Preface

One of the most common problems in the practice of engineering reliability is that of selecting a particular system design among the several options available for achieving some particular performance goal. Often, the goal is a long-lived system. In the case of repairable systems, the goal might be identifying a system design that has as little downtime as possible. The best performance per unit cost is another worthy potential goal. The purpose of this monograph is to provide some guidance on how problems of this type might be formulated and solved. Our approach relies on the relatively new notion of "system signatures." We will introduce the concept here in the context of the well established theory of coherent systems and will seek to provide convincing evidence that the recommended approach to the problems mentioned above is efficacious.

I must admit that I stumbled on the notion of system signatures quite by accident. While visiting the University of Washington in 1982-3, on sabbatical leave from the University of California, Davis, I was, among other things, working on a number of research problems in Reliability Theory. One problem involved trying to understand more deeply the concept of "closure" under the formation of coherent systems. My being in Seattle, where much of the seminal work in this area had been done in the 60s by Birnbaum, Esary, Marshall, Proschan and Saunders, may have subconsciously driven me toward this topic. What was well known at the time was that a (coherent) system in components with increasing failure rates (IFR) is not "closed," that is, is not necessarily IFR, but that the larger class of systems in components with the IFRA (increasing failure rate average) property is closed, that is, these systems are themselves IFRA.

The question that interested me at the time was more or less halfway between these two results – what could one say about the class of systems that did enjoy the IFR closure property? The literature contained a partial answer: k-out-of-n systems in i.i.d. IFR components were known to be IFR.

Expanding upon this result seemed to require a new tool. My immediate goal was to find ways of identifying a direct connection between the failure rate of a system and the common failure rate of its components. The representation theorem in Samaniego [61] expressing the system's failure rate as a multiple of the component failure rate turned out to be the tool that permitted the complete characterization of systems that will be IFR when their components are i.i.d. IFR. That representation had the form $r_T(t) = h(\mathbf{s}, F)r(t)$, where $\mathbf{s}$ is the signature of the system, a vector essentially capturing the influence of the system design on the system's failure rate, and F and r are the underlying distribution and failure rate of the components. So the birth of the signature idea dates back nearly 25 years. In the preceding paragraph, I referred to this notion as "relatively new." In the grand scheme of things, 25 years goes by in a flash, so from that point of view, one could say that signatures are relatively new. My intent, however, was to acknowledge that the notion was not recognized as broadly useful until its properties were carefully studied in their own right. In Kochar, Mukerjee and Samaniego [51], some new preservation theorems were proven, and the comparison of system lifetimes via the properties of their signatures was shown to be feasible and fruitful. These results revealed the potential power and breadth of the concept. Both of the themes referred to above will be presented in detail in Chapter 4 of the present work.

This monograph consists of six substantive chapters on signatures and their applications, together with an opening chapter introducing the topic and a closing chapter summarizing the state of research on signatures and sharing some of my thoughts on future theoretical developments and potential applications of interest. Most of the existing theory on structural reliability is based on Birnbaum, Esary and Saunders' [9] seminal work on multicomponent, two-state systems. As in Birnbaum, et al. [9], our emphasis here will be on binary systems (which are either in a functioning (1) or failed (0) state). My work with the notion of signatures began with the publication of Samaniego [61], a paper entitled "On the Closure of the IFR Class under the Formation of Coherent Systems." In that paper, the signature vector was defined for a coherent system with components having i.i.d. lifetimes with common distribution F. In brief, the signature of such a system in n components is an n-dimensional probability vector $\mathbf{s}$ that is the distribution of the index of the ordered component failure time that corresponds to system failure.

Especially in the last decade, the broad applicability of system signatures has become apparent, and their utility in the comparison of coherent systems and communication networks has been more firmly established. Most recently, we have found that the tool can facilitate the reformulation of heretofore analytically intractable discrete optimization problems in the area of Reliability Economics, providing a mechanism which can make the analytical treatment of these problems feasible. My purpose here is to present a useful overview of work to date on the properties and applications of system signatures with a view toward opening up new potential applications. Some new results will be combined with work available in the literature. The present work is intended to be both comprehensive and unifying. If I succeed in accomplishing these goals, I am convinced that both the scope and depth of application of system signatures in reliability will be substantially enhanced in future work.

I wish to thank my students and coworkers who have participated in many of my studies in this general problem area. These include, in alphabetical order, Debasisis Bhattacharya, Henry Block, Philip Boland, Michael Dugas, Subhash Kochar, Myles Hollander, Michael McAssey, Hari Mukerjee, Moshe Shaked and Eric Vestrup. Many of the ideas presented in this monograph came to life in the course of my conversations with these collaborators, and I express to each of them my deep appreciation for the many stimulating discussions we have had and for each of their contributions to the theory and application of the signature idea. I hope that this attempt to present a coherent and unified version of our collective results does justice to both the work and to these key contributors.

