U.S. DEPARTMENT OF COMMERCE
National Technical Information Service

AD-A032 105

Engineering Design Handbook: Development Guide for Reliability. Part Three. Reliability Prediction

Army Materiel Command Alexandria Va

Jan 76

AMC PAMPHLET

AMCP 706-197

ENGINEERING DESIGN HANDBOOK

DEVELOPMENT GUIDE FOR RELIABILITY

PART THREE

RELIABILITY PREDICTION

DISTRIBUTION STATEMENT A
Approved for public release;
Distribution Unlimited

PRICES SUBJECT TO CHANGE

HEADQUARTERS, US ARMY MATERIEL COMMAND

JANUARY 1976

DEPARTMENT OF THE ARMY
HEADQUARTERS US ARMY MATERIEL COMMAND
5001 EISENHOWER AVENUE, ALEXANDRIA, VA 22333

AMC PAMPHLET
NO. 706-197

6 January 1976

ENGINEERING DESIGN HANDBOOK

RELIABILITY PREDICTION

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

Paragraph Page

CHAPTER 10. RELIABILITY PREDICTION (GENERAL)

CHAPTER 11. MONTE CARLO SIMULATION

CHAPTER 12. RELIABILITY OPTIMIZATION

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS

LIST OF TABLES

LIST OF TABLES

Classified documents may be released on a "Need to Know" basis verified by an official Department of Army representative and processed from Defense Documentation Center (DDC), ATTN: DDC-TSR, Cameron Station, Alexandria, VA 22314.

Comments and suggestions on this handbook are welcome and should be addressed to:

Commander
US Army Materiel Development and Readiness Command
Alexandria, VA 22333

(DA Forms 2028, Recommended Changes to Publications, which are available through normal publications supply channels, may be used for comments/suggestions.)

CHAPTER 1 INTRODUCTION

This handbook reviews the basic ideas and formulas in probability and statistics and shows the kinds of models that might be useful for the reliability of systems. The concept of s-independence is discussed very thoroughly since it is so important in reliability improvements wrought by redundancy.

A large portion of the handbook deals with the effects of redundancy, simply because the calculation of reliability for non-redundant systems is so straightforward (although often tedious). The distinction between redundancy and repair is blurred in practice, especially when a failed unit is replaced by a good inactive unit.

Some of the techniques are presented only in their basic form. References are given for further study. Often the designer and reliability engineer will have better things to do than study sophisticated mathematics. It is usually better to find a person already trained in the subject who can then solve the specialized problems. In those cases the function of this handbook is to provide the designer and reliability engineer with

(1) basic knowledge; so they can converse intelligently with the experts, and
(2) perspective; so they know when to call an expert.

In dealing with mathematics it is important always to remember what mathematics is, and what it isn't. Mathematics *per se* is rules and relationships between abstract concepts. It is always "true" in the sense that it is correct (assuming no rules were violated), but all mathematics is not applicable to everything. It is in applying mathematics to a problem that we get in trouble. We have to choose what kind of mathematics to use, and then to choose what real-world things will be represented by what mathematical concepts. For example, is a particular material adequately representable by elastic, viscoelastic, or viscous equations? Or, is a physical coil of wire representable by a lumped inductance in series with a resistance?

Probability theory is abstract mathematics that can usefully represent many situations. Much of this handbook shows how to represent things by probabilities and how to manipulate those probabilities.

There is little that is new in probability/statistics for reliability. The Bibliography at the end of this chapter gives many references for those who need instruction in those topics. The books are labeled as Elementary, Intermediate, or Advanced. This handbook makes no attempt to rewrite all those books.

BIBLIOGRAPHY

Probability and Statistics Books

AMCP 706-110 through -114, *Experimental Statistics, Sections 1-5*, USGPO (Intermediate).

R. E. Barlow and F. Proschan, *Mathematical Theory of Reliability*, John Wiley & Sons, Inc., N.Y., 1965 (Advanced).

Vic Barnett, *Comparative Statistical Inference*, John Wiley & Sons, Inc., N.Y., 1973 (1975 corrected reprint), (Intermediate, Advanced).

A. M Breipohl, *Probabilistic Systems Analysis*, John Wiley & Sons, Inc., N.Y., 1970 (Elementary, Intermediate).

DA Pam 70-5, *Mathematics of Military Action, Operations and Systems* (Elementary, Intermediate).

A. J. Duncan, *Quality Control and Industrial Statistics*, Richard D. Irwin, Inc., Homewood, Ill., 1965 (Elementary, Intermediate).

W. Feller, *An Introduction to Probability Theory and Its Applications*, Vols. I, II, John Wiley & Sons, Inc., N.Y., Vol. I, 1957, Vol. II, 1966 (Advanced).

J. E. Freund, *Modern Elementary Statistics*, Prentice-Hall, Englewood Cliffs, N.J., 1967 (Elementary).

Gnedenko, Belyayev, and Solovyev, *Mathematical Methods of Reliability Theory*, Academic Press, N.Y., 1969 (Advanced).

P. Hoel, *Introduction to Mathematical Statistics*, John Wiley & Sons, Inc., N.Y., 1962 (Elementary, Intermediate).

Mann, Schafer, and Singpurwalla, *Methods for Statistical Analysis of Reliability and Life Data*, John Wiley & Sons, Inc., N.Y., 1974 (Intermediate, Advanced)

I. Miller and J. E. Freund, *Probability and Statistics for Engineers*, Prentice-Hall, Englewood Cliffs, N.J., 1965 (Elementary).

NBS Handbook 91, *Experimental Statistics*, USGPO 1966 (Intermediate).

E. Parzen, *Modern Probability Theory and Its Applications*, John Wiley & Sons, Inc., N.Y., 1960 (Intermediate, Advanced).

E. Parzen, *Stochastic Processes*, Holden-Day, Inc., San Francisco, 1962 (Advanced).

M. L. Shooman, *Probabilistic Reliability*, McGraw-Hill, N.Y., 1968 (Elementary, Intermediate).

Many of the early reliability texts, and some of the more recent ones which are not mentioned here, have an inadequate or poor introduction to probability and statistics. Most probability/statistics texts are quite adequate.

CHAPTER 2 - REVIEW OF ELEMENTARY PROBABILITY THEORY (DISCRETE)

2-0 LIST OF SYMBOLS

A,B,C,E = sets
A_F, A_G, B_F, B_G = events that units U_A and U_B are Failed or Good
A_i, B_i, C_i, E_i = subsets of A,B,C,D,E
$E\{\}$ = s-expected value of
E_B, E_{HT}, E_{ET} = events of Benign, High Temperature, Electrical Transient environments
E_L, E_S = events of Light and Severe environments
M_i = ith central moment
N_A, N_B, N_E = number of subsets in A,B,E
pmf = probability mass function
$Pr\{\}$ = probability of
s- = denotes statistical definition
μ = mean
σ = standard deviation
σ^2 = variance
Ω = complete sample space
Φ = null event
$\cup$ = union
$\cap$ = intersection

2-1 INTRODUCTION

The question always arises "What is probability?" Some say it is relative frequency; others say it is degree-of-belief; and still others have different concepts. In many good reliability and engineering textbooks (and virtually all mathematical books) probabilities are mathematical concepts which can then be applied to such things as relative frequency and degree-of-belief. The situation is analogous to plane geometry. Plane geometry is a mathematical theory that uses concepts such as point and line. The theory is true (consistent) regardless of what a point or line is taken to be. Plane geometry often is applied successfully to many reasonably flat things in everyday life, and we associate point and line with the everyday concepts.

Probability and statistics are related very closely to each other. The difference between them is not clear to many engineers. Probability theory usually considers the parameters of a general problem as known, then computes numbers (probabilities) about particular sets of events. It goes from the general to the particular. Statistics on the other hand treats actual data and tries to decide what useful things can be done with them and how to get them. It goes from the particular to the general. A statistic is a number obtained from a sample or obtained from manipulating other statistics. In engineering problems one usually uses a mixture of probability and statistics; there is little to be gained in debating which calculations are probabilistic and which are statistical.

2-2 BASIC PROBABILITY RULES

2-2.1 SAMPLE SPACE, SAMPLE POINT, EVENT

These are basic concepts for any probability problem. The sample space is made up of all the sample points. An event is a collection of sample points; it can contain as few sample points as *none*, or as many as *all*. The concepts are best illustrated by examples. See the Bibliography in Chapter 1 for books which can explain the concepts.

Example 1. For one throw of a single die, the sample space is the set of numbers 1, 2, 3, 4, 5, 6; i.e., the sample space is all possible values that can arise. Each value is called a sample point. There are six sample points in the sample space for this example.

Every possible single outcome of an experiment is a sample point. The naming of every sample point is a first step in making a probabilistic model of any problem, although it often is done implicitly. Each sample point also has a probability associated with it. The probability usually is assigned or calculated from known event-probabilities.

In the example of one throw of a single die, the probabilities usually are assigned by defining the die to be "fair"; i.e., each face has an equal probability of appearing. Then the probability assigned to each sample point is 1/6. By definition, the sum of the probabilities for all sample points must be one.

Engineers who use probability often go astray because they do not understand sample-space and assignment of probabilities to each sample point.

Example 2. A coin is tossed three times. What is the sample-space? Let t denote a tail and h a head. Then there are eight sample points in the sample space:

ttt	*htt*
tth	*hth*
tht	*hht*
thh	*hhh*

The event 'First toss is a head' has four sample points: *htt*, *hth*, *hht*, *hhh*. The event "'First toss is a head' ∩ 'Last toss is a tail'" has two sample points: *htt*, *hht*. The event 'First toss is neither a head nor a tail' has no sample points.

2-2.2 NOTATION AND DEFINITIONS

There is no universally accepted and used set of notation. Because the difficulties engineers have with probability are often basic in nature, a notation is selected which is not easily confused with something else, even though it is sometimes cumbersome. The notation and definitions are illustrated in Figs. 2-1 and 2-2.

Φ — The null event; viz., the event contains no sample points.

Ω — The complete sample space; viz., the event contains all the sample points.

$\cup$ — Union, and/or; e.g., $A \cup B$ contains all sample points which are in A and/or in B. (Sometimes + is used.)

$\cap$ — Intersection, both/and; e.g., $A \cap B$ contains only those sample points which are in both A and B. (Sometimes X is used.)

$Pr\{\cdot\}$ — Probability of the event (or sample point) contained in the { }; e.g.,

$Pr\{a\}$ = probability of the sample point a

$Pr\{A\}$ = probability of the event A

$Pr\{\cdot|\cdot\}$ — Conditional probability; probability of the event to the left of the |, given that the event (condition) to the right of the | has occurred; e.g., $Pr\{A|B\}$ is the conditional probability of event A, given that the event B has occurred.

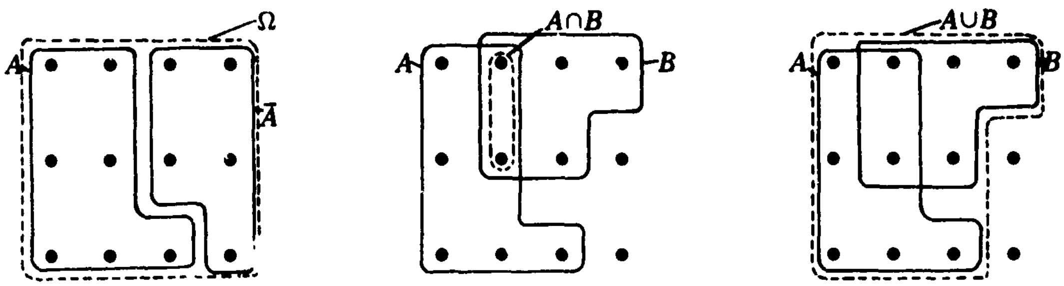
$Pr\{A|B\} \equiv Pr\{A \cap B\}/Pr\{B\}$; $Pr\{B\} \neq 0$.

$Pr\{A|B\}$ is meaningless (contradiction in terms) if $Pr\{B\} = 0$.

mutually exclusive — Two events are mutually exclusive if and only if they have no sample points in common; e.g., A and B are mutually exclusive if and only if $A \cap B = \Phi$.

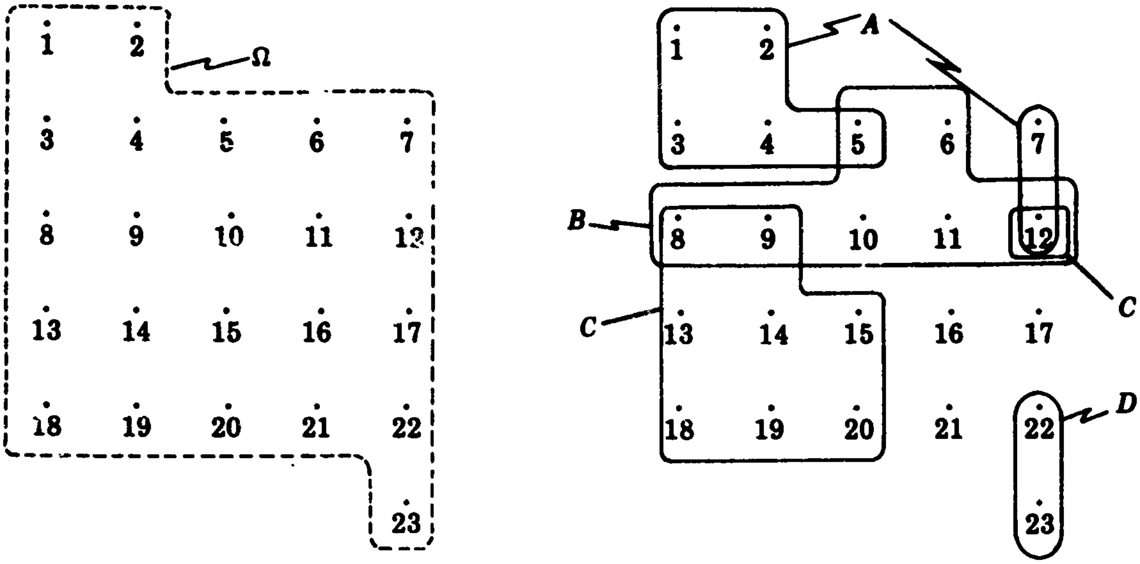
exhaustive — A set of events is exhaustive if and only if the union of the events contains all sample points in the sample space; e.g., A, B, C are exhaustive if $A \cup B \cup C = \Omega$.

