Center for Quality and Productivity Improvement
University of Wisconsin
610 Walnut Street
Madison, Wisconsin 53705

(608) 263-2520
(608) 263-1425 FAX
quality@engr.wisc.edu

Report No. 173

Detecting Malfunctions in Dynamic Systems

George Box, Spencer Graves, Soren Bisgaard,

John Van Gilder, Ken Marko, John James,

Mark Seifer, Mark Poublon, and Frank Fodale

March 1999

CQPI
University of Wisconsin

Report No. 173

Report Series
in Quality and Productivity

Detecting Malfunctions in Dynamic Systems

George Box[1], Spencer Graves[2], Soren Bisgaard[3],
John Van Gilder[4], Ken Marko[5], John James[6], Mark Siefer[7],
Mark Poublon[8], and Frank Fodale[9]

ABSTRACT

Computer controls are increasingly being employed in systems ranging from simple to very complex. A new trend is to extend these computer systems to include monitoring schemes to detect malfunctions. An example is provided by new automobiles sold in the US and Canada. By law they must include "on-board diagnostics" designed to detect certain malfunctions in the powertrain system that may cause excessive emissions. The present article outlines some of the fundamental concepts of system's monitoring and general principles for the design of such monitors.

[1] Research Director, Center for Quality and Productivity Improvement, University of Wisconsin-Madison, USA
[2] Center for Quality and Productivity Improvement, University of Wisconsin-Madison, USA and Productive Systems Engineering, San Jose, CA, USA
[3] Director, Institute for Technology Management, University of St. Gallen, Switzerland
[4] Powertrain Control Center, General Motors, Milford, MI, USA
[5] Project Leader, Advanced Diagnostics, Ford Scientific Research, Dearborn, MI, USA
[6] Ford Scientific Research, Dearborn, MI, USA
[7] Reliability Coordinator, Powertrain Program Management, DaimlerChrysler Technical Center, Auburn Hills, MI, USA
[8] Product Development Specialist, DaimlerChrysler Technical Center, Auburn Hills, MI, USA
[9] DaimlerChrysler Technical Center, Auburn Hills, MI, USA

Detecting Malfunctions in Dynamic Systems

George Box, Spencer Graves, Søren Bisgaard,
John Van Gilder, Ken Marko, John James, Mark Seifer,
Mark Poublon, and Frank Fodale

Computer controls are increasingly being employed in systems ranging from simple to very complex. A new trend is to extend these computer systems to include monitoring schemes to detect malfunctions. An example is provided by new automobiles sold in the US and Canada. By law they must include "on-board diagnostics" designed to detect certain malfunctions in the powertrain system that may cause excessive emissions. The present article outlines some of the fundamental concepts of system's monitoring and general principles for the design of such monitors.

1. Introduction

Our lives, property and safety increasingly depend on ever more complex systems; gains in computer technology have significantly increased this trend. It is therefore of growing importance to surround such systems with monitoring systems that can indicate if the primary system is malfunctioning. A typical example is provided by the auto industry's legally mandated "on-board diagnostics" (OBD). These monitoring systems are intended to detect certain malfunctions in the powertrain as precursors for potential emission problems. The regulations in this area, known as OBD II and originally promulgated by the California Air Resources Board (CARB 1997), have been adopted throughout the US and Canada and are being studied for possible adoption by governmental agencies around the world. Whether mandated or not, it will be important that similar malfunction detection systems be designed for many other systems.

Although the malfunction detection problem superficially may appear to be similar to automatic feedback control, we will in this article show that system's monitoring is different and that techniques and principles originating from the statistical literature are called for. Thus the purpose of this article is to outline the basic concepts involved in monitoring systems such as dynamic modeling, Type I and II errors and run length distributions. We will also outline general principles for the design of system's monitoring including how to evaluate the performance of monitors.

2. What is a Monitoring System?

The basic idea of a monitoring system is shown in Figure 1. It is here exemplified in the context of an on-board diagnostic for automobiles, but the same principles will apply to any other monitoring system. The system to be monitored is shown as accepting inputs and producing outputs generally quantifiable in terms of numbers and sampled at discrete, regular time intervals. This *primary* system may or may not include one or more subsystems for feedback control; such control systems will in this context be considered part of the primary system, not the monitoring system. In automotive applications the primary system could be the entire powertrain system or some component of it, for example the intake manifold.

In parallel with the primary system is a secondary system that we will call the *model module*. It accepts data on inputs to the primary system, and its mission is to model, as closely as possible, the performance of the primary system when functioning properly. The model may or may not be based on physical theory; it may be an empirical model, a neural network or some combination of these, and is often dynamic (i.e. time dependent). Next, outputs from the primary system are compared with those from the model. Differences between the measured outputs of the primary system and the model are called *residuals*. If these show significant discrepancies, it is taken as evidence that the primary system is malfunctioning, and the operator is notified usually by turning on a malfunction indicator light (MIL). The decision about turning on the MIL based on the residuals is done by the decision module, which together with the model constitute the monitoring system.

The problem of deciding when a deviation between the output from the primary system and that of the model is sufficiently large to warrant a malfunction indication is complicated by the fact that the inputs and the primary system itself are subject to random variations and measurement errors. Moreover, the model is seldom if ever completely accurate. Because of these errors — random system's variation, measurement errors and model inadequacy — no matter how well we design the decision module, the malfunction indicator will occasionally be turned on when it shouldn't. Using statistical terminology we will call this a *Type I decision error*. Similarly, sometimes the malfunction indicator will not be turned on when it should. This we will call a *Type II decision error*.

