Machine Monitoring Using Probability Thresholds and System Operating Characteristics

Stephen M. Pollock
Department of Industrial and Operations Engineering
The University of Michigan
Ann Arbor MI 48109-2117, USA

and

Jeffrey M. Alden
G.M. R&D Center
Warren MI 48090

Technical Report 95-14

1. Introduction

We are concerned with detecting a change in the underlying condition of a process, when the available observations are related only probabilistically to this condition. This situation has long been of concern to statisticians, engineers, economists, epidemiologists, etc. In this paper we address the specific context of monitoring a production machine, with the objective of effectively determining when to shut the machine down for maintenance or replacement. Applications to other areas such as quality control, health, military surveillance, or economic analysis should be readily apparent

Before presenting any details, we list some definitions and initial assumptions, in order to clarify the general setting that governs our analysis:

a) There is an underlying time interval that characterizes the operation of the machine. It is convenient to think of this time interval as the "part production cycle time" of a machine that produces discrete parts (although it may also be advantageous to use natural time units such as minutes or hours). All times and intervals are subsequently measured in units of this time interval.

b) The machine can be in only one of two conditions: "good" or "bad" (denoted G and B, respectively). By the "G" condition we mean the machine is operating in such a way that it is "in control" or "normal" or "producing acceptably"; by the B condition we mean it is "out of control" or "failed" or "producing bad parts (scrap)".

c) The machine starts in G but at some operating time T goes to B. Ths is called a "failure event", or more simply, a "failure". The time T, called the *failure time*, is a random variable with known probability density function (p.d.f.) $\phi(t)$, cumulative distribution function $\Phi(t)$, and expected value $E(T)$.

d) Observations of "signals" which are probabilistically related to the machine's condition are made at fixed, pre-determined times.

e) Immediately following any observation one of two possible actions can be made: "do nothing" or "take an action consistent with believing the machine is about to go into (or is

in) B". The latter action is called a *check*.

f) When a check is made production is stopped and the condition of the machine becomes known with certainty. A check that finds the machine in G, called a *false alarm*, returns the machine to operation (in G) after an interval of length g. A check that finds the machine in B, called a *true alarm*, re-sets it to "as-new" condition – or, equivalently, replaces it by a new (identical) machine – after an interval of length b. This event is called a *renewal*. Typically $g < b$ since renewal often requires fixing or replacing something, while checking when the machine is in G may only require a brief inspection to ascertain that it is in fact in G.

g) The process (observations, failures, checking, etc.), which we call "monitoring", continues indefinitely, with successive failure times assumed to be independent and identically distributed (I.I.D.).

All of these assumptions, of course, must be eventually relaxed to conform to the realities of any actual processes under consideration. On the other hand, the important underlying relations among performance measures, and their dependance upon machine parameters and checking strategies, are exhibited using this set of simplifying assumptions.

Our fundamental problem, then, is to determine a policy, i.e. when to check, knowing the complete history of the process, including its age, the elapsed time since the last check and the values of all observations made to date. There are two competing concerns that underly the determination of an effective policy: checking soon enough so that the machine does not operate too long in condition B , while not checking so often that the machine is shut down unnecessarily. In other words, an economic (or other) advantage is gained when a policy raises an alarm that detects the occurrence of B soon after it happens. However, it is also desirable to avoid costly false alarms that result in shutting the machine down to check it when it is still in condition G.

Trading off (or constraining) the costs of delayed failure detection and false alarms is the basis of most quality control and control chart procedures developed over the past sixty

years [see Shewhart (1931), Roberts (1966), Johnson and Leone (1962), Montgomery (1980), Lorenzen and Vance (1986), for a historical perspective and related formulations]. The most common approaches to "optimizing" these procedures [e.g. Moskowitz, Plante and Chun (1989), Saniga (1989)] use economic models that explicitly incorporate costs ascribable to false alarms and delayed checking. Such trade-off analyses and economic approaches depend upon specific policy structures that, although they have intuitive appeal and lead to easy computation or evocative charting methods, can be inefficient or arbitrary in nature.

More important, such methods often essentially ignore what is known about the machine's failure time distribution. In contrast, we explicitly make use of this distribution to present and evaluate a checking policy that is "optimal" according to a broad set of criteria. Our approach is also motivated by a natural inclination to develop checking policies that become, in appropriate limits, those that have been shown in the literature to be optimal for those well-studied situations where either *no* observations or *perfect* observations are made. The former is usually analysed under the rubric of "optimal maintenance policies" [see, for example, Barlow et al. (1963)]; the latter has been the subject of "optimal replacement policies" [see, for an early example, Page (1954)]. These special cases "bracket" the capabilities of any realizable system.

2. Performance Measures

The use of *any* checking policy ultimately results in performance measures of interest to decision makers. Before one can find effective checking policies, then, it is important to define these measures and understand the relations among them. We choose to list these measures in three general categories, defined below, along with parameters and variables that will be used in our analysis.

Type One (False Alarm) Measures.

False alarms are costly since resources are used to process each alarm, and production time is often lost as well. Measures that serve as proxies for these costs include:

$r_f \equiv$ *false alarm rate* $\equiv$ the expected number of false alarms per unit time,

$p_f \equiv$ fraction of total time spent processing false alarms,

$\mu \equiv$ expected number of false alarms until failure,

ARL (sometimes denoted ARL0) $\equiv$ the expected machine operating time until a check, given the machine *starts and remains* in G

Type Two (Late Detection) Measures.

Being slow to stop a machine that is in B leads to the production of bad parts or simply to lost production. Measures of the cost of such delayed failure detection include:

$D \equiv$ (random variable) time between failure and the next check,

$\delta \equiv$ *expected detection time* $\equiv$E(D), also called EDD – "expected delay in detection" – see Marcellus(1993),

$r_t \equiv$ *true alarm rate* $\equiv$ the expected number of true alarms per unit time,

$p_t \equiv$ fraction of total time spent processing true alarms,

ARL1 $\equiv$ the expected machine operating time until a check, given the machine *starts and remains* in B

We note here that, in spite of their popularity in the literature (and in practice), the use of the average run length measures ARL and ARL1 is questionable for two reasons:

a) the hypothetical situation (required for computing ARL) where the machine is "forced" to remain in G until the first alarm is hard to justify. Its interpretation is particularly unclear if a policy allows the machine to fail before the first alarm;

b) ARL1 is defined only for the situation where the machine *starts and remains* in condition B – a situation that is hard to imagine being realized in practice.

This shortcoming has been pointed out before by Woodall (1986), Svoboda (1991) andothers.

Composite Measures.

Other performance measures of operational interest can be expressed as simple functions of Type One and Type Two measures. In these definitions, the term *total time* means operating time plus checking time.

$p_S \equiv$ fraction of total time the machine is in B and producing scrap,

$p_B \equiv$ fraction of total time the machine is in B (either producing scrap or being replaced),

$p_G \equiv$ fraction of total time the machine is in G (i.e. producing usable parts),

$r \equiv$ *total alarm rate* $\equiv$ the expected number of alarms of any type per unit time,

$p_0 \equiv$ fraction of total time spent processing alarms of any type

One can combine performance measures to obtain the overall cost per unit time. Computing this cost rate, however, using the simplest linear model, requires the following parameters:

$K_f =$ fixed cost per false alarm

$V_f =$ cost per unit time spent processing a false alarm, including lost production

$K_t =$ fixed cost per true alarm

$V_t =$ cost per unit time spent when processing a true alarm (i.e. when the machine is stopped and found to be in B), including lost production

$V_d =$ cost per unit time of producing scrap while in condition B, including lost production.

The resulting total cost per unit time, c_T, can then be expressed as:

$$c_T = K_f r_f + V_f p_f + K_t r_t + V_t(p_B - r_t\delta) + V_d r_t p_S. \tag{1}$$

Computing c_T requires knowing all these cost coefficients (minimizing c_T requires knowing at least their ratios), which in many cases are difficult (if not impossible) to obtain. For this reason, and because the method we propose to use is can be implemented without availibility of these costcoefficients, we do not directly pursue cost-minimization. Our approach concentrates, instead, on computing *non-cost* performance measures and using them to choose among various checking policies. Moreover, the policy we produce can be readily shown to provide a decision rule that minimizes c_T.

3. System Operating Charactistics

We are ultimately concerned with providing decision makers with checking policies that:

a) are easy to understand and implement;

b) readily allow input of, and sensitivity analysis with respect to, important parameters, including the moments of the machine failure time, and the discriminatory capabilities of various signal observation devices; and

c) do not require an explicit assessment of the hard-to-estimate cost coefficients listed above.

The means by which we present the consequences of using *any* particular monitoring policy is the *System Operating Curve* or *System Operating Characteristic* (SOC) [see Pollock (1964) and Rapoport et al. (1979) for early development]. The SOC is a simple graphical plot involving two axes – one showing a Type One measure, the other a Type Two (or composite) measure. A single point on the SOC represents a pair of performance measures attainable by using a particular checking policy with a machine characterized by specified parameter values. A family of such points represents the range of output measures attainable by changing one or more policy variables available to the decision maker.

For example, a SOC could be a plot of p_S (the fraction of time the machine is producing scrap) versus p_0 (the fraction of time spent checking alarms of any type). Consider a set of

such SOC's, one for each of three different hypothetical monitoring situations A, B and C, as shown in Figure 1. Each curve represents the set of operating points (i.e., values of p_S and p_0) achievable by using different values of a particular policy variable.

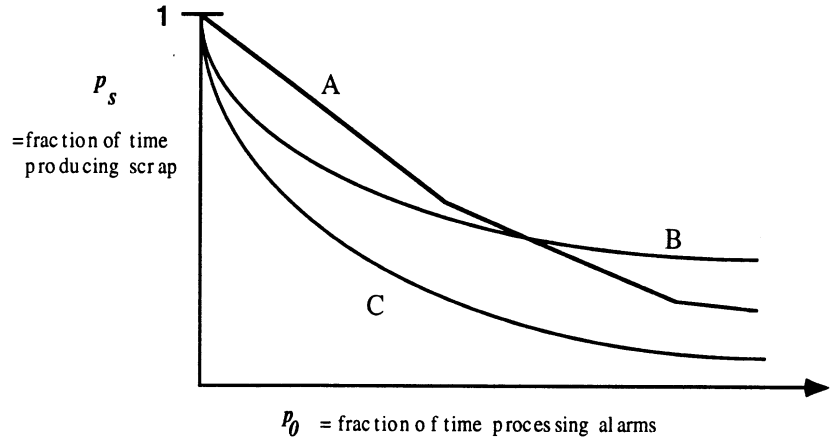

Figure 1: Operating Characteristics (OC) for three hypothetical procedures A, B and C.

Assuming that the costs of implementing A, B and C are all the same, situation C is clearly better than A or B, since it has a lower p_S for any given p_0 or lower p_0 for any given p_S. (Perhaps C allows the monitoring of signals that are not available to either A or B). If only A or B were available, A would be preferred to B when the advantage of achieving early detection (i.e. small p_S) is more important than the disadvantage of having a large fraction of time spent checking alarms.

The SOC is similiar to the *Receiver Operating Characteristic* (ROC) used in telecommunication and signal detection theory, and is also related to the *power curve* of simple hypothesis testing. The word *system* is used to emphasise the fact that it is a combination of the machine, the monitoring device *and* a checking policy that is being represented. Note that equation (1) can be used to directly produce values of c_T for any point on the SOC curve, given appropriate cost coefficients.