partitioning — A set of events is a partition-



(A) Complement (B) Intersection (C) Union

FIGURE 2-1. Example Event Relationships for 2 Events



$\Omega \equiv 1$ through 23
$A \equiv 1$ through 5, 7, 12
$B \equiv 5, 6, 8$ through 12
$C \equiv 8, 9, 12$ through 15, 18 through 20
$D \equiv 22, 23$

Definitions

Examples of Set Relationships

$A \cap B$	= 5, 12
$B \cap C$	= 8, 9, 12
$C \cap A$	= 12
$A \cap B \cap C$	= 12
$A \cap D$	= Φ
$B \cap D$	= Φ
$C \cap D$	= Φ

$\bar{A}$	= 6, 8 through 11, 13 through 23
$\bar{B}$	= 1 through 4, 7, 13 through 23
$\bar{C}$	= 1 through 7, 10, 11, 16, 17, 21 through 23
$\bar{D}$	= 1 through 21
$\bar{A} \cap D$	= 22, 23
$D \subset \bar{A}$	
$D \subset (\bar{A} \cap \bar{B})$	
$\bar{D} \cup (\bar{B} \cap C)$	= 13 through 15, 18 through 20
$22 \in D$	

FIGURE 2-2. Example Event Relationship for 4 Events

	ing of the sample space if and only if the events are all mutually exclusive and the set is exhaustive. (The name comes from the way a set of partitions breaks up a room into smaller rooms, each of which is separate; but every part of the original room is in some smaller room.)
$\bar{}$	Denotes the complement of an event; e.g., $\bar{A}$ is the complement of A.
complement	The complement of an event contains all the sample points in the sample space which are not in the event. A formal definition is $B = \bar{A}$ if and only if $A \cup B = \Omega$ and $A \cap B = \Phi$.
,	Beware of the comma, it is not ordinarily a defined symbol. Often intersection is meant, but one can't be sure.
$Pr\{\cdot;\cdot\}$	Probability of the event to the left of the ";". The events or parameters to the right of the semicolon are known. The notation is often used for emphasis or as a reminder. It is similar to $Pr\{\cdot\|\cdot\}$ except that the event to the right of the "|" is a random one, whereas the event or parameters to the right of the ";" are certain (known exactly).
$\in$	$a \in B$ means that a is a sample point of B.
$\subset$	$A \subset B$ means that A is a subset of B; viz., all sample points of A are also in B, but all sample points of B need not be in A.

2-2.3 RULES, LAWS, AND DEFINITIONS FOR EVENTS

Let A, B, C be any events.

$$A \cup \bar{A} = \Omega \tag{2-1}$$

$$A \cap \bar{A} = \Phi \tag{2-2}$$

$$A \cup A = A \tag{2-3}$$

$$A \cap A = A \tag{2-4}$$

$$A \cup B = B \cup A \tag{2-5}$$

$$A \cap B = B \cap A \tag{2-6}$$

$$A \cup (B \cup C) = (A \cup B) \cup C = A \cup B \cup C \tag{2-7}$$

$$A \cap (B \cap C) = (A \cap B) \cap C = A \cap B \cap C \tag{2-8}$$

$$A \cup (B \cap C) = (A \cup B) \cap (A \cup C) \tag{2-9}$$

$$A \cap (B \cup C) = (A \cap B) \cup (A \cap C) \tag{2-10}$$

$$\overline{(A \cup B)} = \bar{A} \cap \bar{B} \tag{2-11}$$

$$\overline{(A \cap B)} = \bar{A} \cup \bar{B} \tag{2-12}$$

2-2.4 RULES, LAWS, AND DEFINITIONS FOR PROBABILITIES

Let A, B, C be any events; and let

A_i, $i = 1, \ldots, N_A$ be a partitioning of A. (The A_i are mutually exclusive and exhaustive.)

B_i, $i = 1, \ldots, N_B$ be a partitioning of B. (The B_i are mutually exclusive and exhaustive.)

a_j, $j = 1, \ldots, M$ be the sample points in A.

E_i, $i = 1, \ldots, N_E$ be any N events.

$$Pr\{A\} = \sum_1^M Pr\{a_j\} \tag{2-13}$$

$$0 \leqslant Pr\{A\} \leqslant 1 \tag{2-14}$$

$$Pr\{\Phi\} = 0 \tag{2-15}$$

$$Pr\{\Omega\} = 1 \tag{2-16}$$

$$Pr\{A \cup B\} = Pr\{A\} + Pr\{B\} - Pr\{A \cap B\} \tag{2-17}$$

$$Pr\{A \cup B \cup C\} = Pr\{A\} + Pr\{B\} + Pr\{C\} - Pr\{A \cap B\} - Pr\{B \cap C\} - Pr\{C \cap A\} + Pr\{A \cap B \cap C\} \tag{2-18}$$

$$Pr\{A|B\} = Pr\{A \cap B\} / Pr\{B\} \text{ for } Pr\{B\} \neq 0 \tag{2-19}$$

$$Pr\left\{E_1 \cup E_2 \cup \cdots \cup E_{N_E}\right\} = \sum_{i=1}^{N_E} Pr\{E_i\} - \sum_{i=1}^{N_E} \sum_{j=1}^{i-1} Pr\{E_i \cap E_j\} + \sum_{i=1}^{N_E} \sum_{j=1}^{i-1} \sum_{k=1}^{j-1} Pr\{E_i \cap F_j \cap E_k\} - \cdots \pm Pr\{E_1 \cap E_2 \cap \cdots \cap E_N\} \tag{2-20}$$

The first term in Eq. 2-20 is an upper bound; adding terms in succession provides an alternating series of bounds which get increasingly better, until exactness is reached when all terms are used.

$$Pr\{A \cap B\} = Pr\{A|B\} Pr\{B\} = Pr\{B|A\} Pr\{A\} \qquad (2\text{-}21)$$

Eq. 2-21 is a form of Bayes' Theorem.

$$Pr\{A \cap B \cap C\} = Pr\{A|(B \cap C)\} Pr\{B|C\} Pr\{C\} \qquad (2\text{-}22)$$

$$Pr\{A\} = \sum_{i=1}^{N_A} Pr\{A_i\} \qquad (2\text{-}23a)$$

$$Pr\{A\} = \sum_{j=1}^{N_B} Pr\{A|B_j\} Pr\{B_j\} \qquad (2\text{-}23b)$$

$$Pr\{A_i|B\} = \frac{Pr\{B|A_i\} Pr\{A_i\}}{\sum_{j=1}^{N_B} Pr\{B|A_j\} Pr\{A_j\}} \qquad (2\text{-}24)$$

Eq. 2-24 is a form of Bayes' Theorem.

2-3 s-INDEPENDENCE

There are several equivalent definitions of s-independence. From an engineering point of view, the most satisfactory definition is Eq. 2-25.

A and B are s-independent if and only if

$$Pr\{A|B\} = Pr\{A|\bar{B}\} = Pr\{A\}. \qquad (2\text{-}25)$$

That is, the probability of A is the same regardless of whether we know that B has occurred, or has not occurred, or we do not know about B -- B just doesn't make any difference. There are several equations that are logically equivalent to Eq. 2-25, each implying the others. (The second equation in Eq. 2-25 actually is implied by the first one.) The most satisfactory definition from a statistical point of view is Eq. 2-26.

A and B are s-independent if and only if

$$Pr\{A \cap B\} = Pr\{A\} \quad Pr\{B\}. \qquad (2\text{-}26)$$

Eq. 2-26 is defined even for $Pr\{B\} = 0$ or 1 whereas Eq. 2-25 is not. The extension to more than two events is easier with Eq. 2-26.

N events are s-independent if and only if for every intersection of events taken 2, 3, ..., N at a time, the probability of the intersection of those events is the product of the probabilities of the individual events. This can be a complicated concept; see the Bibliography at the end of Chapter 1 for a further discussion.

Example.

Suppose there are 2 units (from one population) in a subsystem and both must fail for the subsystem to fail. If the probability of failure of each is 0.200 and the probability of subsystem failure is 0.200 × 0.200 = 0.0400, then the failure events are s-independent. Even if the probability of subsystem failure were 0.0404 (e.g., 1% above the 0.0400 figure), the failure events could be considered s-independent for engineering purposes.

Suppose that the probability of failure of each unit is 1.00×10^{-3} and th probability of subsystem failure is 1.00×10^{-6}; then the failure events are s-independent. But if the probability of subsystem failure were 0.000401 (0.0004 more, just as in the preceding paragraph), the failure events would in no way be s-independent. When failure probabilities are very small, one must be very careful not to ignore events whose probabilities might ordinarily be neglected.

2-4 CONDITIONAL s-INDEPENDENCE

A very important concept is conditional s-independence; i.e., two (or more) events can be conditionally s-independent, given a particular event. All general theorems on probabilities are valid also for conditional probabilities with respect to any particular event C_i. Thus Eq. 2-25 becomes

$$Pr\{A|(B \cap C_i)\} = Pr\{A|(\bar{B} \cap C_i)\} = Pr\{A|C_i\} \qquad (2\text{-}27)$$

and Eq. 2-26 becomes

$$Pr\{A \cap B \,|C_i\} = Pr\{A|C_i\} \, Pr\{B|C_i\}. \qquad (2\text{-}28)$$

In many engineering situations, if two events A and B (say, failures) are not s-independent, they will be conditionally s-independent, given each event of a set of events which is a partitioning of the sample space.

Example.

A_F, A_G events that unit U_A is failed or good

B_F, B_G events that unit U_B is failed or good

Let the sample points, events, and associated probabilities be as shown in Table 2-1. The probability of each event, as shown, is the sum of the probabilities of each of the sample points in the event.

Are the events A_F, B_F s-independent? To find out, use Eq. 2-26.

$$Pr\{A_F \cap B_F\} = Pr\{(a_f b_f)\} = 0.158$$

$$Pr\{A_F\} \times Pr\{B_F\} = 0.250 \times 0.380 = 0.095$$

They are not the same ($0.158 \neq 0.095$); so the events A_F, B_F are s-dependent.

Suppose there are two possible environments, light (event E_L) and severe (event E_S), and that the new sample space, events, and probabilities are as shown in Table 2-2. The events A_F and B_F are conditionally s-independent as shown by the calculations in Table 2-3. Eqs. 2-28 and 2-19 are used in the calculation.

TABLE 2-1. SAMPLE SPACE FOR EXAMPLE

	B_G 0.620	B_F 0.380
A_G 0.750	$a_g b_g$ 0.528	$a_g b_f$ 0.222
A_F 0.250	$a_f b_g$ 0.092	$a_f b_f$ 0.158

The number associated with each of the 4 sample points ($a_g b_g$, $a_g b_f$, $a_f b_g$, $a_f b_f$) is the probability of that sample point.

The events are defined as

$$A_G \equiv (a_g b_g, a_g b_f)$$
$$A_F \equiv (a_f b_g, a_f b_f)$$
$$B_G \equiv (a_g b_g, a_f b_g)$$
$$B_F \equiv (a_g b_f, a_f b_f)$$

The conditions under which events are conditionally s-independent are sometimes called common-modes,* and the failures which result from severe common-modes are called common-mode failures. This phenomenon is so important it will be illustrated with another example.

Example, Common mode (cause) failure:

Notation:

A_F, B_F = failure events of units U_A and U_B.

S_F = $A_F \cap B_F$, failure event of the system S.

E_B, E_{HT}, E_{ET} = a partitioning of the sample space: event of a Benign Environment, a High-Temperature Environment, and an Electrical-Transient Environment.

Given: The events A_F, B_F are conditionally s-independent, given E_i (i = B, HT, ET).

$$Pr\{A_F|E_B\} = Pr\{B_F|E_B\} = 6 \times 10^{-4}, \quad Pr\{E_B\} = 0.9976$$

$$Pr\{A_F|E_{HT}\} = Pr\{B_F|E_{HT}\} = 1 \times 10^{-2}, \quad Pr\{E_{HT}\} = 2 \times 10^{-3}$$

$$Pr\{A_F|E_{ET}\} = Pr\{B_F|E_{ET}\} = 1 \times 10^{-1}, \quad Pr\{E_{ET}\} = 4 \times 10^{-4}$$

Cursory inspection of the data shows that U_A and U_B are quite reliable if the environment is benign, and that nonbenign environments are rare. We first calculate the unconditional failure probability for U_A and U_B (see Table 2-4). It is negligibly different from the benign conditional failure probability. This leads us to believe, reasonably enough, that the effects of the nonbenign environments are negligible.

But then we calculate the probabilities that both U_A and U_B are failed (see Table 2-4). The situation is now quite different; one of the nonbenign environments is most important.

*Now called "common-cause."