I express my special appreciation to Dr. Robert Launer and to Dr. Harry Chang of the Army Research Office for their sustained support of this project. The present work combines new results developed under ARO support with a reworking and new presentation of years of work on signature-related ideas. My recent research in this area, as well as my work on the present project, was supported by ARO grants ARO19-02-1-0377 and WN11NF05-1-0118. It is also a pleasure to acknowledge other agencies that have supported this work over the years, including the Air Force Office of Scientific Research, The Ford Foundation, The National Security Agency and The National Science Foundation. This monograph was written during a sabbatical leave funded by the University of California, Davis, and I gratefully acknowledge that support as being a critical element in the completion of this project. I would like to thank Brad Efron for inviting me to spend a portion of that leave in the Statistics Department at Stanford University. This provided me with a stimulating yet quiet place to hide out while getting this project off the ground. Finally, I thank the students in my graduate reliability course in Winter Quarter, 2007, for reading the penultimate draft of this monograph and suggesting many improvements. My thanks to Ying Chen, Tammy Greasby, Yolanda Hagar, Jung Won Hyun, Michelle Norris, Clayton Schupp, and Li Zhu.

Francisco J. Samaniego
University of California, Davis
March, 2007

2

Background on Coherent Systems

2.1 Basic Ideas

We will use the term "system" quite freely and regularly, even though it will remain an undefined term throughout this monograph. As we all have some experience with engineered "systems," our use of the term should cause no confusion. Informally, we can think of a system as consisting of a collection of "components," basic constituents which are connected in some fashion to create the whole. We might consider a radio, an automobile, a computer or a cell phone as concrete examples of systems in common use. The main characteristic of our use of the term is that a system works or fails to work as a function of the working or failure of its components. While there are various ways to formalize the notion of a system being partially functioning (for example, a car could technically be driven for a few miles with a flat tire), we will follow the convention established by Birnbaum et al. [9] and consider a system to be either working or failed at any given point in time. To quantify this fact, we assign a 1 to the event that the system works and a 0 to the event that the system fails. The same can be said of each component.

For a system with n components, this idea gives rise to the notion of a state vector, that is, a vector $\mathbf{x} = (x_1, x_2, \ldots, x_n) \in \{0,\ 1\}^n$, where for each i, $x_i = 1$ if the ith component is working and $x_i = 0$ if it is not working. We will be interested in whether or not the system is working when the components are in a specific state. A mapping called the structure function provides the desired link.

Definition 2.1. *Consider the space $\{0,\ 1\}^n$ of all possible state vectors for an n-component system. The* structure function $\varphi : \{0,\ 1\}^n \to \{0,\ 1\}$ *is a mapping that associates those state vectors $\mathbf{x}$ for which the system works with the value 1 and those state vectors $\mathbf{x}$ for which the system fails with the value 0.*

Some examples will help make this concept clear. Most people are familiar with two particular systems that arise frequently in reliability: the series system and the parallel system. The first works only if every component is working, while the second works as long as at least one component is working. The structure function for an n-component series system is given by

$$\varphi(\mathbf{x}) = \prod_{i=1}^{n} x_i \ , \tag{2.1}$$

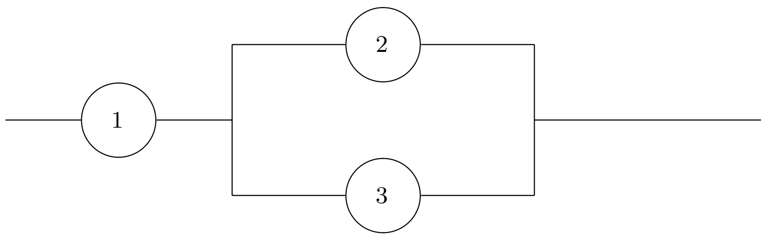
while for a parallel system, we have

$$\varphi(\mathbf{x}) = 1 - \prod_{i=1}^{n}(1 - x_i) \ . \tag{2.2}$$

The two systems above are extreme examples of an important class of systems called "k-out-of-n systems." This label has a bit of ambiguity to it, as it could represent systems that fail upon the kth component failure, but it might also represent systems that work as long as at least k components are working. For that reason, the former are often called k-out-of-n:F systems and the latter are called k-out-of-n:G systems, the "F" standing for the k failed components that ensure system failure and the "G" standing for the k good components that ensure that the system functions. Throughout this monograph, I will make reference to k-out-of-n systems without using the qualifiers F or G. In all instances, I will be referring to what I have called above the k-out-of-n:F systems. Thus, a series system is a 1-out-of-n system and a parallel system is an n-out-of-n system. For hybrid systems such as the system pictured in Figure 2.1, the structure functions will have elements of both of the functions above, that is,