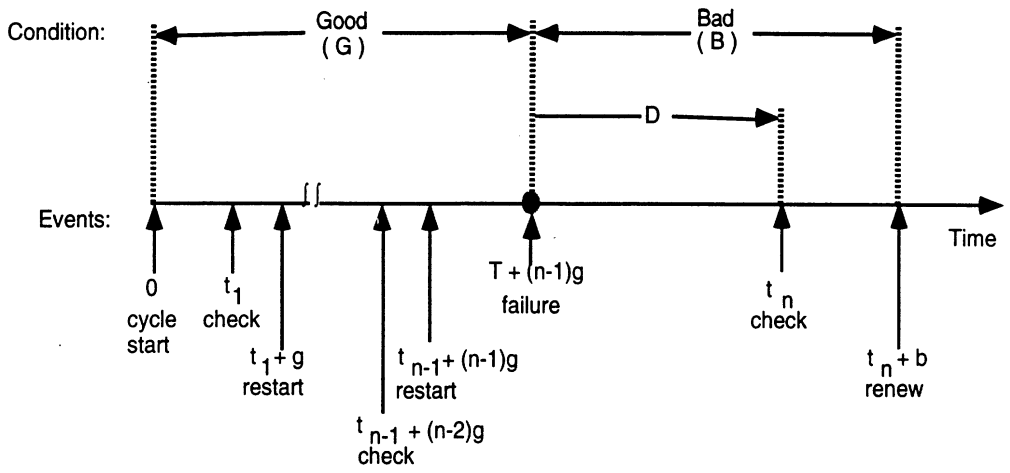

Figure 2: A single cycle with n checks ($n-1$ of which are false alarms). Arrows indicate checking actions and renewals; T = operating time until failure which occurs at time $T+(n-1)g$; D = detection time; t_i = time of i^{th} check (a false alarm for $i < n$); t_n = time condition B is detected; and $t_n + b$ is the time the machine re-enters condition G (which ends the cycle).

4. Relationships Among Performance Measures

We now present general relations among the performance measures that exist for *any* reasonable checking/monitoring procedure. To obtain these, we define the a *cycle* to be the time between renewals. Figure 2 shows a cycle that contains n checks at times $t_1, t_2, \cdots, t_n$. There are $n-1$ checks that find the machine in G (i.e., there are $n-1$ false alarms), each requiring time g. The last check in the cycle finds the machine in B and takes time b. Since the machine is stopped during checks of any type, in any cycle the *operating* time until failure, T, differs from the total time until failure which equals T plus the time spent checking (while in G) prior to failure.

From Figure 2 we see that $\overline{L}$, the expected cycle length when there are n checks (hence $n-1$ false alarms) is

$$\overline{L} = E(T) + g(n-1) + \delta + b. \tag{2}$$

As previously defined, μ is the expected number of false alarms until failure (also called

EFA). An elementary use of the fundamental renewal theorem gives

$$r_f = \frac{\mu}{E(T) + g\mu + \delta + b} \tag{3}$$

and

$$r_t = \frac{1}{E(T) + g\mu + \delta + b}. \tag{4}$$

The ratio of these two equations gives

$$\mu = \frac{r_f}{r_t}. \tag{5}$$

Solving for δ in equation (4) gives

$$\delta = \frac{1}{r_t} - E(T) - g\mu - b = \frac{1}{r_t} - E(T) - g(\frac{r_f}{r_t}) - b. \tag{6}$$

The expected time the machine spends in B is $\delta + b$. Since true alarms occur at a rate r_t,

$$p_B = (\delta + b)r_t, \tag{7}$$

The times to process a false alarm and a true alarm are g and b, respectively. Since these occur at rates r_f and r_t

$$p_f = gr_f. \tag{8}$$

$$p_t = br_t. \tag{9}$$

Finally, by definition,

$$r = r_f + r_t \tag{10}$$

$$p_G + p_B = 1 \tag{11}$$

$$p_S = p_B - p_t \tag{12}$$

$$p_0 = p_t + p_f. \tag{13}$$

Relations (2) through (13) hold for *any* checking procedure. Thus having computed any pair of Type One and Type Two measures (such as r_t and r_f) and knowing $E(T)$, b, and g, allows the calculation of δ, p_B, p_G, p_t and p_f, etc.

For expositional simplicity, in the remainder of this paper we set the lengths of time needed to perform checks to unity, i.e. $b = g = 1$, since computing results for arbitrary b and g values (even when they are zero) is straightforward, as shown in Appendix C.

5. The Basic Monitoring Process

We now present additional assumptions and notation needed to define the monitoring process, determine an optimal checking policy, and to compute performance measures that result from its use. Let

$C_t \equiv$ condition of the machine at time t,

so that

$\Phi(t) \equiv$ the cumulative distribution for the failure time $T = \text{prob.}\{C_t = B | C_0 = G\}$.

$i =$ observation number, $i = 1, 2, ...$

$\tau_i =$ time at which the ith observation is made

Since observations (and therefore checking opportunities) are limited to the times τ_i, it is useful to define the function $f(i)$, the probability that failure occurs between the $(i-1)$st and ith *observation*, so that

$$f(i) \equiv \Phi(\tau_i) - \Phi(\tau_{i-1}), \; i = 1, 2, 3, \ldots, \tag{14}$$

where $\tau_0 \equiv 0$.

The observation made at time τ_i is the random variable X_i, having a (p.d.f.) $f_{X_i}(\cdot)$, depending upon the machine condition as follows:

$$f_{X_i}(x) = \begin{cases} p(x); & \text{if } C_{\tau_i} = G, \; i = 1, 2, \ldots, \\ q(x); & \text{if } C_{\tau_i} = B, \; i = 1, 2, \ldots \end{cases} \tag{15}$$

The random vector $\underline{X}_n$ of observations is defined as as

$$\underline{X}_n \equiv (X_1, X_2, \ldots X_n),$$

and its realization $\underline{x}_n$ is

$$\underline{x}_n \equiv (x_1, x_2 \ldots x_n).$$

Any checking policy can be viewed as being a decision rule to determine whether or not to check the machine at time τ_n given the set of observations $\underline{x}_n$. For example, using the ordinary Shewhart chart (see, e.g. Montgomery [1991]), the decision is based upon only the last observation x_n: if this value falls outside pre-determined control limits then a check (i.e. a "search for an assignable cause") is made. The control limits are policy variables, and varying them will produce an associated SOC.

For Shewhart charts with supplementary runs tests, if K out of the last N observations fall within a pre-specified zone, then a check is made. Here, K and N and the control limits are policy variables that, when varied, will produce the associated SOC. Other policies, such as those using CUSUM or EWMA charts, make use of various functions of some (or all) of the observations $\underline{x}_n$. It is important to note, however, that these policies are in some sense ad-hoc. By contrast, we introduce a class of policies based upon certain optimality conditions.

6. The Probability Threshold Rule (PTR)

A particularly attractive form of checking policy can be based upon a simple proposition: since the observations at times τ_1 through τ_n provide information about machine condition, this information can be used to "update" the probability that $C_{\tau_n} = B$. Specifically, by defining

$$P_n(\underline{x}_n) \equiv \text{prob.}\{C_n = B | C_0 = G, X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n\},$$

we can use this probability as the fundamental element in the definition of the

Probability Threshold Rule (PTR): Check when $P_n(\underline{x}_n)$ first equals or exceeds some *threshold probability* p^*.

If $\phi(t)$ is geometric, it is well known (e.g., see Girshick and Rubin [1952], Shiryaev [1963], or Pollock [1965] for early references, or Pollak [1987] for a more recent one) that the PTR is "optimal" in that it allows one to select the policy variable p^* so as to minimize either:

a) the expected detection time δ for a given value of false alarm rate r_f; or

b) the false alarm rate r_f for a given value of expected detection time δ; or

c) the cost per unit time, c_T, given by equation (1)

From an *operational* point of view this means that given a particular value of p^*, a decision maker can achieve the associated performance level shown as a point on the a SOC. Or, equivalently, from an *analysis* or *strategic* point of view, given the ability to efficiently compute $P_n(\underline{x}_n)$ the decision maker can vary values of p^* to explore trade-offs among various performance measures. For example, if curve C of Figure 1 represents a typical SOC associated with using a PTR policy: large p_S and small p_0 are produced when $p^* \rightarrow 1$, and small p_S and large p_0 are produced when $p^* \rightarrow 0$.

Note that the PTR has only one "free" parameter, p^*.

7. Computing $P_n(\underline{x}_n)$

The computation of $P_n(\underline{x}_n)$ follows from a straightforward use of the definition of conditional probability:

$$\begin{aligned} P_n(\underline{X}_n) &\equiv \text{prob.}\{T \leq \tau_n | \ \underline{X}_n = \underline{x}_n\} \\ &= \frac{\text{prob.}\{T \leq \tau_n \cap \underline{X}_n = \underline{x}_n\}}{\text{prob.}\{\underline{X}_n = \underline{x}_n\}} \end{aligned} \tag{16}$$

where

$$\text{prob.}\{T \leq \tau_n \cap \underline{X}_n = \underline{x}_n\} = \sum_{j=1}^{n} f(j) \Pi_{i=1}^{j-1} p(x_i) \Pi_{k=j}^{n} q(x_k) \tag{17}$$

and

$$\text{prob.}\{\underline{X}_n = \underline{x}_n\} = \text{prob}\{T \leq \tau_n \cap \underline{X}_n = \underline{x}_n\} + \sum_{j=n+1}^{\infty} f(j)\Pi_{i=1}^{n} p(x_i) \tag{18}$$

Substituting (17) and (18) into (16), and dividing numerator and denominator by $\Pi_{i=1}^{n} p(x_i)$, gives

$$P_n(\underline{x}_n) = \frac{\sum_{j=1}^{n} f(j)\Pi_{k=j}^{n} L(x_k)}{\sum_{j=1}^{n} f(j)\Pi_{k=j}^{n} L(x_k) + \overline{F}(n)} \tag{19}$$

where

$$L(x_i) \equiv q(x_i)/p(x_i) \tag{20}$$

is the likelihood ratio for condition B given $X_i = x_i$, and

$$\overline{F}(n) \equiv \sum_{i=n+1}^{\infty} f(i) = \text{prob}\{T > \tau_n\}.$$

Although computing $P_n(\underline{x}_n)$ directly from equation (19) is straightforward, it is advantageous, instead, to use the "odds in favor of condition B" (also called the odds ratio)

$$R_n(\underline{x}_n) \equiv P_n(\underline{x}_n)/(1 - P_n(\underline{x}_n)). \tag{21}$$

(For notational convenience, the argument $\underline{x}_n$ is suppressed for the remainder of this paper, e.g., $R_n(\underline{x}_n)$ is written as R_n.) This odds ratio can be obtained directly from equation (19) as

$$R_n = [\overline{F}(n)]^{-1} \sum_{j=1}^{n} f(j)\Pi_{k=j}^{n} L(x_k). \tag{22}$$

This allows a recursive representation for R_n:

$$R_n = \frac{L(x_n)}{\overline{F}(n)}[\overline{F}(n-1)R_{n-1} + f(n)], \tag{23}$$

which can be confirmed by substitution into equation (22). Equation (23) is an excellent way to compute R_n, and thus $P_n = R_n/(1 + R_n)$, since R_n is calculated from the previously obtained R_n and each new observation x_n by means of a simple addition and multiplication.

Equation (23) clarifies the challenge of computing performance measures associated with the PTR. In particular, consider computing when P_n first equals or exceeds p^* (at which time the PTR produces a check). This is the equivalent of finding the first time that R_n equals or exceeds the "odds threshold"

$$\rho^* \equiv p^*/(1-p^*). \tag{24}$$

When x_{n+1} is replaced by the r.v. X_{n+1}, we see that equation (23) can be viewed as the generator of a Markov Process R_n. This process has as a state space the non-negative real line $\mathbb{R}^+$, with transitions at observation times τ_n governed by the stochastic behavior of X_n, which in turn are governed by the p.d.f.s of equation (15). When an observation of $X_{n+1} = x_{n+1}$ is made, either

(a) R_{n+1} is less than the threshold ρ^*, and the process continues; or

(b) R_{n+1} equals or exceeds ρ^* and a check is made.

8. Performance Measures for Two Limiting Cases:

In this section we "bracket" performance of the PTR for any realizable monitoring system by computing performance measures for two limiting worst-case and best-case extremes of monitoring.