TABLE 2-2. SAMPLE SPACE FOR MODIFIED EXAMPLE

	$B_G E_L$ 0.560	$B_F E_L$ 0.140
$A_G E_L$ 0.630	$p111$ 0.504	$p121$ 0.126
$A_F E_L$ 0.070	$p211$ 0.056	$p221$ 0.014
	E_L 0.700	

	$B_G E_S$ 0.060	$B_F E_S$ 0.240
$A_G E_S$ 0.120	$p112$ 0.024	$p122$ 0.096
$A_F E_S$ 0.180	$p212$ 0.036	$p222$ 0.144
	E_S 0.300	

$E_L = (p111, p121, p211, p221)$; $A_G = (p111, p121, p112, p122)$; $B_G = (p111, p211, p112, p212)$

$E_S = (p112, p122, p212, p222)$; $_F = (p211, p221, p212, p222)$; $B_F = (p121, p221, p122, p222)$

Explanation of notation for p_{ijk}:

i position reserved for event A
j position reserved for event B
k position reserved for event E
1 = "good" for events A and B
2 = "fail" for events A and B
1 = "light" for event E
2 = "severe" for event E

TABLE 2-3. CALCULATIONS TO SHOW EVENTS A_F AND B_F ARE CONDITIONALLY s-INDEPENDENT

Procedure	Example
1. State the sample space, events, and their probabilities.	1. See Table 2-2.
2. State the events to be tested for conditional s-independence and the conditions.	2. A_F, B_F to be conditionally s-independent. E_L, E_S are the conditions.
3. State the equations to be tested. $Pr\{(A \cap B) \mid C_i\} \stackrel{?}{=} Pr\{A \mid C_i\} \, Pr\{B \mid C_i\}$ (2-28)	3. $Pr\{(A_F \cap B_F) \mid E_i\} \stackrel{?}{=} Pr\{A_F \mid E_i\} \, Pr\{B_F \mid E_i\}$ for $i = L, S$
4. Use the definition of conditional probability to find each of the probabilities. $Pr\{A \mid B\} = Pr\{A \cap B\} / Pr\{B\}$ for $Pr\{B\} \neq 0$ (2-19)	4. $Pr\{(A_F \cap B_F) \mid E_i\} = Pr\{A_F \cap B_F \cap E_i\} / Pr\{E_i\}$ (2-29) $Pr\{A_F \mid E_i\} = Pr\{A_F \cap E_i\} / Pr\{E_i\}$ (2-30) $Pr\{B_F \mid E_i\} = Pr\{B_F \cap E_i\} / Pr\{E_i\}$ for $i = L, S$ (2-31)
5. Find the sample points in each of the intersections.	5. $A_F \cap B_F \cap E_L = (p221)$ $A_F \cap B_F \cap E_S = (p222)$ $A_F \cap E_L = (p211, p221)$ $A_F \cap E_S = (p212, p222)$ $B_F \cap E_L = (p121, p221)$ $B_F \cap E_S = (p122, p222)$
6. Find the probabilities by adding the probabilities of the sample points.	6. $Pr\{A_F \cap B_F \cap E_L\} = 0.014$ $Pr\{A_F \cap B_F \cap E_S\} = 0.144$ $Pr\{A_F \cap E_L\} = 0.056 + 0.014 = 0.070$ $Pr\{A_F \cap E_S\} = 0.036 + 0.144 = 0.180$ $Pr\{B_F \cap E_L\} = 0.126 + 0.014 = 0,140$ $Pr\{B_F \cap E_S\} = 0.096 + 0.144 = 0.240$ $Pr\{E_L\} = 0.504 + 0.126 + 0.056 + 0.014 = 0.700$ $Pr\{E_S\} = 0.024 + 0.096 + 0.036 + 0.144 = 0.300$
7. Calculate the conditional probabilities. $Pr\{(A_F \cap B_F) \mid E_i\}$ $= Pr\{A_F \cap B_F \cap E_i\} / Pr\{E_i\}$ (2-29) $Pr\{A_F \mid E_i\} = Pr\{A_F \cap E_i\} / Pr\{E_i\}$ (2-30) $Pr\{B_F \mid E_i\} = Pr\{B_F \cap E_i\} / Pr\{E_i\}$ for $i = L, S$ (2-31)	7. $Pr\{(A_F \cap B_F) \mid E_L\} = 0.014/0.700 = 0.020$ $Pr\{A_F \cap B_F \mid E_S\} = 0.144/0.300 = 0.480$ $Pr\{A_F \mid E_L\} = 0.070/0.700 = 0.100$ $Pr\{A_F \mid E_S\} = 0.180/0.300 = 0.600$ $Pr\{B_F \mid E_L\} = 0.140/0.700 = 0.200$ $Pr\{B_F \mid E_S\} = 0.240/0.300 - 0.800$
8. Check the equations in step 3	for $i = L$: $0.020 \stackrel{?}{=} 0.100 \times 0.200 = 0.020$ *yes* for $i = S$: $0.480 \stackrel{?}{=} 0.600 \times 0.800 = 0.480$ *yes*

The events A_F, B_F are conditionally s-independent, given each of the conditions E_L, E_S. As shown in the previous example, A_F, B_F are not (unconditionally) s-independent.

In systems which use redundancy to achieve very high reliability, the importance of common-mode failures often is overlooked completely. The key nature of conditional *s*-independence ought always to be in the analyst's mind when he uses redundancy.

TABLE 2-4. COMMON MODE (CAUSE) FAILURE CALCULATIONS

Procedure	Example
1. Calculate the $Pr\{A_F\}$ Adapt $$Pr\{A\} = \sum_{j=1}^{N_B} \{A\vert B_j\} Pr\{B_j\} \quad (2\text{-}23b)$$	1. $$Pr\{A_F\} = \sum_{j=1}^{3} Pr\{A_F\vert E_j\}\, Pr\{E_j\} \quad (2\text{-}32)$$ $(6 \times 10^{-4}) \times 0.9976 + (1 \times 10^{-2}) \times (2 \times 10^{-3}) + (1 \times 10^{-1}) \times (4 \times 10^{-4}) = 6.59 \times 10^{-4}$
2. Calculate $Pr\{B_F\}$	2. $Pr\{B_F\} = Pr\{A_F\} = 6.59 \times 10^{-4}$ because A_F and B_F are interchangeable in the probabilities as given.

The unconditional probabilities differ from the benign conditional ones by less than 10 %. (In practice rarely is a low probability of failure known as accurately as within ± 10%.)

Procedure	Example
3. Calculate the conditional probabilities of $A_F \cap B_F$. Adapt Eq. 2-28. $Pr\{(A_F \cap B_F)\vert E_i\} = Pr\{A_F\vert E_i\} Pr\{B_F\vert E_i\}$, for $i = B, HT, ET$.	3. $Pr\{(A_F \cap B_F)\vert E_B\} = (6 \times 10^{-4})^2 = 0.00036 \times 10^{-3}$ $Pr\{(A_F \cap B_F)\vert E_{HT}\} = (1 \times 10^{-2})^2 = 0.1 \times 10^{-3}$ $Pr\{(A_F \cap B_F)\vert E_{ET}\} = (1 \times 10^{-1})^2 = 10 \times 10^{-3}$
4. Calculate $Pr\{A_F \cap B_F\}$. Adapt Eq. 2-23b.	4. $Pr\{A_F \cap B_C\} = (0.00036 \times 10^{-3}) \times 0.9976 + (0.1 \times 10^{-3}) \times (2 \times 10^{-3}) + (10 \times 10^{-3}) \times (4 \times 10^{-4}) = 0.36 \times 10^{-6} + 0.200 \times 10^{-6} + 4 \times 10^{-6} = 4.56 \times 10^{-6}$
5. Calculate $Pr\{A_F\}\, Pr\{B_F\}$	5. $Pr\{A_F\} Pr\{B_F\} = (6.59 \times 10^{-4})^2 = 4.34 \times 10^{-7}$

From step 4 it is seen that virtually the only "cause" of system failure is the common-mode Electrical Transient Environment. From step 5, it is seen that if (unconditional) *s*-independence were to have been assumed, the failure probability of the system would have been underestimated by a factor of 10.

2-5 DISTRIBUTIONS

Very often the sample space is a subset of the integers (or can be put into 1 - 1 correspondence with some of the integers), and the probability to be assigned to a sample point is a function of the integer which corresponds to the sample point. The probability mass function (*pmf*) is the function which assigns a probability to each sample point. This is illustrated in Table 2-5.

2-5.1 RANDOM VARIABLES

When the sample space is associated with the integers, it is convenient to introduce the notion of random variable. For example, the events C_i and E_i in this chapter are random variables, and the probability of the event depends on the integer i. A variable is a random variable if the uncertainty involved with it is important, i.e., if probabilities need to be associated with it. This is an engineering decision; for example, the lengths of posts to be driven in the ground might not be considered random even though they had a spread of ± 10%, whereas the diameters of ball bearings would probably be random variables if their spread was ± 1%.

TABLE 2-5. DISCRETE DISTRIBUTIONS

	Binomial	Poisson*
parameters	p_1, p_2, N $(p_1 + p_2 = 1)$	μ
random variables	n_1, n_2 $(n_1 + n_2 = N)$	n
pmf	$\dfrac{N!}{n_1!n_2!} p_1^{n_1} p_2^{n_2}$	$\dfrac{e^{-\mu}\mu^n}{n!}$
mean μ	p_1N, p_2N	μ
variance σ^2	p_1p_2N	μ
3rd central moment M_3	$Np_1p_2(p_2 - p_1)$	μ
4th central moment M_4	$Np_1p_2(3Np_1p_2 - 6p_1p_2 + 1)$	$\mu(3\mu + 1)$
coefficient of variation $\dfrac{\sigma}{\mu}$	$\left(\dfrac{p_2}{p_1N}\right)^{1/2}$	$\mu^{-1/2}$
coefficient of skewness $\dfrac{M_3}{\sigma^3}$	$\dfrac{p_2 - p_1}{(Np_1p_2)^{1/2}}$	$\mu^{-1/2}$
excess coefficient of kurtosis $\dfrac{M_4}{\sigma^4} - 3$	$-\dfrac{6}{N} + \dfrac{1}{Np_1p_2}$	μ^{-1}

*As is customary, the symbol μ (for mean) is used for the parameter because the parameter happens to be the mean. This is also done in the *s*-normal distribution.

There is nothing mysterious about randomness and random variables. If you need something to be a random variable, it is; if you don't need it to be, it isn't.

2-5.2 MOMENTS

Random variables with *pmf*'s have moments. The two conventional points about which to take moments are the origin and the mean; when taken about the mean, they are called central-moments. The second moment-about-the-mean is the variance (square of the standard deviation).

The *n*th moment, about the origin, of x is the s-expected value of x^n;

$$E\{x^n\} \equiv \sum_i x_i^n pmf\{x_i\} \qquad (2\text{-}33)$$

where $\sum_i$ implies the sum over the domain of x_i. (It is presumed that the series converges absolutely; if not, a textbook ought to be consulted.)

The *n*th moment, about the mean, of x is the s-expected value of $(x-\mu)^n$:

$$E\{(x-\mu)^n\} = \Sigma\, (x_i - \mu)^n\, pmf\{x_i\} \qquad (2\text{-}34)$$

where $\mu \equiv E\{x\}$.

2-5.3 TWO DISTRIBUTIONS

Two common discrete distributions are the binomial and Poisson. Table 2-1 gives their definitions and properties. The Poisson distribution is often used as an approximation for the binomial distribution; it is usually adequate if the Poisson probability $\sum_{n=N+1}^{\infty} pmf\{n\}$ is negligible.

The adaptation (in most places) can be made mechanically as follows:

$$p_1 N \to \mu$$
$$N \to \infty$$
$$p_2 \to 1$$

CHAPTER 3 REVIEW OF ELEMENTARY PROBABILITY THEORY (CONTINUOUS)

3-0 LIST OF SYMBOLS

C = Conditional event
$Cdf\{\}$ = Cumulative distribution function
Cov{ } = Covariance
$E\{\}$ = s-Expected value
$f(x)$ = $pdf\{X\}$
M_i = ith central moment
$pdf\{\}$ = probability density function
$Pr\{\}$ = Probability of
s- = denotes statistical definition
$Sf\{\}$ = Survivor function
Var{ } = Variance
x,y,z = particular values of X,Y (also used as subscripts)
X,Y,Z = random variables
μ = mean
σ = standard deviation
$\int_X$ = integral over the domain of X

3-1 INTRODUCTION

When the sample space is continuous rather than discrete, the theoretical basis of probability theory can become much more sophisticated. However, many relatively simple problems can be solved by a straightforward extension of the concepts in Chapter 2. Only those straightforward concepts are discussed in this volume. Those who need more advanced concepts ought to consult the Bibliography in Chapter 1.

The concept of probability-density needs to be introduced. It is analogous to physical density functions, where continuous variables are being used. For example, a 10-ft long uniform bar which weighs 200 lb has a density of (200 lb)/(10 ft) = 20 lb/ft It is not meaningful to talk about the weight of a point along the bar, only the weight between two points. If the bar is nonuniform, then the density changes from point to point along the bar.

Probability densities can be very misleading because of possible transformations of the variables. For example, if a random variable has a uniform (constant) probability density, the logarithm of that random variable will NOT have a uniform probability density.

The basic rules for probability are quite similar to those for the discrete case, but the notation is usually somewhat different.

3-2 BASIC PROBABILITY RULES

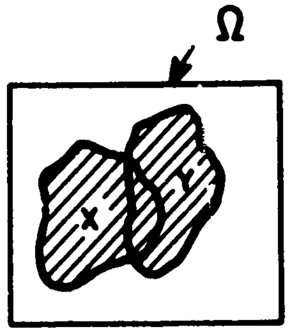
3-2.1 SAMPLE SPACE, EVENT

The sample space is the domain of the random variable (i.e., the values that can possibly be assumed by the random variable) or the domains of the several random variables. For example, the strength of a metal has the domain $(0,\infty)$.

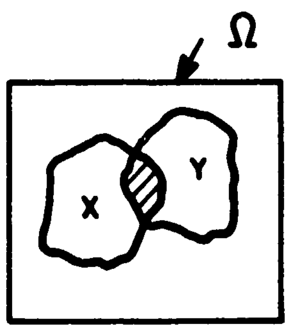
An event is the occurrence of some portion of the sample space. For example, an event might be "Strength > S_0" where S_0 is some constant. Figure 3-1 shows some set rules for continuous space.

3-2.2 NOTATION AND DEFINITIONS

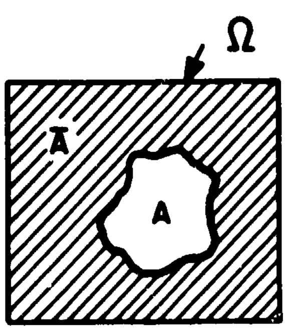
Notation	Definition
capital letter	The name of a random variable.
lower case letter	A specific value of the random variable.
$Pr\{\cdot\}$	Probability of the event in the { }; e.g., $Pr\{X < x\}$ = probability of the event $X < x$
$Pr\{\cdot\|\cdot\}$	Conditional probability; probability of the event to the left of the |, given that the event (condition) to the right of the | has occurred.
$Pr\{\cdot;\cdot\}$	Probability of the event to the left of the semicolon. The events or parameters to the right of the semicolon are known. The notation is often used for emphasis or as a reminder.



(A) Union of X and Y (written X ∪ Y)



(B) Intersection of X and Y (written X ∩ Y)



(C) A set (A) and its complement (Ā)

FIGURE 3-1. Venn Diagrams Showing Set Relationships

$Cdf\{\cdot\}$ — Cumulative distribution function of the variable inside the { }; e.g., $Cdf\{X\} \equiv Pr\{X < x\}$.

$pdf\{\cdot\}$ — Probability density function of the variable inside the { }; it is the derivative of the *Cdf*, if the derivative exists.

