3\cdot 2 + 2+3+3 = 72 \\ V_3 &= 5(9+6+2) - 2\cdot 2 + 3+3 = 83 \\ V_4 &= 5(7+4+3) - 1\cdot 3 + 3 = 70 \\ V_5 &= 5(8+6+3) = 85 \end{aligned}$$

The job chosen to be the last is the fourth in SPT sequence on M_4, ($V_4 = \min_j V_j$), namely J_1. The other jobs are scheduled according to SPT on M_4.

Thus, the optimal permutation schedule is 2-4-5-3-1 with $\sum_{i=1}^{5} C_i = 343$ and takes the form as per Figure 4.

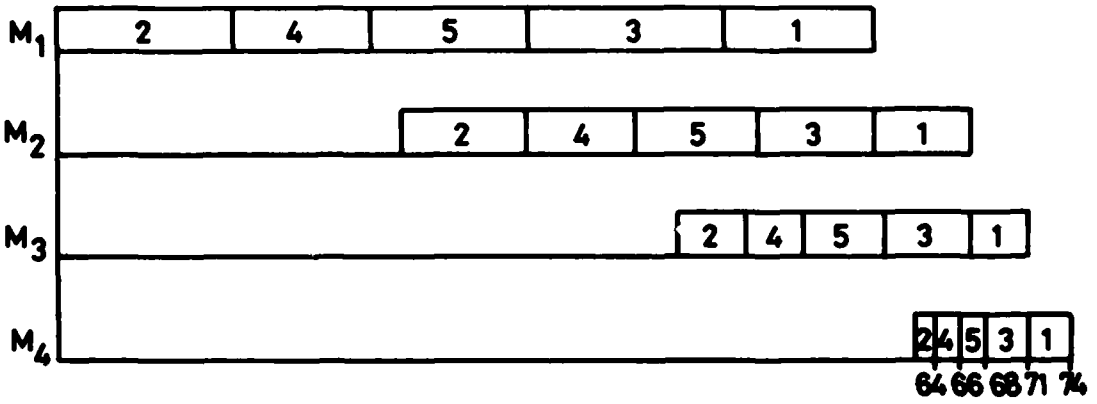


FIGURE 4. Optimal schedule for Example 2.

5. $n/m/P$, NO-WAIT/ΣC_i, $m \geq 2$, WITH A SERIES OF DOMINATING MACHINES

We recall that "no-idle" constraint prescribes for the machines to work continuously without idle intervals, while the no-wait constraint prescribes for the jobs to be processed continuously without waiting times between consecutive machines.

The set of all permutation schedules for an "F, no-wait" problem ($t_{ij} \geq 0$, missing operations are allowed) is not a dominant set. However, we confine our search for optimality to permutation schedules, namely, the problems under discussion in this section, are particular cases of $n/m/P$, no-wait/ΣC_i. Note that for $t_{ij} > 0$ (missing operations are not allowed) a feasible solution for an "F, no-wait" problem is a permutation one, thus "F, $t_{ij} > 0$, no-wait" and "P, $t_{ij} > 0$, no-wait" are equivalent. As was previously pointed out, $n/2/F$, no-wait/ΣC_i is NP-Complete while the compexity of $n/m/F$, no-wait, $t_{ij} > 0/\Sigma C_i$, $m \geq 2$, is an open problem.

THEOREM 5: For an increasing series of dominating machines the two problems $n/m/P$, no-wait/ΣC_i and $n/m/P$, no-idle/ΣC_i have the same optimal sequence.

PROOF: For $n/m/P$, no-wait/ΣC_i with an increasing series of dominating machines, we have

$$C_{[i]} = \sum_{i=1}^{m-1} t_{[1]i} + \sum_{j=1}^{i} t_{[j]m}. \tag{12}$$

It is readily shown that (12) leads to (5) and (6). □

A direct consequence of Theorems 5 and 3 is that the optimal schedule for $n/m/P$, no-wait/ΣC_i with an increasing series of dominating machines is SPT on M_m except for the first job that satisfies

$$\operatorname{Min}_j \left\{ V_j = n \sum_{k=1}^{m-1} \gamma_{jk} + (j-1)\gamma_{jm} - \sum_{i=1}^{j-1} \gamma_{im} \right\}.$$

EXAMPLE 3: A 5/4/P, no-wait/ΣC_i with an increasing series of dominating machines with processing times as per Table 1 (Example 1).

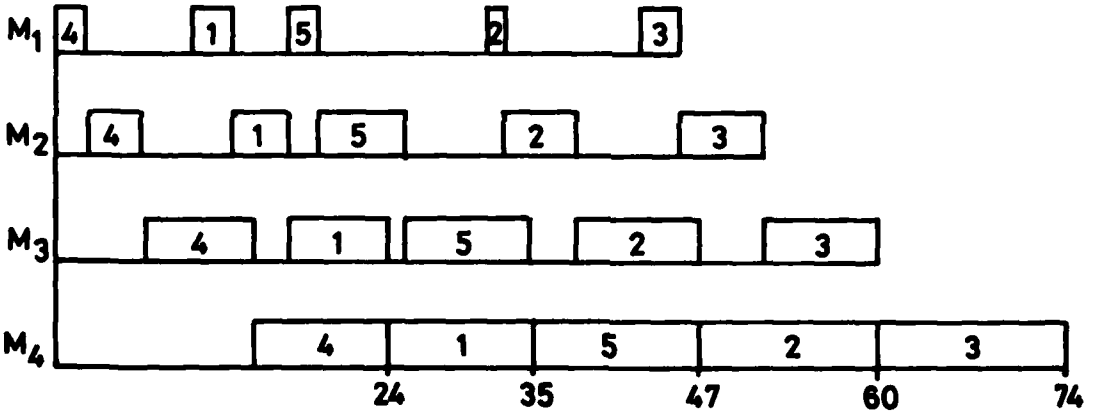


FIGURE 5. Optimal schedule for Example 3.

The calculations and the optimal sequence are the same as in Example 1, but the resulting schedule is as per Figure 5 (compare with Figure 3).

THEOREM 6: The optimal schedule for $n/m/P$, no-wait/ΣC_i, $m \geq 2$, with a decreasing series of dominating machines* is according to SPT on M_1.

*Since $M_k \geq M_{k+1}$, $k = 1, 2, \ldots, m-1$, the no-wait constraint is not effective for the case [illegible] for discussion the two problems $n/m/P$, no-wait/ΣC_i and $n/m/P/\Sigma C_i$ are equivalent.

PROOF: Since $M_k \geq M_{k+1}$, $k = 1, 2, \ldots, m-1$, we have for the optimal schedule

$$\sum_{i=1}^{n} C_i = \sum_{j=1}^{m} t_{[1]j} + t_{[1]1} + \sum_{j=1}^{m} t_{[2]j} + \ldots + \sum_{i=1}^{n-1} t_{[i]1} + \sum_{j=1}^{m} t_{[n]j} \tag{13}$$

$$= \sum_{i=1}^{n} \sum_{j=1}^{m} t_{[i]j} + \sum_{i=1}^{n-1} (n-i) t_{[i]1}.$$

The first term on the right-hand side of (13) is a constant, and the sequence $t_{[1]1} \leq t_{[2]1} \leq \ldots \leq t_{[n-1]1}$ minimizes the second term. Morever, $t_{[n]1} \geq t_{[n-1]1}$, otherwise if $t_{[n]1} \neq \max_i t_{[i]1}$ the value of the second term can be reduced by interchanging the last job with $J_{[j]1}$, $t_{[j]1} = \max_i t_{[i]1}$. □

EXAMPLE 4: A $5/4/P$, no-wait/ΣC_i with a decreasing series of dominating machines with processing times as per Table 2. Since $M_1 \geq M_2 \geq M_3 \geq M_4$ we have a decreasing series of dominating machines and the optimal sequence is according to SPT on M_1, 4-1-5-2-3, with $\sum_{i=1}^{5} C_i = 247$. The optimal schedule takes the form as per Figure 6.

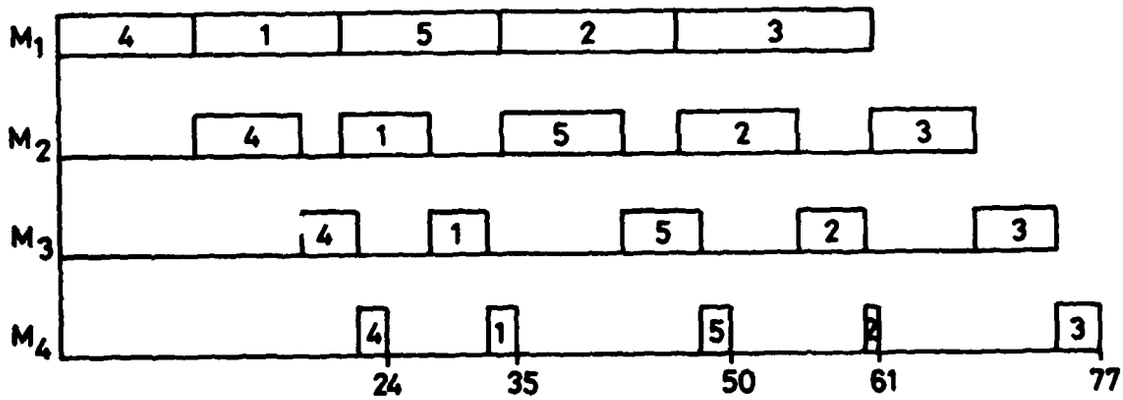


FIGURE 6. Optimal schedule for Example 4.

REFERENCES

[1] Baker, K.R., *Introduction to Sequencing and Scheduling* (John Wiley and sons, New York, 1974).

[2] Conway R.W., W.L. Maxwell and L.W. Miller, *Theory of Scheduling* (Addision Wesley, Reading, Mass 1967).

[3] Garey, M.R., D.S. Johnson and R. Sethi, "The Complexity of Flowshop and Jobshop Scheduling." Mathematics of Operations Research, *1*, 2, 117-129 (1976).

[4] Hefetz, N. and I. Adiri, "The Influence of Missing Operations on Scheduling Problems," Naval Research Logistics Quarterly, *29*, 3 (1982).

[5] Lenstra, J.K., *Sequencing by Enumerative Methods,* (Mathematical Center, Amsterdam, 1977).

[6] Rinnooy-Kan, A.H.G., *Machine Scheduling Problems: Classification, Complexity and Computations* (Nijhoff, The Hague, 1976).

ON THREE BASIC METHODS FOR SOLVING BOTTLENECK TRANSPORTATION PROBLEMS

Ulrich Derigs

Institut Für Ökonometrie und Operations Research
Bonn, West Germany

ABSTRACT

For solving transportation problems essentially three types of methods are known: primal methods, the Hungarian method and the shortest augmenting path method. In this paper we present the specialization of these approaches to the bottleneck transportation problem and report some computational experience.

1. INTRODUCTION

The classical transportation problem (TP) can be formulated in the following way. There are m supply points and n demand points with supply point i capable of supplying amount $a_i (i = 1, \ldots, m)$ and demand point j having demand $b_j (j = 1, \ldots, n)$ with $\sum_i a_i = \sum_j b_j$. Find the least cost transportation pattern from the supply points to the demand points when the cost of transporting a unit amount from i to j is c_{ij}, i.e.,

$$\min \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} \cdot x_{ij} \qquad \text{subject to}$$

$$\text{(1.1)} \qquad \sum_{j=1}^{n} x_{ij} = a_i \qquad (i = 1, \ldots, m)$$

$$\text{(1.2)} \qquad \sum_{i=1}^{m} x_{ij} = b_j \qquad (j = 1, \ldots, n)$$

$$\text{(1.3)} \qquad x_{ij} \geqslant 0 \qquad (i = 1, \ldots, m;\ j = 1, \ldots, n)$$

A related problem is the bottleneck transportation problem (BTP). Here a transportation time t_{ij} is specified between each supply point i and each demand point j. Now it is required to find a transportation pattern which minimizes the total time necessary for transporting the goods from the supply to the demand points. Thus,

$$\text{(1.4)} \qquad \min \max \{t_{ij} | x_{ij} > 0\} \text{ subject to (1.1), (1.2) and (1.3)}$$

Problem (1.4) is often called "time-transportation problem." It occurs in connection with transportation of perishable goods, with the delivery of emergency supplies or when military units are to be sent from their bases to the front.