No Observations: As a "worst case" bound on the SOC, we can consider the limiting situation where observations provide *no* information, which is equivalent to having $p(x_i) = q(x_i)$, so that $L(x_i) = 1$, for $i = 1, 2 \ldots$. In this case, equation (23) reduces to

$$R_n = [\overline{F}(n)]^{-1}[\overline{F}(n-1)R_{n-1} + f(n)], \tag{25}$$

which is a deterministic difference equation with solution (given boundary condition $R_0 = 0$):

$$R_n = \frac{F(n)}{\overline{F}(n)}. \tag{26}$$

From the definition of R_n in equation (21), this gives

$$P_n = F(n), \tag{27}$$

a result that holds for any $F(n)$. Clearly, observations of $\underline{x}_n$ have no effect on P_n, which is simply the cumulative distribution for the failure time T evaluated at $T = \tau_n$.

Two performance measures that can be readily calculated are μ and δ. By definition of the PTR, the (deterministic) time to first check, t_1, is the smallest monitoring time such that the threshold probability is exceeded, i.e.

$$t_1 = \min_{t \in \{\tau_i\}} \{t : \Phi(t) \geq p^*\}. \tag{28}$$

Similarly, the time of the jth check, t_j, can be shown to be

$$t_j = \min_{t \in \{\tau_i\}} \{t : \frac{\Phi(t) - \Phi(t_{j-1})}{1 - \Phi(t_{j-1})} \geq p^*; t > \tau_{j-1},\} \tag{29}$$

where $t_0 = 0$.

The computation of μ and δ becomes straightforward if we assume that the inequalities of expressions (28) and (29) are satisfied as equalities. This situation holds if the checking times t_j are not constrained to be in the set $\{\tau_j\}$ – a reasonable assumption if the monitoring provides no information. In this case, the t_j are easily shown to be given by

$$\Phi(t_j) = 1 - (1 - p^*)^j. \tag{30}$$

The probability that the failure time T is between the $(j-1)$st and the jth check, i.e. that $\{t_{j-1} \leq T \leq t_j\}$, is $\Phi(t_j) - \Phi(t_{j-1})$. Since all previous checks will have produced false alarms, we have

$$\mu = \sum_{j=0}^{\infty} (j-1)[\Phi(t_j) - \Phi(t_{j-1})] = 1/p^* - 1 \tag{31}$$

The time late, given $t_{j-1} \leq T \leq t_j$ is $t_j - T$, so by simple decomposition of expectations

$$\delta = \sum_{j=1}^{\infty} \int_{t=t_{j-1}}^{t_j} (t_j - t)\phi(t)dt \; = E(T) - p^* \sum_{i=1}^{\infty} (1-p^*)^{i-1} \Phi^{-1}[1 - (1-p^*)^i]. \tag{32}$$

This result is equivalent to similar ones contained in the literature, and serves as the basis for most cost-minimizing maintenance policies , for example those in Barlow and Proschan [1965].

Perfect Monitoring. The limiting case of "perfect" information represents another special situation. In this case the supports of $p(x_i)$ and $q(x_i)$ are disjoint: $L(x_i) = 0$ for all i such that $\tau_i < T$; and $L(x_i) = \infty$ for all i such that $\tau_i \geq T$. From equation (19), we see that *any* non-zero threshold is exceeded for all n such that $\tau_n \geq T$. In this case

$$\delta = E_T[\min_{t \in \{\tau_i\}} (t - T : t \geq T)], \tag{33}$$

and the machine is checked only when it fails. From this, it is trivial to show that the true alarm rate is $r_t = 1/(E(T) + b)$, the false alarm rate is $r_f = 0$, the fraction of time spent in processing true alarms is $p_t = b/(E(T) + b)$ and the fraction of time processing false alarms is $p_f = 0$.

9. Behavior of R_n for Geometric Failure Time Using the PTR

Unfortunately, computation of performance measures for the PTR is extremely difficult for general failure time distributions $\Phi(t)$. (For a recent example dealing with a uniform distribution, see Wang [1995]). However, suppose the failure time T has the geometric p.m.f. with expected value $E(T) = 1/h$,

$$\phi(t) = h(1-h)^{t-1}, \; t = 1, 2, 3, \ldots, \tag{34}$$

so that the cumulative distribution is

$$\Phi(t) = 1 - (1-h)^t, \; t = 1, 2, 3 \ldots. \tag{35}$$

Furthermore, assume that observations are made at equal intervals, i.e. at times $\tau_1 = s, \tau_2 = 2s, \tau_3 = 3s, \ldots$. (This assumption is often consistent with machine monitoring practice. However, the determination of a "good", much less"optimal" interval – or, if not at equal

intervals, the set of planned observation times that should be used – is still very much an open research question.) The associated probability that a failure will occur between observation i and $(i-1)$ is readily shown to be

$$f(i) = a(1-a)^{i-1},\ i = 1, 2, 3, \ldots, \tag{36}$$

with cumulative distribution

$$F(i) = 1 - (1-a)^{i},\ i = 1, 2, 3 \ldots, \tag{37}$$

where

$$a \equiv 1 - (1-h)^{s}. \tag{38}$$

Using this distribution, equation (23) reduces to:

$$R_n = \ell(X_n)[R_{n-1} + a], \tag{39}$$

where

$$\ell(X) \equiv L(X)/(1-a) \tag{40}$$

can be interpreted to be a "normalized" likelihood ratio.

In this case the conditions for checking and continuing, respectively, using equation (39), are:

a) if $\ell(x_{n+1}) < \rho^*/(R_n + a)$, continue; and

b) if $\ell(x_{n+1}) \geq \rho^*/(R_n + a)$, check.

The absorption behavior of this process, and in particular the distribution of the (random variable) time until R_N first equals or exceeds the odds threshold ρ^*, has a long and important history of study (see Shiryaev [1978]), resulting in computational (as contrasted to structural) methods for limiting cases or approximations (e.g. Pollak [1985]). In the sections that

follow, we present a Markov Chain approximation that extends these results, and provides an efficient method for computing performance measures of interest.

10. Markov Process Representation for Geometric Failure Time

All performance measures of interest can be obtained by computing the steady state probabilities of a mixed continuous-discrete state Markov Process MPR, created by combining R_n with the machine condition C_n. MPR is defined such that:

a) Transitions occur immediately after observations and any associated checking;

b) The state at the end of the n^{th} transition is denoted as $S_n \in \mathcal{S},\ n = 1, 2, \ldots$;

c) The state space $\mathcal{S}$ is the union of five sub-spaces: three each containing a single state and two each containing elements that are mixed continuous-discrete states.

These sub-spaces of $\mathcal{S}$ are:

$\mathcal{S}_G \equiv \{(R, G) : R \in (0, \rho^*)\}$, set of states where R_n is between 0 and ρ^*, and $C_n = G$;

$\mathcal{S}_B \equiv \{(R, B) : R \in (0, \rho^*)\}$, set of states where R_n is between 0 and ρ^*, and $C_n = B$;

$\mathcal{S}_0 \equiv 0$, renewal state: the state entered after the machine is renewed, i.e., when $R_n = 0$ (or, equivalently, $P_n = 0$) and $C_n = G$;

$\mathcal{S}_G^* \equiv \{(\rho^*, G)\}$, false alarm state: the state entered after checking while the machine is in G, i.e., when $R_n \geq \rho^*$ and $C_n = G$;

$\mathcal{S}_B^* \equiv \{(\rho^*, B)\}$, true alarm state: the state entered after checking while the machine is in B, i.e., when $R_n \geq \rho^*$ and $C_n = B$.

Given the distribution of the random variable X_{n+1} shown by equation (15) and the geometric failure time distribution of equation (36), the transition probabilities among the states in these sets are governed by the evolution of R_n described by equation (39). Having a geometric distribution of failure time is the equivalent of having the probability of the machine condition going from G to B at each transition be the constant a (except from $\mathcal{S}_G^*$ where this probability is zero); this is the key to establishing the Markovian properties of MPR.

We note some of the properties of the Markov Process MPR:

a) it is ergodic, since there is a single closed communicating class of states;

b) the probability of transition from $\mathcal{S}_G^*$ or $\mathcal{S}_B^*$ to $\mathcal{S}_0$ is 1, reflecting the one transition (since $b = g = 1$) needed to check after a false alarm or a true alarm;

c) due to the geometric failure time distribution of equation (35), the single-step transition probability is a for transitions:

   i) from the set $\mathcal{S}_G$ to the set $\mathcal{S}_B \cup \mathcal{S}_B^*$;

   ii) from the state $\mathcal{S}_0$ to the set $\mathcal{S}_B \cup \mathcal{S}_B^*$.

The steady-state probabilities for the singleton states are defined as

$$\begin{aligned}
\pi_0 &\equiv \lim_{n\to\infty} \text{prob}\{S_n = \mathcal{S}_0\}, \\
\pi_G^* &\equiv \lim_{n\to\infty} \text{prob}\{S_n = \mathcal{S}_G^*\}, \\
\pi_B^* &\equiv \lim_{n\to\infty} \text{prob}\{S_n = \mathcal{S}_B^*\}.
\end{aligned}$$

In addition, we define the steady state cumulative distribution functions for the sets $\mathcal{S}_G$ and $\mathcal{S}_B$ as

$$\begin{aligned}
\Pi_G(r) &= \lim_{n\to\infty} \text{prob}\{S_n \in [(t, G) : t \le r]\}, \quad 0 < r < \rho^*, \\
\Pi_B(r) &= \lim_{n\to\infty} \text{prob}\{S_n \in [(t, B) : t \le r]\}, \quad 0 < r < \rho^*,
\end{aligned}$$

and define

$\Pi_G(0) = \Pi_B(0) = 0,$

$\Pi_G(\rho^*) = \lim_{\epsilon \to 0} \Pi_G(\rho^* - \epsilon)$, and

$\Pi_B(\rho^*) = \lim_{\epsilon \to 0} \Pi_B(\rho^* - \epsilon)$.

Performance measures are easily obtained from these steady state probabilities and distributions. In particular, by appealing to the ergodic theorem for Markov Processes, we know that

π_0 = fraction of time the process is in the renewed state.

Thus, $p_r = p_t + p_f = \pi_0$, since MPR is in $\mathcal{S}_0$ for exactly one time unit per cycle.

Similiarly,

$p_f = \pi_G^*$ = expected fraction of time the process is in the false alarm state,

$p_t = \pi_B^*$ = expected fraction of time the process is in the true alarm state.

Given these values for p_f and p_t, the analysis in Section 3 and Appendix C allows computation of all other measures of interest.

The equations needed to calculate π_B^* and π_G^* are given in Appendix A. In particular, equations (A4) and (A5) are special cases of the Fredholm equation of the second kind, which has a long history of theoretical study (e.g., Groetsch (1984) or Brunner (1982)) and numerical means of solution (Schippers (1983)). Indeed these equations have an analogue to those developed by Pollak [1987] to compute an ARL0-type measure. However, as Pollak notes, a solution method is lacking for even the simplest forms of monitoring distributions $p(x)$ and $q(x)$. The computational literature is problem-specific and essentially suggests using transformation of variables and discretization approximations tailored to the problem at hand.

11. Markov Chain Approximation

We now present an approximation useful for finding performance measures for specific cases of $p(x)$ and $q(x)$, by showing that the process MPR, which has *continuous* elements in its state space, can be approximated by a Markov *Chain* ("MCR") with *discrete* state space.

To construct MCR, we identify values of the odds ratio R_n that lie in the interval $(0, \rho^*)$, but are restricted to the finite set

$$\mathcal{S}^1 \equiv \{r_1, r_2, \ldots r_{m-1}\}.$$

The key to this restriction is to find a set of r_i values that "cover" the interval $(0, \rho^*)$ in such a way that the sums over probabilities in $\mathcal{S}^1$ well approximate the integrals over S implicit in equations (A2) to (A5). (Obtaining such a set of r-values is shown, for example, in section 12 for Bernoulli monitoring; at this point we will assume that $\mathcal{S}^1$ is available.)