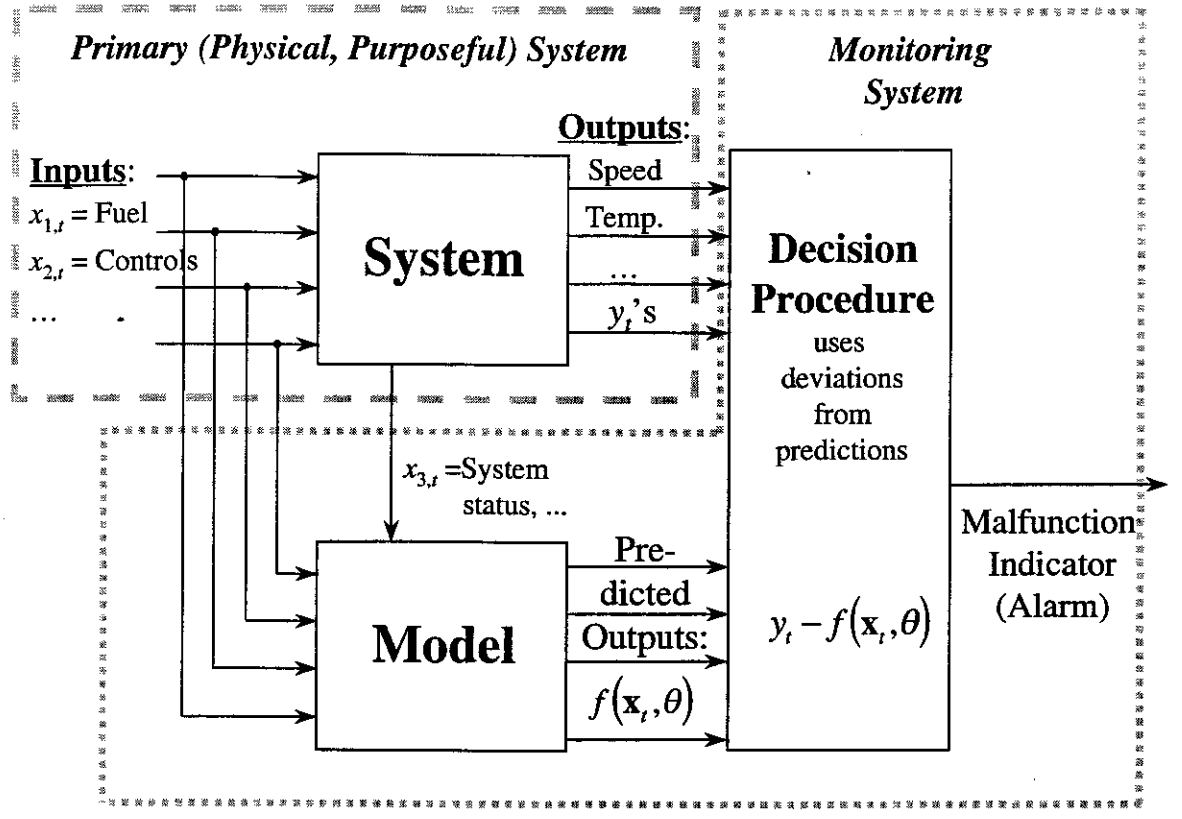

Figure 1. Diagnosing malfunctions in a system.

A little reflection will show that both types of decision errors are undesirable. If too frequent, they can significantly reduce the practical usefulness of a monitor. Table 1 shows the different types of decision errors as a function of the "true" state of the purposeful system. In the automotive context a Type I error means that the owner is required to repair the car when nothing is wrong, and a Type II error that the vehicle might be polluting excessively without the operator being appropriately notified.

An important issue is how to design monitoring systems that are sufficiently quick to respond to real malfunctions thereby avoiding a persistent Type II error while also minimizing the probability of a Type I error. In many cases this requires good predictive models and carefully developed decision procedures. We will in the following discuss a general approach to the selection of decision procedures based on Cuscores (to be described below) that provide monitoring schemes that are sufficiently responsive with an acceptably low error rate.

	Decision	
Truth	"good"	"bad"
good	Correct Classification	Type I error (false alarm)
bad (malfunctioning)	Type II error (failure to detect)	Correct detection (valid alarm)

Table 1. Errors of Type I and II.

3. What Do We Mean by a Good Monitoring System?

As with the design of any system, it is important up front to specify what the objective is and develop criteria that can be used to measure how well a specific implementation meets the goal. For a monitoring system, we would ideally want it never to signal a malfunction when none has occurred, but quickly respond as soon as one has occurred. However, because of random variation this ideal is not completely achievable. In practice the malfunction indicator will, if it is sufficiently responsive, on occasion be turned on when it shouldn't. We may tune it to reduce its sensitivity, but then we run the risk that it may not be turned on when it should. Thus in the design of a malfunction detection system we need to find a compromise between the probabilities of Type I and Type II errors considering, as we shall discuss below, the response time.

The monitor for a dynamic system may repeat its tests more or less continuously. Thus we are interested in a monitoring system that with high probability is sufficiently

quick to respond when a malfunction has occurred. Moreover, we want the monitoring system not to respond, or at least respond with a very small probability over a wide time horizon, when no malfunction has occurred. To evaluate the responsiveness we will therefore consider the *distribution* of the time to an alarm – the *run length*. Whether the primary system is good or bad, the run length distribution describes in terms of a (cumulative) probability distribution how long the monitor runs before turning on the malfunction indicator.

To illustrate the concept of a run length distribution, suppose a diagnostic is testing a component of a powertrain once for each trip; for simplicity let us assume that each of these tests is statistically independent of the previous tests. Suppose further that if the system is good, the probability of turning on the malfunction indicator is only 0.0005 per trip. The probability of turning on the malfunction indicator in n trips is then $1-(1-0.000\,5)^n$. This probability (of turning on the MIL) as a function of the number of trips is shown in Figure 2 as the dotted graph and is what we call the *cumulative run length distribution*. From this we can see that even with this extremely small probability of turning on the MIL in a single trip, after 10,000 trips the probability is virtually 1 of turning on the MIL at least once.

Let us now consider a system that is bad. Here we want the monitoring system to respond quickly. Suppose the probability of *failing* to detect a malfunction in a single trip when it really has occurred is 0.7. (This may be unrealistically high, but we will use this number for illustration.) This is the probability of a Type II error in a single trip. Because we repeatedly perform the test in every subsequent trip, the probability of failing to detect the malfunction in n trips is $(0.2)^n$. Thus as shown in Figure 2 with the solid graph, the chance of *not* detecting a real malfunction will quickly diminish. Notice incidentally that this function is not the cumulative run length distribution but one minus the cumulative run length distribution.