For solving TP we know essentially three types of methods

- primal methods
- the "Hungarian" method
- the shortest augmenting path method.

Nearly every approach for solving TP can be modified to treat BTP.

So for instance the primal method of Hammer [10], [10a], for solving BTP is strongly related to the method of Klein [11] for solving TP. Starting from a feasible solution both methods are looking for "negative cycles" to improve the actual solution.

In this paper we present these three basic approaches for solving BTP and we report some computational experience with these methods.

2. THE HUNGARIAN METHOD

This method for solving BTP was first proposed by Garfinkel and Rao [8]. Inspired by a work of Edmonds and Fulkerson [5] on general bottleneck problems, they called the procedure the *Threshold Method.* The algorithm can be described in the following way: First determine a "good" lower bound $\underline{z}$ for the optimal objective value z. Then define a network $\mathcal{N}(\underline{z})$ in the following way: Let $V = \{s,t\} \cup M \cup N$ with $M = \{1, 2, \ldots, m\}$ and $N = \{m+1, \ldots, m+n\}$. be the nodeset of $\mathcal{N}(\underline{z})$. Now every node $i \in M$ is connected with s by an arc, respectively, each $m + j \in N$ with t. A pair $(i, m+j)$ is connected by an arc only if $t_{ij} \leqslant \underline{z}$ holds.

The arc-capacities $d_{k,l}$ are defined by

$$d_{s,i} := a_i \quad (i = 1, \ldots, m)$$

$$d_{i,m+j} := \infty \quad (i = 1, \ldots, m; j = 1, \ldots, n)$$

$$d_{m+j,t} := b_j \quad (j = 1, \ldots, n).$$

Now a maximum flow $f = (f_{ij})$ from source s to sink t is determined using the labeling method of Ford and Fulkerson [6]. If for the maximal flow value $v = \sum_i a_i$ holds, the optimal solution for the BTP is obtained defining

$$x_{ij} := f_{i,m+j} \ (i = 1, \ldots, m; j = 1, \ldots, n)$$

If $v < \Sigma\, a_i$ the labeling procedure yields simultaneously a minimal cut $(X, \bar{X})$ in $\mathcal{N}(\underline{z})$ with $s \in X$ and $t \in \bar{X}$. This cut has the property

$$t_{ij} \leqslant \underline{z} \Rightarrow (i, m+j) \notin (X, \bar{X})$$

To obtain a solution to (1.1)-(1.3) it is therefore necessary to use an arc $(i, m+j)$ with $t_{ij} > \underline{z}$. Hence, we determine

$$z^* := \min \{t_{ij} \mid i \in X,\ m + j \in \bar{X}\} > \underline{z}.$$

Define $\underline{z} := z^*$ and repeat the process.

After $(m \cdot n)$ iterations, at most, an optimal solution is obtained. The algorithm is summarized in the following flow-chart:

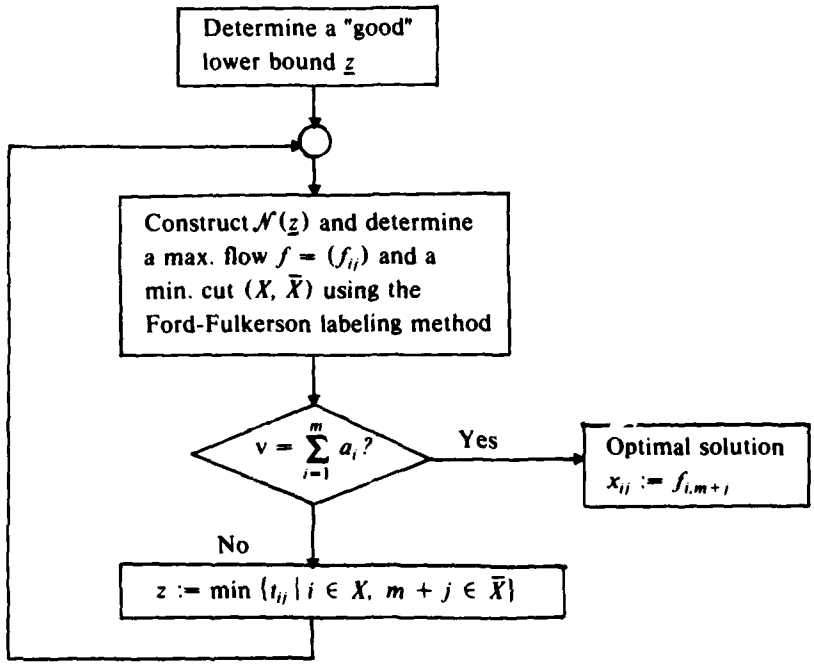


FIGURE 1. Flow-chart of the Hungarian method

During the algorithm the network $\mathcal{N}(\underline{z})$ need not be constructed explicitly. All operations of the labeling procedure can be performed directly on the "time" — matrix $T = (t_{ij})$.

3. A PRIMAL METHOD

The first method for solving BTP proposed by Barsov [1] is a primal algorithm which proceeds "dual" to the Hungarian method in some sense. In the course of the Hungarian method the "threshold" $\underline{z}$ is successively increased until a feasible solution can be found. In this method a feasible solution is always at hand and the threshold-value is successively decreased until no further solution can be found. Starting from a feasible solution $x = (x_{ij})$ with objective value $\bar{z}$ a cost-matrix $D = (d_{ij})$ is defined by

$$d_{ij} := \begin{cases} 0 & \text{if } t_{ij} < \bar{z} \\ 1 & \text{else} \end{cases}$$

Now a TP with cost-matrix D is solved. If the optimal solution $y = (y_{ij})$ has an optimal value $z(y) > 0$ the solution $x = (x_{ij})$ is optimal for BTP. Otherwise

$$z^* := \max \{t_{ij} \mid y_{ij} > 0\} < \bar{z} \text{ holds.}$$

Define $\bar{z} := z^*$ and $x := y$ and repeat the process.

Computational experience shows [8] that this method is inferior to the Hungarian method. Recently, Finke and Smith [7] developed an improved primal method.

Starting from a lower bound $\underline{z}$ and a "good" start solution $x = (x_{ij})$, a cost-matrix $D = (d_{ij})$ is defined by

$$d_{ij} := \left\lfloor \frac{t_{ij}}{\underline{z} + 1} \right\rfloor \text{ with } [x] := \text{greatest integer} \leqslant x.$$

Now the associated TP is solved using the well known MODI-method and $x = (x_{ij})$ as start solution. If the optimal solution $y = (y_{ij})$ has objective value $z(y) = 0$, then y is optimal for BTP. Otherwise, we consider the dual variables u_i, $i = 1, \ldots, m$ and v_j, $j = 1, \ldots, n$ associated with the optimal solution y. Then

$$u_i + v_j \leqslant d_{ij} \qquad i = 1, \ldots, m;\ j = 1, \ldots, n$$

$$y_{ij} > 0 \Rightarrow u_i + v_j = d_{ij}$$

and

$$\underline{z}' := \min \{t_{ij} \mid u_i + v_j > 0\} > \underline{z} \text{ holds.}$$

It can be shown that $\underline{z}'$ is a lower bound for the optimal value for the BTP. Define $\underline{z} := \underline{z}'$ and repeat the process. In an implementation the actual basic solution of the TP is stored as a tree using the list structures proposed by Glover and Klingman [9], Srinivasan and Thompson [12]. Due to this technique primal methods are superior to other methods in the case of TP.

The following flow-chart summarizes the algorithm of Finke and Smith.

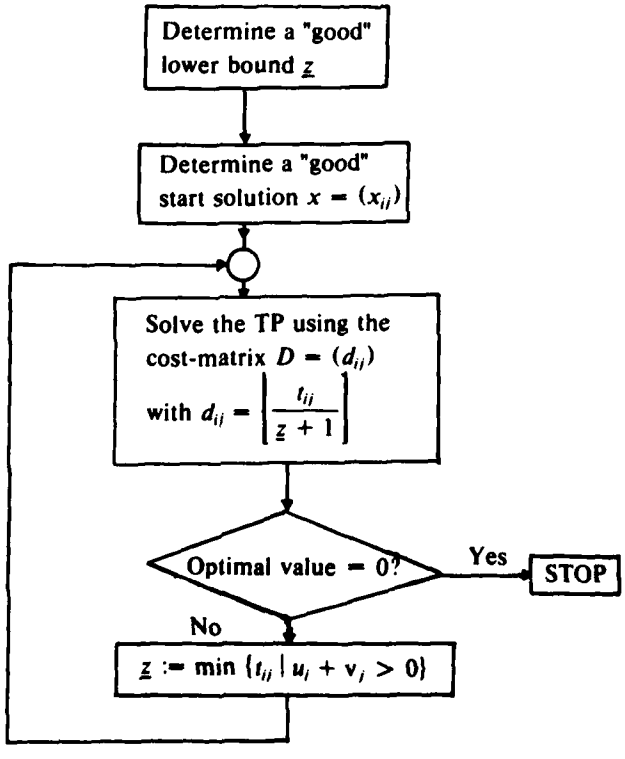


FIGURE 2. Flow-chart of the primal method of Finke and Smith

4. THE SHORTEST AUGMENTING PATH METHOD

This method was proposed by Tomizawa [14] for solving TP and applied to BTP by Derigs and Zimmermann [4]. A theoretical foundation of this method can be found in [2]. During the procedure so-called k-subproblems (P_k) have to be solved for $1 \leqslant k \leqslant m$:

$$(P_k) \qquad \min \max \{t_{ij} | x_{ij} > 0\}$$

$$\sum_{j=1}^{n} x_{ij} = \begin{cases} a_i & i = 1, 2, \ldots, k \\ 0 & \text{else} \end{cases}$$

$$\sum_{i=1}^{m} x_{ij} \leqslant b_j \quad j = 1, \ldots, n$$

$$x_{ij} \geqslant 0 \quad i = 1, \ldots, m;\ j = 1, \ldots, n.$$

The solution for (P_1) is obvious. Starting from an optimal solution $x = (x_{ij})$ for (P_k) with $k = 1, \ldots, m-1$ an optimal solution for (P_{k+1}) is obtained by means of augmenting paths. For this purpose we partition $\{1, 2, \ldots, n\} = N_1 \cup N_2$ with

$$N_1 = \left\{ j \mid \sum_{i=1}^{k} x_{ij} = b_j \right\} \qquad \text{"saturated columns."}$$

Now a sequence Δx of mutually distinct matrix-entries

$$(k+1, j_1),\ (i_1, j_1),\ (i_1, j_2),\ (i_2, j_2),\ \ldots,\ (i_r, j_r),\ (i_r, j_{r+1})$$

is called an augmenting path with respect to $x = (x_{ij})$ if

$$x_{i_q, j_q} > 0 \text{ for } q = 1, 2, \ldots, r \text{ and}$$

$$j_q \in N_1 \text{ for } q = 1, 2, \ldots, r \text{ and } j_{r+1} \in N_2.$$

The capacity of Δx is defined by

$$\text{cap}(\Delta x) := \min \{a_{k+1},\ b_{j_{r+1}} - \sum_{i=1}^{k} x_{i, j_{r+1}},\ \epsilon\} \text{ with}$$

$$\epsilon := \min \{x_{i_q, j_q} | q = 1, 2, \ldots, r\}.$$

Let us define $\bar{x} := x \oplus \Delta x$ by

$$\bar{x}_{ij} := \begin{cases} x_{ij} - \text{cap}(\Delta x) & \text{for } (i,j) = (i_1, j_1), \ldots, (i_r, j_r) \\ x_{ij} + \text{cap}(\Delta x) & \text{for } (i,j) \in \Delta x \setminus \{(i_q, j_q) | q = 1, \ldots, r\} \\ x_{ij} & \text{else} \end{cases}$$

and

$$w(\Delta x) := \max \{t_{ij} | \bar{x}_{ij} > 0\}.$$

Let D_k be the set of all augmenting paths with respect to $x = (x_{ij})$ and $\Delta \bar{x} \in D_k$ with

$$(4.1) \qquad w(\Delta \bar{x}) \leqslant w(\Delta x) \text{ for } \Delta x \in D_k$$

$$(4.2) \qquad \text{cap}(\Delta \bar{x}) = a_{k+1}$$

then $\bar{x} := x \oplus \Delta \bar{x}$ is an optimal solution for (P_{k+1}).