$Sf\{\cdot\}$ — Survivor function; $Sf\{X\} \equiv Pr\{X > x\} = 1 - Cdf\{X\}$ for continuous variables

, — both/and, used as a symbol analogous to intersection; e.g., it is used to denote a joint *pdf*.

; | — Used in *Cdf*, *pdf*, *Sf*, etc., i. a fashion and with a meaning analogous to that for $Pr\{\cdot;\cdot\}$ and $Pr\{\cdot|\cdot\}$.

3-2.3 RULES, LAWS, AND DEFINITIONS FOR PROBABILITY DENSITIES

Let X, Y be suitable random variables with domains $(-\infty,\infty)$.

$$pdf\{X\} > 0 \tag{3-1}$$

$$0 < Cdf\{X\} < 1 \tag{3-2a}$$

$$0 < Sf\{X\} < 1 \tag{3-2b}$$

Let

$$f(x) \equiv pdf\{X\}$$
$$F(x) \equiv Cdf\{X\}$$
$$g(y) \equiv pdf\{Y\}$$
$$G(y) \equiv Cdf\{Y\}$$
$$h(x\ y) \equiv \text{joint } pdf \text{ of } X \text{ and } Y$$
$$H(x\ y) \equiv \text{joint } Cdf \text{ of } X \text{ and } Y$$

then

$$f(x) \equiv \text{marginal } pdf \text{ of } x$$
$$F(x) \equiv \text{marginal } Cdf \text{ of } x$$
$$g(y) \equiv \text{marginal } pdf \text{ of } y$$
$$G(y) \equiv \text{marginal } Cdf \text{ of } y$$

$$F(x) = H(x,\infty) \tag{3-3a}$$

$$G(y) = H(\infty,y) \tag{3-3b}$$

While "h or H" uniquely determines "f or F" and "g or G", "f or F" and "g or G", uniquely determining "h or H" is not true because the form of the *s*-dependence of x and y is not then known.

3-2.4 TRANSFORMATION OF VARIABLES

Let X,Y be two suitable random variables

$$f(x) \equiv pdf\{X\}$$
$$g(y) \equiv pdf\{Y\}$$
$$y = y(x)$$
$$g(y)dy = f(x)dx \tag{3-4a}$$
$$g(y) = f(x)\left|\frac{dx}{dy}\right| \tag{3-4b}$$

The form of Eq. 3-4a is usually easier to remember. Variables can be transformed directly, within a *Cdf*, with no complications at all.

3-2.5 CONVOLUTION

Let

1. Z, X, Y be suitable random variables with domains $(-\infty,\infty)$
2. $Z = X + Y$
3. $w(z) = pdf\{Z\}$
   $f(x) = pdf\{Y\}$
   $g(y) = pdf\{Y\}$
   $h(x,y) = pdf\{X,Y\}$

Then, the convolution formula is

$$w(z) = \int_{-\infty}^{\infty} h(z - y,y)dy = \int_{-\infty}^{\infty} h(x,z - x)dx = \int_{-\infty}^{\infty} h(x,z - x)dx \tag{3-5}$$

If X and Y are *s*-independent, then the convolution formula is

$$w(z) = \int_{-\infty}^{\infty} f(x)g(z - x)du = \int_{-\infty}^{\infty} f(z - y)g(y)dy = \int_{-\infty}^{\infty} f(z - y)g(y)dy \tag{3-6}$$

3-3 *s*-INDEPENDENCE AND CONDITIONAL *s*-INDEPENDENCE

The notion of *s*-independence is analogous to that for discrete distributions.

X,Y are *s*-independent random variables if and only if

$$pdf\{X,Y\} = pdf\{X\}\,pdf\{Y\} \tag{3-7}$$

The concept is the same for conditional *s*-independence. X,Y are conditionally *s*-independent random variables if and only if

$$pdf\{X,Y|C\} = pdf\{X|C\}\,pdf\{Y|C\} \tag{3-8}$$

where C = a condition (event).

Conditional *s*-independence plays a very important role in reliability calculations where redundancy is involved.

3-4 DISTRIBUTIONS

In reliability engineering the most common domain for a random variable is $(0,\infty)$. Examples of variables with the domain $(0,\infty)$ are strength, time, failure rate. In many cases where the domain is $(-\infty,\infty)$, the probabilities associated with $(-\infty,0)$ are negligible and are included only to simplify the mathematics. This is especially true for the *s*-normal distribution wherein negative values of some variables are physically meaningless; but it is convenient to integrate over the whole real line.

Continuous mathematical distributions rarely represent physical phenomena over the entire domain of the variable. Usually, however, the probabilities associated with the disturbing part of the domain are negligible. If they are not, then of course, the model must be reformulated.

3-4.1 MOMENTS

Random variables with *pdf*'s have moments. The two conventional points about which to take moments are the origin and the mean; when taken about the mean, they are called central moments. Two random variables can have joint moments, although only the second is used practically. Let X be the random variable and $f(x) \equiv pdf\{X\}$.

The nth moment (about the origin) of X is the *s*-expected value of x^n:

$$E\{X^n\} \equiv \int_X x^n f(x)dx \tag{3-9}$$

where X implies the integral over the domain

TABLE 3-1. DISTRIBUTIONS

	exponential	s-normal*	lognormal	Weibull	gamma**	uniform
parameters	λ	$\mu, \sigma, -\infty<\mu<\infty$	$\alpha, \beta, B \equiv \exp(1/\beta^2)$	$\alpha, \beta, B_n \equiv \Gamma(1+n/\beta)$, $b_n \equiv B_n/B_1^n$	α, β	$a, b; -\infty<a<b<\infty$
random variable	x	$x, -\infty<x<\infty$	x	x	x	$x; a<x<b$
pdf	$\exp(-\lambda x)$	$\frac{1}{\sigma\sqrt{2\pi}} \exp\left[-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right]$	$\frac{\beta}{x\sqrt{2\pi}} \exp\left\{-\frac{1}{2}\left[\ln\left(\frac{x}{\alpha}\right)^\beta\right]^2\right\}$	$\frac{\beta}{\alpha}\left(\frac{x}{\alpha}\right)^{\beta-1} \exp\left[-\left(\frac{x}{\alpha}\right)^\beta\right]$	$\frac{1}{\alpha\Gamma(\beta)}\left(\frac{x}{\alpha}\right)^\beta \exp\left(-\frac{x}{\alpha}\right)$	$1/(b-a)$
failure rate $\frac{pdf}{Sf}$ (if simple)	λ	goes from 0 to $+\infty$	goes from 0 to a max. then to 0	$\frac{\beta}{\alpha}\left(\frac{x}{\alpha}\right)^{\beta-1}$	monotonic	$1/(b-x)$
median	$(\ln 2)/\lambda$	μ	α	$\alpha(\ln 2)^{1/\beta}$	not tractable	$\frac{1}{2}(a+b)$
mode	0	μ	αB	$\max\left\{\alpha\left(1-\frac{1}{\beta}\right)^{1/\beta}, 0\right\}$	$\max\{\alpha(\beta-1), 0\}$	none
mean μ	$1/\lambda$	μ	$\alpha/B^{1/2}$	αB_1	$\alpha\beta$	$\frac{1}{2}(a+b)$
variance σ^2	$1/\lambda^2$	σ^2	$\alpha^2 B(B-1)$	$\alpha^2 B_1^2 (b_2-1)$	$\alpha^2\beta$	$(b-a)^2/12$
3rd central moment M_3	$2/\lambda^3$	0	$\alpha^3 B^{3/2}(B-1)^2(B+2)$	$\alpha^3 B_1^3 (b_3 - 3b_2 + 2)$	$2\alpha^3\beta$	0
4th central moment M_4	$9/\lambda^4$	$3\sigma^4$	$\alpha^4 B^2 (B-1)^2 (B^4 + 2B^3 + 3B^2 - 3)$	$\alpha^4 B_1^4 (b_4 - 4b_3 + 6b_2 - 3)$	$3\alpha^4$	$(b-a)^4/80$
coefficient of variation σ/μ	1	σ/μ	$(B-1)^{1/2}$	$(b_2-1)^{1/2}$	$1/\beta^{1/2}$	$\frac{b-a}{\sqrt{3}(b+a)}$
coefficient of skewness M_3/σ^3	2	0	$(B-1)^{1/2}(B+2)$	$(b_3 - 3b_2 + 2)/(b_2 - 1)^{3/2}$	$2/\beta^{1/2}$	0
excess coefficient of kurtosis $\frac{M_4}{\sigma^4} - 3$	6	0	$(B-1)(B^3 + 3B^2 + 6B + 6)$	$-3 + \frac{(b_4 - 4b_3 + 6b_2 - 3)}{(b_2-1)^2}$	$6/\beta$	-1.2

Some of the formulas were adapted from Ref. 1

Domain of all parameters and random variables is $(0, \infty)$ unless otherwise stated

For $\beta = 1$ this is the exponential distribution

* As is customary, the symbols μ (for mean) and σ (for standard deviation) are used for the parameters because the parameters happen to be the mean and standard deviation

** Chi square, Erlang, Pearson type III, and Maxwell distributions are special and/or reparameterized cases of the gamma distribution of X. (It is presumed that the integral converges absolutely; if not, a textbook ought to be consulted.)

The nth moment, about the mean, of X is the *s*-expected value of $(X - \mu)^n$:

$$E\{(X-\mu)^n\} = \int_X (x-\mu)^n f(x)dx \qquad (3\text{-}10)$$

where $\mu \equiv E\{X\}$.

Let X and Y be random variables

$$f(x) \equiv pdf\{X\}$$
$$g(y) \equiv pdf\{Y\}$$
$$h(x,y) \equiv pdf\{X,Y\}$$
$$\mu_x \equiv E\{X\}$$
$$\mu_y \equiv E\{y\}$$

then $\text{Var}\{X\} \equiv E\{(x-\mu)^2\}$

and (3-11)

$$\text{Cov}\{X,Y\} \equiv E\{(x-\mu_x)(y-\mu_y)\}$$
$$\equiv \int_X \int_Y (x-\mu_x)(y-\mu_y)h(x,y)dxdy \qquad (3\text{-}12)$$

The linear-correlation coefficient is defined as

$$\rho \equiv \frac{\text{Cov}\{X,Y\}}{[\text{Var}\{X\}\text{Var}\{Y\}]^{1/2}} \qquad (3\text{-}13)$$

3-4.2 DISTRIBUTIONS AND THEIR PROPERTIES

The most popular distribution for time-to-failure or time-between-failures is the exponential. There are two reasons for this popularity.

1. The distribution fits many data without doing too much violence to an engineering concept of goodness-of-fit.
2. The failure rate is a constant, and thus the distribution is very tractable.

The most popular distribution for material properties, device parameters, and generalized "stresses", "potentials", and "currents" is the *s*-normal distribution. There are two reasons for its popularity.

1. The distribution fits many data without doing too much violence to an engineering concept of goodness-of-fit.
2. The distribution is so tractable, has no parameters for the basic distribution, and convolves into itself.

Most distributions can be transformed into something that looks different by a linear transformation of the variable. Custom, more than anything else, determines what the standard form is. If a linear transformation $X = aU + b$ is applied to a distribution, the mean and variance are transformed as follows:

$$E\{X\} = aE\{U\} + b \qquad (3\text{-}14a)$$
$$\text{Var}\{X\} = a^2 \text{Var}\{U\} \qquad (3\text{-}14b)$$

There are usually several ways of writing the parameters of a distribution, e.g., a scale parameter can be used in the form λx or x/α (where x is the random variable and α,λ are parameters). The forms in Table 3-1 are chosen to be useful to reliability engineers.

REFERENCE

1. W. G. Ireson, Ed., *Reliability Handbook*, McGraw-Hill Book Company, Inc., N.Y. 1966.

CHAPTER 4 REVIEW OF ELEMENTARY STATISTICAL THEORY

4-1 INTRODUCTION

This chapter presents some of the statistical concepts which are useful in a reliability context. The Bibliography at the end of Chapter 1 gives elementary, intermediate, and advanced texts on probability and statistics. It is not the purpose of this chapter to write another textbook on statistics.

The purpose of statistics is to help people analyze real data and draw reasonable conclusions from them. In reliability engineering, the function of statistics most often will be in showing an engineer what he does NOT know from the data; i.e., statistics will provide an engineer with a feeling for the uncertainty in the conclusions he wants to draw from the data.

The few concepts of statistics that are important in reliability ought to be carefully learned. It is better not to use them than to use them incorrectly.

4-2 ESTIMATION OF PARAMETERS

It is usually convenient to summarize a mass of data by stating a distribution from which they might well have come. This usually is done by choosing a distribution (on the basis of previous ideas, simplicity, massaging of the data, or something else) and then estimating the parameters of the distribution. There are several popular methods of estimating parameters; they are not detailed here--but *Part Six, Mathematical Appendix and Glossary*, shows estimation methods for many of the popular distributions.

The important thing about an estimate is its properties, not how you got it. In these days of readily available computers, the cost of making estimates whose properties are good and well known is negligible compared to the cost of getting the original data.

4-2.1 s-EFFICIENT ESTIMATOR

For engineers, s-efficiency is what estimation is all about. Any estimator uses a statistic; that statistic has properties such as a mean value and a variance. The s-efficiency of an estimator is measured by the second moment of the estimator taken about the true value. If the estimator is s-biased (par. 4-2.3), then this second moment is "variance + (bias)^2". If the estimator is s-unbiased (zero bias), s-efficiency is measured by the variance of the estimator. For a fixed sample size, the smaller the variance of the estimator, the more s-efficient it is.

There is a lower bound to the variance of an estimator--the Cramer-Rao lower bound. s-Efficiencies often are measured relative to the Cramer-Rao lower bound; if this s-efficiency is 100 percent, that's as s-efficient as one can get. Most estimators used in reliability work are quite s-efficient.

s-Efficiency is perhaps the most desirable property of an estimator. It tells you how good or bad your estimate is likely to be.

4-2.2 s-CONSISTENT ESTIMATORS

An s-consistent estimator is one which "approaches" the true value as the sample size "goes to infinity". The reason for the quote marks is that the phrases are loose expressions of complicated mathematical concepts; for a more exact definition, consult a textbook. s-Consistency is a very desirable attribute of an estimator. Virtually all estimators in use in reliability work are s-consistent.