Figure 2 illustrates several of the problems involved in designing an appropriate monitoring system. First, even with an exceedingly small probability (e.g., 0.0005) of making a Type I error (turning on the malfunction indicator when nothing is wrong) in any single trip, the probability of making a Type I error in many trips can be substantial.

Excessive false MILs are worse than useless because (*a*) people will ignore the warning lights, failing to repair defective systems, while (*b*) a high rate of false MILs may undermine the enforceability of the law (i.e. if it is too high any enforcement effort may be successfully challenged in court). In automobiles the problem of frequent false MILs can be particularly serious because some of the tests are conducted very frequently. For some OBD II monitors, tests are conducted once per trip and a car may easily make more than 10,000 trips as a good vehicle providing an opportunity for many false alarms. Other automotive diagnostics may run several times per trip, which can make the problem worse. Some types of OBD II mandated monitors, such as engine misfire monitors, are running continuously during trips making an astronomical number of tests in the lifetime of a vehicle. Thus unless we design the malfunction indicator with extremely small Type I error probability for each test, the MIL can actually do more harm than good.

Now let us consider the risk of Type II errors. Here time is usually on our side. As shown in Figure 2, even with an extremely high probability (e.g., 0.7 in our example above) of not detecting a malfunction in a single trip, the chance of not noticing a malfunction reduces to near zero levels after only a few trips. Thus the issue becomes one of asking what are the consequences of receiving a delayed notification. If the monitor is for a critical

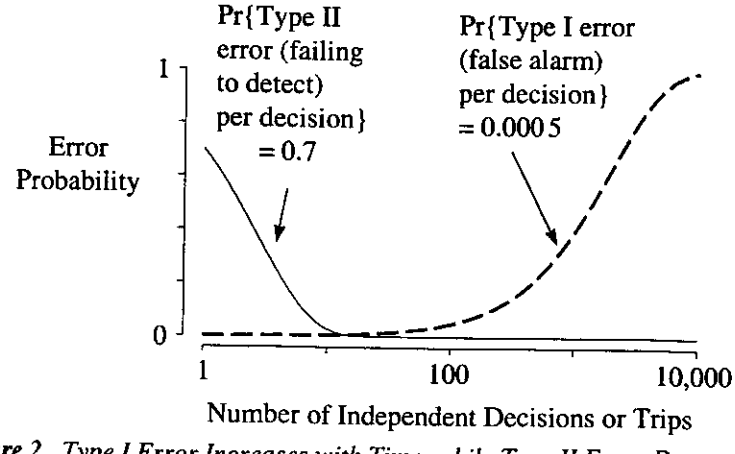

Figure 2. Type I Error Increases with Time, while Type II Error Decreases with Time.

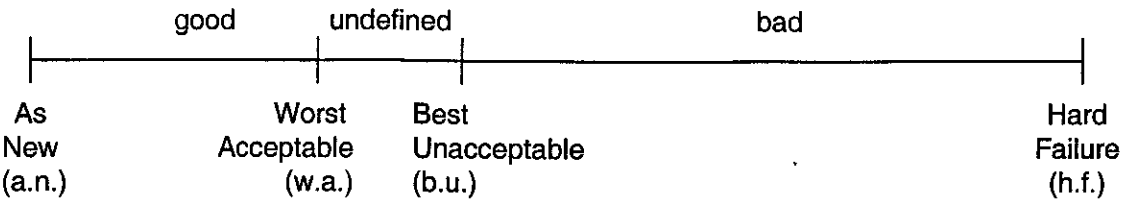

Figure 3. A Range of Possible Conditions of a System.

function of a nuclear power plant, a delayed malfunction indication may mean that signs of an impending catastrophe go unnoticed. In that case we clearly need to design monitoring systems with a quick response time and low risk of making a Type II error even in a single test. However, in the automotive context where malfunctions are less catastrophic it would seem reasonable that the required response time goals should be judged relative to the time it likely will take the average owner to get a vehicle repaired once the MIL has been turned on. Moreover, the cumulative damage to the environment caused by the delay is serious but not catastrophic. Thus, it would seem prudent to balance the probability of making Type I and II errors in favor of reducing the chance of Type I errors. As shown in Figure 2 with repeated tests the Type II error probability will be reduced and the malfunction eventually discovered although not necessarily on the first trip after the malfunction has occurred.

In the previous discussion of run length distributions, we described the probability of detecting a malfunction when a system is *good* and called that a Type I error. Further we discussed the probability of not detecting a malfunction when the system being monitored is *bad* and called that a Type II error. In the automotive context the situation is somewhat more complicated. Systems can assume a continuum of states between "as new" through what we will call "worth acceptable" and "best unacceptable" to "hard failure," as illustrated in Figure 3. We assume that the condition of a vehicle at any point in time can be represented as a point on this continuum. To be precise, the conditions of new vehicles will not be a single point but will follow some distribution clustered around the "as new" point. For any point on this continuum, we can, at least in principle, think of a run length distribution (i.e. the probability of turning on the malfunction indicator within any number of observations). The collection of these run length distribution curves can be represented by a surface as displayed in Figure 4, popularly called the *waterfall chart.*

To better appreciate Figure 4 we have in Figure 5 sliced it for a constant condition of the system. This figure emphasizes the cumulative distribution function of the run length for a system that is constantly in a "best unacceptable (b.u.)" state. For this case, the probability of detecting a b.u. malfunction in one trip is over 50 percent, and the probability of detecting the malfunction in at most two trips is over 95 percent.