Given the finite elements of $\mathcal{S}^1$, we define a finite state space $\mathcal{R}$ for MCR

$$\mathcal{R} = \mathcal{S}_0 \cup \mathcal{S}_G^* \cup \mathcal{S}_B^* \cup \mathcal{R}_G \cup \mathcal{R}_B.$$

The first three sub-spaces are the singletons previously defined in Section 10 (corresponding to renewal, false alarm and true alarm states, respectively), and the last two are

$$\begin{aligned} \mathcal{R}_G &\equiv \{r_i : r_i \in \mathcal{S}^1,\ C_n = G,\ i = 1, 2, \ldots m-1,\ n = 1, 2, \ldots\} \\ \mathcal{R}_B &\equiv \{r_i : r_i \in \mathcal{S}^1,\ C_n = B,\ i = 1, 2, \ldots m-1,\ n = 1, 2, \ldots\}. \end{aligned}$$

Thus $\mathcal{R}_G$ represents a discrete subset of R_n values while the machine is in G, and $\mathcal{R}_B$ represents a discrete subset of R_n values when the machine is in B.

A simple arbitrary numbering of states allows us to represent MCR as a $(2m+1)$-state ergodic Markov Chain, which we will refer to as "MC," with

$\sigma_n \equiv$ the state of MC after the n^{th} transition;

state space $I_{2m+1} \equiv \{0, 1, 2, \ldots, 2m\}$; and

transition matrix P with elements $[P]_{ij} = p_{ij} \equiv \text{prob}\{\sigma_n = j | \sigma_{n-1} = i\}$ for $i, j \in I_{2m+1}, n = 1, 2, \ldots;$

Details of the relation between MCR and MC finding the values of p_{ij} are contained in Appendix B. Given P and defining

$$\pi_i \equiv \lim_{n \to \infty} \text{prob}\{\sigma_n = i\}, \; i = 0, 1, \ldots 2m,$$

the steady-state probability vector $\pi \equiv \{\pi_0, \pi_1, \pi_2, \ldots, \pi_{2m}\}$ can be obtained by solving the set of linear equations:

$$\pi = \pi P \tag{41}$$

$$\pi = \pi \underline{1} \tag{42}$$

where $\underline{1} \equiv \{1\ 1\ 1\ \ldots 1\}^t$ is the transpose of the unit $(2m+1)$-vector.

Performance measures can be immediately obtained from these equations since π_0 is immediately given, and $\pi^*_G = \pi_m$ and $\pi^*_B = \pi_{2m}$. The next sections show specific results for two examples: monitoring Bernoulli and Normal observations.

12. Markov Chain MCR for Bernoulli Observations and Geometric Failure Time

The "discretization" of MPR to MCR depends in general on the monitoring distributions $p(x)$ and $q(x)$. This section deals with Bernoulli observations, by which we mean that the observations $X_n \in \{0, 1\}, n = 1, 2, \ldots,$ and

$$p(x) = \begin{cases} 1 - \alpha & \text{if } x = 0, \\ \alpha & \text{if } x = 1, \end{cases}$$

$$q(x) = \begin{cases} \beta & \text{if } x = 0, \\ 1 - \beta & \text{if } x = 1. \end{cases}$$

This situation is a form of classical hypothesis testing: $x = 0$ is "evidence" of condition G (e.g. no defect in an observed manufactured product) and $x = 1$ is evidence of condition B

(e.g. a defect is observed). Thus α is analogous to an "error of the first kind," and β to an "error of the second kind."

The likelihood ratio is

$$L(x) = \begin{cases} \beta/(1-\alpha) & \text{if } x = 0, \\ (1-\beta)/\alpha & \text{if } x = 1. \end{cases}$$

By defining

$$\begin{aligned} w_0 &\equiv \frac{\beta}{(1-\alpha)(1-a)}, \\ w_1 &\equiv \frac{1-\beta}{\alpha(1-a)}, \end{aligned}$$

equation (39) can be written

$$R_n = \begin{cases} w_0(R_{n-1}+a) & \text{if } X_n = 0, \\ w_1(R_{n-1}+a) & \text{if } X_n = 1. \end{cases} \tag{43}$$

For a realistic problem,

a) the value of a (the probability that the machine goes from G to B on any transition) is usually quite small;

b) α and β, the "misclassification" errors in a single observation of x, are both less than .5, so that $\alpha + \beta < 1$.

In order to identify a useful set $\mathcal{S}^1$ we can consider the evolution of the R_n process of equation (43) when the process starts with $R_n = 0$ (i.e., $P_0 = 0$). The possible values of R_n that can be generated after the first three observations, assuming none exceed ρ^*, are given in Table 1:

Observation Number n	Possible R_n Values
1	w_0
	w_1
2	$(a+w_0)w_0$
	$(a+w_0)w_1$
	$(a+w_1)w_0$
	$(a+w_1)w_1$
3	$(a+(a+w_0)w_0)w_0$
	$(a+(a+w_0)w_0)w_1$
	$(a+(a+w_0)w_1)w_0$
	$(a+(a+w_0)w_1)w_1$
	$(a+(a+w_1)w_0)w_0$
	$(a+(a+w_1)w_0)w_1$
	$(a+(a+w_1)w_1)w_0$
	$(a+(a+w_1)w_1)w_1$

Table 1: Possible values of R_n after $n = 1, 2, 3$ observations.

After h observations the maximum number of possible distinct values of R_n that could be generated is clearly $2^{h+1} - 2$. However, an important effect reduces this number, which allows a practical means of discretizing MCR to MC: as h becomes large, some values of R_h satisfy $R_h \geq \rho^*$, in which case either state $\mathcal{S}_G^*$ or $\mathcal{S}_B^*$ occurs and R_{h+1} becomes 0.

Thus we can generate $\mathcal{S}^1 \equiv \{r_1, r_2, \ldots, r_{m-1}\}$ by simply computing all possible R_n-values achievable over a "horizon" of h observations of $X_n, n = 1, 2, \ldots, h$. The advantages of this approach are:

a) only feasible values of r_i are generated,

b) it actually represents the R_n process for $n = 1, 2, \ldots, h$ over the horizon h, and

c) the accuracy of the discrete approximation can be improved by increasing h.

An algorithm for generating m and $\mathcal{S}^1$ is:

1. Set $S_0 = \{0, \rho^*\}$, a horizon $h \geq 1$ and $n = 1$.

2. Generate the set $s_0 = \{w_0(r_i + a)$ for all $r_i \in S_{n-1}$ such that $w_0(r_i + a) < \rho^*\}$.

3. Generate the set $s_1 = \{w_1(r_i + a)$ for all $r_i \in S_{n-1}$ such that $w_1(r_i + a) < \rho^*\}$.

4. Set $S_n = \{r : \ r \in s_0 \cup s_1$ and $r \notin \cup_{i=0}^{n-1} S_i\}$.

5. Increment n by 1.

6. If $n < h$ then go to step 2.

7. Set $\mathcal{S}^1 = \cup_{i=1}^{n} S_i$, $m = |\mathcal{S}^1| + 1$ and stop.

At termination $\mathcal{S}^1$ will be a set of $m - 1$ distinct r-values which, when sorted by increasing value, can be labeled $r_1, r_2, \ldots, r_{m-1}$. Computational experience (see section (13)) has shown that even though m increases nearly exponentially in h, the accuracy of the discrete approximation, for a wide range of parameters, becomes excellent for $h \leq 10$.

The key to converting MCR into MC is identifying, for any possible realized value of R_n that is not an element of $\mathcal{S}^1$, the "closest" element to it. Thus the entire evolution of the process, and in particular values of R_n for $n \geq h$, can approximately contained within the states $\{0, \ \mathcal{S}_G^*, \ \mathcal{S}_B^*, \ \mathcal{S}^1 \times G$ and $\mathcal{S}^1 \times B\}$.

A formal procedure for doing this is to define $W_0^* = \{0 \cup \mathcal{S}^1 \cup \rho^*\}$ with 0 as its 0^{th} element (i.e., $r_0 = 0$) and ρ^* as its m^{th} element (i.e., $r_m = \rho^*$), and define the indices

$$\begin{aligned} J_0(i) &= \text{the index of the closest element in } W_0^* \text{ to } R_{n+1} \text{ given } R_n = r_i \text{ and } x_{n+1} = 0 \\ &= \arg \min_{k \in 0,1,\ldots m} \{|w_0(r_i + a) - r_k|\} \end{aligned}$$

and

$$\begin{aligned} J_1(i) &= \text{the index of the closest element in } W_0^* \text{ to } R_{n+1} \text{ given } R_n = r_i \text{ and } x_{n+1} = 1 \\ &= \arg \min_{k \in 0,1,\ldots m} \{|w_1(r_i + a) - r_k|\}. \end{aligned}$$

The associated probability transition matrix P can be created (see Appendix B) from the two probability transition submatrices P^G and P^B corresponding to state transitions placing the machine in G and B, respectively, as shown in Figure 5.

The composition of these submatrices is:

$$\left[P^G\right]_{ij} = \begin{cases} 1-\alpha & \text{if } j = J_0(i),\ i \in \{0,1,2,\ldots,m-1\}, \\ \alpha & \text{if } j = J_1(i),\ i \in \{0,1,2,\ldots,m-1\}, \\ 0 & \text{other } i \in \{0,1,\ldots,m-1\},\ j \in \{0,1,\ldots,m\}, \end{cases} \tag{44}$$

$$\left[P^B\right]_{ij} = \begin{cases} \beta & \text{if } j = J_0(i)+1,\ i \in \{0,1,\ldots,m-1\}, \\ 1-\beta & \text{if } j = J_1(i)+1,\ i \in \{0,1,\ldots,m-1\}, \\ 0 & \text{other } i \in \{0,1,\ldots,m-1\},\ j \in \{1,2,\ldots,m\} \end{cases} \tag{45}$$

(46)

13. Numerical Results for Bernoulli Monitoring

To obtain numerical results for Bernoulli monitoring we first generate the set $\mathcal{S}^1$ using the algorithm of the preceding section. Figures 6a, 6b, and 6c show how $m = |\mathcal{S}^1|$ increases with increasing horizon h for various values of α, β, a, and ρ^*. These figures show that the number of states becomes nearly exponential in h only for large ρ^*, small a, and large α and β, conditions that are not likely to be attained in practice.

We next compute, using equations (41) and (42), the steady-state probability vector π. Figures 7a and 7b show $pi_i, i = 0, 1, ...m$, plotted against the values of r_i that were generated by the algorithm, for $\alpha = \beta = 0.2$, $a = 0.01$, and $\rho^* = 0.2$.

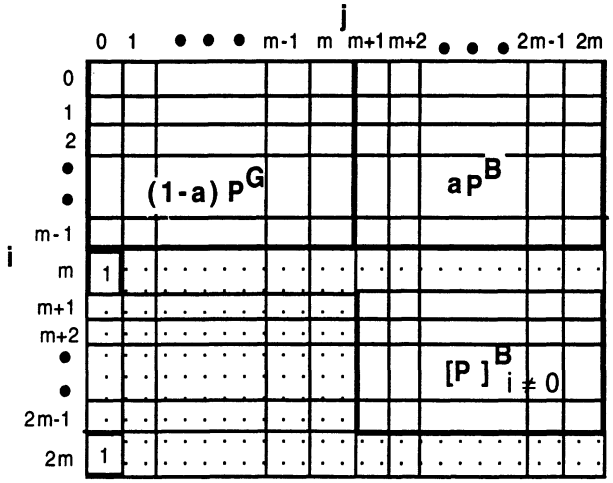

Figure 5: Schematic representation of the transition matrix P from state i to state j showing the use of submatrices P^G and P^B. $[P^B]_{i\neq 0}$ denotes the submatrix created by removing the row associated with $i = 0$ from P^B. Shaded regions represent zero transition probabilities.

Note the "jumpy" nature of the steady-state probabilities and the "gaps" between the values of r_i. This behavior suggests that

a) any solution approach which uses a continuous approximation to the state space would be unwieldy, and

b) using a simple evenly spaced "grid" over the r axis to represent the possible r_i values

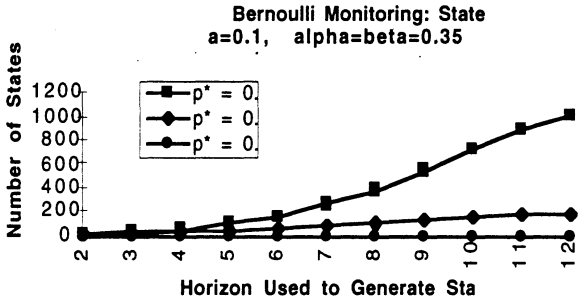

Figure 6a: The number of states generated increases as the horizon used to generate the r-state space increases.