$$\varphi(\mathbf{x}) = x_1[1 - (1 - x_2)(1 - x_3)] \ . \tag{2.3}$$

Fig. 2.1. A series-parallel system in three components

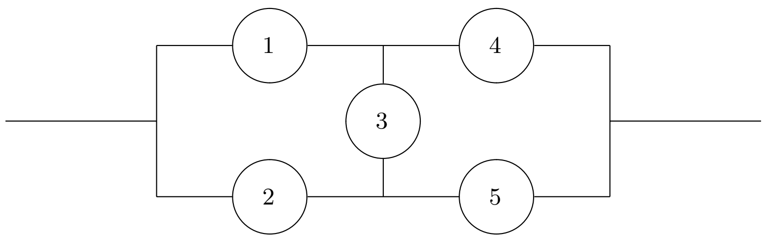


The structure function for a k-out-of-n system is most easily represented as:

$$\varphi(\mathbf{x}) = 0 \text{ if } \sum_{i=1}^{n} x_i \leq n-k \text{ and } \varphi(\mathbf{x}) = 1 \text{ if } \sum_{i=1}^{n} x_i \geq n-k+1 \,. \tag{2.4}$$

A fourth example is the five-component bridge system displayed in Figure 2.2 below. Its structure function $\varphi(\mathbf{x})$, which follows the figure, is substantially more complex than those that precede it.

Fig. 2.2. A bridge system in 5 components



The structure function of the bridge system shown above is given by

$$\begin{aligned}\varphi(\mathbf{x}) = {} & x_1x_4 + x_2x_5 + x_1x_3x_5 + x_2x_3x_4 - x_1x_2x_3x_4 - x_1x_2x_3x_5 \\ & - x_1x_3x_4x_5 - x_1x_2x_4x_5 - x_2x_3x_4x_5 + 2x_1x_2x_3x_4x_5\end{aligned} \tag{2.5}$$

The reader might rightly wonder how the latter structure function was obtained. I placed it in the text at this point because it provides me with the opportunity to make some useful comments. Regarding how the function was obtained, one could use a rational, orderly process (resembling the inclusion-exclusion principle which is discussed later in this chapter) to account first for the ways in which the system will necessarily work – e.g., if components 1 and 4 are working or if components 1, 3 and 5 are working, etc. – and then compensating for their two-way, three-way and four-way intersections. We will introduce shortly an approach that results in (2.5) in a conceptually and practically simpler way. The fact remains that the structure function of a five-component bridge system is a complex object, no matter how you get it. You can imagine struggling with the structure function of a system of order 20. Dealing with the structure functions of systems of order 100 seems almost imponderable. In situations where systems are being compared or where an optimal system is sought relative to some fixed criterion, it is clear that the indexing of the class of systems by their structure functions complicates rather than simplifies the problem. A further complication is that any relabeling of a system's components gives rise to a new function that is equivalent to the original $\varphi(\mathbf{x})$ but looks different. One would have to inspect the two functions side by side to verify that the two representations have components that were in one-to-one correspondence. This, then, constitutes the first bit of motivation one might have for considering other ways of characterizing system designs. But structure functions are useful in their own way, and we will study them further before moving on. In particular, we will utilize them as a vehicle for focusing on the class of systems we will refer to as "coherent."

If an engineer is designing a system for performing a particular function, there are two basic requirements he or she would impose upon any design considered for use. First, the system would not contain any component whose functioning has absolutely no influence on whether or not the system works. If the vector $(x_1, \ldots, x_{i-1}, a, x_{i+1}, \ldots, x_n)$ represents a state vector for the n components of an arbitrary system for which $x_i = a \in \{0,\ 1\}$, then component i is said to be *irrelevant* if the system's structure function φ has the property that $\varphi(x_1, \ldots, x_{i-1}, 0, x_{i+1}, \ldots, x_n) = \varphi(x_1, \ldots, x_{i-1}, 1, x_{i+1}, \ldots, x_n)$ for all possible values of $(x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n) \in \{0,\ 1\}^{n-1}$. If an n-component system contained a component that was irrelevant, that component would be removed from the design since there is a simpler system (of order no larger than $n-1$) that can provide identical performance.

