Chapter One

Introduction

1.1 THE BASIC QUESTION

In this book, we study the problem of active failure detection in dynamical systems. Failure detection in some form is now part of essentially every complex device or process. In some applications the detection of failures, such as water losses in nuclear reactors or engine problems on an aircraft, is important for safety purposes. The detection of failure leads to emergency actions by operators or computers and results in a quick and controlled shutdown of the system. In other situations, such as on space missions, the detection of failures results in the use of back-up or alternative systems. These are the more dramatic examples and are often what one first thinks of when hearing of a failure. But in today's society failure detection also plays a fundamental role in managing costs, promoting efficiency, and protecting the environment. It is often much more economical to repair a part during a scheduled maintenance than to have a breakdown in the field. For example, failure may mean that a part or subsystem is not performing to specification, resulting in increased fuel consumption. Detecting this failure means that a scheduled repair can be made with savings of both resources and money. Failure can also mean that a part or subsystem is not performing as expected and that if allowed to continue the result could be a catastrophic failure. But again detection of this type of failure means that repairs can be initiated in an economical and convenient manner. It is much easier to repair a weakened pump than to have to clean up a major sewage spill.

A number of specific examples from applications are in the cited literature. In chapters 2 and 3 we shall use a couple of intuitive examples to motivate some of the ideas that follow. Simpler academic examples will be used to illustrate most of the key ideas and algorithms. Then in the later chapters we shall include some more detailed examples from application areas.

Because of the fundamental role that failure detection plays, it has been the subject of many studies in the past. There have been numerous books [3, 69, 4, 26, 39, 70, 27] and survey articles [83, 44, 1, 38, 34, 68, 35, 36, 37] dedicated to failure detection. The book by Chen and Patton [26] in particular gives an up to date overview of the subject.

Most of these works are concerned with the problem of *passive failure detection*. In the passive approach, for material or security reasons, the detector has no way of acting upon the system. Rather, the detector can only monitor the inputs and the outputs of the system and then try to decide whether a failure has occurred, and if possible of what kind. This decision is made by comparing the measured input-output behavior of the system with the "normal" behavior of the system. The passive approach is often used to continuously monitor the system, although it can also be used to make periodic checks. One simple example of a passive failure detection system is the one that monitors the temperature of your car engine. If the engine gets too hot a warning light may come on. The detector does nothing but passively estimate the engine temperature and compare it to the maximum allowable temperature.

A major drawback with the passive approach is that failures can be masked by the operation of the system. This is true, in particular, for controlled systems. The reason for this is that the purpose of controllers, in general, is to keep the system at some equilibrium point even if the behavior of the system changes. This robustness property, which is clearly desired in control systems, tends to mask abnormal behaviors of the system. This makes the task of failure detection difficult, particularly if it is desired to detect failures that degrade performance. By the time the controller can no longer compensate for the failure, the situation may have become more severe, with much more serious consequences. An example of this effect is the well known fact that it is harder for a driver to detect an underinflated or flat front tire in a car that is equipped with power steering. This trade-off between detection performance and controller robustness has been noted in the literature and has led to the study of the integrated design of controller and detector. See, for example, [60, 80]. A more dramatic example occurred in 1987 when a pilot flying an F-117 Nighthawk, which is the twin-tailed aircraft known as the stealth fighter, encountered bad weather during a training mission. He lost one of his tail assemblies but proceeded back and landed his plane without ever knowing that he was missing part of the tail. The robustness of the control system in this case had the beneficial effect of enabling the pilot to return safely. However, it also had the effect that the pilot did not realize that his aircraft had reduced capability and that the plane would not have performed correctly if a high-speed maneuver was required.

But the problem of masking of failures by the system operation is not limited to controlled systems. Some failures may simply remain hidden under certain operating conditions and show up only under special circumstances. For example, a failure in the brake system of a truck is very difficult to detect as long as the truck is cruising along the road on level ground. It is for this reason that on many roads, just before steep downhill stretches, there are signs asking truck drivers to test their brakes. A driver who disregarded these signs would find out about a brake failure only when he needed to brake going downhill. That is, too late to avoid running off the road or having an accident.