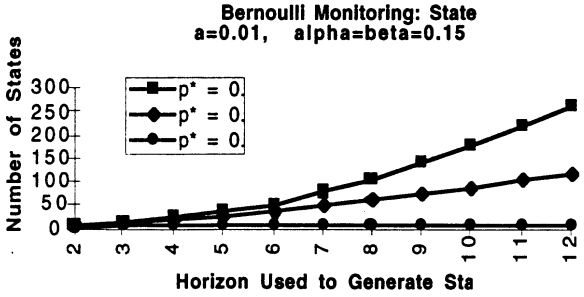

Figure 6b: Increasing mean time between failures by decreasing a from 0.1 (as in Figure 6a) to 0.01 (above) increases the number of generated states.

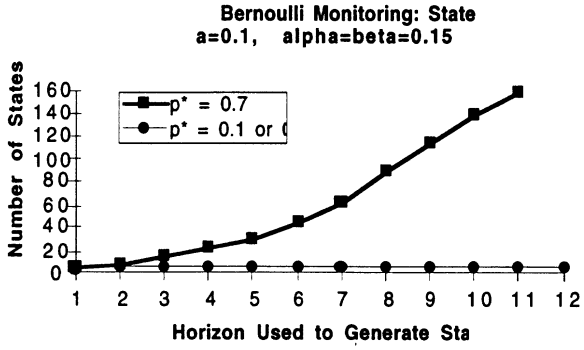

Figure 6c: Decreasing observation information content by increasing α and β from 0.15 (as in Figure 6a) to 0.35 (above) increases the number of generated states.

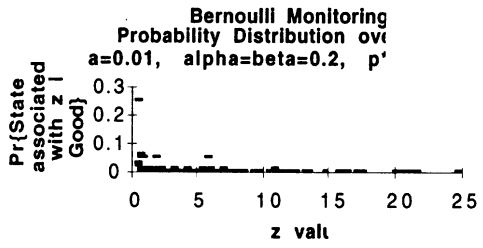

Figure 7a: The probability mass function $[\pi_g]_i$ associated with each r_i value generated by the solution proceedure when $\alpha = \beta = 0.2$, $a = 0.01$, $\rho^* = 0.2$, and a horizon of $h = 8$ is used to generate the state space.

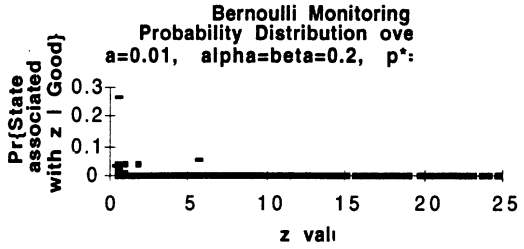

Figure 7b: Increasing the horizon h from 8 (as in Figure 7a) to 12 (above) introduces some additional r_i values, but has a minor effect on the probabilities associated with the old r_i values.

would be inefficient due to the resulting inclusion of highly unlikely (or even impossible) values of r_i.

Figure 7b shows that increasing the horizon h from 8 to 12 (which increases the number of generated states, and thus the computational efort) has little effect on the probabilities associated with the r_i values.

Figure 8 illustrates how computational accuracy (represented by the computed value of π_0) generally stabilizes, and for a wide range of parameter settings produces a maximum absolute percent error of less than 5 percent, for $h \geq 7$. For this reason, we use $h = 7$ for the remaining computational results.

The resulting performance measures can be shown by means of a SOC. In particular, Figures 9a, 9b and 9c each show curves that plot $p_S = \Pr\{\text{producing scrap}\}$ versus $\pi_0 = \Pr\{\text{down for checking}\}$ for selected parameter values.

Figure 9a shows that for a fairly non-informative sensor ($\alpha = \beta = 0.3$), varying ρ^* from .01 to .1 produces a wide range of operating points (i.e. possible values of p_S and π_0). Figure 9b shows the effect of increasing sensor sensitivity to $\alpha = \beta = 0.3$. Figure 9c shows that with an even more sensitive sensor ($\alpha = \beta = 0.1$), only a few operating points are possible for ρ^* between 0.01 and 0.5. Indeed, for $a = 0.1$ there is only one feasible operating point ($p_S = 0.00747$, $\pi_0 = 0.146$) in this range of ρ^*.

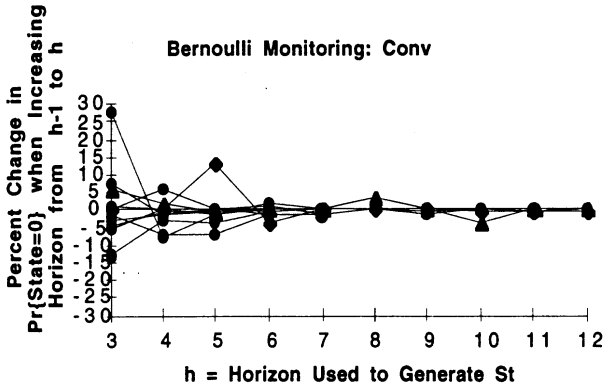

Figure 8: Percent change in π_0 as the horizon used to generate the r-state space increases from $h-1$ to h. The figure displays 18 curves corresponding to all possible parameter settings such that $\alpha = \beta \in \{0.15, 0.25, 0.35\}$, $a \in \{0.01, 0.1\}$, and $\rho^* \in \{0.1, 0.4, 0.7\}$. Six of the parameter settings had zero percent change for all h shown. The two worst cases are plotted using diamonds ($\alpha = \beta = 0.35, a = 0.01, \rho^* = 0.4$) and triangles ($\alpha = \beta = 0.35, a = 0.01, \rho^* = 0.1$).

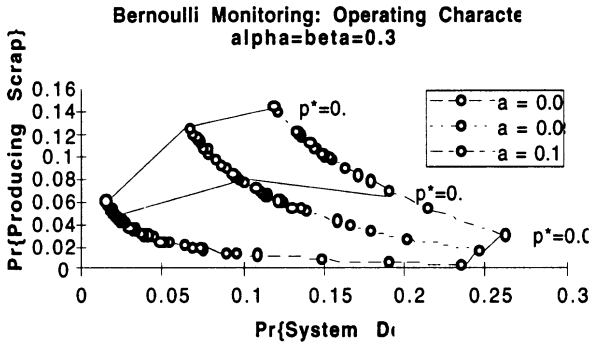

Figure 9a: System Operating Characteristics under Bernoulli monitoring for Pr{Producing Scrap} versus Pr{System Down} generated by varying ρ^* from 0.01 to 0.5 in steps of 0.01. There are three curves, one for each $a \in \{0.01, 0.05, 0.1\}$ with $\alpha = \beta = 0.3$ in each case. Any apparent non-convexity of these curves is due to round off errors in π_G^* and π_S^* which are used to calculate Pr{producing scrap} and Pr{system down}. Points associated with the same value of ρ^* are connected by solid lines for $\rho^* = 0.01$, 0.3 and 0.5.

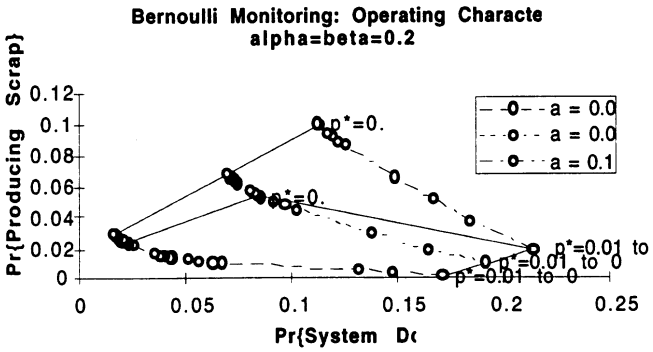

Figure 9b: System Operating Characteristics of Figure 9a but with $\alpha = \beta$ decreased from 0.3 to 0.2. Note that increasing the failure rate tends to collapse ranges of smaller ρ^* into one operating point, for example $\rho^* \in (0.01, 0.3)$ produces a wide range of operating points when $a = 0.01$, but produces only *one* operating point when $a = 0.1$. Comparing this figure with Figures 9a and 9c shows this collapse is more pronounced with smaller α and β.

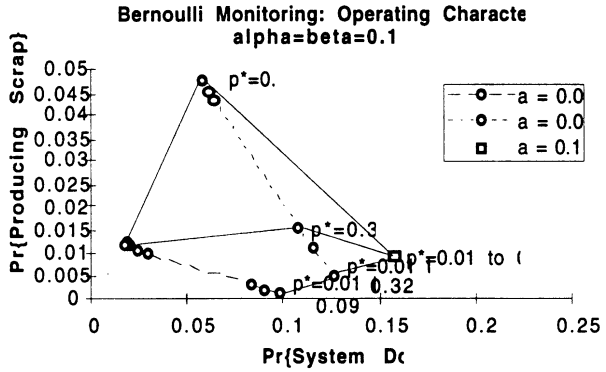

Figure 9c: System Operating Characteristics of Figure 9a with $\alpha = \beta$ reduced from 0.3 to 0.1. Note the extreme collapse of operating points to just one for all $\rho^* \in (0.01, 0.5)$ associated with a high failure rate ($a = 0.1$) and informative sensors.

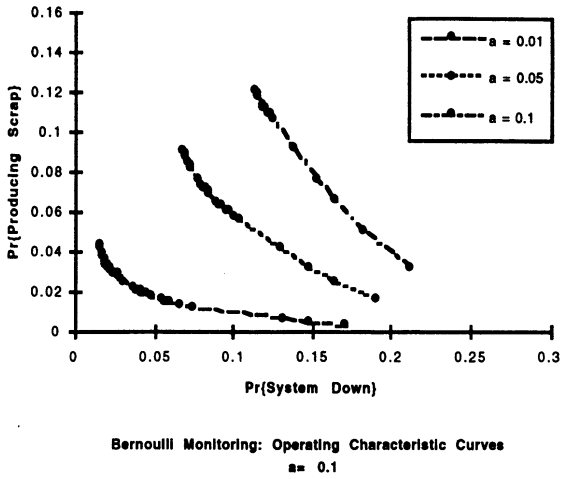

Figure 10: System Operating Characteristics with an asymmetric sensor: $\alpha = 0.3$, $\beta = 0.2$. Decreasing β from 0.3 (as in Figure 9a) to 0.2 (above) improves both performance measures.

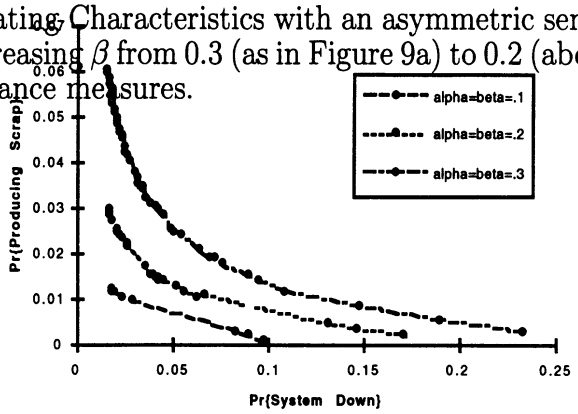

Figure 11: Decreasing α and β of a symetric sensor from 0.3 to 0.1 dramatically improves the System Operating Characteristic for scrap production versus system down time.

Figure 10 shows a SOC for an "asymmetric" sensor with $\alpha = 0.3$ and $\beta = 0.2$. Improving the sensor error probability $\beta = \Pr\{x = 0|G\}$ from $\beta = 0.3$ (shown in Figure 9a) to .2 improves both measures plotted on the SOC: decreasing β reduces the detection time as well as the probability the system is down: a lower β gives greater confidence that observing $x = 0$ implies condition G. This reduces the upward drift of the R_n process for any given set of "zero" observations, which delays the time to a false alarm.