A system that actually changed from a working state to a failed state upon the replacement of a failed component with a working one would be baffling indeed. One innate feature of systems with which we are familiar is that their failure coincides with the failure of some component. If we were monitoring a system over time, we would note that, as components begin to fail, the system may continue to work for a while but, eventually, one of the components that had remained working will prove to be critical to the system's functioning, and the system will fail upon the failure of that component. Fixing a failed component might get the system working again, but in no instance would we see a system, while working on the basis of k functioning components, fail as a result of the act of replacing a failed component with a working component. We call a system *monotone* if fixing a failed component cannot make the system worse. Symbolically, a monotone system has a structure function for which $\varphi(\mathbf{x}) \leq \varphi(\mathbf{y})$ whenever $\mathbf{x} \leq \mathbf{y}$, where the latter vector inequality is understood to be applied component-wise. These two natural properties of engineered systems form the basis for the following:

Definition 2.2. *A system is said to be* coherent *if each of its components is relevant and if its structure function is monotone.*

The definition above severely restricts the number of possible functions mapping $\{0,\ 1\}^n$ into $\{0,\ 1\}$ that could play the role of a structure function for a coherent system. For example, of the 256 possible functions mapping $\{0,\ 1\}^3$ into $\{0,\ 1\}$, there are only 5 that correspond to coherent systems. Definition 2.2 requires, for example, that the structure function of every coherent system satisfy the conditions $\varphi(\mathbf{0}) = 0$ and $\varphi(\mathbf{1}) = 1$. If either of these conditions were violated, the monotonicity property would imply that every component of the system is irrelevant. In spite of the fact that our restricting attention to coherent systems substantially reduces the number of possible structure functions, it remains true that the number $Z(n)$ of coherent systems of order n grows rapidly with n. There are 2 coherent systems of order 2, 5 of order 3 and 20 of order 4. That the number grows exponentially in n is clear from the fact that, to any system of order n, one may attach a new component either in series or in parallel, immediately doubling the number of coherent systems. This implies that there are more than a billion coherent systems of order 30. Counting the exact number of coherent systems of order n is a difficult open problem, though there exist published bounds for the rate of its growth. A notable upper bound was obtained by Kleitman and Markowsky [50], who addressed the equivalent calculation known as “Dedekind's problem” in the area of enumerative combinatorics:

$$\log_2 Z(n) \leq \{1 + O[(\log_2 n)/n]\} \binom{n}{[n/2]}, \tag{2.6}$$

where $\binom{n}{k}$ is the number of combinations of n things taken k at a time and $[x]$ represents the greatest integer less than or equal to x. The size of $Z(n)$ makes the identification of optimal designs in various settings somewhat imposing. In Chapter 7, we will illustrate a methodology that circumvents this difficulty.

Before proceeding to our discussion of an alternative summary of a coherent system's design, we will introduce an additional element of the Birnbaum, Esary and Saunders framework which will play an important role in the sequel. Let us focus for a moment on coherent systems of order n. A set of components P is said to be a *path set* if the system works whenever all the components in the set P work. It is clear that the set of all n components is a path set. If A is a path set, then any set B that has A as a proper subset will be a path set as well. The path sets of special interest are those that contain no proper subsets that are also path sets. Such a set is called a *minimal path set*. We will denote the minimal path sets of a coherent system as $P_1, P_2, \ldots, P_r$. If we examine the bridge system shown in Figure 2.2, we see that the minimal path sets consist of the collection $\{\{1,4\},\{1,3,5\},\{2,5\},\{2,3,4\}\}$. This collection has two interesting properties:

(i) No minimal path set is a proper subset of any other, and
(ii) The algebraic union of all minimal path sets is the set of all the system's components.

It is possible to characterize all coherent systems of a given order n by these two properties of its minimal path sets. Since, by (ii), every component from 1 to n is a member of at least one minimal path set, the relevance of every component is guaranteed. The monotonicity of the system corresponding to a fixed collection of minimal path sets can be argued as follows. If component k is not working and the system is also not working, then the structure function φ will either remain equal to 0 or will increase to 1 when component k is replaced by a working component. On the other hand, if the system is working, there is a minimal path set P whose components are all working. Since any set of components which contains P will also be a path set, it follows that the set $\{P \cup \{k\}\}$ is a path set and that the system's structure function will remain equal to 1 when component k is replaced by a working component.

There is a natural relationship between path sets and sets of components whose failure will guarantee that the system fails. A set of components C is said to be a *cut set* if the system fails whenever all the components in the set C fail. A cut set is *minimal* if it has no proper subset that is also a cut set. The relationship between cut sets and path sets is evident from the following facts: if P is a minimal path set and A is a proper subset of P, then A^c is a cut set, and if C is a minimal cut set and B is a proper subset of C, then B^c is a path set. Neither A^c nor B^c need be minimal. The family of minimal cut sets has properties analogous to (i) and (ii) above, viz., properties (iii) and (iv) below.

(iii) No minimal cut set is a proper subset of any other, and
(iv) The algebraic union of all minimal cut sets is the set of all the system's components.

In Table 2.1, coherent systems of order 4 are identified by their minimal cut sets.