An alternative to passive detection, which could avoid the problem of failures being masked by system operation, is *active detection*. The active approach to failure detection consists in acting upon the system on a periodic basis or at critical times using a test signal in order to detect abnormal behaviors which would otherwise remain undetected during normal operation.

The detector in an active approach can act either by taking over the usual inputs of the system or through a special input channel. An example of using the existing input channels is testing the brakes by stepping on the brake pedal. One class of applications using special channels is when the system involves a collection of pipes or tubes and a fluid or gas is being pumped through the pipes. A substance is injected into the flow in order to determine flow characteristics and pipe geometry. A specific example is the administration of dyes using intravenous injection when conducting certain medical imaging studies. The imaging study lasts for a certain period of time. Since many people react to the dyes, it is desired to keep both the total amount of dye and the rate at which the dye is injected small, consistent with getting sufficient additional resolution.

The active detection problem has been less studied than the passive detection problem. The idea of injecting a signal into the system for identification purposes, that is, to determine the values of various physical parameters, has been widely used and is a fundamental part of engineering design. But the use of extra input signals specifically in the context of failure detection was introduced by Zhang [90] and later developed by Kerestecioğlu and Zarrop [49, 48, 50]. These works served as part of the initial motivation for our study. However, these authors consider the problem in a very different context, which in turn leads to mathematical problems that are very different from those we consider in this book. Accordingly, we have chosen not to review them here.

There are major efforts under way in the aerospace and industrial areas to try to get more extended and more autonomous operation of everything from space vehicles to ships at sea. Regular and extensive maintenance is being replaced by less frequently scheduled maintance and smaller crews. This is to be accomplished by large numbers of sensors and increased software to enable the use of "condition-based maintenance." Active failure detection will play an increasingly important role both in the primary system and in back-up systems to be used in the case of sensor failures.

Before beginning the careful development in chapter 2, we will elaborate a little more on the ideas we have just introduced in this section.

1.2 FAILURE DETECTION

Failure detection consists of deciding whether a system is functioning properly or has failed. This decision process is based on measurements of some of the inputs and outputs of the system. In most cases, these measurements are obtained from sensors placed on or around the system and from knowledge of some of the control inputs.

Given the measurements, the problem is then to decide if the measurement data are consistent with "normal functioning" of the system.

There are two ways of approaching this problem. One is to define a set of input-output trajectories consistent with normal operation of the system. These trajectories are sometimes called the *behavior* of the system. Failure detection then becomes some type of set inclusion test. The other approach consists of assigning a probability to each trajectory and then using probabilistic arguments to build a test. But even in the first approach the notion of probability is often present because without it there is in general no way of defining a set of normal trajectories without being overly conservative. What we often do is to exclude "unlikely" trajectories from this set by selecting an a priori threshold on the likelihood of the trajectories that we admit into the set. Indeed, under the assumption that the observations result from the model, an abnormal behavior is nothing but an unlikely event. There are numerous variations on these two approaches. The choice of which to use is influenced by the nature of the problem.

In *model-based* failure detection, the normal (nonfaulty) behavior of the system is characterized using a mathematical model, which can be deterministic or stochastic. This model then defines an *analytical redundancy* between the inputs and the outputs of the system which can be tested for failure detection purposes. The use of analytical redundancy in the field of failure detection originated with the works of Beard [5] and Jones [46], and of Mehra and Peschon [59] in the stochastic setting. A good picture of the early developments is given in the survey by Willsky [83].

This book develops a model-based approach for several classes of models which consist of differential, difference, and algebraic equations. Accurate models of real physical systems can become quite complex, involving a variety of mathematical objects including partial differential equations (PDEs). However, it is intrinsic to the problem of failure detection that the tests have to be carried out either in real time or close to it. The whole point of failure detection is to determine that a failure has occurred in time to carry out some type of remedial action. Thus, while some calculations, such as design of the detector, can be done off-line, the actual detection test must usually be able to be carried out on-line. To accomplish this, the model used for failure detection purposes in most cases is linear. Nonlinear effects are often included in the noise and model uncertainty effects. In addition, either the models are finite dimensional, or in the case of differential equations, the dimension of the state space is finite. This often requires some type of approximation process if the true underlying models are infinite dimensional. We illustrate this in chapter 4 when we consider differential equation models that include delays.