4-2.3 s-BIAS

s-Bias is the difference between the s-expected (mean) value of an estimator (for a fixed sampling plan) and the true value. It enters the measure of s-efficiency (par. 4-2.1); as long as the s-bias is less than about 50 percent of the standard deviation, the contribution of the s-bias can be neglected. Being s-unbiased is nice for theoretical work, but it is vastly overrated as a criterion for goodness of reliability estimators. The main reason for this is that if $\hat{\theta}$ is an s-unbiased estimator of θ, $f(\hat{\theta})$ is an s-biased estimator of $f(\theta)$ unless $f(\cdot)$ is a linear function. The most widespread misunderstanding of this principle is involved in the estimate for the variance of an s-normal distribution. The S^2 statistic, $S^2 \equiv SS/(N-1)$

--where SS is the sum of squares of deviations about the sample mean, and N is the number of items in the sample—is an s-unbiased estimator of σ^2 (the true value of the variance), but S is an s-biased estimator of σ. (The square root function is not linear.) Another example is $1/\lambda$, the reciprocal parameter for an exponential distribution. An s-unbiased estimator for $1/\lambda$ is the sample mean, but the reciprocal of that estimator is an s-biased estimator of λ. (The reciprocal is not a linear function.)

How is an engineer to know what function of the parameter ought to be s-unbiased? He doesn't. In general, reliability engineers can ignore s-bias of estimators; they need only be concerned about s-efficiency.

4-2.4 UNCERTAINTY

Any estimates of parameters ought to be accompanied by an estimate of the uncertainty involved. Two common methods of indicating uncertainty are the covariance matrix and s-confidence intervals. The reliability engineer need not know how to get them, only how to use them.

4-3 TESTS OF s-SIGNIFICANCE

The most important thing about s-significance is what it isn't; it is not "engineering importance". s-Significance is concerned with tests that are run to see if one thing is different from another. A statistical model is formulated and measurements (tests) are made on the sample(s) to measure the difference in the items of the sample. For example, does heat-treating method A produce better fatigue properties than heat-treating method B? Usually the statistical hypothesis is made that there is no difference. Then the statistical distribution of the test statistic is calculated. In the example, the test statistic might be the difference in average fatigue-strengths at 10^7 cycles of stress. The value of that test statistic for the sample(s) is measured and compared with the distribution. If a value as large as observed would occur only 0.1 percent of the time or less, the effect (difference) is not likely to have been a chance observation, but is likely to be due to one method being better than another. If the value of the test statistic for the sample(s) would be exceeded 40 percent of the time, then it is not likely that one method is better than another. The percentage chosen (0.1 percent, 40 percent, etc.) is called the s-significance level. In practice, engineers want the effect to be s-significant at a 20 percent level or less.

Regardless of the outcome of the statistical test, the engineer wants the effect to be of engineering importance. It is possible to take a sample small enough so that no matter what the actual difference is, it will not be s-significant because the uncertainties due to too few data overwhelm all other considerations. On the other hand, it is also possible to take so much data that the difference will be s-significant, no matter how small the effect. Tests of s-significance suffer from being equivalent to point estimates. Engineers would rather estimate the difference between two methods and the uncertainty in that estimate. This procedure is discussed in par. 4-4 on s-confidence statements.

4-4 s-CONFIDENCE STATEMENTS

As with s-significance there is an important difference between the engineering and statistical concepts. s-Confidence is a statistical concept with a very special, exact meaning. Don't use the concept without understanding that meaning.

An example statement is a good way to understand the concept.

> "The true improvement in fatigue strength (method B over method A) lies between −1.7 and +10.9 kips/in.^{2} at a 90 percent s-confidence level."

The 90 percent s-confidence level means that 90 percent of the times that one goes through the statistical manipulations as done for this example, the resulting statement will be correct; 10 percent of the time it will be wrong. The −1.7 and +10.9 kips/in.^{2} are called the s-confidence limits.

For a given set of sample measurements, the higher the s-confidence level is, the wider the s-confidence limits will be.

An engineer might look at the s-confidence statement and say, "Even if the improvement in fatigue strength were as good as the top limit, it wouldn't be too useful. We need an improvement of at least 20 kips/in.^{2}" There is probably little point, then, in running more tests. However, if he says, "All we need is 5 kips/in.^{2} improvement," he undoubtedly would want to run more tests to pin down the improvement more exactly.

s-Confidence is not engineering confidence, although the concepts are related.

4-5 GOODNESS-OF-FIT TESTS

When a particular distribution is assumed to represent a set of data, a natural question arises, "How good is the fit of the distribution to the data?" There are several statistical tests that can be performed. Some are peculiar to the distribution itself, and some can be applied to any distribution. The two most popular ones for application to any distribution are the Chi-Square and the Kolmogorov-Smirnov tests.

A goodness-of-fit test is equivalent to a test of *s*-significance (par. 4-3) and has all the difficulties associated with *s*-significance tests. That difficulty--briefly--is that it is possible to take so few data that it is impossible to reject any distribution, and it is possible to take so many data that every distribution will be rejected.

What is needed is a test for fit that answers an engineering question, such as, "If I use this distribution for interpolation, how bad will my answers be?" Unfortunately, such tests are not available. Therefore, a considerable amount of engineering judgment must be used in reckoning goodness-of-fit.

4-6 SAMPLES AND POPULATIONS

In practical situations the population, about which statistical inferences are to be made, is determined by the method in which the sample for testing was drawn. The use of historical data is fraught with extreme danger this way. For example, electrolytic capacitors that were derated to 50 percent or less were more reliable than those derated to, say, 70 percent of their rating; results like that were obtained in reliability studies of armed forces equipment in the 1950's. Was this sample taken from all kinds of designers, or was it taken from only a subset of designers? For example, if the designers whose equipment was measured were such that conservative designers put electrolytic capacitors in cool places and careless designers put them in hot places, the population of designers does not include those who put very derated electrolytics in hot places nor those who put mildly derated ones in cool places.

Probably the most controversial situation of samples vs populations concerns the relationship of cigarette smoking to health. Samples were taken of smokers and nonsmokers, etc., but from what population were the people a statistically random sample?

A more frequently occurring difficulty is testing a small sample of parts and then implicitly hoping that the small sample represents the population which will be obtained from several suppliers month after month.

For really important tests, the engineer has to decide what are the possibly important effects and then find an appropriate statistician to help with sampling.

4-7 IFR AND DFR DISTRIBUTIONS

Sometimes it is difficult to determine a distribution of lifetimes of a unit. It may, even then, be feasible to decide that the failure rate of the unit is always increasing (IFR → Increasing Failure Rate) or always decreasing (DFR → Decreasing Failure Rate). If a distribution is known to be IFR or to be DFR, bounds can be put on the failure behavior. One of these bounds is provided by the Constant Failure Rate distribution and its associated relationships.

For example, the Weibull and Gamma distributions (see Table 3-1 for notation) are IFR when the shape parameter β is greater than 1 and DFR where it is less than 1. Both have constant failure rates when the shape parameter is 1. The *s*-normal distribution is IFR; the lognormal distribution is neither (at first the failure rate increases, then it decreases).

A general discussion of IFR and DFR distributions is given in Ref. 1; DFR distributions are discussed in detail in Ref. 1. Bounds on reliability parameters are given in Refs. 2-5. Refs. 6, 7 discuss the conditions under which systems:

1. Made up of IFR elements, are themselves IFR.

2. Made up of DFR elements, are themselves DFR. Ref. 8 shows how to test a sample to see if it comes from a distribution with a monotonic failure rate, and if so, whether it is IFR or DFR.

Even though this mathematical material is available in the literature, it is not clear how valuable it can be to the reliability engineer. An experienced statistician ought to be consulted before applying any of the results. The reliability engineer must also use his judgment in deciding how much less stringent the restrictions for this theory really are, than just to blithely assume one of the conventional distributions.

Generally speaking, the decisions about hardware will not be radically different regardless of which of several distributions is chosen to represent the life of the units. If that conclusion is not true, then the engineer is in serious trouble because he needs more information than he has.

REFERENCES

1. F. Proschan, "Theoretical Explanation of Observed Decreasing Failure Rate", *Technometrics* 5, No. 3, 375-83 (1963).
2. R. E. Barlow and A. W. Marshall, "Bounds for Distributions with Monotone Hazard Rate", D1-82-0247, Boeing Scientific Research Laboratories, 1963.
3. R. E. Barlow and A. W. Marshall, "Tables of Bounds for Distributions with Monotone Hazard Rate", D1-82-0249, Boeing Scientific Research Laboratories, 1963.
4. R. E. Barlow and F. Proschan, "Comparison of Replacement Policies, and Renewal Theory Implications", D1-82-0237, Boeing Scientific Research Laboratories, 1963.
5. R. E. Barlow and F. Proschan, *Mathematical Theory of Reliability*, John Wiley and Sons, New York, 1964.
6. J. D. Esary and F. Proschan, "Relationship Between System Failure and Component Failure Rates", *Technometrics* 5, No. 2, 183-9 (1963).
7. R. E. Barlow, A. W. Marshall, and F. Proschan, "Properties of Probability Distributions with Monotone Hazard Rate", *Annals of Mathematical Statistics* 34, No. 2, 375-89 (1963).
8. F. Proschan and R. Pyke, *Tests for Monotone Failure Rate*.

CHAPTER 5 SOME ADVANCED MATHEMATICAL TECHNIQUES

5-0 LIST OF SYMBOLS

n = number of states
s- = denotes statistical definition
S_i = system-state i
t = time
u = time at which in-repair unit fails; regeneration point
λ_{ij} = transition rate from S_i to S_j

5-1 INTRODUCTION

The approach to reliability wherein transition distributions from one state to another are all general is not tractable, because there are no simple instants of time at which past history can be ignored. The best that can be done in the general case is to give a complicated algorithm for calculating probability of transition at any time. Therefore, everyone uses simplifying assumptions of some sort. A few of the mathematical techniques that are useful in the simplification process are mentioned here. None were discovered or invented for reliability analysis; they are well-known (to mathematicians) techniques Refs. 1 and 2 give more details on many of them. Handbooks such as Ref. 6 also show these and other techniques; Ref. 7 is an example of a textbook which teaches some of these techniques.

5-2 MARKOV PROCESSES

There are several kinds and generalizations of Markov processes, but only the most simple process will be discussed here. For more details, see Refs. 1 and 2 and the Bibliography at the end of Chapter 1.

5-2.1 SYSTEM STATE

The system is presumed to be in one of a set of states and can go from one state to another. The state of a system is a description of its condition. The analyst can choose the way a state is characterized. Consider this example. Suppose a system consists of three subsystems, each of which can be adequately described by one of the following four conditions: Good, Degraded, Failed waiting for repair, In repair. Further suppose that the state of the system is characterized adequately by giving the states of each of the three subsystems. Then there are 4 X 4 X 4 = 64 possible states of the system. A state of this system consists of the specification of the states of each of its three subsystems, e.g., Good, In repair, Good. When the state of a subsystem changes, the state of the system will change.

5-2.2 MARKOV CHAINS

Suppose the states of the system are specified, e.g., $S_1, ..., S_n$, then there are n states. It is presumed that the probability of going from one state to another depends only on those two states, and no others; past history is wiped out. For any two states, the transition rate is a constant. The transition rate λ_{ij} from state S_i to state S_j corresponds to a failure rate for an exponential process in that it is a ratio of a probability density function to a Survivor function. Many of the λ_{ij} for a system are usually zero, because certain transitions are not possible, by the very nature of the particular system. In the example in par. 5-2.1, just repaired subsystems might always be Good, never Degraded. Then, a subsystem could never go from "In-repair" to "Degraded", but it could go from "In-repair" to "Good" or from "Degraded" to "In-repair". The λ_{ij} can be put in a matrix form.

Many special cases have been worked out in the literature. Refs. 3-5 are likely sources of material.

Considerable simplification of the theory is possible when only the steady-state behavior of the system is of concern, not the transient (start-up) behavior.

In practice, the number of system states must be severely limited in order for the analysis to be tractable.

5-3 LAPLACE TRANSFORMS

The Laplace transform is perhaps the most popular transform for engineers; they use it often in solving differential equations.

The Laplace transform is very closely related to the Laplace-Stieltjes transform and to the Fourier transform. The Moment Generating function and the Characteristic function are also related to the Laplace transform, although statistics texts seem rarely to point this out. (The Characteristic function is, formally, the Fourier transform; and the Moment Generating function is, formally, the Laplace transform.) The Stieltjes form of the Laplace transform has fewer difficulties with "existence" than does the Laplace transform, although in practical reliability work, "existence" of integrals and *pdf's* is rarely a difficulty. In the remaining discussion, the phrase, Laplace transform, includes all the related transforms and functions.

The Laplace transform changes differentiation and integration into multiplication and division by the transform-variable. In reliability analysis, another of its properties is even more important. The Laplace transform of the sum of several *s*-independent random variables is the product of the individual Laplace transforms of the random variables. Thus convolution is transformed to multiplication.

When the equations of the system are expressed in Laplace transforms, the steady state ($t \rightarrow \infty$) behavior can be found easily without inverting the transforms.

The Laplace transform of the answer in a reliability problem often can be obtained in a closed form, albeit usually unwieldy. The difficulty arises because inversion is rarely feasible in closed form; then numerical inversion must be used.

5-4 REGENERATION POINTS

The big advantage of assuming constant transition rates, is that every time-instant is a regeneration (renewal) point. Statistically ... the system (when in a particular ...) has no memory as to how long it has been in that state; each instant is just like every other instant.

If general statistical distributions are used, this is no longer simply the case. The trick in an analysis is to find (or invent) some time instants which have this regeneration property; once you know that the system is at this time instant, its past history can be forgotten. One way of finding suitable regeneration points is to introduce an extra time variable to help describe the state of the system.

For example, suppose a system of two units is in one of the following three states:

1. One unit operating, other in-standby
2. One unit operating, other in-repair
3. One unit in-repair, other waiting-for-repair.

The unit is in state two at time = t; introduce the time = u at which the in-repair unit fails; at time = 0 the operating unit was put into operation. With u as an extra variable, time = u is a regeneration point; the state probabilities do not depend upon the history of the system prior to u.

Of course, the introduction of extra variables complicates the analysis, but, at least, some equations can be written down. This supplementary variable technique is used in the literature, e.g., Ref. 5, in order to "solve" reliability problems where random variables have unspecified distributions. Virtually all problems when stated this way will involve the sums of *s*-independent random variables; so Laplace transforms will ordinarily be used in the solution of the problem (see par. 5-3).