Table 2.1. Coherent Systems of Order 4

System	Minimal cut sets
1	{1}, {2}, {3}, {4}
2	{1}, {2}, {3, 4}
3	{1}, {2, 3}, {2, 4}
4	{1}, {2, 3}, {2, 4}, {3, 4}
5	{1}, {2, 3, 4}
6	{1, 2}, {1, 3}, {1, 4}
7	{1, 2}, {1, 3}, {1, 4}, {2, 3}
8	{1, 2}, {1, 3}, {1, 4}, {2, 3}, {2, 4}
9	{1, 2}, {1, 3}, {1, 4}, {2, 3}, {2, 4}, {3, 4}
10	{1, 2}, {1, 3}, {2, 4}, {3, 4}
11	{1, 2}, {2, 4}, {3, 4}
12	{1, 2}, {3, 4}
13	{1, 2}, {1, 3}, {1, 4}, {2, 3, 4}
14	{1, 2}, {1, 3}, {2, 3, 4}
15	{1, 2}, {1, 3, 4}, {2, 3, 4}
16	{1, 2}, {1, 3, 4}
17	{1, 2, 3}, {1, 2, 4}
18	{1, 2, 3}, {1, 2, 4}, {1, 3, 4}
19	{1, 2, 3}, {1, 2, 4}, {1, 3, 4}, {2, 3, 4}
20	{1, 2, 3, 4}

Some of the 20 systems in Table 2.1 are easily recognizable. Clearly, system 1 is the series system in four components, system 9 is the 2-out-of-4 system, system 19 is the 3-out-of-4 system and system 20 is the parallel system. System 10 is the system formed by connecting a pair of two-component series systems in parallel, while system 12 is formed by connecting a pair of two-component parallel systems in series. Systems 1 - 5 can be viewed as the systems one obtains by adding a new component (1) in series to the five coherent systems of order 3, while systems 6, 16, 17, 18 and 20 are the systems that can be formed by adding a new component (1) in parallel to the five coherent systems of order 3.

Since all possible coherent systems of order 4 are listed in Table 2.1 according to their minimal cut sets, one might wonder what a list of the corresponding minimal path sets would look like. First, it should be mentioned that the 20 collections above can be thought of as the list of all possible minimal path sets of a particular coherent system, as they are an exhaustive list of collections of sets satisfying (i) and (ii) above. With that interpretation, system 1 would be the parallel system in four components, since if each component serves as a minimal path set, the system works if at least one component is working. Note, then, that the minimal cut sets of the series system are precisely the minimal path sets of the parallel system. This is an example of the "duality" of two coherent systems. Coherent system A is the *dual* of coherent system B if the minimal path sets of A are the minimal cut sets of B (and, similarly, the minimal cut sets of A are the minimal path sets of B). The relationship between dual coherent systems can be made precise through their respective structure functions.

Definition 2.3. *If A and B are dual systems, then their structure functions are related by the equation*

$$\varphi^A(\mathbf{x}) = 1 - \varphi^B(\mathbf{1} - \mathbf{x}) . \tag{2.7}$$

A state vector $\mathbf{x}$ is called a cut vector if $\varphi(\mathbf{x}) = 0$ and is called a path vector if $\varphi(\mathbf{x}) = 1$. Now if P is a minimal path set of system B, and $\mathbf{x}^*$ is the corresponding path vector, that is, $x_i^* = 1$ for all $i \in P$ and $x_i^* = 0$ for all $i \in P^c$, then $1 - \mathbf{x}^*$ is clearly a cut vector of B's dual system A since, by (2.7), $\varphi^A(\mathbf{1} - \mathbf{x}^*) = 0$. The set P corresponds to the zeros in the vector $\mathbf{1} - \mathbf{x}^*$ and is thus a cut set of system A. In addition, P is in fact a minimal cut set of A since if any zeros in that vector were turned into ones, the number of ones in the vector $\mathbf{x}^*$ would be diminished. Because P is assumed to be a minimal path set, φ^B would be zero for the diminished vector, implying that φ^A would be one for the augmented version of $\mathbf{1} - \mathbf{x}^*$. Thus, any set smaller than P cannot be a cut set for the system A. In short, the equation (2.7) guarantees that any minimal path set of a system is a minimal cut set of the system's dual and vice versa. This reasoning justifies the claim that for any collection of minimal cut sets defining a coherent system, the same collection can be regarded as the minimal path sets which define coherent systems of that order. Since the minimal path sets of a given system are the same as the minimal cut sets of its dual, one can infer the minimal path sets of a given system of order 4 from Table 2.2. For example, the minimal path sets of system 4 in Table 2.1 are $\{1,2,3\}$, $\{1,2,4\}$, $\{1,3,4\}$, the cut sets of its dual, system 18.