In the simplest case of a model given by a dynamical system, we would have a deterministic system with a known initial condition. In this case the set of normal behaviors would be reduced to a single trajectory. The failure detection test in this case would be very simple, but this situation does not correspond to real-life cases encountered in practice.

A first step in building more realistic model-based normal behavior sets is to consider that the initial condition of the model, characterizing the behavior set, is unknown. To illustrate, suppose that for a continuous-time differential dynamical system, all the information we have is summarized in the system equations

$$\dot{x} = Ax + Bu, \tag{1.2.1a}$$

$$y = Cx + Du, \tag{1.2.1b}$$

where u and y are, respectively, the measured input and output of the system, x is the state, and A, B, C, D are considered known. In the corresponding discrete-time case, the system equations would be

$$x(k+1) = Ax(k) + Bu(k), \tag{1.2.2a}$$

$$y(k) = Cx(k) + Du(k). \tag{1.2.2b}$$

This way of introducing uncertainty in the model is reasonable because the initial condition of the model usually corresponds to the internal state of the system, which is not directly measured. This approach also leads to simple tests for the inclusion of observed data in the set of normal behaviors. For example, in the discrete-time case, the set of input-outputs satisfying (1.2.2) can be characterized in terms of a set of linear dynamical equations involving only measured quantities u and y. To illustrate, suppose we denote the time shift operator by z. Then the system (1.2.2) can be expressed as follows (using the shift operator is equivalent to taking the z transform of the system, which is the discrete analogue of taking a Laplace transform of a continuous-time system):

$$\begin{pmatrix} -zI + A \\ C \end{pmatrix} x = \begin{pmatrix} -B & 0 \\ -D & -I \end{pmatrix} \begin{pmatrix} u \\ y \end{pmatrix}. \tag{1.2.3}$$

Thus if $H(z)$ is any polynomial matrix in z such that

$$H(z) \begin{pmatrix} -zI + A \\ C \end{pmatrix} = 0, \tag{1.2.4}$$

and the matrix-valued polynomial $G(z)$ is defined by

$$G(z) = H(z) \begin{pmatrix} B & 0 \\ D & -I \end{pmatrix}, \tag{1.2.5}$$

then the relation

$$G(z) \begin{pmatrix} u \\ y \end{pmatrix} = 0 \tag{1.2.6}$$

must hold. The analytical redundancy relations (1.2.6) are also called *parity checks*. They are easy to test at every time step, but they are not unique. In the actual implementation of this approach, the choice of the test is made so as to account for unmodeled model uncertainties, and the result is tested against a threshold. See [53] for one such approach.

The other main method for testing the inclusion of observed data in the set of normal behaviors is to use an *observer*. Observers play a fundamental role in control theory. Given a dynamical system, an observer is a second dynamical system which takes the inputs and outputs of the first system as inputs and whose state (or output) asymptotically approaches the state (part of the state) of the first system. This convergence takes place independently of the initial conditions of the original system and the observer system. If the model is assumed to be perfectly known and only the initial condition is unknown, the observer residual, which is the difference between the measured output and its prediction based on past measurements, converges exponentially to zero. Thus this residual can be used for failure detection testing. Such tests are called observer based. In the continuous-time setting, for example, the observer-based residual generator for system (1.2.1) can be constructed as follows:

$$\begin{aligned} \dot{\hat{x}} &= A\hat{x} + Bu - L(y - C\hat{x}), \quad \hat{x}(0) = 0, \\ r &= y - C\hat{x} - Du, \end{aligned}$$

where r denotes the residual and L is a matrix chosen such that $A + LC$ is Hurwitz (all eigenvalues have a negative real part) to assure the convergence of r to zero.