An augmenting path $\Delta x \in D_k$ with property (4.1) can be determined using a modified Dijkstra method. If Δx does not fulfill (4.2) we split (P_{k+1}) into two subproblems (P^1_{k+1}) and (P^2_{k+1}) with $a^1_{k+1} := \text{cap}\,(\Delta x)$ and $a^2_{k+1} := a_{k+1} - a^1_{k+1}$. Thus, (P_{k+1}) can be solved in a finite number of steps.

Computational experience shows that the method can be improved with the aid of the following starting procedure. First we determine a "good" lower bound $\underline{z}$ for the optimal value z and a "good" partial solution $\tilde{x} = (\tilde{x}_{ij})$, i.e.

$$\sum_{j=1}^{n} \tilde{x}_{ij} \leqslant a_i \qquad , i = 1, \ldots, m \tag{4.3}$$

$$\sum_{i=1}^{m} \tilde{x}_{ij} \leqslant b_j \qquad , j = 1, \ldots, n \tag{4.4}$$

$$\tilde{x}_{ij} \geqslant 0 \qquad , i = 1, \ldots, m,\ j = 1, \ldots, n \tag{4.5}$$

with the property

$$\tilde{x}_{ij} > 0 \Rightarrow t_{ij} \leqslant \underline{z} \tag{4.6}$$

such that

$$\sum_{i=1}^{m} \sum_{j=1}^{n} \tilde{x}_{ij} \text{ is large.}$$

A partial solution $\tilde{x}$ with (4.3) - (4.6) is called $\underline{z}$-feasible. Computational tests have shown that it is not useful here to determine a $\underline{z}$-feasible solution with maximal possible sum. Some simple heuristics will give solutions which are within 5% from optimality much faster and, with respect to the running time of the complete procedure, this is of advantage. The complete procedure is summarized in the following flow chart.

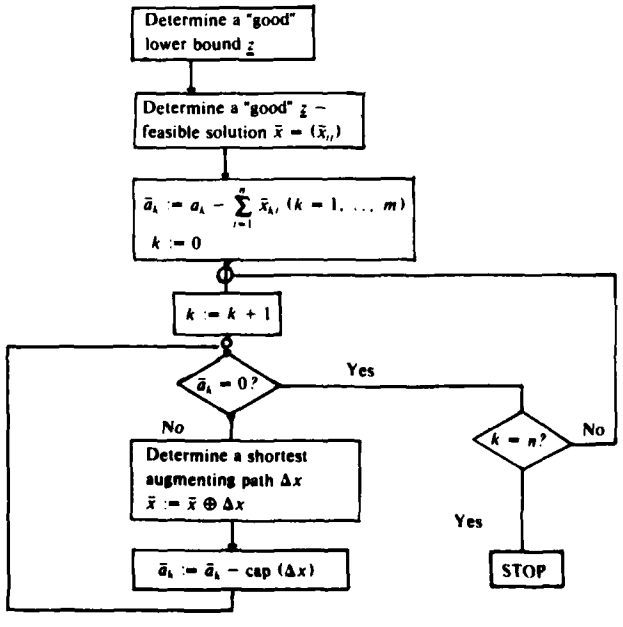


FIGURE 3. Flow-chart of the shortest augmenting path method.

5. LOWER BOUNDS AND STARTING PROCEDURES

All three methods presented above start with a lower bound $\underline{z}$ for the optimal value z. Computational experience shows that the bound proposed by Garfinkel and Rao is favorable. Row-thesholds z_i and column-thresholds z^j are computed with

$$z_i := \min \max \{t_{ij} | x_{ij} > 0\} \quad \text{subject to}$$

$$\sum_{j=1}^{n} x_{ij} \geqslant a_i$$

$$0 \leqslant x_{ij} \leqslant b_j \quad \text{for } i = 1, \ldots, m$$

and z^j defined analogously.

Then $\underline{z} := \max \{z_1, z_2, \ldots, z_m, z^1, z^2, \ldots, z^n\}$ is a lower bound. This bound is easy to calculate and yields a good estimation for the optimal value in most cases.

The primal method of Finke and Smith needs a good starting solution $x = (x_{ij})$. They propose the following method. First a $\underline{z}$-feasible partial solution $\tilde{x} = (\tilde{x}_{ij})$ is constructed. Such a partial solution is used for the shortest augmenting path method, too. *A* $\underline{z}$-feasible solution can be obtained in the following way. For every row i (column j) the number T_i (T^j) of feasible matrix entries (i,j) subject to $t_{ij} \leqslant \underline{z}$ is computed. Now the row i_0 (column j_0) with minimal $T_{i_0} > 0$ ($T^{j_0} > 0$) and positive a_{i_0} (b_{j_0}) is determined. If no such row (column) exists the partial solution is completed. Otherwise, determine in row i_0 (column j_0) the feasible entry (i_0, j_0) with minimal number T^{j_0} (T_{i_0}) and positive b_{j_0} (a_{i_0}). Then define

$$\tilde{x}_{i_0,j_0} := \min \{a_{i_0}, b_{j_0}\}$$

$$a_{i_0} := a_{i_0} - \tilde{x}_{i_0,j_0}$$

$$b_{j_0} := b_{j_0} - \tilde{x}_{i_0,j_0}.$$

Update T_i and T^j and repeat the process.

This way a good z-feasible partial solution $\tilde{x} = (\tilde{x}_{ij})$ is determined. If $\tilde{x} = (\tilde{x}_{ij})$ fulfills (1.1) and (1.2) then it is an optimal solution for BTP. If the partial solution is not optimal in the second step of the starting procedure proposed by Finke and Smith, the remaining supply is distributed using a north-west-corner rule. Finke and Smith call their procedure threshold-totals-method.

The following flow-chart summarizes the starting procedures for the three methods.

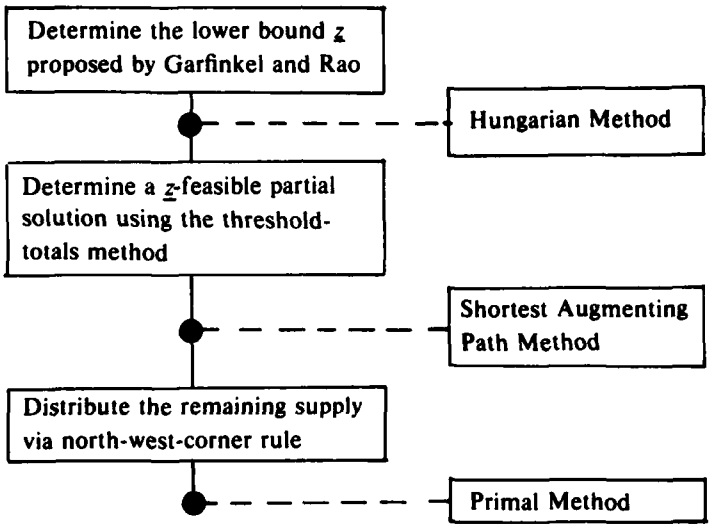


FIGURE 4. Flow-chart of the starting procedures.

6. COMPUTATIONAL RESULTS

FORTRAN IV implementations of all three methods were tested on a CDC CYBER 76 of the Computer Center of the University of Cologne. We used the following codes:

HUNGAR - a version of the Hungarian method which Professor Garfinkel kindly made available to us

PRIMAL - a version of the primal method which Professor Finke kindly made available to us

SAP - an improved version of the BTP-code which is listed in [4]. The improvement concerns mainly the starting procedure.

The HUNGAR-code uses the maximal cardinality NL of expected postive x_{ij}'s in any column as input data. The running time of the program is highly dependent on the choice of NL since the value and position of the x_{ij}'s are stored in NL × N matrices. Thus, the choice of smaller "NL" decreases the running time. But if the actual number exceeds NL, termination occurs. Choosing NL = M/2 all problems were solved.

We considered rectangular as well as quadratic examples. The integer coefficients of the time matrix T, the supply vector a and the demand vector b were generated by a machine independent uniformly distributed pseudo random number generator in the interval $[1, 2^{31}-1]$. Afterwards these numbers were transformed into the intervals $[1,b]$ with $b = 10, 100, 1000, 10000, 2^{31}-1$. Twenty-five examples were generated for each combination to calculate the mean running time. Numerical experience showed that the running time of HUNGAR and SAP for solving rectangular problems with $m > n$ is less than for the equivalent transposed problem with $m < n$. Therefore, we recommend that rectangular problems always be solved in the form with $m > n$.

The following table shows the mean running time for quadratic problems with $m = n = 100$.

TABLE 1 — *Mean Running Time in CPU-Seconds for (100 × 100) BTP*

a_i, b_j \ t_{ij}		1-10	1-100	1-1000	1-10000	1-($2^{31}-1$)
1-10	SAP	.112	.177	.177	.192	.181
	PRIMAL	.106	.124	.124	.133	.126
	HUNGAR	.226	.400	.383	.478	.504
1-100	SAP	.107	.120	.144	.171	.150
	PRIMAL	.094	.106	.105	.106	.111
	HUNGAR	.332	.582	.611	.588	.672
1-1000	SAP	.089	.140	.188	.121	.223
	PRIMAL	.092	.098	.119	.106	.114
	HUNGAR	.333	.439	.548	.603	.713

The mean running time of PRIMAL is significantly better than the other's. For SAP this is simply caused by a single "ill conditioned" problem for which the running time is more than tenfold the running time of the other 24 examples of the group whose running time is comparable to those of PRIMAL.

SAP and PRIMAL are shown to be relatively insensitive to the range of the parameters a_i, b_j and t_{ij} while the running time of HUNGAR is more than doubled when the range for t_{ij} is increased from $b = 10$ to $b = 2^{31} - 1$.

Then we modified the problems subject to

$$t_{1j} = t_{i1} = 0 \text{ for } i = 1, \ldots, m \text{ and } j = 1, \ldots, n \text{ and}$$
$$a_1 = b_1 = \max\{\max\{a_i\}, \max\{b_j\}\}.$$

For these perturbated problems the starting procedure has no effect and thus the computational behavior of the "pure" algorithms is shown.

Table 2 shows that PRIMAL and HUNGAR are more dependent on the quality of the starting procedure and the range of the time values t_{ij} than SAP.