Table 2.2. Duality among systems of order 4

System	Dual
1	20
2	17
3	16
4	18
5	6
7	14
8	15
9	19
10	12
11	11
13	13

The closer look at the list of dual systems in Table 2.2 is instructive. An example of the reasoning which leads to the identification of a system's dual is as follows. Note that system 4 in Table 2.1 has minimal cut sets equal to $\{1\}$, $\{2,3\}$, $\{2,4\}$ and $\{3,4\}$. Now consider the system that has these four sets as its minimal path sets, that is, the dual of system 4. Recognizing that each minimal cut set of this dual system must render all its minimal path sets inoperable, we may identify the sets $\{1,2,3\}$, $\{1,2,4\}$ and $\{1,3,4\}$ as the minimal cut sets of the dual system. These cut sets satisfy conditions (iii) and (iv) above, and, in fact, correspond to the minimal cut sets of system 18 in Table 2.1. Thus, systems 4 and 18 are duals of each other. A further insight one can glean from Table 2.2 is the fact that a system can be its own dual; among systems of order 4, systems 11 and 13 in Table 2.2 have this property.

Carrying out the process exemplified above to identify the dual of system 11 in Table 2.1 brings out an interesting idiosyncrasy of any attempt to classifying coherent systems by their minimal path or cut sets (an idiosyncrasy which carries over to their classification by their structure functions). The system with minimal path sets $\{1,2\}$, $\{2,4\}$ and $\{3,4\}$ (which were the cut sets of system 11) has minimal cut sets $\{1,4\}$, $\{2,3\}$ and $\{2,4\}$. The reader might find it disturbing that this collection of cut sets is not among the list of all possible collections of cut sets in Table 2.1 for systems of order 4. The reason for their apparent absence is the way that the components are labeled.

In the latter collection, if we were to relabel component 3 as component 1 and component 1 as component 3, we see that the resulting sets would be precisely the cut sets listed for system 11 in Table 2.1. Thus, system 11 is its own dual. A more important insight gained from this little exercise is that the labeling of components matters, so that two seemingly different systems might in fact be the same after its components are relabeled. On the other hand, while the minimal cut sets of systems 6 and 11 in Table 2.1 have a similar appearance, they are qualitatively different (since, for example, all three minimal cut sets of system 6 have component 1 in common), so that no relabeling can convert them into the minimal cut sets of system 11. The fact that relabeling of components can disguise a coherent system, making it look like a potentially different system, is a special difficulty that renders path sets, cut sets and structure functions (which share this property) quite imperfect as indexes for the class of coherent systems. The signature of a coherent system introduced in the next chapter does not suffer from this imperfection.

As has been noted earlier, the direct computation of the structure function for an arbitrary coherent system can be algebraically cumbersome. Fortunately, there is a simple connection between the structure function of a coherent system and its minimal path sets and minimal cut sets. In order for a system to work, it must be the case that all the components of at least one minimal path set are working. Similarly, the system will work if and only if at least one of the components in every minimal cut set is working. These observations provide the following tools for the computation of the structure function of a coherent system. Let the minimal path sets of the system be $P_1, \ldots, P_r$, and for each such set, define path structure function $p_j(\mathbf{x})$ as

$$p_j(\mathbf{x}) = \prod_{i \in P_j} x_i \ . \tag{2.8}$$

Now $p_j(\mathbf{x}) = 1$ precisely when every component in P_j is working. It follows that the structure function of the system may be represented as

$$\varphi(\mathbf{x}) = 1 - \prod_{j=1}^{r} (1 - p_j(\mathbf{x})) \ . \tag{2.9}$$

Equation (2.9) can also be written in the following equivalent and revealing form:

$$\varphi(\mathbf{x}) = \max_{\{1 \le j \le r\}} p_j(x) = \max_{\{1 \le j \le r\}} \min_{\{i \in P_j\}} \{x_i\} \tag{2.10}$$

Inspection of equations (2.9) or (2.10) confirms that the structure function is 1 if and only if there is at least one minimal path set for which all components are working. As an example of the approach above, let's revisit the calculation of the structure function for the bridge system in Figure 2.2. As we have noted, the minimal path sets for this system are $\{1, 4\}$, $\{2, 5\}$, $\{1, 3, 5\}$ and $\{2, 3, 4\}$. From (2.9), we thus obtain

$$\varphi(\mathbf{x}) = 1 - (1 - x_1x_4)(1 - x_2x_5)(1 - x_1x_3x_5)(1 - x_2x_3x_4) . \tag{2.11}$$

Noting that the constants cancel out of this expression and using the idempotency of the x variables (that is, $x_i^2 = x_i$), one can quite readily reduce the expression in (2.11) to that in (2.5).