In practice, the residual is tested against a threshold to account for uncertainties. The freedom in the choice of the observer is used for robustness purposes. One such method can be found in [28].

It turns out that the observer-based detection and parity check methods, which historically have been developed independently, are in fact very similar. As is shown in [56], in the discrete-time case, the parity check test is equivalent to an observer-based test where the observer is taken to be deadbeat (L is chosen so that $A + LC$ is nilpotent).

Assuming that all the uncertainties are concentrated in the initial condition alone does not correspond to the reality of most applications. Indeed, the application of the parity check and observer-based methods requires a delicate robustification stage which is constructed using ad hoc methods. In fact, it is the combination of the model and the thresholding test that defines the set of normal behaviors of the system, and not just the dynamic model.

In a model-based approach, rather than focusing on the use of thresholds, it is more natural, and also more informative from a theoretical point of view, to capture the uncertainties using the model itself. A first step in this direction would be to consider additive noise. For example, in the continuous-time setting a model with additive noise might take the form

$$\dot{x} = Ax + Bu + M\nu, \tag{1.2.7a}$$

$$y = Cx + Du + N\nu, \tag{1.2.7b}$$

where ν represents the additive input noise.

In the deterministic setting, the simplest type of noise would correspond to a completely unknown input ν. In this case the set of normal behaviors would be characterized again in terms of a linear model but with both the initial condition and one or more inputs unknown. The test of inclusion for this set of behaviors turns out to be straightforward to build. Both a residual-based method (using unknown input observers or the eigenstructure assignment method developed by Patton and coworkers; see chapter 4 of [26]) and parity check tests (constructed for a reduced model) can be used.

However, the use of unknown additive noise and an unknown initial condition for the model can lead to a very conservative test. That is, it can lead to considering a set of normal behaviors that is much larger than the actual set of normal behaviors. The reason for this is that we usually have some information on the state and on the inputs and outputs of the system. It is rare in a physical system to have a parameter that can take any value a priori. One way to exploit information about variables is to use stochastic modeling and consider the initial condition and the inputs to be random variables. This approach does not directly define a set of normal behaviors. Rather, it associates a probability with each trajectory. The failure detection tests applies a threshold to this probability to decide whether or not a failure has occurred. A set of normal behaviors is implicitly defined as those of high enough probability to exceed the threshold.

Stochastic modeling allows the designer to incorporate realistic information concerning additive noises, in terms not only of the mean and variance, but also of the frequency spectrum. The construction of failure detection tests for such models was introduced in [59], where a complete solution was given based on the Kalman filter. In this approach, the decision is based on statistical tests performed on the Kalman filter's innovation, which in the absence of a fault must be of zero mean, white, and have known variance. In a sense, this method is again observer based because a Kalman filter is just a special case of an observer where the observer gain is tuned to satisfy certain stochastic properties. The innovation here plays the role of the residual in the observer-based method. The difference is that the thresholding test is not ad hoc but corresponds to a statistical property.

Considering a model that contains both unknown inputs and inputs modeled by stochastic processes allows for both robustness and performance, as has been shown in [81], where Kalman filters for the descriptor system were used to deal with unknown inputs. A unified theory of residual generation for models containing unknown inputs and inputs modeled by stochastic processes was developed in [61], where a complete solution was also presented.

Even though it is possible to model many system uncertainties in terms of additive noises, this approach usually leads to conservative designs. More realistic and less conservative models can be constructed by allowing model uncertainty and bounded noise terms in the model. This has been studied in recent years in the context of estimation theory, where the objective is to construct an estimate of the state of the system based on input-output measurements, and is commonly referred to as robust filtering. Robust filtering originated with the works of Bertsekas and Rhodes [8], Schweppe [78, 79], and Kurzhanski and Valyi [52]. It was developed for estimating the states of dynamical models corrupted by unknown but bounded disturbances and noises. In [8] the energy bound is studied for characterization of the uncertainties. For system (1.2.7) on $[0, T]$, for example, the bound would take the form