Ref. 7 discusses renewal theory in detail.

REFERENCES

1. DA Pam 70-5, *Mathematics of Military Action, Operations and Systems*,
2. M. L. Shooman, *Probabilistic Reliability*, McGraw-Hill Book Company, Inc., N.Y., 1968.
3. Gnedenko, Belyayev, and Solovyev, *Mathematical Methods of Reliability Theory*, Academic Press, N.Y., 1969.

4a. *Proceedings of the Annual Symposia on Reliability*, 1966-1971.

4b. *Proceedings of the Annual Reliability and Maintainability Symposia*, 1972-present.

5. *IEEE Transactions on Reliability*.
6. G. A. Korn, T. M. Korn, *Mathematical Handbook for Scientists and Engineers*, McGraw-Hill Book Company, Inc., N.Y., 1968.
7. W. Feller, *An Introduction to Probability Theory and its Applications*, Vols. I, II, John Wiley & Sons, Inc., N.Y.; Vol. I, 1968, Vol. II, 1966.

CHAPTER 6 CREATING THE SYSTEM RELIABILITY MODEL

6-0 LIST OF SYMBOLS

k-out-of-n:F	=	special kind of system, see par. 6-3.2
k-out-of-n:G	=	special kind of system, see par. 6-3.2
MTF	=	Mean Time to Failure
$MTBF$	=	Mean Time Between Failures
s-	=	denotes statistical definition
t	=	time, time-to-failure
$X,Y,Z,A,B,S,...$	=	events or elements on a dependency diagram
Ψ_i	=	subsets of Ψ; Ψ is any event or set
$\Psi',\Psi'',\Psi_{(i)}$	=	events related to Ψ; Ψ is any event
$\bar{\Psi}$	=	not Ψ complement of Ψ; Ψ is any event or set
AND, OR	=	logical operators (AND $\rightarrow \cap$; OR $\rightarrow \cup$)
$\Delta,\nabla,\circ,\square$	=	symbolic elements for a dependency diagram; see par. 6-2.3.1

6-1 INTRODUCTION

In order to compute the reliability measures of a system, it is necessary to develop a reliability model of the system. A reliability model consists of some combination of a reliability block diagram or Cause-Consequence chart, a definition of all equipment failure and repair distributions, a definition of the up-state rules, and a statement of spares and repair strategies. This chapter is written from the point of view of reliability diagrams, because historically the material has been presented that way.

A reliability block diagram is obtained from a careful analysis of the manner in which the system operates, i.e., the effects on overall system performance of failures of the various parts that make up the system; the support environment and constraints, including such factors as the number and assignment of spare parts and repairmen; and the mission. Careful consideration of these factors yields a set of rules (which will be referred to as "up-state rules") which define satisfactory operation of the system (system up) and unsatisfactory operation (system down), as well as the various ways in which these can be achieved. If a system operates in more than one mode, a separate reliability diagram must be developed for each one (Refs. 1 and 2).

A considerable amount of engineering analysis must be performed in order to develop a reliability model. The engineer proceeds as follows.

(1) Develop a functional block diagram of the system based on his knowledge of the physical principles governing system operation and behavior.

(2) Develop the logical and topological relationships between functional elements of the system.

(3) Use the results of performance evaluation studies to determine the extent that the system can operate in a degraded state. This information might be provided by outside sources.

(4) Define the spares and repair strategies (for maintained systems). The spares strategy defines the spares allocated to the system and, in the case of multiple failures, defines the order in which spares are to be used. The repair strategies define the number of repairmen and the order in which they are to be used in the case of multiple failures.

This chapter presents a description of the engineering analysis procedures, mathematical block-diagramming techniques, and other procedures used to construct reliability models.

6-2 ENGINEERING ANALYSIS

6-2.1 INTRODUCTION

Before the reliability model can be constructed, the system must be analyzed. A functional block diagram and a dependency diagram, which define the logical and topological relationships between functional elements and their inputs and outputs, must be developed. These diagrams can be developed for electrical, electromechanical, and mechanical systems — the underlying principles are the same for all (Refs. 2 and 3).

Basically, the functional block diagram must contain the following items:

1. A clear identification of all functions and repetitive functions.
2. Input-output relationships between functions. For electronic systems, this takes the form of signal flow from input to output. Usual and alternate modes must be shown.
3. A clear indication of where power supplies or power sources are applied to the system.
4. Description of switching arrangements and the sequence in which alternate modes are used.

The dependency diagram schematically represents the logical interdependencies of the functional elements of the system and illustrates step-by-step how an input is processed to produce the output signal or mechanical action (Refs. 2 and 4).

Notes and attachments can be used to provide more detailed information on a specific system than can be portrayed directly on the dependency diagram. An alphameric code ought to be established which correlates the dependency diagram with the functional block diagram.

The reliability block diagram for the case of reliability without repair can be derived directly from the dependency diagram using the techniques of Boolean algebra. For repairable systems, simple modifications that describe the spares and repair strategies must be made to the basic block diagram.

6-2.2 FUNCTIONAL BLOCK DIAGRAMS

Functional block diagrams must be developed to provide descriptive coverage from system to subassembly levels. The information contained in them and in the detailed circuit and mechanical descriptions of the system can be used to develop a reliability model. The functional block diagrams, circuit diagrams, mechanical descriptions, dependency diagrams, and reliability block diagrams are related by means of an alphameric coding scheme.

Notes and attachments to the functional block diagrams must (1) provide more-detailed information than can be portrayed directly on the functional block diagrams, and (2) describe functional relationships whose complexity precludes direct listing. Typical attachments to the functional block diagrams include timing diagrams, switching rules, and descriptions of complex interconnections between functions.

Several levels of functional block diagram might be required. System-level functional block diagrams show the relative locations of the highest level functional elements in the system, their interconnections, relation to the external environment, power levels, and points of access to external systems. Basic system mechanical layout information (such as physical boundaries) is superimposed on the system functional block diagram.

Depending on the system being described, several levels of intermediate functional block diagrams might be required. The intermediate-level functional block diagrams are identical in structure and format to the system diagrams, but describe the system in greater detail. When basic equipment layout information is available, it is superimposed on the intermediate-level block diagrams.

Many systems require several levels of mechanical descriptions. At the overall coverage level, gross physical details are superimposed on the system block diagram. At intermediate levels, more-detailed physical features are defined. This is important because hardware boundaries are needed to specify equipment configurations for which reliability must be computed. The definition of physical configuration is important when repairable systems are being analyzed because the repair times are a function of accessibility and ease of handling, which are physically related parameters.

The structure of the functional block diagrams and the physical descriptions depend on the system. A tank, for example, has a very well-defined physical structure and functional block diagram. On the other hand, a tropospheric-scatter communications system has large, interconnected units dispersed over a site area.

6-2.2.1 Discrete Systems

A discrete system has precisely defined mechanical and electrical boundaries, and it occupies a limited, well-defined volume. Examples of such systems are rifles, artillery projectiles, tanks, and helicopters. A functional and mechanical description of a discrete system usually can be prepared in a straightforward manner. The reliability block diagrams usually are derivable readily from the descriptions.

A traditional radio receiver is an example of a simple discrete system; see Fig. 6-1 (Ref. 5). The system-level functional block diagram describes the functional elements of the system and defines the signal flow and interconnections between the functional elements. All functional blocks are numbered and are keyed to the blocks of the reliability model.

A more complex discrete system is the infrared (IR) camera in Fig. 6-2 (Ref. 6). This system contains mechanical, optical, and electrical subsystems. These subsystems can be completely described by functional block diagrams of different levels of complexity. For example, the mirrors can be described by a single level block diagram, while the IR detector may require several levels of functional block diagrams and detailed circuit schematics for a complete description.

A tank is an example of an even more complex discrete system; it contains mechanical, electromechanical, and electronic components and subsystems. Because of the way a tank is structured, a simple functional block diagram which places the functions in a simple geometrical order with a signal flow from input to output cannot be drawn. The system-level block diagram of a main battle tank is shown in Fig. 6-3 (Ref. 7).

6-2.2.2 Dispersed Systems

In a dispersed system the components are dispersed over an area and often fit together in a complicated way that requires multiplexing of signal paths and feedback. It may be difficult to describe such a system with a single set of functional block diagrams; a more complex representation might be required.

A tropospheric-scatter system is a good example of a dispersed system (Ref. 8). Tropospheric-scatter transmission systems are used to extend line of sight communication systems by using atmospheric refraction to transmit high-frequency waves beyond the horizon. Direct transmission between two terminal stations located beyond the optical horizon is obtained by the scattering properties of the troposphere. Since the transmission properties of the atmosphere randomly fluctuate, many properties of a tropospheric-scatter system are statistical. This complicates the functional description of the system be-

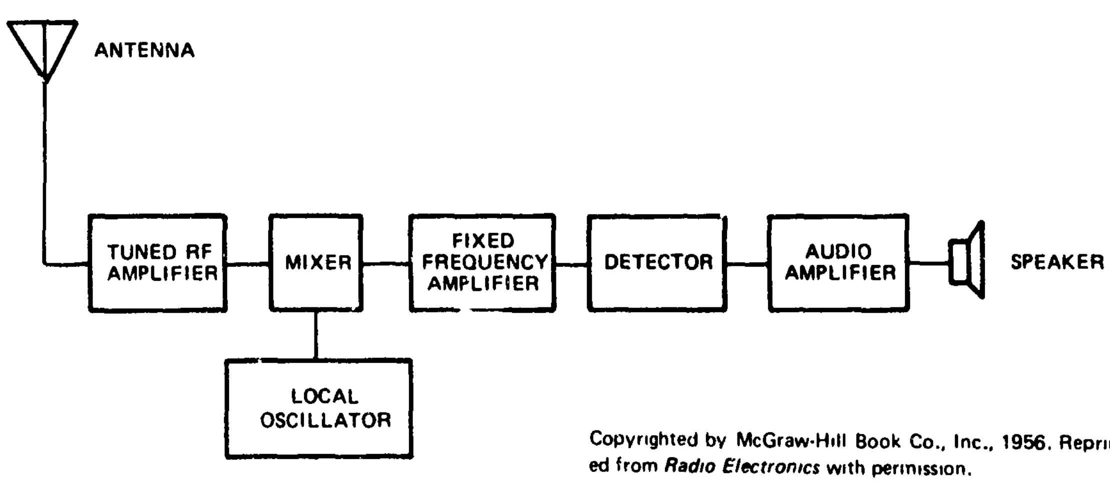


FIGURE 6-1. Radio Receiver Functional Block Diagram[5]

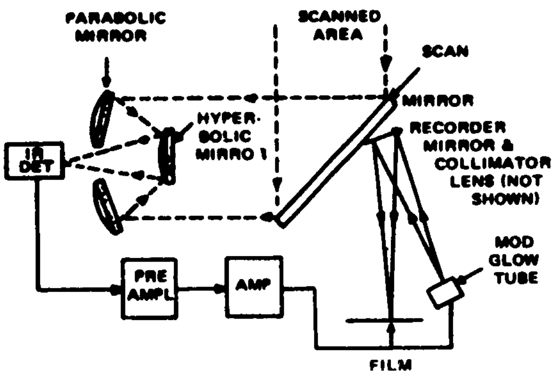


FIGURE 6-2. Infrared Camera Functional Block Diagram[6]

cause the properties of the transmission path, which is external to the system hardware, affect system reliability. Therefore, the transmission medium also must be described in the system functional block diagram.

A summary of the items making up a tropospheric-scatter system functional description follows:

1. Geographical deployment plan
2. Station layout plan
3. System layout plan
4. Shelter layout plan
5. Antenna layout plan
6. Channeling plan
7. Frequency allocations plan
8. Equipment lists
9. Tabulation of system and equipment characteristics
10. Functional block diagrams of equipment and systems at each station
11. Signal dependency diagrams
12. System interface diagrams
13. Individual functional block diagrams.

The reliability model for this system is very complex. Several reliability models will be required to compute system reliability and the reliability of individual equipments.

A System Layout Plan and an Equipment Functional Diagram for one station are described in Figs. 6-4 and 6-5.

6-2.3 DEPENDENCY DIAGRAMS

6-2.3.1 Definition of Terms

A dependency diagram pictorially defines the logical, electrical, and topological interrelationships between the events and functional elements in a system (Refs. 2 and 4). The terms used in the previous sentence are defined as follows:

1. The logical interrelationships between functional elements are the rules governing the interplay between input and output signals or forces. These rules can best be expressed by Boolean equations.
2. The electrical interrelationships describe the flow of electrical energy between functions. A good example is a traditional signal flow diagram.
3. The topological relationships express the geometric structure of the system. This is very important because, frequently, the components comprising a function are physically located in different parts of the system, even in different equipment cabinets. Therefore, the system geometry must be carefully defined.