We can also get useful information by slicing Figure 4 the other way for a particular number of trips, as illustrated in Figure 6. This is in the statistical literature called the "power function." It shows how the probability of detecting a malfunction (within the indicated number of trips) varies with the condition of the system. The ideal power function would be 0 (zero) on the good side (to the right) of the "midpoint" condition and 1 (one) on the bad side (to the left). Unfortunately, this ideal is never possible because there is uncertainty in any decision. This is reflected in Figure 6 by the fact that the power function (at any fixed run length) is not a step change. As the measurement error and random variability in the evaluation process increases, the power function becomes flatter. This merely quantifies the increasing difficulty of detecting a signal (a malfunction) in greater noise.

The performance of any monitor can in principle be summarized as in Figures 4-6. These figures will in general be different for different monitors and can be used to compare how different monitoring schemes respond to a system that spends its entire life in one state. For a better monitor, the probability of indicating a malfunction will increase faster with both time and severity of the condition of the system. Conversely, as the signal-to-noise ratio decreases, or the processing efficiency of the algorithm decreases, the slopes in Figures 4-6, in both the time and condition dimensions, become more gentle. For monitoring emission controls, a gentle slope can be a problem because OBD II requires a fairly steep increase from "worst acceptable" to "best unacceptable" in the power curves of Figure 6. If the slope becomes too gentle, the monitor under consideration may not be useable.

With these concepts defined we can now discuss what we mean by an effective monitoring system. Essentially the answer will depend on what the malfunction detection system is going to be used for. In general the requirements can be specified in terms of the properties of the run length distributions for given states of the system. As indicated above if the monitored system is a nuclear power plant, we will need to specify certain minimum require-

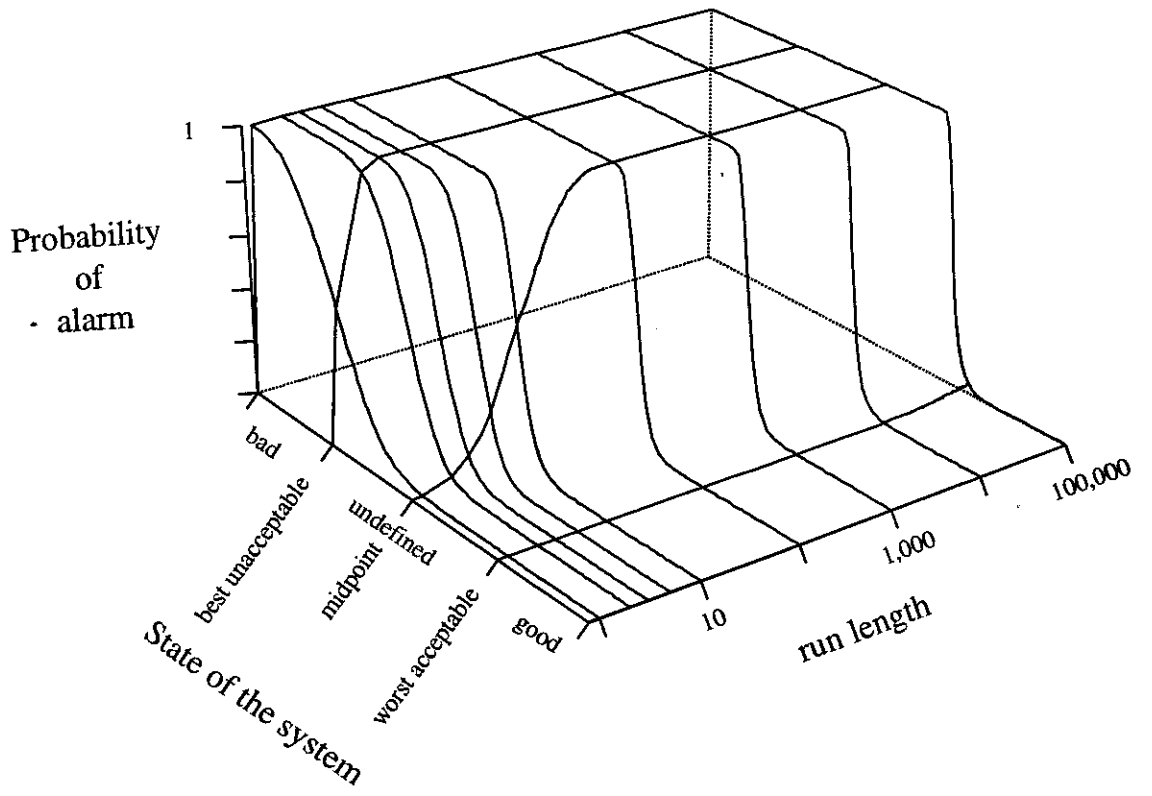

Figure 4. *Probability of Indicating a Malfunction within a Given Time for a Particular Condition of the System.*