Similar considerations apply to the development of the structure function using the properties of minimal cut sets. In this case, we denote the minimal cut sets of the system of interest to be $C_1, \ldots, C_k$, and we define cut structure functions as

$$c_j(\mathbf{x}) = 1 - \prod_{i \in C_j} (1 - x_i) . \tag{2.12}$$

Now $c_j(\mathbf{x}) = 1$ precisely when the minimal cut set C_j contains at least one working component. Since the system works if and only if every minimal cut set contains at least one working component, it follows that the system's structure function has the alternative representation

$$\varphi(\mathbf{x}) = \prod_{j=1}^{k} c_j(\mathbf{x}) . \tag{2.13}$$

Equation (2.13) can be written in the following equivalent form:

$$\varphi(\mathbf{x}) = \min_{\{1 \le j \le k\}} c_j(\mathbf{x}) = \min_{\{1 \le j \le k\}} \max_{\{i \in C_j\}} \{x_i\} \tag{2.14}$$

The discussion above, and in particular, the representations of the structure function in (2.9) and (2.13) make it apparent that a coherent system may be thought of as a parallel system in which each element is a series system in the components of a minimal path set. Similarly, the system can be thought of as a series system in which each element is a parallel system in the components of a minimal cut set.

Given two systems with structure functions φ_1 and φ_2 respectively, it is clear that the second system performs better than the first if $\varphi_1(\mathbf{x}) \le \varphi_2(\mathbf{x})$ for all $x \in \{0, 1\}^n$, since this inequality implies that the first system will always fail when the second system does. Two useful results concerning coherent systems are easily established by comparing appropriate structure functions. The first is that no system can perform better than the parallel system nor worse than the series system. This follows from the self-evident inequalities

$$\prod_{i=1}^{n} x_i \le \varphi(\mathbf{x}) \le 1 - \prod_{i=1}^{n} (1 - x_i) . \tag{2.15}$$

The second result is both more interesting and more useful. It concerns the effect of redundancy in system designs. If one has an n-component system and had the opportunity to enhance its performance by incorporating redundancy through the addition of n more components, one might reasonably ask which of two options would be preferable: (1) backing up every component, essentially replacing every component in the original design by a parallel system in two components, or (2) backing up the entire system by an identical system of order n (placed in parallel with the original system). The following theorem establishes that option (1) is better, documenting the well known engineering principle that componentwise redundancy is always better than systemwise redundancy.

Theorem 2.1. *Let φ be the structure function of a coherent system of order n. Then for any $\mathbf{x}$ and $\mathbf{y} \in \{0,1\}^n$,*

$$\varphi(1-(1-x_1)(1-y_1),\ldots,1-(1-x_n)(1-y_n)) \geq 1-(1-\varphi(\mathbf{x}))(1-\varphi(\mathbf{y})) \quad (2.16)$$

Proof. The structure function of the left-hand side of (2.16) is obtained by adopting the view that the system with componentwise redundancy can be considered to have the same structure as the original system, but with each component replaced by a parallel subsystem in two components. The ith subsystem works when $[1-(1-x_i)(1-y_i)] = 1$ and fails when $[1-(1-x_i)(1-y_i)] = 0$. Since the inequalities $1-(1-x_i)(1-y_i) \geq x_i$ and $1-(1-x_i)(1-y_i) \geq y_i$ hold for all i, it follows from the mononicity of φ that

$$\varphi(1-(1-x_1)(1-y_1),\ldots,1-(1-x_n)(1-y_n)) \geq \varphi(\mathbf{x})$$

and

$$\varphi(1-(1-x_1)(1-y_1),\ldots,1-(1-x_n)(1-y_n)) \geq \varphi(\mathbf{y}) .$$

The latter inequality implies that

$$\varphi(1-(1-x_1)(1-y_1),\ldots,1-(1-x_n)(1-y_n)) \geq \max\{\varphi(\mathbf{x}),\varphi(\mathbf{y})\},$$

an inequality that is equivalent to (2.16). ∎

2.2 The Reliability of a Coherent System

Consider a coherent system in n independent components. If we fix a time t at which the system is examined, we may treat the ith component as working with probability p_i, that is, we may let $p_i = P(X_i = 1)$, where X_i is a Bernoulli variable representing the random state of the ith component at time t. We will define the *reliability* of a system at time t as the probability that it is working at that time. This probability will be denoted by $h(\mathbf{p})$ and can be computed from the structure function as

$$h(\mathbf{p}) = P(\varphi(\mathbf{X}) = 1) = E\varphi(\mathbf{X}) . \quad (2.17)$$