TABLE 2 — *Mean Running Time in CPU-Seconds for the Perturbated* (40 × 40) Problems

a_i, b_j \ t_{ij}		1-10	1-100	1-1000	1-10000	1-($2^{31}-1$)
1-10	SAP	.014	.105	.122	.117	.121
	PRIMAL	.015	.118	.369	.446	.448
	HUNGAR	.016	.269	.608	.720	.686
1-100	SAP	.014	.153	.196	.202	.169
	PRIMAL	.014	.133	.389	.483	.504
	HUNGAR	.026	.436	.790	.922	.988
1-1000	SAP	.014	.161	.179	.217	.164
	PRIMAL	.015	.132	.398	.483	.506
	HUNGAR	.025	.422	.834	.937	.983

Then we applied the algorithms to bottleneck assignment problems i.e., problems with $a_i = b_j = 1$.

TABLE 3 — *Mean Running Time in CPU-Seconds for (100 × 100) Bottleneck Assignment Problems*

	t_{ij}	1-10	1-100	1-1000	1-10000	$1 - (2^{31} - 1)$
Assignment Problem	SAP	.074	.074	.074	.073	.071
	PRIMAL	.083	.092	.100	.096	.105
	HUNGAR	.055	.103	.143	.140	.155

For assignment problems SAP needs at most $n/2$ iterations while PRIMAL will perform a number of degenerate pivot operations. Again PRIMAL and SAP show a more robust nature with respect to the range of the time parameters t_{ij} than HUNGAR.

More computational results can be found in [3] and [4].

7. CONCLUSIVE REMARKS

In this paper we presented three basic methods for solving the bottleneck transportation problem. It is beyond the scope of this study to give an entire review of all the different methods which are available for solving BTP. Nevertheless, we think that the presented procedures are representative in the sense that the main approaches for tackling combinatorial optimization problems are specialized to the bottleneck transportation problem.

The computational tests indicate that PRIMAL and SAP outperform HUNGAR in general. The optimal choice between PRIMAL and SAP is data dependent and should be specially made for every problem to be solved. If it can be expected that the start-heuristic is performing well, PRIMAL is recommended. Otherwise, SAP is of advantage.

We want to close our discussion by referring to another computational study. Werner [15] compared versions of SAP and PRIMAL on special problems the data of which was defined by some structured transportation networks incorporating time tables for transportation vehicles, waiting-times and different regions with different magnitudes of demand and supply. Werner reports that the running time for PRIMAL was highly dependent on the distribution of data and the more the data was not uniform the more SAP became superior.

ACKNOWLEDGMENT

The author is grateful to Professor R. S. Garfinkel and Professor G. Finke who kindly made available their FORTRAN IV codes for numerical comparisons.

REFERENCES

[1] Barsov, A.R., *What is Linear Programming?*, Moscow 1959 (in Russian).
[2] Burkard, R.E., U. Zimmermann, "Weakly Admissible Transformations for Solving Algebraic Assignment and Transportation Problems," Mathematical Programming Studies , *12*, 1-18 (1980).
[3] Derigs, U, "Engpaβ-Zielfunktion and Zeit/Kosten-Trade-offs beim Transportproblem, Thesis," Universität Köln, Germany (1980).
[4] Derigs, U., and U. Zimmermann, "An Augmenting Path Method for Solving Linear Bottleneck Transportation Problems," *Computing*, *22*, 1-15 (1979).
[5] Edmonds, J., and R. Fulkerson, "Bottleneck Extrema," Journal of Combinatorial Theory, *8*, 299-306 (1970).
[6] Ford, L.R., and R. Fulkerson, *Flows in Networks* (Princeton University Press, Princeton, New Jersey, 1962).
[7] Finke, G., and P.A. Smith, "Primal Equivalents to the Threshold Algorithm," Operations Research Verfahren, *31*, 185-198 (1979).
[8] Garfinkel, R.S., and M.R. Rao, "The Bottleneck Transportation Problem," Naval Research Logistics Quarterly, *18*, 465-472 (1971).
[9] Glover, F., D. Karney, and D. Klingman, "The Augmented Predecessor Index Method for Locating Stepping Stone Paths and Assigning Dual Prices in Distribution Problems," Transportation Science, *6*, 171-179 (1972).
[10] Hammer, P.L., "Time Minimizing Transportation Problems," Naval Research Logistics Quarterly, *16*, 345-367 (1969).
[10a] Hammer, P.L., "Communication on 'the Bottleneck Transportation Problem,' and 'Some Remarks on the Time Transportation Problem,'" Naval Research Logistics Quarterly, *18*, 487-490 (1971).
[11] Klein, M., "A Primal Method for Minimal Cost Flows with Applications to the Assignment and Transportation Problems," Management Science, *14*, 205-220 (1967).
[12] Srinivasan, V., and G.L. Thompson, "Accelerated Algorithms for Labeling and Relabeling of Trees, with Applications to Distribution Problems," Journal of the Association for Computing Machinery, *19*, 712-726 (1972).
[13] Srinivasan, V., and G.L. Thompson, "Algorithms for Minimizing Total Cost, Bottleneck Time and Bottleneck Shipment in Transportation Problems," Networks, *1*, 179-194 (1972).
[14] Tomizawa, N., "On Some Techniques Useful for Solution of Transportation Network Problems," Networks, *1*, 179-194 (1972).
[15] Werner, R., "Beschreibung, Programmierung und Vergleich neuer Verfahren zur Lösung von Bottleneck-Transportproblemen," Thesis, Hochschule der Bundeswehr, Hamburg, Germany (1980).

SEQUENCING n JOBS ON TWO MACHINES WITH SETUP, PROCESSING AND REMOVAL TIMES SEPARATED

Dileep R. Sule

Louisiana Tech University
Ruston, Louisiana

ABSTRACT

The paper extends the machine flow-shop scheduling problem by separating processing time into setup, processing and removal times.

In a recent paper Yoshida and Hitomi [2] extended the classic two machine scheduling problem first introduced by Johnson [1]. They allowed setup times to be independent of the processing times. This paper is the further extension of the Yoshida and Hitomi model; it allows for separation of processing time into setup time, processing time and removal time for each job on each machine.

For example, consider a machine shop environment. Operations associated with each job on each machine when the machine/operator is available could be summarized as follows:

1. Setup time independent of the unit to be processed. This operation consists of activities such as obtaining the blueprints, procuring the necessary tools, fetching the required jigs and fixtures and setting them on the machine.

2. Setup time that is a unit dependent. This operation includes the time required to set the unit in the jigs and fixtures and to adjust the tools as required.

3. Processing time.

4. Removal time dependent on the unit. This operation includes the times for activities such as disengaging the tools from the unit, and releasing the unit from the jigs and fixtures.

5. Removal time independent of the unit. This operation includes activities such as dismantling the jigs, the fixtures and/or tools, inspecting/sharpening of the tools, returning them to the central depository, cleaning the machine and the adjacent area.

Since activities 2, 3 and 4 are unit dependent, their times could be combined and designated as the processing time. However, times for the activities 1 and 5 are independent of the unit to be processed.

Formulation of the Problem

Let,

P_i^m = processing time of job i on machine m. This includes times for the activities 2, 3 and 4 described before. $i = 1, 2, \ldots n$, $m = 1, 2$.

S_i^m = setup time independent of the unit, i.e., activity 1, for job i on machine m.

R_i^m = removal time, i.e., activity 5, for job i on machine m.

C_i^m = completion time of job i on machine m.

In addition, in the mathematical development, it is assumed that the jobs are designated so that the job in the ith position of the processing sequence is job i. Since the setup is independent of the unit, if there exists an idle time on machine II, setup on machine II can be done before the unit is available from machine I. The unit is available from machine I when processing on the unit is completed; however, the machine is not available for the next job until the removal operation on the machine is finished. Figure 1 shows these activities graphically.

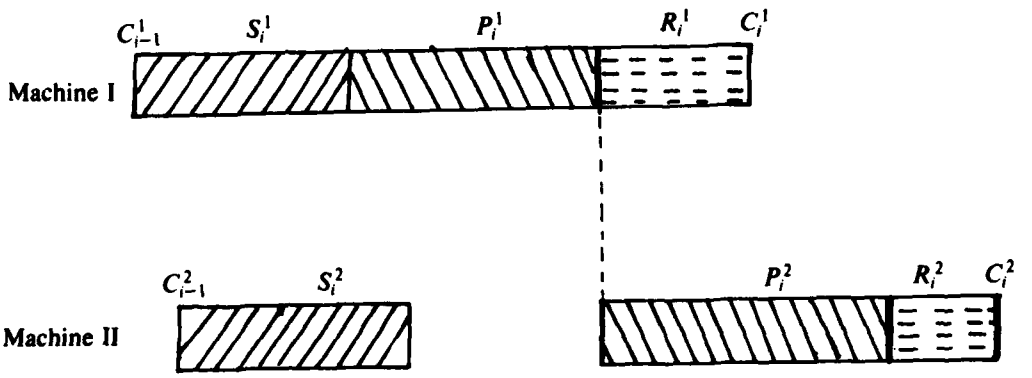


FIGURE 1. Graphical illustration of two machine problem.

Completion time for job i on machine I is given by,

$$C_i^1 = \sum_{j=1}^{i} S_j^1 + P_j^1 + R_j^1 \tag{1}$$

Completion time for job i on machine II is

$$C_i^2 = (C_{i-1}^1 + S_i^1 + P_i^1) + P_i^2 + R_i^2 \tag{2}$$

if $C_{i-1}^2 + S_i^2 < C_{i-1}^1 + S_i^1 + P_i^1$ (see Figure 1)

or

$$C_i^2 = C_{i-1}^2 + S_i^2 + P_i^2 + R_i^2$$

whenever

$$C_{i-1}^2 + S_i^2 \geqslant C_{i-1}^1 + S_i^1 + P_i^1. \tag{3}$$

The combination of (2) and (3) gives

$$C_i^2 = \text{Max}\,[C_i^1 - R_i^1 - S_i^2,\ C_{i-1}^2] + S_i^2 + P_i^2 + R_i^2. \tag{4}$$

By successive application of (4) using (1), the total elapsed time is given by

$$T = C_n^2 = \text{Max}\left\{\max_{0 \leqslant u \leqslant n}\left[\sum_{i=1}^{u}(S_i^1 - S_i^2 + P_i^1) - \sum_{i=1}^{u-1}(P_i^2 + R_i^2 - R_i^1)\right],\ 0\right\} + \sum_{i=1}^{n}(S_i^2 + P_i^2 + R_i^2). \tag{5}$$

The optimal schedule can be found by using Johnson's method to solve a two machine flow-shop problem where the processing times of job i on machines I and II are $S_i^1 - S_i^2 + P_i^1$ and $P_i^2 + R_i^2 - R_i^1$, respectively.

REFERENCES

[1] Johnson, S.M., "Optimal Two and Three-Stage Production Schedules with Setup Times Included," Naval Research Logistics Quarterly, *1*, 1, 61-68 (1954).

[2] Yoshida, T. and K. Hitomi, "Optimal Two-Stage Production Scheduling with Setup Times Separated," AIIE Transactions, *11*, 3, 261-263 (1979).

A NOTE ON THE MAXIMIN VALUE OF TWO-PERSON, ZERO-SUM GAMES*

Alvin E. Roth

Mellon Professor of Economics
University of Pittsburgh
Pittsburgh, Pennsylvania

1. INTRODUCTION

The theory of two-person, zero-sum games has occupied a special place in the general theory of games ever since its introduction by von Neumann and Morgenstern [17]. They presented the theory of two-person, zero-sum games as a natural extension of the theory of rational choice under uncertainty, as modeled by the maximization of an expected utility function. Their conclusion was that rational players should always play their maximin strategies in such games, and should regard the maximin payoff as the "value" of the game.†

Subsequent authors have questioned both the validity and generality of these conclusions. Ellsberg [4], for instance, points to gaps in the arguments which von Neumann and Morgenstern present in support of their conclusions, and raises questions related to the interpretation of the naure of the outcomes of the game, while Aumann and Maschler [1] discuss issues which arise in passing from the extensive to the normal form of a game. This latter discussion has generated a lively controversy (cf. Aumann and Maschler [2], Davis [3], Owen [7], Taylor [13]). McClennen [6] also examines gaps in the arguments presented by von Neumann and Morgenstern, and, like Aumann and Maschler, concludes that the prescription that rational players should choose maximin strategies cannot be derived directly from the principle that rational individuals are utility maximizers.