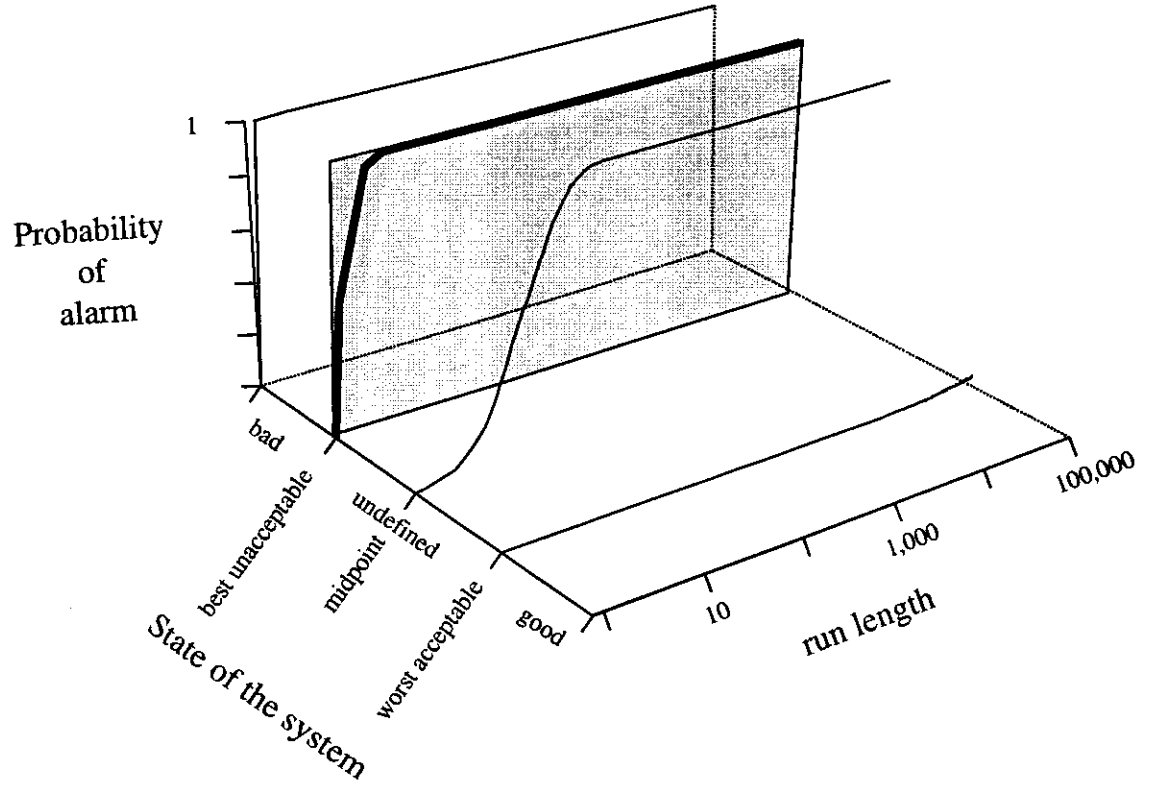

Figure 5. *The Cumulative Distribution Function Is the Probability of turning on the MIL for a Particular Condition of the System.*

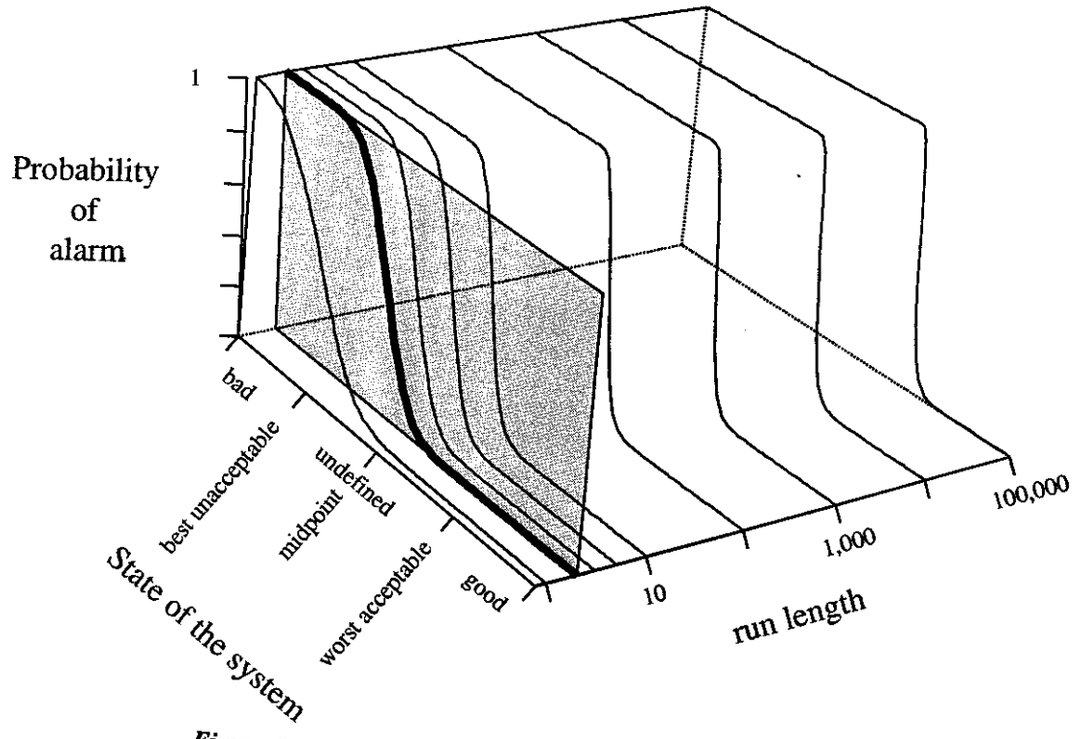

Figure 6. Power of the Monitor at a Particular Run Length.

ments for the responsiveness of the system as expressed through the run length distributions in case of an abrupt malfunction. For the design and fine tuning of the malfunction indicator system, we then need to balance this with the risk of a false alarm. Likewise if it is a malfunction indicator for an automotive powertrain, or some component of it, we will have to specify certain minimum requirements in terms of coordinate points we want the waterfall curve to go through. For a specific application, that will have to be negotiated and settled between the interested parties. In general for a particular monitoring system it is in principle possible to compute the waterfall chart, or at least selected coordinate points on this surface, to evaluate how well a proposed system meets stated specifications. Moreover, any monitoring system will include parameters and thresholds that can be adjusted to achieve as close as possible certain run length goals.

4. Designing Simple Static Monitoring Systems

Above we discussed how to set requirement specifications via the run length distribution for a monitoring system. Let us now turn to the construction of the monitoring system itself. As indicated in Figure 1, it consists of two modules, a model and a decision function.

Suppose the observations from the system to be monitored are random, Normally distributed and serially independent with mean $\mu_{0,t}$ when system is good (worst acceptable) and $\mu_{1,t}$ when it is bad (best unacceptable), where $\mu_{1,t}>\mu_{0,t}$. The subscript t in $\mu_{j,t}$, $j = 0, 1$, allows for the possibility that the system is dynamic.