$$(x(0) - x_0)^T P_0^{-1}(x(0) - x_0) + \int_0^T |\nu|^2 dt < d. \tag{1.2.8}$$

Causal estimation is estimation based on only current and past information. It turns out that the solution to the causal estimation problem for (1.2.7) with the noise bound (1.2.8) is intimately connected to the Kalman filter for this system, where ν is interpreted as a unit-variance zero-mean white noise process. In fact, the estimate and the associated error covariance matrix given by the Kalman filter parameterize an ellipsoid which gives the set of x's consistent with the inequality (1.2.8) in the deterministic problem. The connection between these two problems should not be a total surprise, because in the stochastic setting the left hand side of the inequality (1.2.8) can be interpreted as the negative of the log-likelihood of the noise process.

The study of other types of constraints, for example, instantaneous bounds on the norm of $\nu(t)$, has proved to be more difficult, and only conservative solutions have been proposed. The problem is that pointwise bounds lead to sets that have "corners" and are not even strictly convex. Thus the set of x corresponding to consistent state trajectories is not strictly convex, and the existing analysis requires that bounding ellipsoids must be used. See [78].

In all of the work mentioned so far, it was assumed that the nominal model was perturbed only by additive noises. However, in many real systems the equations themselves are not known exactly or may change with time. There is always some variability in components. In recent years Petersen and Savkin [72, 76], Sayed [77], and El Ghaoui *et al*. [40, 41, 42] have proposed various methods for handling the case where the true model includes both perturbations to the model dynamics and additive unknown signals. They consider models of the form

$$\dot{x} = (A + \delta A)x + (B + \delta B)u + M\nu, \tag{1.2.9a}$$

$$y = (C + \delta C)x + (D + \delta D)u + N\nu, \tag{1.2.9b}$$

where ν represents the additive noise and $(\delta A, \delta B, \delta C, \delta D)$ represent the uncertainties of the system matrices. In real applications different system entries are subject to different amounts of perturbation and some, such as zeros, may not undergo any perturbation at all. Thus it is reasonable to assume that $\delta A = J_1 \Gamma K_1$ where J_1, K_1 provide structure and scaling to the perturbation and Γ is some sort of arbitrary but bounded perturbation. For technical reasons, it turns out to be useful to replace Γ by $\Delta(I - H\Delta)^{-1}$ to give

$$\begin{pmatrix} \delta A & \delta B \\ \delta C & \delta D \end{pmatrix} = \begin{pmatrix} J_1 \\ J_2 \end{pmatrix} \Delta(I - H\Delta)^{-1} \begin{pmatrix} K_1 & K_2 \end{pmatrix},$$

because under certain types of bounding on Δ this leads to a quadratic problem.

In many respects the failure detection problem is close to the estimation problem, and many of the techniques developed for estimation have been used for failure detection. For example, H_∞ filtering has been used in [31, 33, 54], a game theoretic approach in [28], and a set-valued estimation approach by Savkin and Petersen [75] and Petersen and McFarlane [71]. The uncertainty model adopted in this book is closely related to that of [75, 71], which is also sometimes called model validation. This uncertainty model allows for fairly realistic modeling of uncertainties with reasonable conservatism and leads to failure detection tests of acceptable complexity.

Another important method of model-based failure detection uses identification. In the case of parameter identification, the failure detection test is based on the distance between the identified parameter and a nominal parameter corresponding to normal operation of the system. Parameter identification again can be considered as an estimation problem but here the dependence on the parameter is no longer linear and specific estimation methods must be used. We do not use these methods in this book, nor do we use any of the subspace identification techniques recently introduced in the literature; see, for example, [2].

Finally, there have been attempts at using nonlinear models for the purpose of failure detection. Up until now, there have not been any systematic design procedures proposed which could be used efficiently in practice for all nonlinear systems. See the survey paper [35] by Frank. In the special case of bilinear systems, the study can be carried somewhat further; see, for example, [51, 87, 85, 89, 88]. In this book, we initially avoid using nonlinear models. When we have nonlinearities in the system, we try to model them as uncertainties. This approach can be quite conservative when we are dealing with large nonlinearities but it allows us to use powerful tools from linear algebra in applying our methodology. In chapter 4, we introduce an optimization-based method which does allow us to consider some special classes of nonlinear systems.