Figure 11 compares different sensors when $a = 0.1$, dramatically showing the advantage of having a more sensitive sensor.

Figure 12 shows an interesting alternative form of an SOC. The two attributes are p_S and $\Pr\{\text{Producing good product}\} = \pi_0 + p_G$. The latter measure is important since checking

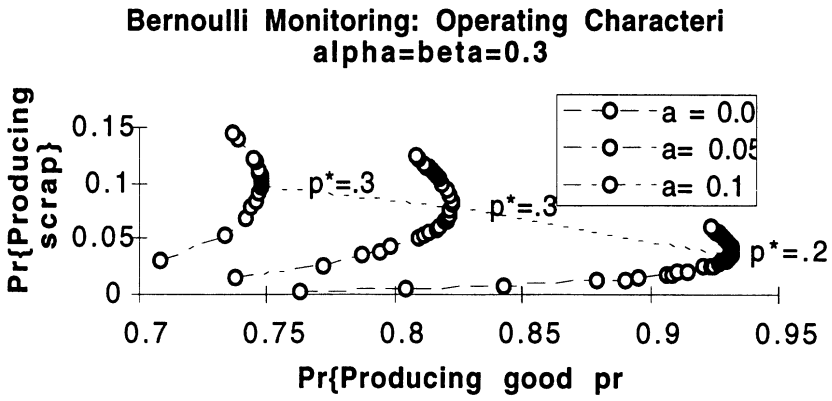

Figure 12: Operating Characteristic Curve for scrap production versus good production. "Better" is towards the point (1,0): no scrapping and always producing good product (no down time). Note the non-optimal operating points above the dotted line: when $a = 0.01$, for example, for each ρ^* above 0.26 there exists a ρ^* below 0.26 with the same throughput of good product *and* a lower scrap rate. Similar "optimality threshold boundaries" for $a = 0.05$ and $a = 0.1$ are near 0.32 and 0.36, respectively.

time is taken from production capacity even though less scrap is produced. This SOC shows operating points (above the dotted line) that will *never* be optimal with respecttothese measures. For example when $a = 0.01$, there exits for each value of ρ^* above 0.26 another ρ^* below 0.26 that produces the *same* throughput of good product with a *lower* scrap rate.

14. Elements of $\tilde{P}$ for Normal monitoring.

For Normal monitoring the observations X_n are independent normally distributed random variables depending only on machine condition. In particular,we asume the distibutions of X_n in equation(15) aregiven by

$$p(x) = \frac{1}{\sqrt{2\pi}} e^{-x^2/2}$$

$$q(x) = \frac{1}{\sqrt{2\pi}} e^{-(x-\mu)^2/2}.$$

The likelihood ratio is thus

$$L(x) = q(x)/p(x) = e^{\mu x - \mu^2/2}$$

and equation (22) becomes

$$R_{n+1} = \gamma e^{\mu X_n}[a + R_n], \tag{47}$$

where

$$\gamma \equiv \frac{e^{-\mu^2/2}}{1-a}. \tag{48}$$

Again, we first assume a set of r-values $\{r_0 \equiv 0, r_1, r_2, \ldots, r_{m-1}, r_m \equiv \rho^*\}$ is given. Using the governing equation (refeqRnnorm) given $r_n = y$, the conditional distibution of R_{n+1} is:

$$\begin{aligned} \text{prob.}\{R_{n+1} \le r | r_n = y\} &= \text{prob.}\{\gamma e^{\mu X_n}(a+y) \le r\} \\ &= \text{prob.}\{X_n \le \tfrac{1}{\mu} \ln \frac{r}{\gamma(a+y)}\}. \end{aligned}$$

When the machine in condition is G, X_n has pdf $p(x)$. Thus the elements of P^G are given by:

$$\begin{aligned}
[P^G]_{ij} &= \text{prob.}\{R_{n+1} = R_j \mid R_n = r_i\} \\
&= \text{prob.}\{R_{n+1} \le r_j \mid r_n = r_i\} - \text{prob.}\{R_{n+1} \le r_{j-1} \mid R_n = r_i\} \\
&= \Phi\left(\tfrac{1}{\mu} \ln \frac{r_j}{\gamma(a+r_i)}\right) - \Phi\left(\tfrac{1}{\mu} \ln \frac{r_{j-1}}{\gamma(a+r_i)}\right) \qquad i,j = 1,2,\ldots,m-1. \\
[P^G]_{mj} &= \begin{cases} 0 & \text{j=1,2,...,m.} \\ 1 & \text{j=0} \end{cases} \\
[P^G]_{im} &= \text{prob.}\{R_{n+1} \ge \rho^* \mid R_n = r_i\} = 1 - \Phi\left(\tfrac{1}{\mu} \ln \frac{\rho^*}{\gamma(a+r_i)}\right) \qquad i = 1,2,\ldots,m-1 \\
[P^G]_{0j} &= \Phi\left(\tfrac{1}{\mu} \ln \frac{r_j}{a\gamma}\right) - \Phi\left(\tfrac{1}{\mu} \ln \frac{r_{j-1}}{a\gamma}\right) \qquad j = 1,2,\ldots,m-1 \\
[P^G]_{0m} &= 1 - \Phi\left(\tfrac{1}{\mu} \ln \frac{\rho^*}{a\gamma}\right)
\end{aligned}$$

where $\Phi(\cdot)$ is the standard normal cumulative distribution.

Similarly, while the system is in condition B, X_n has pdf $q(x)$, and so

$$
\begin{aligned}
[P^B]_{ij} &= \Phi\left(\frac{1}{\mu}\ln\frac{r_j}{\gamma(a+r_i)}-\mu\right)-\Phi\left(\frac{1}{\mu}\ln\frac{r_{j-1}}{\gamma(a+r_i)}-\mu\right) \quad i,j=1,2,\ldots,m-1\\
[P^B]_{mj} &= \begin{cases} 0 & \text{j=1,2,...,m} \\ 1 & \text{j=0} \end{cases}\\
[P^B]_{im} &= 1-\Phi\left(\frac{1}{\mu}\ln\frac{\rho^*}{\gamma(a+r_i)}-\mu\right) \qquad i=1,2,\ldots,m-1\\
[P^B]_{0j} &= \Phi\left(\frac{1}{\mu}\ln\frac{r_j}{a\gamma}-\mu\right)-\Phi\left(\frac{1}{\mu}\ln\frac{r_{j-1}}{a\gamma}-\mu\right) \qquad j=1,2,\ldots,m-1\\
[P^B]_{0m} &= 1-\Phi\left(\frac{1}{\mu}\ln\frac{\rho^*}{a\gamma}-\mu\right)
\end{aligned}
$$

Solving equations (41) and (42) depends crucially upon obtaining a set S^1 such that these linear equations have numerically stable solutions. Because of the peculiar nature of the matrices P^G and P^B the task of generating such a set is not straightforward. Since X_n is a *continuous* random variable, however, a re-stating of equations (41) and (42) is possible. This re-formulation allows a solution to be obtained by means of existing techniques from numerical analysis. In particular, we can define the elements of r_i to be

$$r_i = i\delta = \frac{i\rho^*}{m} \qquad i=0,1,2,\ldots,m$$

so that δ is the interval between m+1 equally spaced points from $R_n = 0$ to $R_n = \rho^*$. By taking the limit as $\delta \to 0$, this allows the definition of continuous analogues of the steady-state probability vectors Π. To do this, we define $f_g(r)$ and $f_b(r)$ to be the steady state probability density functions satisfying:

$$
\begin{aligned}
\pi_0 f_g(r)\delta &= \text{prob.}\{r < R \le r+\delta \cap \text{system is in condition } G\}\\
\pi_0 f_b(r)\delta &= \text{prob.}\{r < Z \le r+\delta \cap \text{system is in condition } B\}
\end{aligned}
$$

and define the vectors

$$
\begin{aligned}
g_i &\equiv \Pi_i \qquad i=1,2,\ldots,m-1\\
b_i &\equiv \Pi_{i+m}
\end{aligned}
$$

Thus $f_g(r_i)\delta = [\Pi_g]_i$ and $f_b(r_i)\delta = [\Pi_b]_i$. Equation (41) can then be written

$$[\Pi_g]_j = (1-a)\sum_{i=1}^{m}[\Pi_g]_i P_{ij}^G + (1-a)P_{0j}^G \quad j = 1,2,\ldots,m-1 \tag{49}$$

$$[\Pi_g]_m = (1-a)\sum_{i=1}^{m}[\Pi_g]_i P_{im}^G + (1-a)P_{0m}^G. \tag{50}$$

Using the definition of $f_g(.)$, the first set of these becomes:

$$f_g(r_j)\delta = (1-a)\sum_{i=1}^{m} f_g(r_i)\delta P_{ij}^G + (1-a)P_{0j}^G \qquad j = 1,2,\ldots,m-1 \tag{51}$$

From the definitions of P_{ij}^G and P_{0j}^G given in above, it can be readily shown that, for small δ,

$$[P^G]_{ij} \cong p\left(\frac{\delta}{\mu}\ln\frac{r_j}{\gamma(a+r_i)}\right)\frac{d}{dr_j}\left[\frac{1}{\mu}\ln\frac{r_j}{\gamma(a+r_i)}\right] \tag{52}$$

$$= \frac{\delta}{\mu r_j}p\left(\frac{1}{\mu}\ln\frac{r_j}{\gamma(a+r_i)}\right), \qquad \text{i,j=1,2,...,m-1} \tag{53}$$

and

$$P_{0j}^G \cong \frac{\delta}{\mu r_j}p\left(\frac{1}{\mu}\ln\frac{r_j}{\gamma}\right). \tag{54}$$

Dividing equation (51) by δ and taking the limit as $\delta \to 0$ gives

$$f_g(r) = (1-a)\int_0^{\rho^*} f_g(x)\frac{1}{\mu r}p\left(\frac{1}{\mu}\ln\frac{r}{\gamma(a+x)}\right)dx + (1-a)\frac{1}{\mu r}p\left(\frac{1}{\mu}\ln\frac{r}{a\gamma}\right) \quad 0 < r < \rho^*$$

By a similar argument,

$$f_b(r) = \int_0^{\rho^*}[f_b(x) + af_g(x)]\frac{1}{\mu r}q\left(\frac{1}{\mu}\ln\frac{r}{\gamma(a+x)}\right)dx + a\frac{1}{\mu r}q\left(\frac{1}{\mu}\ln\frac{r}{a\gamma}\right) \quad 0 < r < \rho^*$$

Equations () and () are the continuous equivalents of equations (41) and (42). The solution method is to first solve equation () for $f_g(r)$, and then solve equation () for $f_b(r)$. Substituting $P_{im}^G = 1 - \sum_{j=1}^{m-1} G_{ij}$ and $P_{0m}^G = 1 - \sum_{j=1}^{m-1} P_{0j}^G$ into equation (50) and then using the continuous approximations for G_{ij} and g_j, we can derive the alarm probabilities:

$$[\Pi_g]_m = (1-a)\int_0^{\rho^*} f_g(x)\int_{\rho^*}^{\infty}\frac{1}{\mu r}p\left(\frac{1}{\mu}\ln\frac{r}{\gamma(a+x)}\right)dr\,dx + (1-a)\int_{\rho^*}^{\infty}\frac{1}{\mu r}p\left(\frac{1}{\mu}\ln\frac{r}{a\gamma}\right)dr \tag{55}$$

$$[\Pi_b]_m = \int_0^{\rho^*} [f_b(x)+af_g(x)] \int_{\rho^*}^{\infty} \frac{1}{\mu r} q\left(\frac{1}{\mu} \ln \frac{r}{\gamma(a+x)}\right) dr\, dx + a \int_{\rho^*}^{\infty} \frac{1}{\mu r} q\left(\frac{1}{\mu} \ln \frac{r}{a\gamma}\right) dr. \quad (56)$$

Finally, the normalizing equation equivalent to equation (42) is

$$\pi_0^{-1} = 1 + \int_0^{\rho^*} f_g(x)dx + \int_0^{\rho^*} f_b(x)dx + [\Pi_g]_m + [\Pi_b]_m. \quad (57)$$