We shall assume initially that the difference between the means for worst acceptable and best unacceptable $d_t = \mu_{1,t} - \mu_{0,t}$ is a constant $d_t = d$. In other words, if the system turns bad, the observations will tend to be higher by, on average, an amount d. The model for the output of the primary system when it is "good" then takes a particularly simple form

$$y_t = \mu_{0,t} + a_t, \quad t = 1, 2, \ldots,$$

where is the observed quantity at time t, and a_t is independent, Normally distributed error for time t with mean 0 and variance σ^2. Let $\hat{y}_t$ be a prediction of $\mu_{0,t}$ obtained by whatever means, e.g., a least squares regression using explanatory variables such as engine rpm and load. If we compare this prediction to the observed output of the primary system y_t we get the residuals $\hat{a}_t = y_t - \hat{y}_t$.

Based on the assumptions above, a decision function is based on computing the cumulative sum defined as

$$Q_t = \sum_{i=1}^{t} (y_i - \hat{y}_i) = \sum_{i=1}^{t} \hat{a}_i .$$

To appreciate how this quantity Q_t works, imagine that the system for a while is "good". In that case the residuals $\hat{a}_t$ will be random Normally distributed quantities with mean zero (assuming $\hat{y}_t$ is unbiased, i.e., has mean $\mu_{0,t}$). Because the residuals have mean zero, Q_t will fluctuate around zero. Now suppose that at time T the system mean changes to $\mu_{1,t}$. Beginning at that time, with each new observation, the positive fixed quantity $d = \mu_{1,t} - \mu_{0,t}$ will be added to the quantity Q_t, aside from the random deviations. This will in turn imply that Q_t for $t > T$ instead of fluctuating around zero suddenly will take off and increase more or less linearly.

In practice better monitors will be obtained by subtracting a constant $d/2$ from the residuals so that we instead monitor through the centered Cusum

$$Q_t = \sum_{i=1}^{t} (y_i - \mu_{0,t} - d/2),$$

where d usually is the difference between "good" and "bad", i.e. $d = \mu_{1,t} - \mu_{0,t}$ (Box and Luceño 1997). When the system is good, the centered Cusum Q_t will, apart from random variability, decline linearly by an amount $d/2$ with each observation. When the system is bad, Q_t will tend to increase by $d/2$ with each observation. If the system is good for a long period of time, Q_t will tend to become a huge negative number. Therefore, if we judged a malfunction by the level of Q_t, we would usually have to wait a long time for Q_t to become positive after the system became bad. Instead, a malfunction is indicated when the centered Cusum Q_t has increased from its previous minimum by some quantity h. The time of the previous minimum is essentially our best estimate of the time of when the malfunction began.

To simplify the evaluation of the state of the system, we use

$$Q_t^+ = \max\{0, (y_t - \mu_{0,t} - d/2) + Q_{t-1}^+\}.$$

Each time Q_t achieves a new minimum, Q_t^+ becomes zero. Therefore any increase in Q_t over that minimum corresponds to an increase in Q_t^+ over 0. In other words, Q_t^+ represents the increase of Q_t over its previous minimum.

The distribution of the time required for Q_t^+ to exceed a particular threshold h can be summarized in a waterfall chart similar to Figure 4. Increasing the threshold pushes the waterfall back while decreasing the threshold pulls it forward. To obtain the desired run length properties for a given application we need to fine tune the threshold h. If h is small the malfunction indicator will be very responsive to real malfunctions but at the expense of too many Type I errors (false alarms). On the other hand, if we set h to a relatively high value, we will reduce the risk of Type I errors, but at the same time the system will be less responsive to a real change in the level. Finding the appropriate balance can be done in part by Monte Carlo simulation, although in some simple cases the run length distribution may also be found analytically.

5. Cuscore Statistics

In the example above, the cumulative sum was the function of the data that provides the best indicator of a step change in the process (Box and Luceño 1997); "best" here means that the monitor is very quick to set an alarm when the system is bad but quite slow to set a false alarm (i.e., if the system is good). Recall that it was assumed that the system was operating at a good level $\mu_{0,t}$ until a change occurred; after that, the system operated at a bad level $\mu_{1,t}$. In that case we say that the *signal* is a step increase, and $d = \mu_{1,t} - \mu_{0,t}$ is the *signal level*. Because the signal was a simple step change buried in Normal independent noise a_t, the appropriate function of the data is the Cusum. In general the appropriate function of the data to consider in a specific case will depend on what kind of signal we are trying to detect in what kind of noise.

For a slightly more complicated example suppose that the powertrain system is good, combustion is complete, and the catalytic converter eliminates most residual emissions. Then the measured emissions, y_t, is a constant plus random noise. A model for the good system is therefore

$$y_t = \mu_{0,t} + a_t,$$

where as above it is assumed that a_t is Normally distributed with mean zero and variance σ^2. Further suppose that when the system malfunctions the emissions become proportional to the throttle position x_t. Thus a model for the bad system is

$$y_t = \mu_{0,t} + \theta x_t + a_t,$$

where θ is a constant. Therefore this model says that the *signal* we are looking for is x_t and $\theta\,(\theta \geq 0)$ is the *signal level*.

For an outline of how the appropriate Cuscore is arrived at in this case, suppose we are receiving a stream of errors a_t's, which are independent Normally distributed with zero mean and variance. The joint probability density function of the errors accumulated up to time t is then given by

$$f(a_1, \ldots, a_t \mid \sigma^2) = \text{constant} \times \sigma^{-t} \exp\left[-\frac{1}{2}\sum_{i=1}^{t}\left(\frac{a_i}{\sigma}\right)^2\right].$$

Now suppose for simplicity that σ is known. When we take the logarithm we get

$$\ln[f(a_1,\ldots,a_t \mid \sigma^2)] = c_0 - \frac{1}{2\sigma^2}\sum_{i=1}^{t} a_i^2$$

where c_0 is a constant (varying with t but not θ). From the system model, we have $a_t = y_t - \mu_{0,t} - \theta x_t$. The logarithm of the joint probability density function of observations up to time t is

$$l_t = \ln[f(a_1,\ldots,a_t \mid \mu_0,\theta,\sigma^2)] = c_0 - \frac{1}{2\sigma^2}\sum_{i=1}^{t}(y_i - \mu_{0,i} - \theta x_i)^2 .$$