Before continuing, it is important to comment on our use of the word *conservative*. In much of the failure detection literature an approach being conservative means that either it will avoid making false detection of a failure at the risk of missing some failures or it will avoid missing failures at the risk of false detection of failure (a false alarm). When the active approach presented here is appropriate, that is, the noise bounds and other assumptions hold, the approach provides guaranteed detection. There are no false alarms and no failures are missed. Here the conservatism arises in the size of the auxiliary test signal. When the set of allowable disturbances is increased on applying the theory and algorithms, the computed auxiliary signal may be larger than is required by the original set of disturbances.

1.3 FAILURE IDENTIFICATION

The failure identification problem goes a step beyond failure detection. It concerns not only deciding whether or not a failure has occurred but also, if a failure occurs, determining what kind of failure has occurred. This requires modeling the behaviors of the system input-output trajectories for every possible failure, in addition to the behavior associated with the normal (nonfaulty) system.

In the model-based approach, very often failed systems are modeled in the same way as the normal system but with additional additive inputs representing the effects of various faults. These inputs are assumed either arbitrary and a priori unknown, or constant with known or unknown values, or stochastic. This type of model was already considered in the work of Beard [5].

Even in the case of a single failure, failure detection can be improved by taking into account the model of the failure. For example, by assuming that a failure corresponds to an additive constant input, Willsky and Jones [84] have developed an interesting detection test based on the generalized likelihood ratio or GLR method.

Even when different failures are modeled as unknown inputs, it may be possible under some conditions to identify which failure has occurred if they enter the system in different directions. A nice geometrical theory has been developed by Massoumnia [57]. See also [58]. This characterization is useful in both the deterministic and the stochastic settings. See also [61].

In practice, except for certain types of incipient faults, which are those that appear gradually, the failure cannot be modeled efficiently as an additive input. Consider for example a sensor failure in a system. In the absence of failure the output measurement might be given by

$$y = cx_i + \nu, \tag{1.3.1}$$

where ν represents the measurement noise, x_i is a component of the state measured by the sensor, c is the sensor gain, which is assumed known, and y is the output of the sensor. When the sensor fails, the sensor gain becomes zero. Thus (1.3.1) becomes

$$y = \nu. \tag{1.3.2}$$

The way unknown input failure modeling could be used in this context would be to have

$$y = cx_i + \nu + f \tag{1.3.3}$$

represent the behavior of the failed system, where f is assumed to be totally arbitrary. This works out because we can take

$$f = -cx_i, \tag{1.3.4}$$

which makes (1.3.3) equivalent to (1.3.2). Note, however, that this is a conservative approach, because if we have any information about ν (for example, boundedness or statistical properties) (1.3.2) and (1.3.3) are not equivalent. In (1.3.2) we have some information about y, whereas in (1.3.3) y is totally arbitrary.

Another approach to failure identification is to consider a separate model associated with each failure. The models need not even have the same state dimensions. This multimodel approach allows realistic modeling of failures in many applications. For example, it would perfectly capture the situation in the sensor failure problem discussed above. The multimodel approach has been used in particular in the GLR context by Willsky; see, for example, chapter 2 of [3].

In this book, we use the multimodel approach for modeling system failures.

1.4 ACTIVE APPROACH VERSUS PASSIVE APPROACH

There are basically two approaches to failure detection and isolation. One is the passive approach, where the detector monitors input-outputs of the system and decides whether a failure has occurred, and if possible of what kind. A passive approach is used for continuous monitoring and when the detector has no way of acting upon the system. The other approach is the active approach, where the detector acts upon the system on a periodic basis or at critical times, using a test signal called the *auxiliary signal*, over a test period, in order to exhibit possible abnormal behaviors. The decision on whether or not the system has failed, and the determination of the type of failure, are made at the end of this period. Sometimes the tests permit early detection, that is, the decisions are made before the end of the test period. The major theme of this book is the design of auxiliary signals for failure detection.