The dependency diagrams can be very helpful in deriving reliability block diagrams. A reliability model for reliability without repair can be derived directly from these diagrams using Boolean algebra techniques. In simple systems, ordinary functional diagrams are sufficient to derive the reliability model. The dependency diagram can become very complex for large systems. Therefore, it should be constructed at a system level which permits the reliability model to be derived but does not expand the diagram to the point where it becomes cumbersome to use. A dependency diagram would never be drawn at the circuit schematic level, for example. The dependency diagram requires standard formatting rules, which minimize the chance of error when deriving the reliability model

6-2.3.2 Standard Formatting Rules

A standard set of dependency-diagram

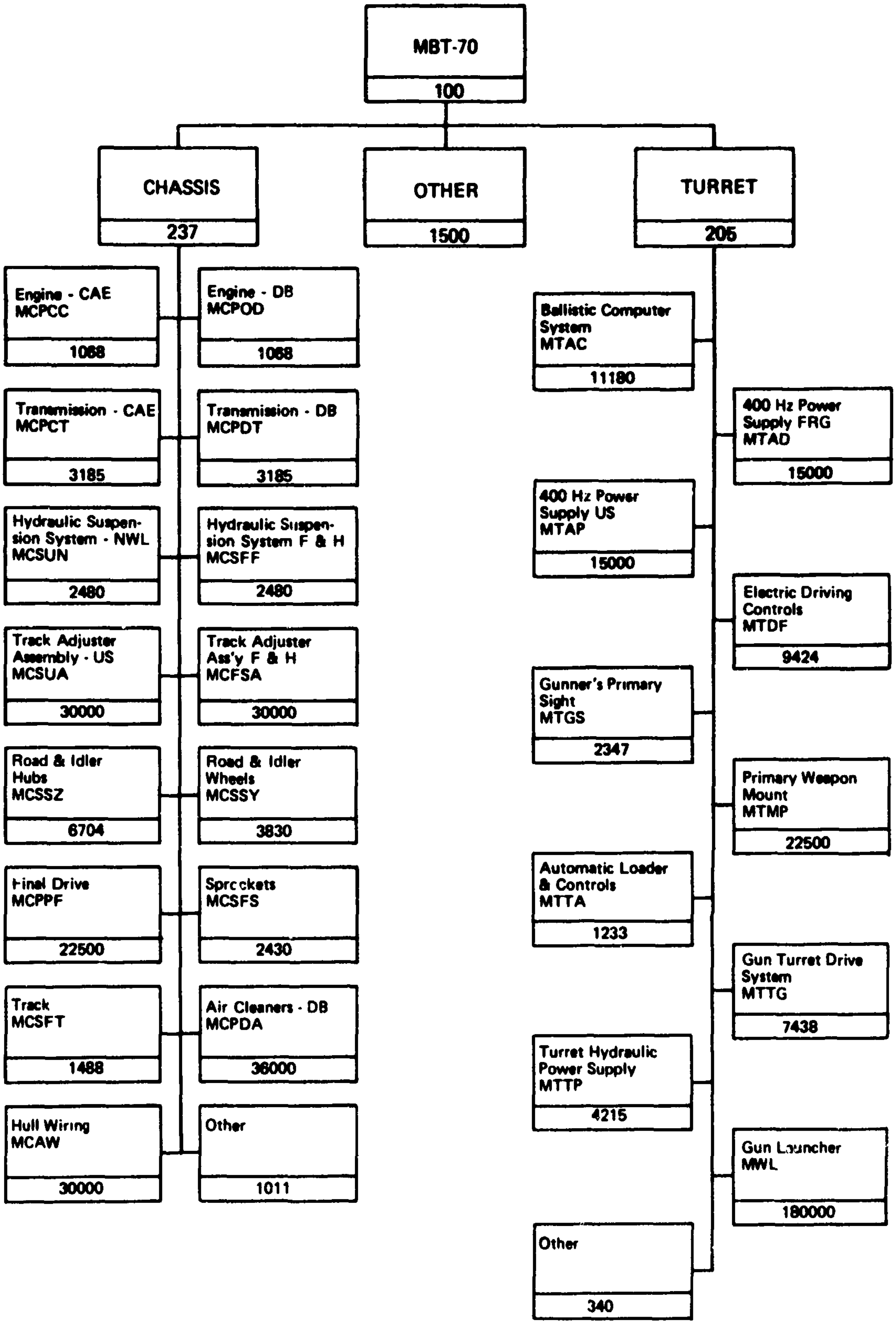
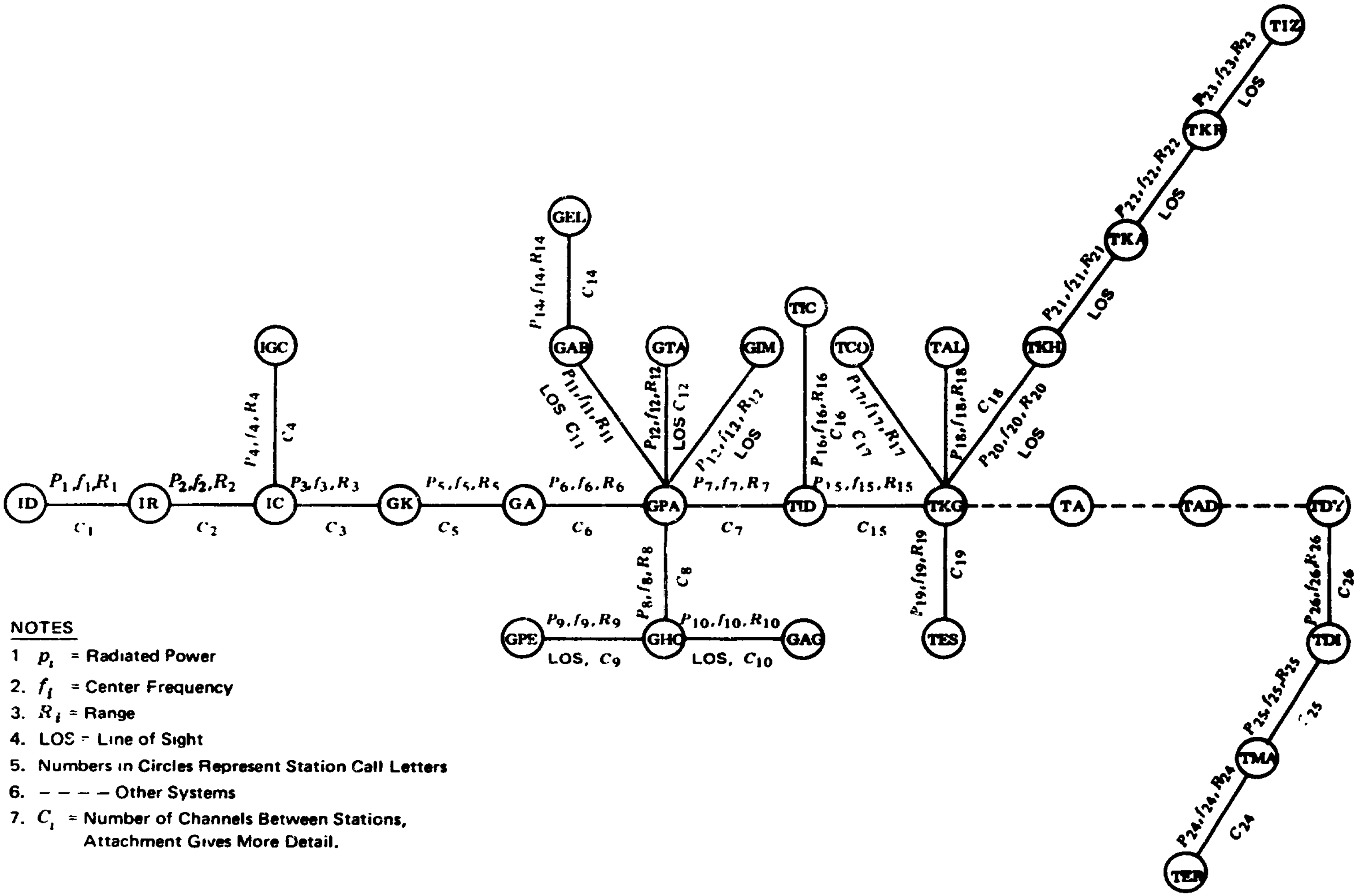


FIGURE 6-3. Functional Diagram of the MBT-70 Tank[7]



NOTES

1. p_i = Radiated Power
2. f_i = Center Frequency
3. R_i = Range
4. LOS = Line of Sight
5. Numbers in Circles Represent Station Call Letters
6. – – – – Other Systems
7. C_i = Number of Channels Between Stations, Attachment Gives More Detail.

FIGURE 6-4. Tropospheric Scatter System Layout Plan (Ref. 8)

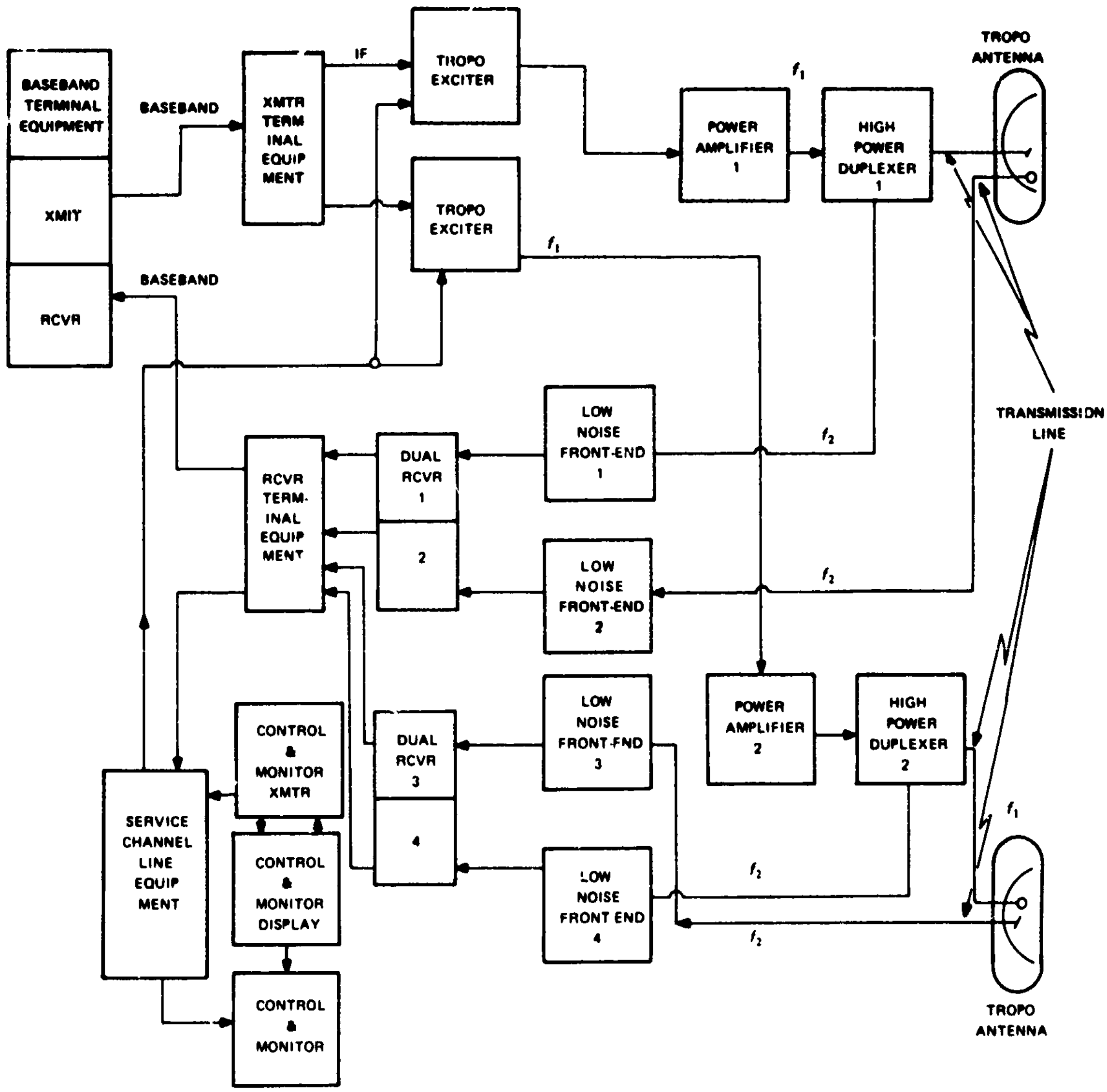


FIGURE 6-5. Equipment Functional Diagram for Tropo Terminal, Station X[8]

formatting rules is required to show unambiguously the logical relations between system functions. (This entire subparagraph is adapted from Refs. 2 and 4.) To be useful, the formatting rules should be uniform, i.e., the same set of symbols and rules must be usable at all levels of system disclosure.

The basic symbolic elements of the dependency diagram are described:

- Δ or ∇ The triangle indicates the existence of a dependency on another event. The apex of the triangle points toward the event which is depended upon.
- ○ The circle placed on a dependency line (in a particular column) indicates the existence of functional element represented by that column.
- □ The square represents an event or multiplicity of events (action or available output) which results from the proper operation of a specific group of functional elements and the availability of specific events.

By use of these basic symbols, a dependency diagram can be developed. The dependency diagram symbolically illustrates the interdependencies between the functional elements and events in the system. The dependency diagram maps the functional interactions of a system into a dependency structure.

In addition to the basic symbols, the dependency diagram also makes use of:

1. Event entries (headings)
2. Functional element entries (headings)
3. Data rows
4. Notes and signal specifications
5. Procedure column.

All of these contain information which is useful for the generation of reliability models.

The column headings list the name and location of all events and functional elements associated with the dependency diagram. Each event and functional element is identified by means of an alphameric code.

The event entries can indicate:

1. Inputs from external equipment
2. Important internal events
3. Outputs to external equipment
4. Terminal events such as outputs from recorder, PPI scope, or headphone set.

If the events are to be observed, such as at test points, the point of observation is indicated in the event entry column. If events are to be measured, the points of measurement are indicated. Specifications or descriptions for the event are referenced by a number located in a box at the base of the column heading. The physical location of each functional element and event is identified at the top of each column. The combinatorial rules governing groups of events and functional elements can be summarized in the headings.

A set of standard interpreting rules for logical, mechanical, electrical, and topological interrelationships between functional elements and events in a system must be used in the dependency diagram. The distinction between topological, electrical, logical, and mechanical considerations is crucial in the formatting of complex systems.

Topological relationships depict the physical interconnections between functional elements. Electrical interrelationships indicate functional signal processing interactions between elements. Logical dependencies indicate the Boolean relationships among functional elements. Mechanical dependencies indicate mechanical interactions between elements in a mechanical system.

The three basic symbols (triangle, circle, square) are combined in various ways to form the dependency structure. The resultant event and the functional elements and dependencies upon which it depends are connected by means of the horizontal dependency lines.

There are nine standard rules for interpreting the structure of the dependency chart for reliability model derivation, i.e.,

1. If a circle (functional element) appears in a specific column several times, it represents only one physical entity.

2. Only AND dependencies can be depicted on a single dependency line.

3. Output events dependent upon a specific functional element are placed to the right of the symbol representing that element. Input events to that element appear to the left of the element.

4. Both logical (in the Boolean sense) AND and OR dependencies can be represented in the vertical direction.

5. The vertical lines demarking the columns delimit physical bounds on the functional (electrical and mechanical) interdependencies. Several event boxes labeled separately and drawn in the same vertical column represent a group of signals which enter the same physical terminal. If the events are drawn one each in a group of adjacent columns, they represent signals that enter different physical terminals of the same functional element.

6. If separately labeled events are drawn in the same column and a dependency triangle is placed under each, the events represent electrically (or mechanically) distinct signals, even though they may be imposed at the same physical point. (Distinct signals or forces are separated by time as well as frequency.) If a single dependency triangle is placed under the group of events, they are electrically (mechanically) similar.