For a fixed set of parameters (μ_0,θ), the joint probability density function f provides information about the probability of obtaining any hypothetical data y_i, $i = 1,\ldots, t$. However, once specific observations y_i, $i = 1,\ldots, t$, have been obtained (and therefore now are fixed values) we can turn the equation around and consider it a function of the (unknown) parameters, and in particular of θ. Specifically, given the data, we can ask which parameter value θ maximizes the function and hence makes this particular set of observations y_i, $i = 1,\ldots, t$, most likely. When the function is used this way it is called a *likelihood function* and plays an important role in obtaining "most likely" estimates of parameters in general statistical analysis. In particular *maximum likelihood estimates* are obtained by finding the parameter values that maximize the likelihood function, or what is equivalent, l_t the logarithm of the likelihood function. Thus as the likelihood function in most cases is continuously differentiable and unimodal, the parameter value that makes the first derivatives of the log likelihood function $\partial l_t/\partial\theta$ zero, is the maximum likelihood estimate.

Now suppose we have a system that for some time has been operating as "good" and generated observations y_t, $t = 1, 2, \ldots$. Thus while the system is "good" the most likely estimate of the parameter value for θ is approximately zero. Therefore differentiating the log likelihood function, with respect to θ and evaluating the derivative for $\theta = 0$ will give a slope that except for random variability is zero when the system is good. The first derivative of the logarithm of the likelihood function with respect to a parameter is called the *Fisher score function* and is defined as

$$Q_t = \left.\frac{\partial l_t}{\partial\theta}\right|_{\theta=0} = \frac{1}{\sigma^2}\sum_{i=1}^{t} a_{i0} r_i ,$$

where $r_i = -\left.\dfrac{\partial a_i}{\partial\theta}\right|_{\theta=0}$. In what follows we will refer to this Q_t as the *Cuscore*.

Now suppose suddenly at time T the system changes such that $\theta > 0$. The subsequent data y_t obtained and incorporated into the likelihood will now force the location of the maximum of the likelihood function to change. Specifically the new derivative terms $a_{t0}r_t$ in the Cuscore obtained after T will no longer be more or less zero since the maximum has moved. (Recall that a_{t0} and r_t are evaluated assuming the system is good.) These terms will in fact rapidly begin to add up and cause the Cuscore to increase, signaling that the system at about time T has changed. In this case where we are looking for a change of θ from to $\theta = 0$ to $\theta > 0$, we will have that

$$r_t = -\left.\frac{\partial a_t}{\partial\theta}\right|_{\theta=0} = -\left.\frac{\partial(y_t - \mu_0 - \theta x_t)}{\partial\theta}\right|_{\theta=0} = x_t ,$$

which implies that

$$Q_t = \frac{1}{\sigma^2}\sum_{i=1}^{t} a_{i0} r_i = \frac{1}{\sigma^2}\sum_{i=1}^{t}(y_i - \mu_0)x_i .$$

Note that if we set $x_t = 1$ (and $\sigma = 1$) for all t then this Q_t is identical to the simple Cusum for detecting a step change we discussed above.

The presence of the throttle x_t in this formula allows the monitor to adjust to changes in the information content of individual observations. For low throttle, there is very little difference between good and bad conditions of the system. Therefore, any difference between the observation and what we expect if the system is good, $(y_t - \mu_{0,t})$, provides little information about a possible malfunction. With high throttle, however, good and bad systems are expected to behave differently, which means that a difference between y_t and $\mu_{0,t}$ will provide information about a possible malfunction. The Cuscore theory says that the product of the deviation from predicted $(y_t - \mu_{0,t})$ and x_t provides an optimal assessment of the relative information in each observation. Therefore this monitor provides a more informative and hence more powerful alternative to typical "enable" conditions currently used in industry, which essentially convert the x_t's to 0's and 1's according to whether the operating conditions are considered to be sufficiently informative regarding the malfunction of interest.

The ideas outlined here can be extended to monitoring complex dynamic systems where the predicted current state may be obtained from complex mathematical models, solution to differential equations, or any other dynamic model such as a neural network. Monitors for multivariate systems with multiple failure modes may require theory beyond that discussed here. Whatever is done, however, requires clear definitions of good and bad, and some ideas of how good systems become bad, whether by abrupt failure or by gradual deterioration. This question becomes step one of an eight-step process for developing a diagnostic, which we now will describe.

6. An Eight Step Process for Developing a Monitoring System

Above we outlined the general idea of how to develop a Cuscore for detection of a malfunction. In this section we briefly outline how this scheme can be implemented in practice for a particular application. We will provide more details in a forthcoming publication.

Step 1. Define "bad" and "good" system conditions: Initially when developing a malfunction indicator the objectives given to engineers may be ill-defined or impossible. In such cases, the first step toward successful design is to redefine the problem so it is clear and appears to be technologically feasible. For monitor design, this step corresponds to making sure the "worst acceptable" and "best unacceptable" points in Figure 3 are clearly defined and sufficiently distinct. We also need to list plausible alternative failure modes (e.g., abrupt jump vs. gradual deterioration), partly to provide a basis for studying robustness of alternative monitors in Step 5.

Step 2. Collect data on both good and bad ("worst acceptable" and "best unacceptable") conditions: Ideally, it is best to have data from both good and bad systems. In practice this is unfortunately not always possible. For systems without a recorded reliability history, failure modes may only be characterized by speculation. Engineers may have only "as new" systems with which to experiment; these could be substantially better than both "worst acceptable" and "best unacceptable." In such cases, the engineers need to develop a method for obtaining data that simulate "worst acceptable" and "best unacceptable" conditions, e.g., by transforming test data from "as new" systems. For some monitors, physical experiments will be required. Statistical techniques for the design of experiments, see Box, Hunter and Hunter (1978), can often help engineers organize and manage this task and make the data gathering as efficient as possible.