Equations () and () are Fredholm equations of the second kind, which have been examined extensively in the literature. The key to their solution is the nature of their kernels, that is the behavior of

$$\begin{aligned} k_g(x,r) &= \frac{1}{\mu r} p\left(\frac{1}{\mu} \ln \frac{r}{\gamma(a+x)}\right) \\ &= \frac{1}{\mu r\sqrt{2\pi}} e^{-\frac{1}{2\mu^2} \ln^2 \frac{r}{\gamma(a+x)}} \end{aligned}$$

and

$$k_b(x,r) = \frac{1}{\mu r\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{1}{\mu} \ln \frac{r}{\gamma(a+x)} - \mu\right)^2}$$

In order to obtain a numerical solution to equation (FG), a FORTRAN code was written using a subroutine (NAG, 1983) that solves this linear non-singular Fredholm integral equation of the second kind using the method of El-Gendi(1969). The function $f_g(.)$, appearing on both sides of equation (FC), is approximated by truncating a Chebychev series to form an n^{th} order polynomial. The subroutine solves for the resulting polynomial coefficients $c_i^g, i = 1, 2, \ldots, n$ such that

$$f_g(r) \approx \sum_{i=1}^{n} c_i^g \cos\left((i-1)\cos^{-1}\left(\frac{2r}{\rho^*} - 1\right)\right)$$

with the property

$$\int_{r=0}^{\rho^*} f_g(r)\, dr \approx \sum_{i=1;\, i \text{ odd}}^{n} \frac{c_i^g \rho^*}{1-(i-1)^2}.$$

The subroutine also provides $f_g(x_i)$ evaluated over the set of Chebyshev points x_i where

$$x_i = \frac{\rho^*}{2}\left(1 + \cos\left(\frac{\pi(i-1)}{n-1}\right)\right) \qquad i = 1, 2, \ldots, n.$$

Equation (FB) is then solved, using the same NAG subroutine, by approximating $f_b(.)$ with a different Chebychev series and using $f_g(.)$ obtained above. This produces the coefficients $c_i^b, i = 1, 2, \ldots, n$, such that

$$f_b(r) \approx \sum_{i=1}^{n} c_i^b \cos\left((i-1)\cos^{-1}\left(\frac{2r}{\rho^*} - 1\right)\right),$$

with the property

$$\int_{r=0}^{\rho^*} f_b(r)\, dr \approx \sum_{i=1;\, i \text{ odd}}^{n} \frac{c_i^b \rho^*}{1-(i-1)^2}.$$

The performance measures $[\Pi_g]_m$ and $[\Pi_b]_m$ are obtained by trapezodial approximation of the integrals in equations (55) and (56) using the values of $f_g(x_i)$ and $f_b(x_i)$ at the Chebyshev points x_i, $i = 1, 2, \ldots, n$. Finally, π_0 is obtainedi by substituting equations (), (), and the values of $[\Pi_g]_m$ and $[\Pi_b]_m$ into equation (28).

15. Numerical Results for Normal Monitoring

Figure 8.1 shows the probability density function $\pi_0 f_g(r)$ for Normal monitoring. Compared to Bernoulli monitoring (Figure 7.2) this distribution is smooth and well behaved except near zero where, it can be shown that $\lim_{r\to 0} f_g(r) = 0$.

Figure 8.2 shows the Operating Characteristic Curves for $p_B = \Pr\{\text{Producing scrap}\}$ versus $\pi_0 = \Pr\{\text{Producing good product}\}$ for two values of a (0.05 and 0.1) while fixing $\mu = 1.5$. As in Figure 7.4a, there is an improvement in the OC with smaller a. Since a is the expected number of inter-monitoring intervals until system failure, a can be reduced by either increasing the actual life of a machine or by decreasing the inter-monitoring interval.

Figure 8.3 shows OC sensitivity to changing μ, the shift in the expected observation value when the system fails. As μ increases the power of the sensor to discriminate between conditions G and B increases and this improves the OC curve. This has obvious implications in evaluating sensors with different μ values.

Figure 8.4 shows the alternative OC curves for Pr{Producing scrap} versus Pr{Producing good product} for $a = 0.05$ and $\mu \in \{0.5, 1.0, 1.5\}$. As in Figure 7.7, if this OC represents a significant trade-off, then there is a wide range of probability thresholds that can be ignored when selecting an operating point. For example, when $\mu = .5$ and $a = 0.05$ all p^* above 0.15 can be ignored

16. Conclusion

The value of the System Operating Characteristic as a tool for evaluation of monitoring procedures and policies has yet to be established. We believe, however, that it is an important and evocative tool for the comparison of policies and the comparison of alternative observation technologies. The structure set forth in this paper provides a method for the computation of critical measures such as the expected detection time and total alarm rate (δ and r), needed to express various SOCs. For the case of general sampling functions $p(x)$ and $q(x)$, Appendix A presents the equations that need to be solved to find key performance measures. In a companion paper we explore the effectiveness of promising numerical techniques that can be used to efficiently obtain SOC's.

Acknowledgments

This research was supported jointly by General Motors Research Laboratories and the National Science Foundation under contract # SES9108956. The support of Walter Albers, Jr., who was instrumental in arranging for this unique funding collaboration, is greatly appreciated. The research and presentation was also appreciably strengthened by many substantive discussions with Richard Marcellus. Much of the computation was assisted by Wei-Ching Wang and Julie Simmons.

APPENDIX A: Steady State Properties of MPR

In this Appendix, we show how the Chapman-Kolmogorov (C-K) equations can be used to write expressions for steady-state probabilities and distributions for MPR. Although the notation and details may appear formidable, the method is a straightforward extension of the use of C-K equations for finding steady-state solutions to a finite state Markov chain. In the development that follows, it might be helpful to refer to the schematic flow diagram of Figure A1. In this diagram, the "states" $\{\mathcal{R}_G\}$ and $\{\mathcal{R}_B\}$ refer to R-values in the open interval $(0, \rho^*)$ while the machine is in condition "G" and "B," respectively. S_G^*, S_B^*, and 0 are the singleton "false alarm," "true alarm," and "renewal" states as discussed in Section 10. The labels on the transition arrows to singleton states represent governing probabilities. The arrows to $\{\mathcal{R}_G\}$ and $\{\mathcal{R}_B\}$), represent the complementary distribution functions $\overline{P}$ and $\overline{Q}$ associated with exceeding the threshold ρ_*, given machine condition G and B, respectively.

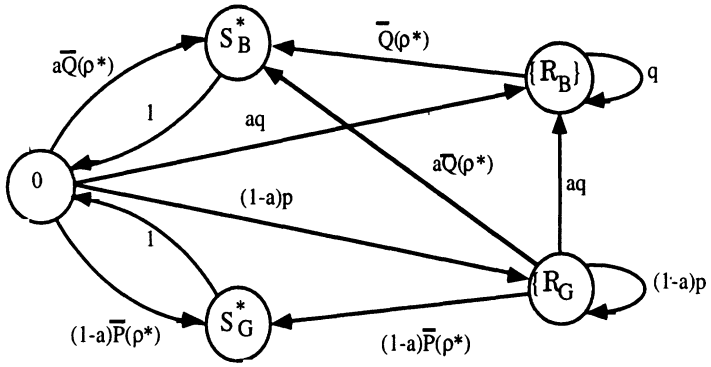

Figure A1: Schematic representation of transitions among the states in MCR. Note that 0, S_B^* and S_G^* are singleton states, while $\{\mathcal{R}_G\}$ and $\{\mathcal{R}_B\}$ represent a continuum of states in the open interval $(0, \rho^*)$.

Consider the computation of $\Pi_G(r)$. By definition

$$\Pi_G(r) = \lim_{n\to\infty} \text{prob}\{S_n \in [(t, G)] : 0 < t \leq r\}, \qquad , 0 < r < \rho^*$$

$$= \lim_{n\to\infty} \text{prob}\{0 < R_n \le r \cap C_n = G\},$$

which by conditioning on the value of R_{n-1}, and noting that $C_n = G$ is only possible if $C_{n-1} = G$, gives

$$\Pi_G(r) = \int_{y=0}^{\rho^*} \lim_{n\to\infty} \text{prob}\{0 < R_n \le r \cap C_n = G | R_{n-1} = y \cap C_{n-1} = G\}$$
$$\times \text{prob}\{R_{n-1} \in (y, y+dy) \cap C_{n-1} = G\}. \qquad (A1)$$

The first probability in (A1) can be obtained by using:

a) equation (23) which governs the behavior of R_n when $R_{n-1} = y$, and

b) prob.$\{C_n = G | C_{n-1} = G\} = 1 - a$ which is independent of the value of R_{n-1}.

Thus

$$\Pi_G(r) = \int_{y=0}^{\rho^*} \lim_{n\to\infty} (1-a)\ \text{prob}\{0 < \ell(X_{n-1})(y+a) \le r | C_{n-1} = G\}$$
$$\times \text{prob}\{R_{n-1} \in (y, y+dy) \cap C_{n-1} = G\}.$$

By definition, the second probability in this expression is

$$\lim_{n\to\infty} \text{prob}\{R_{n-1} \in (y, y+dy) \cap C_{n-1} = G\} = d\Pi_G(y), \qquad 0 < y \le \rho^*.$$

Hence, after taking the limit as $n \to \infty$,

$$\Pi_G(r) = \int_{y=0^+}^{\rho^*} (1-a)\ \text{prob.}\{0 < \ell(X_{n-1})(y+a) \le r | C_{n-1} = G\} d\Pi_G(y)$$
$$+(1-a)\pi_0\ \text{prob.}\{0 < \ell(X_{n-1}a) \le r | C_{n-1} = G\}.$$

By defining the region $\mathcal{C}(r,t) \equiv \{x :\ \ell(x) < r/(a+t)\}$, $\mathcal{C}(\rho^*, x_n)$ becomes the set of "continuation" values of the observation x_n. Using this, and the fact that the p.d.f. for X_{n-1}, given $C_{n-1} = G$, is $p(x)$,

$$\Pi_G(r) = (1-a) \int_{y=0}^{\rho^*} \int_{x\in\mathcal{C}(r,y)} p(x)\, dx\, d\Pi_G(y) + (1-a)\pi_0 \int_{x\in\mathcal{C}(r,0)} p(x)\, dx. \qquad (A2)$$

Similarly, it can be shown that

$$\begin{aligned}
\Pi_B(r) &= a\int_{y=0}^{\rho^*}\int_{x\in\mathcal{C}(r,y)} q(x)\,dx\,d\Pi_G(y) + a\pi_0\int_{x\in\mathcal{C}(r,0)} q(x)\,dx \\
&\quad + \int_{y=0}^{\rho^*}\int_{x\notin\mathcal{C}(\rho^*,y)} q(x)dxd\Pi_B(y) \qquad \text{(A3)} \\
\pi_G^* &= \int_{y=0}^{\rho^*}\int_{x\notin\mathcal{C}(\rho^*,0)} (1-a)p(x)\,dx\,d\Pi_G(y) + \pi_0\int_{x\notin\mathcal{C}(\rho^*,0)} (1-a)p(x)\,dx. \qquad \text{(A4)} \\
\pi_B^* &= \int_{y=0}^{\rho^*}\int_{x\notin\mathcal{C}(\rho^*,0)} aq(x)\,dx\,d\Pi_G(y) + \pi_0\int_{x\notin\mathcal{C}(\rho^*,0)} aq(x)\,dx. \\
&\quad + \int_{y=0}^{\rho^*}\int_{x\notin\mathcal{C}(\rho^*,y)} q(x)\,dx\,d\Pi_B(y) \qquad \text{(A5)}
\end{aligned}$$

and, finally

$$\begin{aligned}
\pi_0 &= \pi_G^* + \pi_B^* \\
&= \int_{y=0}^{\rho^*}\int_{x\notin\mathcal{C}(\rho^*,0)} [aq(x) + (1-a)p(x)]\,dx\,d\Pi_G(y) \\
&\quad + \int_{y=0}^{\rho^*}\int_{x\notin\mathcal{C}(\rho^*,y)} q(x)\,dx\,d\Pi_B(y) + \pi_0\int_{x\notin\mathcal{C}(\rho^*,0)} [aq(x) + (1-a)p(x)]\,dx. \qquad \text{(A6)}
\end{aligned}$$