There is thus a gap between the rationality assumptions which insure that an individual evaluates single person decision problems as a utility-maximizer, and the assumptions which insure that he evaluates two-person, zero-sum games in terms of the maximin value. Two distinct issues are involved here, since the maximin value is intended to be used both to evaluate alternative games and to identify maximin strategies as "rational" choices. This paper addresses the first of these issues.

One approach to studying the maximin value is the axiomatic approach of Vilkas [15] and Tijs [14], who consider arbitrary functions defined on games and present axioms which are

*This work was supported by National Science Foundation Grant SOC75-21820 to the Institute for Mathematical Studies in the Social Sciences, Stanford University, and by National Science Foundation Grant SOC78-09928 to the University of Illinois.

†Strictly speaking, this conclusion applies only to games having saddlepoints in the set of admissable strategies. Such games are sometimes referred to as determined games.

uniquely satisfied by the maximin value. This paper takes a different, but related, approach, and shows that the maximin value of a two-person, zero-sum game can be interpreted as a rational individual's von Neumann-Morgenstern utility for playing the game, and demonstrates necessary and sufficient conditions on the individual's preferences over games for this to be the case. That is, we will consider conditions which an individuals's preferences must obey *in addition* to those which insure that he is a utility maximizer, in order for his utility function to coincide with the maximin value. This approach has been used to study other classes of games: e.g., cooperative games with side payments (Roth [8], [9]), simple games (Roth [10]), and bargaining games without side payments (Roth [11], [12]). The concluding section of this paper will discuss briefly the relationship between the results obtained here and these other results.

It should perhaps be emphasized at this point that it is not the purpose of this paper to further explore the mathematical properties of the maximin value, which are well understood. The questions which have been raised in some of the papers cited above concern, rather, the interpretation which can be placed on the maximin value as part of a theory of rational (i.e., utility-maximizing) behavior. The purpose of this paper is to further explore this latter issue.

2. THE MAXIMIN VALUE AS A VON NEUMANN-MORGENSTERN UTILITY

A two-person, zero-sum game g will be denoted by a triple $g = (S_1, S_2, u)$, where S_1 and S_2 are arbitrary sets, and u is a function such that $u : S_1 \times S_2 \to R^2$, and such that, for any $(s,t) \in S_1 \times S_2$, $u_1(s,t) = -u_2(s,t)$. The interpretation is that in the game g, the player in position i ($i = 1, 2$) will have available a set of strategies* S_i, and will compete for prizes over which his preferences are represented by the utility function u_i. His opponent's preferences will be precisely opposite his own. (Note that in general it is not necessary that the players compete for a single set of physical prizes.† It is sufficient that the results of the game be such that the two players are never in agreement over which of two outcomes is preferable.)

This formulation defines the game independently of the individuals who will play it. Consequently, it is meaningful to consider the preferences of some individual over the positions in a game, or in different games. Letting G be some class of two-person, zero-sum games, and letting $N = \{1, 2\}$ be the set of positions, we will be considering an individual's binary preference relation P defined on $N \times G$, the set of positions in a game. For $i, j \in N$ and $g, g' \in G$, the statement $(i,g)P(j,g')$ should be read "it is preferable to play position i in game g than to play position j in game g'." We will interpret P as a strict preference relation, and define an indifference relation I and a weak preference relation W in the usual way.‡ We will assume that the preference relation is defined over the mixture set of lotteries generated by $N \times G$, and that it satisfies the rationality conditions necessary for the existence of an expected utility function v.

*Typically the strategy set S_i could be the set of all probability mixtures of some finite set of pure strategies.

†Two person, zero-sum games are sometimes interpreted as games in which, for every pair of strategy choices, one player's gains are precisely equal to the other's losses in terms of some physical commodity, such as money. Under this interpretation, the game would only be zero-sum in the rare case that the two players have precisely opposed preferences for lotteries involving money. In our formulation, the game specifies the utility payoffs to be faced by the players. For a given pair of individuals to play a specified game, the monetary rewards to each would have to be adjusted in terms of each individual's utility function for money.

‡So, for $a, b \in N \times G$, aIb if and only if neither aPb nor bPa, and aWb if and only if either aPb or aIB.

Since we are interested in preferences over a possibly infinite set of alternatives (e.g., let G be the set of all matrix games) it will be simplest to refer to the axiomatization of utility due to Herstein and Milnor [5], rather than to von Neumann and Morgenstern's original axiomatization. Denote by $[p(i,g); (1-p)(j,h)]$ the lottery which with probability p has an individual play position i in game g, and with probability $1-p$ play position j in game h. Then we assume that the preference relation is defined on a mixture set containing all such lotteries, and that this preference relation has the properties of transitivity, continuity, and substitutability sufficient for the existence of an expected utility function (Herstein and Milnor [5]). That is, there exists a function v (unique up to origin and scale) which preserves preference ($v(i,g) > v(j,h)$ iff $(i,g)P(j,h)$) and evaluates the utility of a lottery by its expected utility ($v[p(i,g); (1-p)(j,h)] = pv((i,g)) + (1-p)v((j,h))$.

The fact that we are considering preferences defined over games, which are themselves defined in terms of an expected utility function, puts some additional restrictions on the allowable preferences. In particular, for any real number k, let g_k be the (degenerate) game defined by $g_k = (S_1,S_2,u)$ such that $|S_1| = |S_2| = 1$, and $u_1(S_1,S_2) = k$. Thus g_k is the game* in which each player has only one strategy to choose, which results in the player in position 1 receiving a utility of k, and the player in position 2 receiving a utility of $-k$. Since k is an expected utility, we know precisely how an individual's preferences should behave over alternative games g_k. Formally, we assume that the set G contains the set of degenerate games, and impose the following conditions on preferences over degenerate games, which have the effect of embedding the space of utilities into the space of games.

(1) $\qquad (1,g_1)P(1,g_0).$

This simply states that it is preferable to play position 1 in the degenerate game g_1 which awards the player in position 1 a utility of 1 than to play position 1 in the game g_0 and get a utility of 0.

(2) For any real numbers c and k such that $c \geqslant 1$,

$$(1,g_k)I\left[\frac{1}{c}(1,g_{ck}); \left(1-\frac{1}{c}\right)(1,g_0)\right] \text{ and } (1,g_0)I\left[\frac{1}{2}(1,g_k); \frac{1}{2}(1,g_{-k})\right].$$

The first part of this condition states that a player is indifferent between receiving a utility of k for certain in the degenerate game g_k, or participating in a lottery which gives him a utility of ck with probability $1/c$ and a utility of 0 otherwise. The second part of the condition states that a player is indifferent between receiving a utility of k or $-k$, each with probability 1/2, or receiving a utility of 0 for certain. This is precisely what is meant by the statement that the function u is an expected utility function.

Condition (1) permits us to normalize† the utility function v in the natural way, and we henceforth take $v(1,g_0) = 0$, and $v(1,g_1) = 1$. Condition (2) then completes the embedding of utilities into the space of games, giving us the following result.

PROPOSITION 1: For any real number k, $v(1,g_k) = k$.

*Technically speaking, our definition allows there to be more than one game g_k, since we could allow *different* one-element strategy sets. This difference is inessential, and can be ignored.

†Recall that v is determined only up to choice of origin and scale.

PROOF: Since v is a utility function, the first part of (2) implies that, for $c \geq 1$, $v(1,g_k) = v[1/c(1,g_{ck}); (1 - (1/c))(1,g_0)]$. But v is an expected utility function, so the utility of a lottery is its expected utility; i.e., $v[1/c(1,g_{ck}); (1 - (1/c))(1,g_0)] = (1/c)v(1,g_{ck}) + (1 - (1/c))v(1,g_0) = (1/c)v(1,g_{ck})$. Thus, for any k, $v(1,g_{ck}) = cv(1,g_k)$ for $c \geq 1$. For $0 \leq c < 1$, let $d = 1/c$, and $k' = ck$. Then $v(1,g_{dk'}) = dv(1,g_{k'})$, so $v(1,g_{ck}) = cv(1,g_k)$ for any k and any $c \geq 0$. In particular, for any $k \geq 0$, $v(1,g_k) = kv(1,g_1) = k$. But the second part of (2) implies that $v(1,g_k) = -v(1,g_{-k})$ for any k, and so $v(1,g_k) = k$ for all k, as was to be proved.

So far we have imposed conditions on preferences over degenerate games which reflect the fact that they are associated with the underlying expected utility function, but we have yet to impose any conditions which reflect the zero-sum nature of all games in G. The following conditions will suffice for our purposes.

(3) For all $g \in G$, $(1,g_0) I \left[\frac{1}{2}(1,g); \frac{1}{2}(2,g)\right]$.

This condition states that a player is indifferent between getting a utility of zero for certain in the degenerate game g_0 or participating in a lottery which gives him an equal probability of playing either position in any two-person, zero-sum game g. The motivation for this requirement is the idea that if a given zero-sum game g yields an advantage to a player in one of the positions, then it must yield a corresponding disadvantage to the player in the other position. An immediate consequence of condition (3) is the following:

PROPOSITION 2: For all $g \in G$, $v(1,g) = -v(2,g)$.

PROOF: (3) implies that $v(1,g_0) = v[1/2(1,g); 1/2(2,g)] = (1/2)v(1,g) + (1/2)v(2,g)$. But $v(1,g_0) = 0$, so $v(1,g) = -v(2,g)$.

To state the next condition, consider a game $g = (S_1,S_2,u)$ and a given position $i \in N$. for every strategy $s \in S_i$, define $k(s) = k(g,i,s) = \inf_{t \in S_j} u_i(s,t)$, where $j \neq i$.* Then in addition to the conditions already imposed on the preferences, we require the following.

(4) For any $g \in G$, $i \in N$, and $s \in S_i$, $(i,g) W (i,g_{k(s)})$.

If the game g were such that S_i contained only a single strategy, then condition (4) would simply be a version of the "sure thing" principle, which states that any prospect is at least as desirable as the worst outcome which can result from that prospect. As it stands, the condition also reflects that individual i is free to choose any strategy s in S_i. Condition (4) implies the following proposition.

PROPOSITION 3: For any $g \in G$,

$$v(1,g) \geq \sup_{s \in S_1} \inf_{t \in S_2} u_1(s,t),$$

and

$$v(2,g) \geq \sup_{s \in S_2} \inf_{t \in S_1} u_2(t,s).$$

*So $k(g,1,s) = \inf_{t \in S_2} u_1(s,t)$, and $k(g,2,s) = \inf_{t \in S_1} u_2(t,s)$.

That is, the utility of playing a game is at least as large as the "security level" which the game provides. No further assumptions are needed to permit us to reach the following conclusions.

PROPOSITION 4: For any $g \in G$,

$$\inf_{t \in S_2} \sup_{s \in S_1} u_1(s,t) \geq v(1,g) \geq \sup_{s \in S_1} \inf_{t \in S_2} u_1(s,t).$$

PROOF: The right hand inequality is simply the first part of Proposition 3, while the left hand inequality follows from Proposition 2 and the second part of Proposition 3, together with the fact that $u_1(s,t) = -u_2(s,t)$.

A game g is said to have a *saddlepoint* if there exist strategies $s \in S_1$ and $t \in S_2$ which achieve the inf sup and sup inf of Proposition 4, and make them equal. For such games, Proposition 4 has the following immediate consequence.