Step 3. Develop models for both "good" and "bad" conditions: A typical model, in this context, includes a characterization both of what we can predict and of the unpredictable or random portion:

(observations) = (predictions) + (noise).

or

$$y_t = \mu_t + a_t, \qquad (1)$$

where y_t = observations, μ_t = predictions, and a_t = noise. To develop a sensitive monitoring system with good run length properties, we need, as illustrated in our examples, models both of the good system and of the malfunction we are interested in detecting. The better the models are in describing the physical system the smaller the noise will be. Thus it is of critical importance to develop good models to achieve good run length properties. In fact, we believe it is in this area that engineers can make the most important contributions to the design of reliable monitors. In turn the model building process depends critically on good experimental design, see Step 2.

Step 4. Base the diagnostic on likelihood: The Cuscore function discussed in Section 5 was arrived at by differentiating the log likelihood function. The *Cuscore* is then based on an accumulation over time of the Fisher efficient *score* function. For computational purposes, it is convenient to rewrite the Cuscore in the equivalent recursive form

$$Q_t = Q_{t-1} + W_t,$$

where W_t = derivative of the log likelihood function for a single observation with respect to a parameter representing the severity of a malfunction. Many diagnostics signal a malfunction when a certain number exceeds a threshold. We can not directly use Q_t in this way. This follows because most systems begin with a long period of good operation, during which Q_t would generally drift farther and farther away from the threshold. When a malfunction finally occurred, Q_t could not reach the threshold in a timely fashion. We correct for this by focusing on the excess of Q_t over its minimum to date, which can be written as

$$Q_t^+ = \max\left\{0, Q_{t-1}^+ + W_t\right\}.$$

This is called a *one-sided Cuscore*; see Sections 4 and 5.

Step 5. Consider how the characteristics of a monitor vary with the detection threshold h: One more step is required to turn a diagnostic such as a Cuscore into a useable monitor: For illustration, we use a Cusum statistic to detect a change in level. We must select a threshold h so we signal a malfunction when the diagnostic exceeds h. The effect of adjusting h can be described in terms of its effect on the run length distribution and specifically on "waterfall" charts produced by Monte Carlo simulation similar to that in Figure 2. Increases and decreases in h will change the probability of an alarm for all states of the system and for all run lengths. Thus h is essentially a tuning parameter the engineers can use to achieve specific run length properties desired from the malfunction detection system.

Step 6. Does a feasible solution exist? If after extensive fine-tuning of h it becomes clear that the required run length properties cannot be achieved with the current monitor, we have four options: (*a*) The simplest and most effective approach in many cases might be to return to Step 3 to try to develop a better model for the system, thereby reducing the noise and increasing the signal. (*b*) If that is not feasible, it may be necessary to return to Step 2 and con-

duct further experiments. More carefully planned experiments may lead to better models and an increased signal to noise ratio after repeating Step 3. (*c*) In other cases, it may be necessary to revisit the definitions of "worse acceptable" and "best unacceptable" performance (Step 1). If the difference between good and bad can be increased, it can simplify the detection problem. (*d*) Finally, in many cases, the "as new" system is better than the "worst acceptable," and a monitor might have an unacceptable false alarm rate for the "worst acceptable" system but an acceptable rate for many systems between "as new" and "worst acceptable." In such cases, a careful analysis of the performance of the monitor under typical patterns of gradual deterioration may indicate that the overall performance of the monitor might be acceptable.

Step 7. Evaluate the monitor in real systems with multiple thresholds: When Steps 1-6 produce a monitor that seems to perform satisfactorily, it must still be evaluated in real systems. This is required because real data are not likely to follow the simple distributional assumptions we may specify in simulation studies. Thus an artificial simulation study may not adequately describe the performance of a real system. Testing should include the generation of multiple alarms, both false and valid. For OBD, the false alarm rate should be so low that none might be observed during testing. We balance these conflicting needs by testing at artificially low thresholds and then selecting the final threshold by extrapolation.

Step 8. Consider how engineering design procedures can be improved for the future: Most engineers and most design groups design more than one product. Some leading engineering companies now make explicit efforts to translate their experiences with each project into improvements in the procedures used to design other products. Thus this step is included to assure continuous organizational learning and carryover to future product development cycles.

7. Summary and Conclusions

The increasing use of computer controls on systems from simple to highly complex provides new opportunities for the detection of impending or actual malfunctions and for simplifying the work of maintaining complex systems. Many problems can be diagnosed using data already available to the microprocessor and used for control purposes. In such cases, the only substantive cost may be the engineering time required to understand and apply basic concepts of monitoring. In other cases, the value of more timely and accurate identification of impending or current malfunctions can justify the cost of installing and using additional sensors to collect data on the condition of the system.

Cutting edge developments in this area are being driven in part by legal mandates to detect, for example, malfunctions in emission controls on new automobiles. However, there are clear needs for systems to monitor the status of components of many other systems such as aircraft, nuclear power plants, and pacemakers.

To be useful, monitoring schemes must provide reasonably quick response to actual or impending malfunctions with a low probability of falsely declaring a malfunction during the "good" life of a system. Monitors that perform well according to these criteria can be derived using statistical tools including the "Cuscore Principle" and the eight step process for developing a diagnostic, described in this article.

Acknowledgements

This report is based on research supported by the Low Emissions Technologies Research and Development Partnership (LEP) of DaimlerChrysler, Ford, and General Motors through a contract with the University of Wisconsin's Center for Quality and Productivity Improvement.

References

Box, G. E. P., Hunter, W. G. and Hunter, J. S. (1978), *Statistics for Experimenters*, NY: Wiley and Sons.

Box, G. E. P., and Luceño, A. (1997) *Statistical Control by Monitoring and Feedback Adjustment*, NY: Wiley and Sons.

CARB (1997) "Malfunction and Diagnostic System Requirements—1994 and Subsequent Model-Year Passenger Cars, Light-Duty Trucks, and Medium-Duty Vehicles and Engines" (OBD II), with modifications effective as of September 25, 1997, sec. 1986.1 of Title 13,California Code of Regulations. Sacramento, CA: California Air Resources Board; available from http://www.arb.ca.gov/msprog/obdprog/obdprog.htm.