The structure of the active failure detection method considered here is illustrated in figure 1.4.1. The auxiliary signal v injected into the system to facilitate detection is part (or all) of the system input used by the detector for the period of testing. The signal u denotes the remaining inputs measured on-line, just as the outputs y are measured on-line. In some applications the time trajectory of u may be known in advance, but in general the information regarding u is obtained through sensor data in the same way that it is done for the output y.

In order to simplify this introductory discussion, suppose that there is only one possible type of failure. Then in the multimodel approach we have two sets of input-output behaviors to consider and hence two models. The set $\mathcal{A}_0(v)$ is the set of input-outputs $\{u, y\}$ associated with model 0 of normal behavior. The set $\mathcal{A}_1(v)$ is the set of input-outputs asociated with model 1 of behavior when failure occurs. These sets represent possible/likely input-output trajectories for each model. Note that while model 0 and model 1 can differ greatly in size and complexity, the variables u and y have the same dimension in both models.

The problem of auxiliary signal design for guaranteed failure detection is to find a

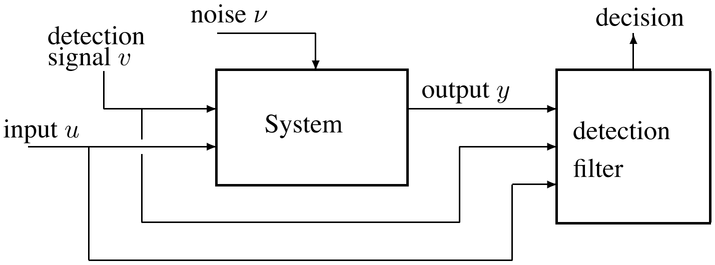

Figure 1.4.1 Active failure detection.

reasonable v such that

$$\mathcal{A}_0(v) \cap \mathcal{A}_1(v) = \emptyset.$$

That is, any observed pair $\{u, y\}$ must come from only one of the two models. Here "reasonable v" means a v that does not perturb the normal operation of the system too much during the test period. This means, in general, a v of small energy applied over a short test period. However, depending on the application, "reasonable" can imply more complicated criteria.

Figures 1.4.2 through 1.4.4 should give a clear picture of the situation. When v is zero, the two sets $\mathcal{A}_0(v)$ and $\mathcal{A}_1(v)$ usually overlap. In particular, when the two sets are associated with linear models, they both contain the origin, as illustrated in figure 1.4.2.

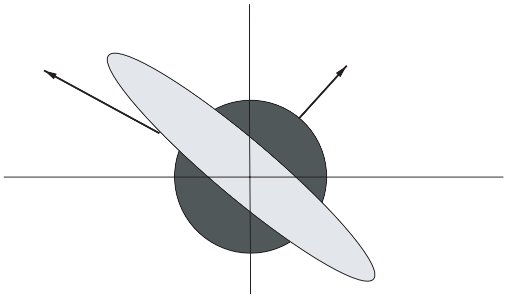

Figure 1.4.2 Auxiliary signal equals zero.

When a signal v other than zero is used, the two sets are moved somewhat apart as illustrated in figure 1.4.3. Increasing the size of v moves the sets further, and at some point they become disjoint as in figure 1.4.4. At this point, we have an auxiliary signal which can be used for guaranteed failure detection. We call such an auxiliary signal *proper*.

The main objective of this book is to present a methodology for the construction of *optimal* proper auxiliary signals in the multimodel context. The auxiliary signal design problem in the multimodel setting has been studied in the past, in particular by Zhang [90] and by Kerestecioğlu and Zarrop [49, 50]. See also the book [48].

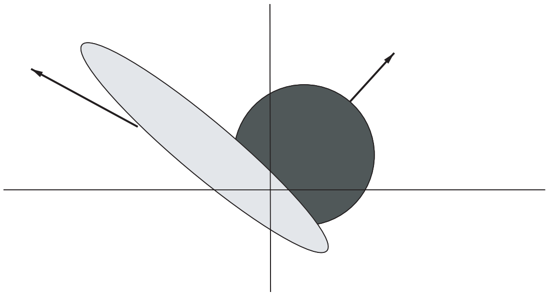

Figure 1.4.3 Small auxiliary signal.