PROPOSITION 5: If $g \in G$ has a saddlepoint, then

$$v(1,g) = \max_{s \in S_1} \min_{t \in S_2} u_1(s,t) = \min_{t \in S_2} \max_{s \in S_1} u_1(s,t).$$

The famous Minimax Theorem of von Neumann [16] establishes that if g is the mixed extension of a finite two-person, zero-sum matrix game, then g has a saddlepoint and Proposition 5 applies.

DISCUSSION

The previous section establishes conditions on an *individual's preferences* over games which yield the maximin value as his utility for a given game. Insofar as these conditions are consistent with our other notions about zero-sum games, this shows that the maximin value is consistent, *both formally and substantively*,* with the notion of rationality embodied in utility maximization. The "machinery" necessary to establish this result shoiuld not, however, be permitted to obscure the essential simplicity of the argument.

The two substantive conditions on preferences are (3) and (4). Condition (4) serves to establish a lower bound on the utility of playing the game from either position. Condition (3), by fixing the utility for one position in the game with respect to the utility for the other position, combines with (4) to produce an upper bound on the utility. For a game with a saddlepoint these bounds coincide, and so the utility is determined.

Since the maximin value is often interpreted as being an unnecessarily conservative assessment of a game (cf. Ellsberg's [4] "reluctant duelist"), it is interesting to note that conditions (3) and (4) actually prevent the utility function from reflecting an excessively conservative attitude. In particular, if g is a game *without* a saddlepoint, then the utility of playing g in at least one of the positions must be strictly greater than the maximin value for that position. This is because Proposition 2 requires that $v(1,g) = -v(2,g)$, and, in a game without a saddlepoint, the maximin values for each position are not the negatives of one another.

*Any numerical index for certain events can of course be extended to a function which preserves expected value over lotteries. The purpose of this paper is to investigate what substantive assumptions about preferences need to be made in order to interpret this extension as an expected utility function.

Note that if the class G does include games which have no saddlepoint, then the conditions imposed in the previous section do not completely determine the utility function v. To do so, it would be necessary to precisely specify a player's attitude towards the additional uncertainty present in games with no saddlepoint. The same phenomena occurs in studying cooperative games, in which it is necessary to specify a player's attitude towards *strategic* risk (cf. Roth [9]).

As a final observation, note that, although the maximin value is linear in degenerate games, it is not linear in arbitrary games. Let g and h be finite matrix games of the same dimensions, let p be a probability (i.e., $p \in [0,1]$), and let $g' = pg + (1-p)h$ (where we employ the usual conventions of matrix arithmetic). Then each element of the matrix g' is the expected value of the corresponding element of the random matrix defined by the lottery $[pg;(1-p)h]$. We might conjecture that a player would be indifferent between playing a given position in the game g' or taking the same position in the lottery. But for preferences obeying the conditions considered in this paper, this is not the case. That is, $v[p(i,g);\ (1-p)(i,h)]$ equals $pv(i,g) + (1-p)v(i,h)$ and is in general not equal to $v(i,g')$, for $i = 1,2$.* In this respect, the preferences considered here over zero-sum games resemble the preferences over bargaining games considered in Roth [11], [12] more than the preferences over games with sidepayments considered in Roth [8], [9].

REFERENCES

[1] Aumann, R.J. and M. Maschler, "Some Thoughts on the Minimax Principle," Management Science, *18*, 5, Part II, 54-63 (1972).

[2] Aumann, R.J. and M. Maschler, "Response to Taylor's Resolution of a Paradox: Mini-Max Rehabilitated," Management Science, *20*, 9, 1316 (1974).

[3] Davis, M., "Some Further Thoughts on the Minimax Principle," Management Science, *20*, 9, 1305-1310 (1974).

[4] Ellsberg, D., "Theory of the Reluctant Duelist," American Economic Review, *46*, 909-923 (1956).

[5] Herstein, I. N. and J. W. Milnor, "An Axiomatic Approach to Measurable Utility," Econometrica, *21*, 291-297 (1953).

[6] McClennen, E. F., "Some Formal Problems with the von Neumann and Morgenstern Theory of Two-Person, Zero-Sum Games, I: The Direct Proof," Theory and Decision, *7*, 1-28 (1976).

[7] Owen, G. [1974], "A Discussion of Minimax," Management Science, *20*, 9, 1316-1317 (1974).

[8] Roth, A.E., "The Shapley Value as a von Neumann-Morgenstern Utility," Econometrica, *49*, 657-664 (1977).

[9] Roth, A.E., "Bargaining Ability, the Utility of Playing a Game, and Models of Coalition Formation," Journal of Mathematical Psychology, *16*, 153-160 (1977).

[10] Roth, A.E., "Utility Functions for Simple Games," Journal of Economic Theory, *16*, 481-489 (1978).

*Since the maximin value is linear in degenerate games, the simplest example of this phenomenon can occur only when when at least one of the players has at least two strategies. So let g and h be the 1×2 matrix games $g = (0,1)$; $h = (1,0)$ where we follow the usual convention that rows and columns indicate the strategy choices of players 1 and 2 respectively, and the matrix entries represent player 1's utility payoffs. Then $v(1,g) = v(1,h) = 0$, and so $v([1/2(1,g);\ 1/2(1,h)]) = 0$. But letting g' be the "expected" matrix $g' = (1/2)g + (1/2)h = \left|\frac{1}{2}\ \frac{1}{2}\right|$, we find that $v(1,g') = 1/2$.

[11] Roth, A.E., "The Nash Solution and the Utility of Bargaining," Econometrica, *46*, 587-594, 983 (1978).
[12] Roth, A.E., *Axiomatic Models of Bargaining* (Springer, New York, 1979).
[13] Taylor, P.T., "Resolution of a Paradox: Mini-Max Rehabilitated," Management Science, *19*, 4, Part I, 466-467 (1972).
[14] Tijs, S.H., "A Characterization of the Value of Zero-Sum Two-Person Games," Naval Research Logistics Quarterly, *28*, 153-156 (1981).
[15] Vilkas, E.I., "Axiomatic Definition of the Value of a Matrix Game," Theory of Probability and its Applications, 9, 304-397 (1963).
[16] von Neumann, J. "Zur Theorie der Gesellschaftsspiele," Mathematische Annalen, *100*, 295-320 (1928).
[17] von Neumann, J. and O. Morgenstern, *Theory of Games and Economic Behavior* (Princeton University Press, 3rd edition, 1953).

A NOTE ON A COMPUTATIONAL MODEL FOR A DATA/VOICE COMMUNICATION QUEUEING SYSTEM

Richard M. Feldman

Industrial Engineering Department
Texas A&M University
College Station, Texas

Carroll A. Claybaugh

Operations Directorate
Defense Communications Agency
Arlington, Virginia

ABSTRACT

Certain types of communication nodes can be viewed as multichannel queueing systems with two types of arrival streams. Data arrivals are characterized by high arrival and service rates and have the ability to queue if all service channels are busy. Voice arrivals have small arrival and service rates and do not have the ability to wait when the channels are full. Computational procedures are presented for obtaining the invariant probabilities associated with the queueing model.

1. INTRODUCTION

The steady growth of the data processing industry and the telephone network over the last four decades has introduced computer-communications networks as efficient transportation vehicles for the remote sharing of information data bases. Recent Defense Communications Agency studies have shown the desirability of a network which integrates voice, interactive, and bulk data for the 1980's (Rosner [10], Schmitz, Saxton, Huang and White [11]). Several additional studies relating to information processing growth in the next few decades portend new data/voice services with substantially increased data flows (Rich and Schwartz [8], Rosner [9]). There are several different time division integration methodologies proposed for combining voice and data demands over the available channel bandwidth (Coviello and Vena [2], Fischer and Harris [4], Frank and Gitman [5]). A competitive allocation scheme allows the data and voice calls to compete for time slots using a first-come, first-served scheme.

Simulation models have been developed to study competitive allocation schemes; however, because of their expense and the fact that rare events are of interest, simulation studies are not necessarily appropriate. Analytical models have also been developed to describe competitive allocation (Bhat and Fischer [1], Fischer [3], Fischer and Harris [4]). The analytical models developed to date all share the common drawback of not being tractable for problems of reasonable size. The purpose of this paper is to present an analytical model of the dynamic

movement of voice and data calls within an integrating device and give a tractable solution procedure for the model. The allocation scheme is that used by Bhat and Fischer [1]. The methodology for the solution scheme combines procedures developed by Neuts [6,7] and Wong, Griffin, and Disney [12].

2. THE MODEL

There are two classes of calls that arrive to the service center: data calls and voice calls. The arrival process of data calls is assumed to be Poisson with rate λ_1 and the arrival process of the voice calls is assumed Poisson with rate λ_2; the two arrival processes being independent. Although each data call actually consists of a random number of discrete bits, the length of each data call is assumed to be adequately approximated by a continuous exponential random variable with mean $1/\mu_1$ and the length of the voice call is assumed exponential with mean $1/\mu_2$. The characteristics of data calls are frequent arrivals and short length whereas the voice calls have infrequent arrivals with a long service time. A buffer is available that can store incoming data calls when all channels are busy. The size of the buffer is such that no effective limit is placed on the queue size. Voice calls, however, may not form a queue so that any incoming voice call is lost whenever all channels are busy. The service center has c channels using a FIFO discipline.

The data/voice integrating system is modeled as a Markov process with a two dimensional state space given by

$$E = \{(n,m): m = 0,1, \ldots, c \text{ and } n = 0,1, \ldots\}.$$

The vector $\mathbf{p}$ (ordered lexicographically) denotes the steady state probabilities where $p(n,m)$ is the steady state probability that there are n data calls in the system and m voice calls in the system. For ease of notation the vector $\mathbf{p}$ is partitioned as

$$\mathbf{p} = (\mathbf{p}_0, \mathbf{p}_1, \ldots) \tag{1}$$

where the mth component of $\mathbf{p}_n$ is $p(n,m)$. The vector $\mathbf{p}$ is the solution to the equation

$$\mathbf{p}Q = \mathbf{0}$$

$$\mathbf{p}\mathbf{1} = 1 \tag{2}$$

where $\mathbf{1}$ and $\mathbf{0}$ are vectors of all ones and zeros, respectively, and Q is the transition rate matrix for the Markov process representing the data/voice system. The matrix Q is infinite and is given by

$$Q = \begin{bmatrix} A_0 & \Lambda & & & & & \\ M_1 & A_1 & \Lambda & & & & \\ & \ddots & \ddots & \ddots & & \bigcirc & \\ & & M_{c-1} & A_{c-1} & \Lambda & & \\ & & & M_{c-1} & A_c & \Lambda & \\ & \bigcirc & & & M_c & A_c & \Lambda \\ & & & & & \ddots & \ddots & \ddots \end{bmatrix} \tag{3}$$

where the submatrices are defined, for $i,j,n = 0, 1, \ldots, c$, by

$$\Lambda(i,j) = \begin{cases} \lambda_1 & \text{if } i = j, \\ 0 & \text{otherwise;} \end{cases} \tag{4}$$

$$M_n(i,j) = \begin{cases} n\mu_1 & \text{if } i = j \text{ and } j \leqslant c - n, \\ (c - j)\mu_1 & \text{if } i = j \text{ and } j > c - n, \\ 0 & \text{otherwise;} \end{cases} \tag{5}$$

$$A_n(i,j) = \begin{cases} \lambda_2 & \text{if } i = j - 1 \text{ and } i \leqslant c - n - 1, \\ \mu_2 & \text{if } i = j + 1, \\ a_n(i) & \text{if } i = j, \\ 0 & \text{otherwise;} \end{cases} \tag{6}$$

where $a_n(i)$ is such that the row sum is zero.