The function $h(\mathbf{p})$ is multilinear, that is, it is linear in every p_i. For the bridge system pictured in Figure 2.2, the system reliability is given by

$$\begin{aligned}h(\mathbf{p}) &= E(\varphi(\mathbf{x}))\\ &= E(X_1X_4 + X_2X_5 + X_1X_3X_5 + X_2X_3X_4 - X_1X_2X_3X_4 - X_1X_2X_3X_5\\ &\quad - X_1X_3X_4X_5 - X_1X_2X_4X_5 - X_2X_3X_4X_5 + 2X_1X_2X_3X_4X_5)\\ &= p_1p_4 + p_2p_5 + p_1p_3p_5 + p_2p_3p_4 - p_1p_2p_3p_4 - p_1p_2p_3p_5\\ &\quad - p_1p_3p_4p_5 - p_1p_2p_4p_5 - p_2p_3p_4p_5 + 2p_1p_2p_3p_4p_5\ .\end{aligned}$$

When components are identically distributed, we have $p_i \equiv p$, and the reliability function h simplifies. For the bridge system in i.i.d. components, the reliability function can be written in terms of this common p, and reduces to

$$h(p) = 2p^2 + 2p^3 - 5p^4 + 2p^5\ . \tag{2.18}$$

In the i.i.d. case, we refer to h as the *reliability polynomial*. For the n-component series, parallel and k-out-of-n systems, the respective reliability polynomials are given by

$$h_1(p) = p^n, \quad h_2(p) = 1 - (1-p)^n \quad \text{and} \quad h_3(p) = \sum_{i=0}^{k-1} \binom{n}{i} (1-p)^i p^{n-i}\ . \tag{2.19}$$

For complex systems, computing the system's reliability can be cumbersome, even under the i.i.d. assumption. A useful tool for making this computation (and as we shall see, for bounding $h(p)$ above and below by partial sums) is the so-called inclusion-exclusion formula. It is simply the general formula for calculating the union of a collection of (possibly overlapping) events. For two events, it is usually called "the addition rule." The general formula is given below. For a proof, see Feller [35].

Theorem 2.2. *Let $A_1, A_2, \ldots, A_n$ be n events, that is, subsets of the sample space of a random experiment. Then the probability that at least one of the events occurs is given by*

$$\begin{aligned}P\left(\cup_{i=1}^n A_i\right) &= \sum_{i=1}^n P(A_i) - \sum_{i<j} P(A_i \cap A_j) + \sum_{i<j<k} P(A_i \cap A_j \cap A_k)\\ &\quad - \cdots \pm P\left(\cap_{i=1}^n A_i\right)\ .\end{aligned} \tag{2.20}$$

If we denote the ith summation of the inclusion-exclusion formula by S_i, we may write

$$P\left(\cup_{i=1}^n A_i\right) = \sum_{i=1}^n (-1)^{i+1} S_i\ . \tag{2.21}$$

Further, partial sums of the formula in (2.21) can be shown to provide increasingly precise upper and lower bounds for the probability of interest. For instance,

$$P\left(\cup_{i=1}^n A_i\right) \le S_1, \quad P\left(\cup_{i=1}^n A_i\right) \ge S_1 - S_2, \quad P\left(\cup_{i=1}^n A_i\right) \le S_1 - S_2 + S_3, \quad \text{etc.}$$

Let's apply this tool in (2.20) to compute the reliability function for the coherent system of order 4 with minimal path sets $\{1\}, \{2,3\}, \{2,4\}, \{3,4\}$, that is, for system 4 in Table 2.1. Note that the inclusion-exclusion formula is precisely tailored for this calculation since a coherent system will function if and only if all the components in at least one of its minimal path sets are working. Let A_1, A_2, A_3 and A_4 be the events that all the components in each of the four minimal path sets above are working. Then the reliability function $h(\mathbf{p})$ of the system is equal to $P\left(\cup_{i=1}^{4} A_i\right)$. For simplicity, let's assume that each component has the same probability p of working. Then

$$\begin{aligned} h(p) &= S_1 - S_2 + S_3 - S_4 \\ &= (p + 3p^2) - (6p^3) + (3p^4 + p^3) - (p^4) \\ &= p + 3p^2 - 5p^3 + 2p^4 \ . \end{aligned} \tag{2.22}$$

More generally, consider a system in n i.i.d. components, and let p be the common probability that any given component is working at a fixed point in time. We will say that the reliability polynomial is in "standard form" when it is written as

$$h(p) = \sum_{i=1}^{n} d_i p^i \ . \tag{2.23}$$

The polynomial in (2.22) is in standard form. The other form we will be interested in is the so-called "pq form," where h is written as

$$h(p) = \sum_{i=1}^{n} c_i p^i q^{n-i} \ . \tag{2.24}$$

In Chapters 6 and 8, we discuss when and why one might prefer one form over the other.