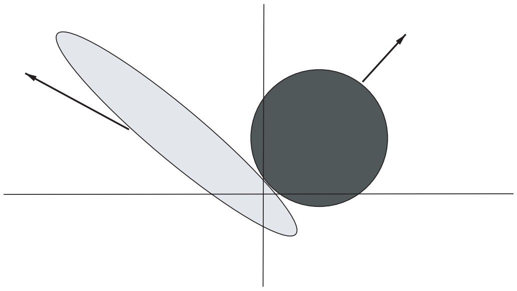

Figure 1.4.4 Proper auxiliary signal.

Our approach, however, differs in many ways from theirs, which was directly motivated by the work on auxiliary signal design for identification purposes and aimed at constructing stationary stochastic auxiliary signals.

We do not restrict our attention to the stationary case; *au contraire*, we focus on short detection intervals where the effects of initial conditions cannot be ignored. In our scenario, the test is to be applied at specific times on a short time interval, perturbing system operation as little as possible, and guaranteeing that if the system has failed, at the end of the test period, the detector discovers the failure.

Our approach is fundamentally deterministic and uses the set membership approach (even if the sets can be based on likelihood thresholding) and we seek guaranteed detectability, a concept that was first introduced in [62]. The work presented in this book follows some of the ideas presented in [62]. However, the models used here are different, thus yielding a very different mathematical theory.

We consider models with different types of uncertainties and in particular realistic model uncertainties, which are often encountered in practice. We have a system theoretic approach and use well established tools such as Riccati equations that allow us to handle very large multivariable systems. The methodology we develop for the construction of the optimal auxiliary signal and its associated test can be implemented easily in computational environments such as Scilab [16, 25] and MATLAB. Moreover, the on-line detection test that we obtain is similar to some existing tests based on Kalman filters and is easy to implement in real time.

This book represents the culmination of several years of research. During this time the results have naturally evolved and have become more general and more powerful. Some of this development and some of the material in this book has appeared in the papers [62, 66, 21, 22, 20, 67, 63, 17, 18, 65, 64, 24].

1.5 OUTLINE OF THE BOOK

In chapter 2, we present the type of models we consider and, in particular, the way that uncertainty can be accounted for in our approach. We also present on-line detection tests associated with these models and discuss their implementation.

The main problem we consider in this book is the construction of the auxiliary signal and its use in failure detection. This problem is considered in chapter 3, where we develop a complete theory and discuss implementation issues. Chapter 3 provides a careful development of the two-model case. The approach of chapter 3 can be applied to more than two models by performing a sequence of two-model tests. When there are more than two possibilities, care must be taken in interpreting each of the sequential tests. This is discussed in chapter 4.

Chapter 4 presents a different approach to the problem of auxiliary signal design. This method is based on numerical optimization. It is less efficient in dealing with large systems, which can be handled by the method presented in chapter 3, but it allows for the consideration of more general models and more complicated constraints. In chapter 3, only linear finite-dimensional models subject to a particular class of uncertainties are considered. In chapter 4, we allow for certain nonlinear models, delays, and more general types of uncertainties. In addition, in chapter 4, we consider an arbitrary finite number of failure models and show how to construct minimal energy proper signals to test for several different failures at once. These simultaneous tests are much more efficient than separate sequential tests. The approach of chapter 4 can also be used for a variety of constrained problems. For example, it is shown how to construct the auxiliary signal of smallest norm for the important case where the auxiliary signal also satisfies a pointwise bound.

Chapter 5 briefly discusses some of the open questions and problem areas for auxiliary signal design.

In chapter 6 we give a collection of programs written in the Scilab language that carry out the algorithms of chapter 3. As described in chapter 6, these are not to be considered as polished industrial grade software. They have been tested on a number of examples.