3. SOLUTION PROCEDURE

The general structure of the matrix Q is identical to the type investigated by Neuts [6] and similar to the finite system investigated by Wong, Griffin and Disney [12]. The solution method as given by Neuts [6] is a two step process. First, it is necessary to find the matrix R such that

$$R^2 M_c + R A_c + \Lambda = 0. \tag{7}$$

The matrix R gives the relationship

$$\mathbf{p}_{c+k} = \mathbf{p}_{c-1} R^{k+1} \quad \text{for } k = 0, 1, \ldots. \tag{8}$$

The second step is to let $\hat{\mathbf{p}} = (\mathbf{p}_0, \ldots, \mathbf{p}_{c-1})$ and then solve for $\hat{\mathbf{p}}$ using equations

$$\hat{\mathbf{p}} T = 0 \tag{9}$$

and

$$\hat{\mathbf{p}} \mathbf{1} + \mathbf{p}_{c-1} R (I - R)^{-1} \mathbf{1} = 1$$

where

$$T = \begin{bmatrix} A_0 & \Lambda & & & \\ M_1 & A_1 & \Lambda & & \\ & \ddots & \ddots & \ddots & \\ & & & & \Lambda \\ & & & M_{c-1} & A_{c-1} + R M_c \end{bmatrix}$$

It will turn out that for this communication problem, the matrix R of equation (7) is easy to obtain whereas $\hat{\mathbf{p}}$ from equation (9) gives computational difficulties which will be overcome by using a technique utilized by Wong, Griffin and Disney [12].

The matrix R is lower triangular since the matrices M_c, A_c, and Λ are lower triangular. Therefore, the diagonal elements of R^2 are simply the square of the diagonal elements of R. Thus, the following equation results when considering the diagonal elements in (7) for $i = 0, \ldots, c$

$$r_{ii}^2 (c - i)\mu_1 - r_{ii}[\lambda_1 + i\mu_2 + (c - i)\mu_1] + \lambda_1 = 0. \tag{10}$$

The solution of (10) yields

$$r_{cc} = \lambda_1/(c\mu_2 + \lambda_1)$$

and for $i = 0, \ldots, c - 1$

$$r_{ii} = \{\lambda_1 + i\mu_2 + (c - i)\mu_1 - [(\lambda_1 + i\mu_2 + (c - i)\mu_1)^2 - 4\lambda_1(c - i)\mu_1]^{1/2}\}/\{2(c - i)\mu_1\}. \tag{11}$$

The off-diagnonal elements are not quite as straight forward but by considering the i,j element with $i > j$, equation (7) yields

$$(c - j)\mu_1 \sum_{k=j}^{i} r_{ik} r_{kj} + \sum_{k=0}^{c} r_{ik} A_c(k,j) = 0$$

which in turn yields

$$(c - j)\mu_1 [r_{ij}(r_{ii} + r_{jj}) + \sum_{k=j}^{i-1} r_{ik} r_{kj}] + (j + 1)\mu_2 r_{i,j+1} = [\lambda_1 + j\mu_2 + (c - j)\mu_1] r_{ij}$$

and thus for $i > j$

$$r_{ij} = \{(j+1)\mu_2 r_{i,j+1} + (c-j)\mu_1 \sum_{k=j}^{i-1} r_{ik} r_{kj}\}/\{\lambda_1 + j\mu_2 + (c-j)\mu_1 (1 - r_{jj} - r_{ii})\} \tag{12}$$

where the sum in equation (12) is defined as zero if $j = i - 1$. Equation (11) and (12) give an easy iterative procedure where the diagonal elements of R are first computed, then the elements such that $j = i - 1$ are computed then the elements such that $j = i - 2$, etc. In this manner the exact expression for R is obtained and the solution of equation (9) remains.

The obvious difficulty with (9) is that for a typical system in which $c = 48$, the dimension of T is 2352×2352. In order to obtain a solution to (9) it will be reduced to solving a 49×49 system.

Equation (9) can be rewritten as

$$\mathbf{p}_0 A_0 + \mathbf{p}_1 M_1 = \mathbf{0} \tag{13}$$

$$\mathbf{p}_{k-1}\Lambda + \mathbf{p}_k A_k + \mathbf{p}_{k+1} M_{k+1} = \mathbf{0} \quad \text{for } k = 1, \ldots, c - 2 \tag{14}$$

$$\mathbf{p}_{c-2}\Lambda + \mathbf{p}_{c-1}(A_{c-1} + RM_c) = \mathbf{0}. \tag{15}$$

Each of the submatrices in the above equation are of dimension $(c + 1) \times (c + 1)$. New matrices B_k are defined with dimension $(2c + 2) \times (2c + 2)$ by

$$B_k = \begin{bmatrix} -A_k \Lambda^{-1} & I \\ -M_{K+1}\Lambda^{-1} & 0 \end{bmatrix} \quad \text{for } k = 1, \ldots, c - 2.$$

It now follows from (14) that

$$(\mathbf{p}_{k-1}, \mathbf{p}_k) = (\mathbf{p}_k, \mathbf{p}_{k+1})B_k \tag{16}$$

and thus

$$(\mathbf{p}_0, \mathbf{p}_1) = (\mathbf{p}_{c-2}, \mathbf{p}_{c-1})B_{c-2}B_{c-3} \cdots B_1. \tag{17}$$

Combining equations (13) and (15) with (17) results in

$$\mathbf{p}_{c-1}([A_{c-1} + RM_c]\Lambda^{-1}, I)B_{c-2} \cdots B_1 \begin{pmatrix} A_0 \\ M_1 \end{pmatrix} = \mathbf{0}. \tag{18}$$

Equation (18) gives a solution for $\mathbf{p}_{c-1}$ that is unique up to a multiplicative constant. Any such solution is obtained and equation (16) is used to obtain $\mathbf{p}_{c-2}, \mathbf{p}_{c-3}, \ldots, \mathbf{p}_0$. The norming equation of (9) is then used and the steady state probabilities are determined.

A unique combination of two previously determined solutions techniques thus yields a computational procedure that can be used to solve a class of problems in the telecommunications field. Typical measures of effectiveness can easily be determined in the same manner as in Neuts [6]. Thus, the utilization of analytical models for such telecommunication systems is now feasible.

ACKNOWLEDGMENTS

The authors appreciate the suggestions made by M. F. Neuts and R. L. Disney regarding solution procedures.

REFERENCES

[1] Bhat, U.N. and M.J. Fischer, "Multichannel Queueing Systems with Heterogeneous Classes of Arrivals," Naval Research Logistics Quarterly, *23*, 271-282 (1976).
[2] Coviello, G.J. and P.A. Vena, "Integration of Circuit/Packet Switching by a SENET Concept," National Telecommunication Conference, New Orleans, *75*, 42.12-42.17 (1975).
[3] Fischer, M.J., "Two Queueing Systems Sharing the Same Finite Waiting Room," Naval Research Logistics Quarterly, *25*, 667-679 (1978).
[4] Fischer, M.J. and T.C. Harris, "A Model for Evaluating the Performance of an Integrated Circuit and Packet Switched Multiplex Structure," IEEE Transactions on Communications, Com-*24*, 195-202 (1976).
[5] Frank, H. and I. Gitman, "Integrated DOB Voice and Data Networks," Report, AD-A039329, National Technical Information Service, Springfield, VA (March 1977).
[6] Neuts, M.F., "Further Results in the M/M/1 Queue with Randomly Varying Rates," Opsearch, *15*, 158-168 (1978).
[7] Neuts, M.F., "The M/M/1 Queue with Randomly Varying Rates," Opsearch, *15*, 139-157 (1978).
[8] Rich, M.A. and M. Schwartz, "Buffer Sharing in Computer-Communication Network Noder," IEEE Transactions on Communications, Com-*25*, 958-970 (1977).
[9] Rosner, R.D., "A Digital Data Network Concept for the Defense Communications Agency." National Telecommunications Conference, New Orleans, *75*, 42.1-42.7 (1975).
[10] Rosner, R.D., "Large Scale Network Design Consideration," International Computer Communication Conference, Stockholm, Sweden, *74*, 189-197 (1974).

[11] Schmitz, H.G., T.L. Saxton, C.C. Huang and J.A. White, "Applications of Associative Processing Techniques to an Integrated Voice/Data Switched Network," Report AD-A040636, National Technical Information Service, Springfield, VA (June 1976).
[12] Wong, B., W. Giffin and R.L. Disney, "Two Finite M/M/1 Queues in Tandem: A Matrix Solution for the Steady State," Opsearch, *14*, 1-18 (1977).

A NOTE ON THE INFLUENCE OF MISSING OPERATIONS ON SCHEDULING PROBLEMS

N. Hefetz and I. Adiri

TECHNION—Israel Institute of Technology
Haifa, Israel

ABSTRACT

This paper attempts to resolve the existing confusion concerning missing operations. Scheduling problems are classified in two groups: (i) null-continuous (NC)—comprising the problems where an optimal schedule remains optimal on replacement of arbitrarily small processing times (existing operations) with zeros (missing operations); (ii) null-discontinuous (NDC)—comprising those problems which are not null-continuous.

A "zero processing time" of an operation refers to either of the two following contingencies: (i) An actual operation whose processing time tends to zero. Thus, if t_{ij} denotes the processing time of operation 0_j in job J_i, then for a sufficiently small positive number ϵ, scheduling problems with $t_{ij} = \epsilon$ and $t_{ij} = 0$ have the same optimal schedules (algorithms), and ϵ may reasonably be replaced with zero to facilitate calculations, and (ii) A nonexisting (missing) operation.

Since an infinitesimal (arbitrarily small) processing time operation (i) has a starting time while a missing operation (ii) has not, the two types have a different effect on a scheduling problem and must be differentiated to prevent ambiguity. Accordingly, we propose to designate an infinitesimal processing time operation as ϵ, and a missing operation as a zero.

Scheduling problems involving operations with arbitrarily small processing times (existing operations where, for all practical purposes, the length may be considered as zero, but have starting times) are basically the same as those with the usual strictly-positive processing times and do not merit separate consideration.

Whether a discipline (flowshop or openshop) allows missing operations or not depends on its definition, without clear preference for one definition over another. However, for the sake of understanding, the definition (whatever it is) must be known and accepted. We propose to define flowshop and openshop disciplines allowing missing operations. (Note, [2] and [3] define flowshop allowing missing operations while [4], [9] and [14] implicitly assume that flowshop does not allow missing operations.) Accordingly, F or 0 in the notation $n/m/\gamma/\delta$*, $\gamma \in \{F, 0\}$ comes under that definition and the processing time of the 0_j operation of job J_i, $(i = 1, 2, \ldots, n;\ j = 1, 2, \ldots, m)$, is a nonnegative integer, $t_{ij} \geq 0$—positive if the operation exists and zero if it does not. In the case of a flowshop, or an openshop, where all n jobs have m operations—i.e., where the processing times of all mn operations are strictly positive—

*The notation is that proposed by Lenstra [9] and Rinnooy Kan [14].

the constraint $t_{ij} > 0 \; \forall \; (i,j)$ is added, and the designation becomes $n/m/\gamma$, $t_{ij} > 0/\delta$, $\gamma \in \{F, 0\}$.

Panwalker, Smith and Woollam [11], demonstrated that the commonly held view that there exists an optimal permutation schedule for $n/3/F/C_{\max}$ and $n/m/F$, no-wait/$C_{\max}$, $m \geqslant 3$, turns out to be correct only for positive processing times.* (The latter fact was recognized by Baker [2].)