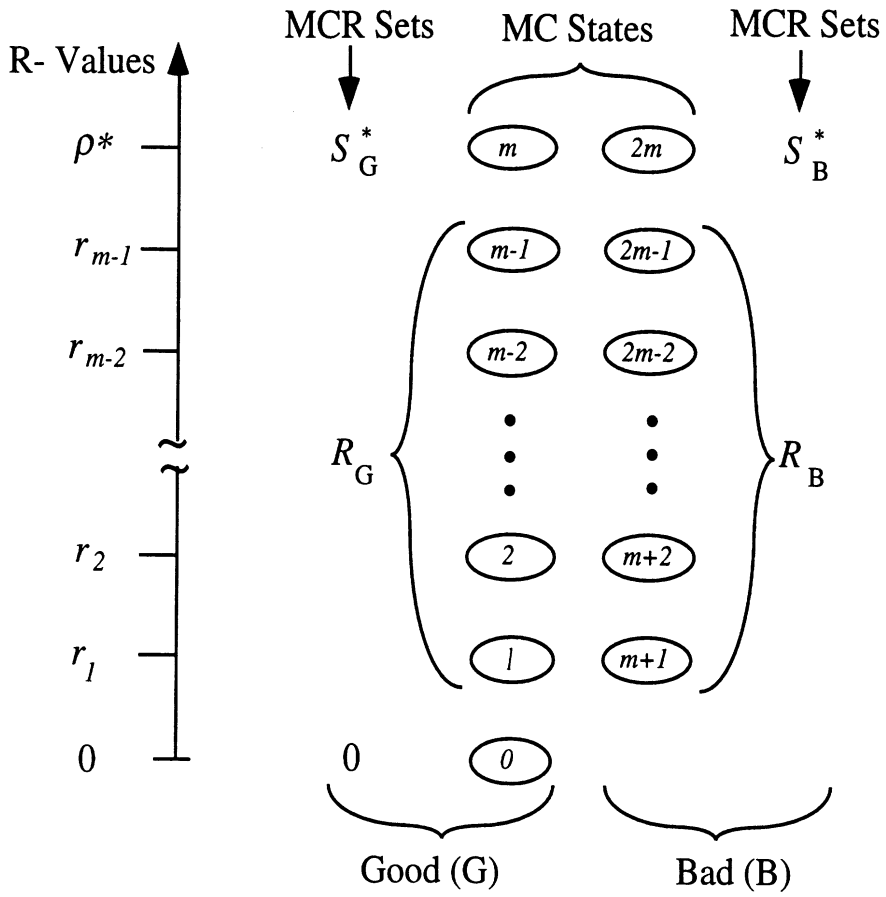

Figure B1: Schematic representation of the MC states, the MCR sets, and the R-values generated from equation (23).

APPENDIX B: Correspondance between Formulations MC and MCR

The correspondance between elements of $I_{2m+1} \in \{0, 1, 2, \ldots, 2m\}$ of the chain MC and of the elements MCR which lie in the set $\mathcal{S}^1 \times \{G, B\}$ are shown in Figure B1.

State 0 is the "starting" state of MC. Since it represents the situation where $P_n = 0$ (and thus $R_n = 0$) it is also the "renewal" state, with machine condition G; state m is the false alarm state, since $R_n = \rho^*$ with machine condition G; state $2m$ is the true alarm state, since $R_n = \rho^*$ with machine condition B. The set $\mathcal{R}_G$ represent states where R_n lies between 0 and ρ^* with machine condition G; $\mathcal{R}_B$ represents states where R_n lies between 0 and ρ^* with

machine condition B.

Transition probabilities for MC, i.e. among the states in I_{2m+1}, are obtained by noting that:

a) when $R_{n+1} \geq \rho^*$, the state entered after transition $n+1$ is either m or $2m$, depending whether machine condition is either G or B;

b) once in state m or $2m$, since $b = c = 1$, the next transition is into state 0 with probability 1;

c) for any value of $R_n < \rho^*$, if the machine condition is G then with probability a condition B will hold on the next transition;

d) the only transition from condition B to condition G must be made via the true alarm state $2m$.

Thus, transitions from alarm states to the renewal state are

$$\begin{aligned} p_{m,0} &= 1, \\ p_{2m,0} &= 1, \\ p_{m,j} &= 0 \qquad \text{if } j \neq 0, \\ p_{2m,j} &= 0 \qquad \text{if } j \neq 0, \end{aligned}$$

and $p_{ij} = 0$ for $i = m+1, m+2, \ldots, 2m-1;\ j = 0, 1, 2, \ldots, m$.

For the rest of the elements of P, we define the $m \times m$ sub-matrices P^G and P^B, such that

$$\begin{aligned} \left[P^G\right]_{ij} &= \text{prob.}\{\sigma_n = j | \sigma_{n-1} = i \cap C_n = G\} \qquad \text{for } i = 0, 1, \ldots, m-1;\ j = 0, 1, \ldots, m, \\ \left[P^B\right]_{ij} &= \text{prob.}\{\sigma_n = j + m | \sigma_{n-1} = i \cap C_n = B\} \quad \text{for } i = 0, 1, \ldots, m-1;\ j = 1, 2, \ldots, m, \end{aligned}$$

The elements of these submatrices are given by the nature of the distributions $p(\cdot)$ and $q(\cdot)$ of equation (15). In terms of these submatrices, the remainding elements of P are given

by:

$$p_{ij} = (1-a)[P^G]_{ij} \qquad \text{for } i = 0,1,\ldots,m-1;\ j = 0,1,\ldots,m$$
$$p_{ij} = a[P^B]_{i,j-m} \qquad \text{for } i = 0,1,\ldots,m-1;\ j = m+1,\ldots,2m,$$
$$p_{ij} = [P^B]_{i-m,j-m} \qquad \text{for } i = m+1,m+2,\ldots,2m-1;\ j = m+1,m+2,\ldots,2m.$$

The first equation reflects transitions from $C_{n-1} = G$ to $C_n = G$; the second represents transitions from $C_{n-1} = G$ to $C_n = B$; the third equation reflects transitions while the machine condition is B. Matrix P is shown schematically in Figure **.

APPENDIX C: Performance Measures for Arbitrary b and g

In our development, we assumed checking times of $b = g = 1$. Calculating performance measures for arbitrary b and g is described here.

Recalling that $\overline{L}$ is the expected cycle time, we can re-express equation (**) as:

$$\overline{L}(b, g) = E(T) + g(n - 1) + \delta + b \tag{58}$$

which explicitly incorporates as arguments the checking times b and g;for condition B or G, respectively. By definition, $p_f(b, g)$ is then the associated fraction of time the machine is in the false alarm state. It is clear that, when $b = g = 1$, $p_f(1, 1)\overline{L}(1, 1)$ is the resulting expected number of false alarms per cycle. For arbitrary b and g, the cycle time is $\overline{L}(1, 1)$ plus $b - 1$ (for the check while in B) plus $p_f(1, 1)\overline{L}(1, 1)(g - 1)$ (for the false alarm checks). Hence, the expected cycle time for arbitrary b and g can be written

$$\overline{L}(b, g) = \overline{L}(1, 1) + (b - 1) + p_f(1, 1)\overline{L}(1, 1)(g - 1).$$

Using this result, the fraction of time spent in the false alarm state is

$$p_f(b, g) = \frac{p_f(1, 1)\overline{L}(1, 1)g}{\overline{L}(b, g)},$$

and the fraction of time spent in the true alarm state is

$$p_t(b, g) = \frac{b}{\overline{L}(b, g)}.$$

References

Barlow, R. E., Hunter, L. C. and Proschan, F.,"Optimum Checking Procedures", *J. Soc. Indust. Appl. Math.*, 11, 1078-1095 (1963)

Barlow, R. E., and Proschan, F., *MAthematical Theory of Reliability*, Wiley, New York (1965)

Boland, P.J. and Proschan, F. , "Periodic Replacements With Increasing Minimal Repair Costs at Failure Time', *Operations Research*, 8, 1183-1189 (1982)

Brunner, H. "A Survey of Recent Advances in the Numerical Treatment of the Volterra Integral and Integro-differential Equations", *Journal Comput. Appl. Math*, 8, 213-229 (1982).

Duncan, A. J., "The Economic Design of $\overline{X}$ Charts Used to Maintain Current Control of a Process," *Journal of the American Statistical Association*, 51, 228–242, (1956)

Girshik, M.A. and Rubin, H. "A Bayes Approach to a Quality Control Model," *Ann. Math. Stat.*, Vol. 23, 114-125 (1952)

Groetsch, C. W., "The theory of Tikhonov Regularization for Fredholm Equations," 104p, Boston Pitman Publication (1984).

Johnson, N.L. and Leone, F.C., "Cumulative Sum Control Charts: Mathematical Principles Applied to Their Construction and Use," Parts I, II, III. *Industrial Quality Control*, Vol. 18, 15-21; Vol. 19, 29-36; Vol 20, 22-28, (1962).

Lorden, G., "Procedures for Reacting to a Change in Distribution," *Ann. Math. Stat.*, 1897-1908, (1971).

Lorenzen, T. J. and L. C. Vance , "The Economic Design of Control Charts: A Unified Approach", *Technometrics* 28, 3–10, (1986)

Marcellus, R. L. and Jasmani, Z., "A Comparative Study of Cusum Control Charts and Bayesian Process Control," in *Productivity and Quality Management Frontiers - III*, Institute of Industrial Engineers, Norcross, GA. (1991)

Montgomery, D.C. "The Economic Design of Control Charts: A Review and Literature Survey," *Journal of Quality Technology*, Vol. 12, No. 2 75-87 (1980).

Montgomery, D C. *Introduction to Statistical Quality Control*, second edition, John Wiley and Sons, New York. (1991),

Moskowitz, H., Plante, R. and Chun, Y.H. "Economic Design of Continuous Shift Model $\overline{X}$ Process Control Charts," Krannert Graduate School of Management, Purdue University (1989).

Page, E. S., "Continuous Inspection Schemes," *Biometrika* 41, 100–115, (1954)

Pollak, M. "Average Run Lengths of an Optimal Method of Detecting a Change in Distribution," *Ann. Stat.*, Vol. 15, No. 2, 749-779 (1987).

Pollak, M. and Siegmund, D., "Approximations to the Expected Sample Size of Certain Sequential Tests," *Ann. Stat.*, Vol. 6, 1267-1282, (1975).

Pollak, M. "Optimal Detection of A Change in Distribution," *Ann. Stat.*, Vol. 13, No. 1, 206-227 (1985).

Pollock, S.M., "Minimum Cost Checking Using Imperfect Information," *Management Science*, Vol. 13, No 7, pp 206-227 (1965).

Rapoport, A. and G. J. Burkheimer, "Parameters of Discrete Time Models of Detection of Change," *Management Science* 19(9), 973-984, (1973)

Roberts, S.W., "A Comparison of Some Control Chart Procedures," *Technometrics*, Vol. 8, 411-430 (1966).

Schippers, H., "Multiple Grid Methods for Equation of the Second Kind," Amsterdam, Mathematisch Centrum, 133p (1983)

Shewhart, W.A., "*The Economic Control of the Quality of Manufactured Product*," Macmillan, New York, (1931).

Shiryaev, A.N., "*Optimal Stopping Rules*," Springer-Verlag, New York, (1978).

Shiryaev, A.N. "On Optimum Methods in Quickest Detection Problems," *Prob. Appl.*, Vol. 8, 22-46 (1963).

Svoboda, L., "Economic Design of Control Charts: A Review and Literature Survey," in *Statistical Process Control in Manufacturing*, Marcel Dekker, Inc., New York. (1991)

Woodall, W. H.,"Weaknesses of the Economic Design of Control Charts," *Technometrics* 28(4), 408–409, (1986)

Woodall, W. H., "The Statistical Design of Quality Control Charts," *The Statistician* 34, 155–160,(1985)