1. NULL-CONTINUOUS AND DISCONTINUOUS SCHEDULING PROBLEMS

The general rule is that optimal schedule remain unchanged under small changes in the close neighborhood of the given processing times, providing there is no replacement of ϵ with zero (deletion of an infinitesimal operation). Where such replacement is resorted to, some problems remain unaffected while in others the optimal schedule undergoes total transformations; in the latter category, an optimal algorithm and schedule for $t_{ij} > 0 \; \forall (i,j)$ are not optimal for $t_{ij} \geqslant 0$. Moreover, there are problems for which there exists an efficient optimal algorithm (belonging to P) for the first case ($t_{ij} > 0 \; \forall \; (i,j)$) while for the second ($t_{ij} \geqslant 0$), the problem is NP-Complete.

DEFINITIONS

Null-continuous (NC) scheduling problem — one where an optimal schedule remains optimal on replacement of arbitrarily small processing times with zeros and vice versa. Thus, an optimal algorithm and schedule for $t_{ij} > 0 \; \forall \; (i,j)$ are also optimal for $t_{ij} \geqslant 0$.

Null-discontinuous (NDC) scheduling problem — one which is not null-continuous. Thus, an NDC problem defines two different problems, one with strictly positive processing times ($t_{ij} > 0 \; \forall \; (i,j)$—missing operations are not allowed) and the other where zero processing times are allowed ($t_{ij} \geqslant 0$—missing operations are allowed).

Informally, a problem where all operations with arbitrarily small processing times for every machine can be shifted to the beginning or the end of the schedule without affecting the measure of performance or violating the constraints—is NC. To show that a problem is NDC, it suffices to work out an example where replacement of arbitrarily small processing time with a zero results in a different optimal schedule.

The following numerical examples of NC and NDC problems demonstrate the fact that an NDC problem defines two different problems—one where $t_{ij} > 0$ and the other where $t_{ij} \geqslant 0$.

EXAMPLE 1:

NC instance — $5/2/F/C_{\max}$† with processing times as per Table 1.

*[11] was brought to our attention by the referee. At the time of writing the first version of the paper, we were unaware of [11] and independently reached the same conclusions.

†The $n/2/F/C_{\max}$ and $n/2/F$, no-wait/$C_{\max}$ problems are NC and NDC, respectively.

TABLE 1 — *Processing Times*

Operation (machine) \ Job	1	2	3	4	5
1	4	ϵ	ϵ	3	2
2	2	2	2	ϵ	4

The problem is NC since arbitrarily small processing time operations on the first and second machines can be shifted to the beginning and the end of the schedule, respectively, without violating any constraint or affecting C_{max}. Thus, optimal schedules constructed by Johnson's algorithm [8] remain optimal after replacement of ϵ with zeros.

NDC Instance—$5/2/F$, no-wait/C_{max} with processing times again as per Table 1. For $t_{ij} > 0\ \forall\ (i,j)$ Gilmore and Gomory's algorithm [5], [13] yields an optimal schedule as presented in Figure 1. The algorithm itself is not affected by replacing ϵ with zeros, but the resulting schedule is no longer optimal (Figure 2).

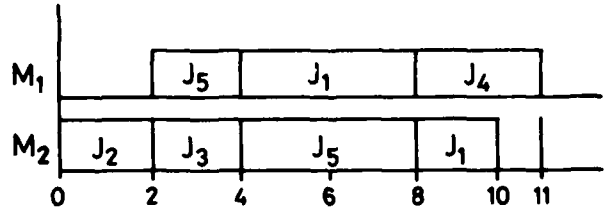


FIGURE 1. Optimal schedule for $5/2/F$, no-wait, $t_{ij} > 0/C_{max}$, ($\epsilon > 0$)

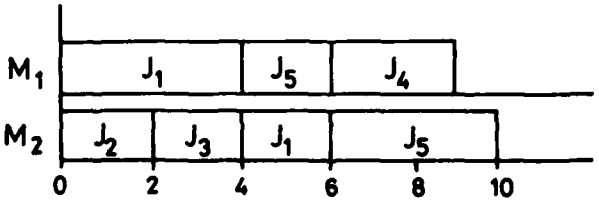


FIGURE 2. Optimal schedule for $5/2/F$, no-wait/C_{max}, ($\epsilon > 0$)

NP-Completeness of a scheduling problem is proved by reducing a known NP-Complete problem to an instance of the problem in question that in many cases contains zero processing times. In the light of the preceding discussion only if the problem in question is NC then results obtained for $t_{ij} \geqslant 0\ \forall (i,j)$ are also true for $t_{ij} > 0$, and vice versa; if NDC, the problem

may refer either to $t_{ij} > 0\ \forall(i,j)$ or $t_{ij} \geqslant 0$ with a different approach and specific optimal algorithm and schedule in each case. Thus, before zero processing times are allowed, the problem in question must be characterized as NC or NDC.

2. CLASSIFICATION AND COMPLEXITY OF NC AND NDC PROBLEMS

A first attempt has been made to classify some of the scheduling problems into NC and NDC, and to examine the complexity of each problem (the results are obtained in Table 2). As explained previously, each NDC problem defines two different problems, one with strictly positive processing times ($t_{ij} > 0\ \forall(i,j)$) and the other with nonnegative ones, ($t_{ij} \geqslant 0$). The fact that for a particular problem, the case where $t_{ij} > 0\ \forall(i,j)$ is a special case of $t_{ij} \geqslant 0$ and, thus, its complexity is lower than or equal to that of the latter (theoretically speaking, all combinations are possible) is demonstrated in Table 2.

TABLE 2 — *Classification and Complexity of NC and NDC Problems*

	Problem	Complexity	
		$t_{ij} > 0$	$t_{ij} \geqslant 0$
NDC	$n/2/F$, no-wait/C_{max}*	$0(n^2)$, [5]	Unary NP-Complete, [15]
	$n/3/F$, no-wait/C_{max}	?†	Unary NP-Complete, [4]
	$n/4/F$, no-wait/C_{max}	Unary NP-Complete, [12]	Unary NP-Complete, [12]
	$n/2/F, r_i \geqslant 0/C_{max}$	Binary NP-Complete, [10]	Binary NP-Complete, [10]
	$n/2/F$, tree/C_{max}‡	Binary NP-Complete, [10]	Binary NP-Complete, [10]
	$n/2/F$, tree'/C_{max}§	$0(n \log n)$, [16]	?
	$n/3/F/C_{max}$	Unary NP-Complete, [10]	Unary NP-Complete, [4]
	$n/2/0$, no-wait/C_{max}*	Unary NP-Complete, [15]	Unary NP-Complete, [15]
	$n/2/J, n_i = 2$, no-wait/C_{max}*	Unary NP-Complete, [15]	Unary NP-Complete, [15]
	$n/2/F/\Sigma C_i$	Unary NP-Complete, [4]	Unary NP-Complete, [4]
	$n/2/F$, no-wait/ΣC_i	?†	Unary NP-Complete, [4]
	$n/2/0$, no-wait/ΣC_i	?	Unary NP-Complete, [1]
	$n/2/0/\Sigma C_i$	?	Unary NP-Complete,[1]
	$n/2/J/n_i = 2$, no-wait/ΣC_i	?	Unary NP-Complete, [4]
	$n/1$/seq.dep./C_{max} (Travelling salesman)	Unary NP-Complete, [14]	Unary NP-Complete,[14]
NC	$n/2/F/C_{max}$	$0(n \log n)$, [8]	
	$n/2/J, n_i \leqslant 2/C_{max}$	$0(n \log n)$, [7]	
	$n/m/0/\delta$	Depends on the number of machines (m) and the regular measure of performance (δ).	

*Flowshop, jobshop and openshop disciplines are defined allowing zero processing times.

†Prize carrying open problem, [6], [9], [10].

‡tree—Tree precedence relations where job J_i may start only after job J_j has been completed.

§tree' - Tree precedence relations where, on each machine, job J_i may start only after J_j on this machine has been completed.

ACKNOWLEDGMENT

The authors wish to thank Professor D. S. Johnson of Bell Laboratories, Murray Hill, for his constructive comments.

BIBLIOGRAPHY

[1] Achugbue, J.O. and F.Y. Chin, "Scheduling the Openshop to Minimize Mean Flow Time," Technical Report TR79-8, Department of Computer Science, University of Alberta, Canada (1979).

[2] Baker, K.R., *Introduction to Sequencing and Scheduling* (John Wiley and Sons, New York, 1974).

[3] Conway, R.W., W.L. Maxwell and L.W. Miller, *Theory of Scheduling* (Addison-Wesley, Reading, Mass., 1967).

[4] Garey, M.R., D.S. Johnson and R. Sethi, "The Complexity of Flowshop and Jobshop Scheduling," Mathematics of Operations Research, *1*, 117-129 (1967).

[5] Gilmore, P.C. and R.E. Gomory, "Sequencing a One-State Variable Machine: a Solvable Case of the Travelling Salesman Problem," Operations Research, *12*, 655-679 (1964).

[6] Graham, R.L., E.L. Lawler, J.K. Lenstra and A.H.G. Rinnooy Kan, "Optimization and Approximation in Deterministic Sequencing and Scheduling: A Survey," Mathematical Center, BW 82/77, Amsterdam (1977).

[7] Jackson, J.R., "An Extension of Johnson's Results on Job Lot Scheduling," Naval Research Logistics Quarterly, *3*, 201-203 (1956).

[8] Johnson, S.M., "Optimal Two- and Three-Stage Production Schedules with Setup Times Included," Naval Research Logistics Quarterly, *1*, 61-68 (1954).

[9] Lenstra, J.K., "Sequencing by Enumerative Methods," Mathematical Center, Amsterdam (1977).

[10] Lenstra, J.K., A.H.G. Rinnooy Kan and P. Brucker, "Complexity of Machine Scheduling Problems," Annals of Discrete Mathematics, *1*, 343-362 (1977).

[11] Panwalkar, S.S., M.L. Smith and C.R. Woollam, "Counterexamples to Optimal Permutation Schedules for Certain Flowshop Problems," Naval Research Logistics Quarterly, *28*, 339-340 (1981).

[12] Papadimitriou, C.H. and P.C. Kanellakis, "Flow-Shop Scheduling with Limited Temporary Storage," To appear in Journal of the Association of Computing Machinery.

[13] Reddi, S.S. and C.V. Ramamoorthy, "On the Flow-Shop Sequencing Problem with No Wait in Process," Operational Research Quarterly, *23*, 323-331 (1972).

[14] Rinnooy, Kan A.H.G., *Machine Scheduling Problems: Classifications, Complexity and Computations* (Nijhoff, The Hague, 1976).

[15] Sahni, S. and Y. Cho, "Complexity of Scheduling Shops with No Wait in Process," Technical Report 77-20, Computer Science Department, University of Minnesota, Minneapolis (1977).

[16] Sidney, J.B., "The Two-Machine Maximum Flow Time Problem with Series-Parallel Precedence Constraints," Faculty of Management Sciences, University of Ottawa, Canada (1977).

INFORMATION FOR CONTRIBUTORS

The NAVAL RESEARCH LOGISTICS QUARTERLY is devoted to the dissemination of scientific information in logistics and will publish research and expository papers, including those in certain areas of mathematics, statistics, and economics, relevant to the over-all effort to improve the efficiency and effectiveness of logistics operations.

Manuscripts and other items for publication should be sent to Professor Herbert Solomon, Editor, Naval Research Logistics Quarterly, Department of Statistics, Sequoia Hall, Stanford University, Stanford, California 94305.

Manuscripts submitted for publication should be typewritten, double-spaced, and the author should retain a copy. Refereeing will be expedited if four copies of the manuscript are submitted.

A short abstract (not over 400 words) should accompany each manuscript. This will appear at the head of the published paper in the QUARTERLY.

There is no authorization for compensation to authors for papers which have been accepted for publication. Authors will receive reprints of their published papers.

NAVAL RESEARCH
LOGISTICS
QUARTERLY

SEPTEMBER 1982
VOL. 29, NO. 3
NAVSO P-1278

CONTENTS

ARTICLES


DATE
ILME