7. A plus sign (+) on the dependency diagram indicates that some group of functional elements and events are related in a logical OR fashion.

8. A small circle (○) or dot placed on the dependency diagram above the square representing an event indicates that the functional elements providing inputs to that event are related in a logical AND fashion.

9. Dummy Events: If groups of events are related in a complex manner that is difficult to describe using the listed rules, or if the resulting descriptions are ambiguous, a dummy event can be used. All of the event outputs feed as inputs to the dummy event. The Boolean relation or logical rule governing the interaction between the elements is stated in the column heading above the dummy event and just above the box representing the event.

These rules establish the dependency diagram as a device for describing the topological, mechanical, logical, and electrical relationships which govern the operation of a system. A number of examples presented to illustrate the application of these rules follow:

A. *Simple Series Dependency*. The simple series dependency for a single functional element is shown in Fig. 6-6. The small circle above the square (which represents the Z output) indicates an AND series relationship between X, Y, and Z. This representation may be extrapolated to a group of series functional elements.

B. *Parallel Inputs* (Figs. 6-7 through 6-10). Several possible combinations can occur. The events A_1, A_2, and A_3 enter functional block S through the same terminal or different terminals, they are electrically (mechanically) similar or electrically (mechanically) different, and the event A_4, depends upon A_1, A_2, and A_3 in a logical AND or logical OR fashion. Eight different dependency diagrams can be drawn.

1. *Identical Inputs, Same Terminal, AND Dependency* (Fig. 6-7). Standard rules 2, 3, 5, 6, 8, and 9 apply.

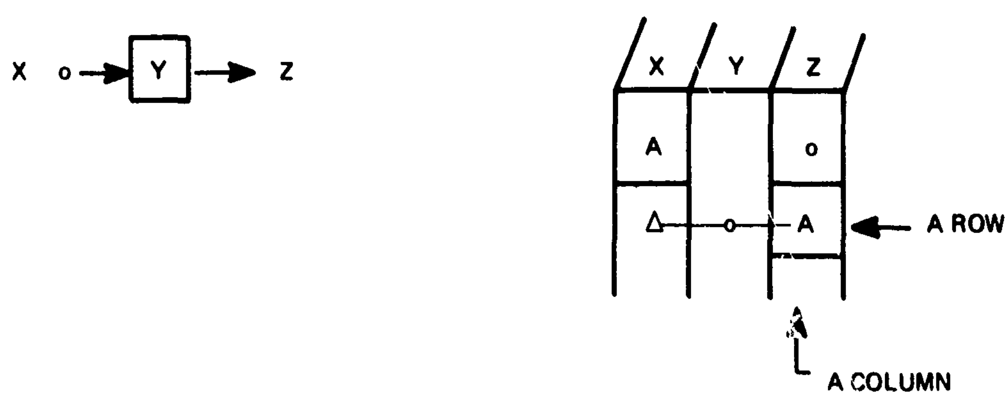


FIGURE 6-6. Simple Series Dependency[2]

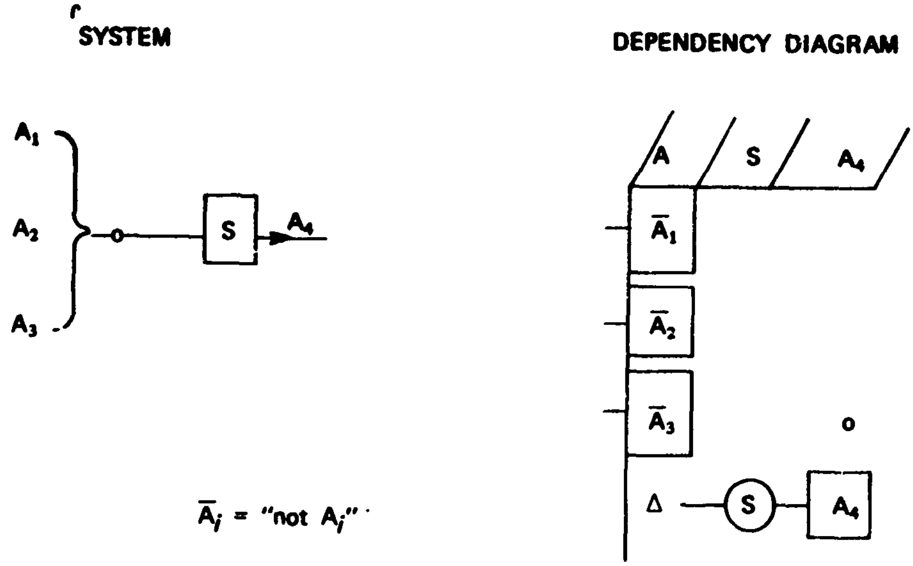


FIGURE 6-7. Identical Electrical Signals, Same Terminal, AND Dependency[2]

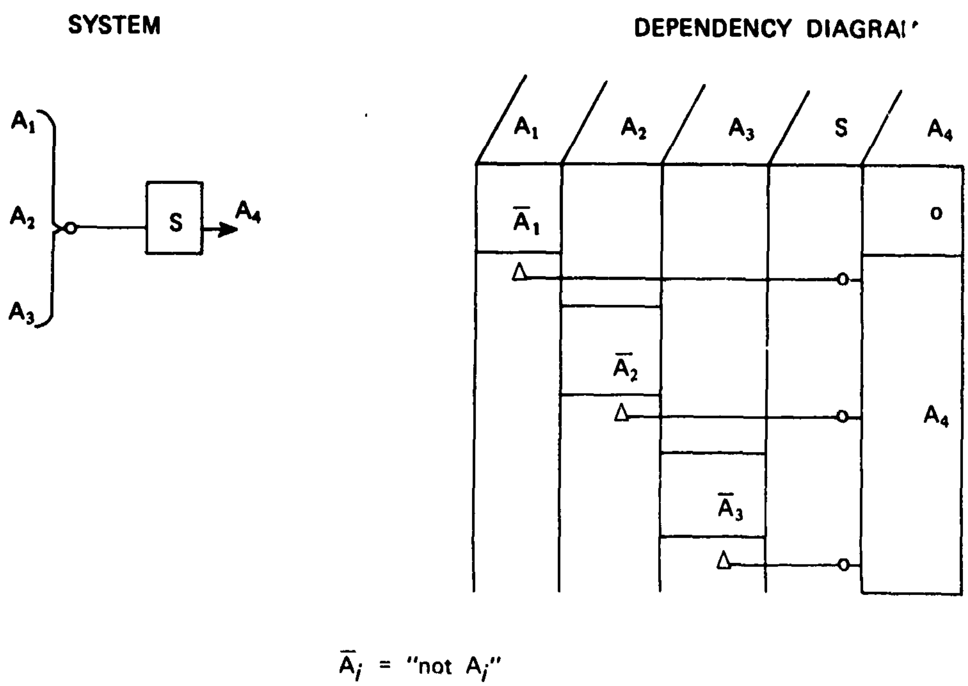


FIGURE 6-8. Identical Electrical Signals, Different Terminal, AND Dependency[2]

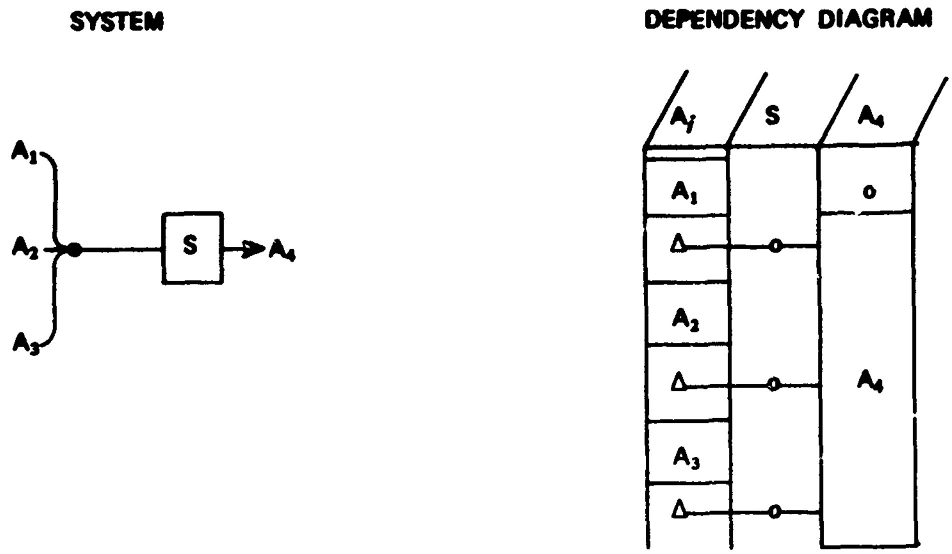


FIGURE 6-9. Different Electrical Signals, Same Terminal, AND Dependency[2]

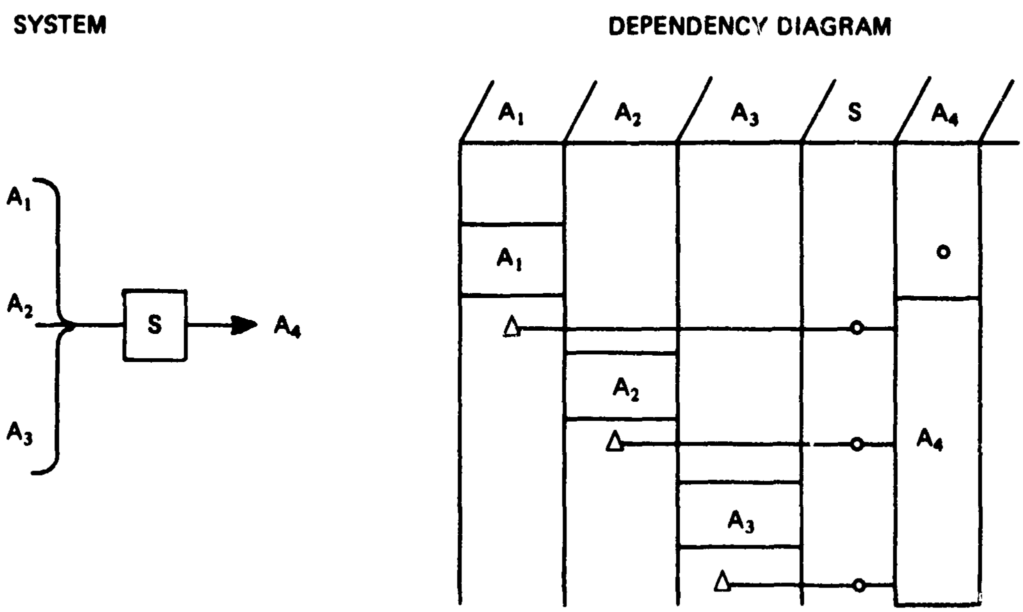


FIGURE 6-10. Different Physical Terminals, Electrically Different Signals, AND Dependency[2]

2. *Identical Inputs, Same Terminal, OR Dependency*. According to Rule 7, a (+) sign would be placed above the A_3 box because of the OR dependency. If the logical rule were combinatorial, the statement *m* (*n*), meaning *m* of *n*, would be placed next to this (+) sign.

3. *Identical Inputs, Different Terminals, AND Dependency* (Fig. 6-8). Standard rules 1, 3, 4, 5, 6, 8, and 9 apply.

4. *Identical Inputs, Different Terminals, OR Dependency*. An OR sign (+) would be placed above A_4 by Rules 7 and 4.

5. *Different Inputs, Same Terminal, AND Dependency* (Fig. 6-9). Signals A_1, A_2, and A_3 are different electrically (frequency or time wise). Rules 1, 2, 3, 5, and 6 apply.

6. *Different Inputs, Same Terminal, OR Dependency*. An OR sign (+) would be placed above A_4 by Rules 7 and 4.

7. *Different Physical Terminal, Different Inputs, AND Dependency* (Fig. 6-10). A_1, A_2, and A_3 are different.

8. *Different Physical Terminal, Different Inputs, OR Dependency*. An OR symbol would be placed above A_4.

C. *Large Numbers of Functional Branches in Parallel (Contractions)*. In this situation, a functional element B interfaces with N parallel branches, consisting of M elements in series (Fig. 6-11(A)). The format of the dependency diagram depends on whether or not the branches are identical and whether or not the functional elements within each branch are identical. Several cases must be considered:

1. All MN functional elements are different.
2. All elements in a given branch are identical, but each branch is different.
3. All elements in a given branch are different, but each parallel branch is the same.
4. All elements are identical.

Under certain circumstances, when large numbers of elements are involved, contractions can be used to simplify the dependency diagram. Examples follow:

Case 1: All MN elements are different. No contractions are possible.

Case 2: All elements in a given branch are identical, but each branch is different. The branch can be contracted by means of a multiple column contraction, Fig. 6-11(B). E represents a functional block composed of F, G, and H in series. E and its composition are described in the column heading. The resultant dependency diagram is Fig. 6-11(C).

Case 3: All elements in a particular branch are different, but all branches are identical. The multiple row contraction can be used, but not the multiple column contraction.

Case 4: All elements in all rows are identical. A further contraction is possible. This is called the multiple row contraction and is illustrated in Fig. 6-11(D). The N in the lower right hand corner of the event box indicates the number of parallel branches that are represented. This contraction is only possible when all the D_N outputs are impressed upon a single functional entity.

6-2.3.3 Examples

Several examples illustrate the wide variety of systems whose operation can be represented by dependency diagrams:

1. A simplified tropospheric-scatter system (electronic)
2. A relay (electromechanical)
3. A packaged speed reducer (mechanical).

A block diagram and dependency chart are given for each system.

A. *A Simplified Tropospheric Scatter System (Electronic)*. The functional block diagram of the receive functions of a tropospheric scatter system is given in Fig. 6-12 and its dependency diagram in Fig. 6-13 (Refs. 2 and 8). The dependency diagram is drawn at the system level for simplicity. Diagrams also can be drawn for each of the functions. The functional block diagram is only one of the several descriptive techniques required for a tropospheric scatter system; however, a detailed system description which includes geographical deployment plan, station, layout plan, system layout plan, etc., is not presented here.

B. *A Relay (Electromechanical)*. The functional block diagram of a relay is shown in Fig. 6-14 and its dependency diagram is shown in Fig. 6-15 (Refs. 2 and 9). The relay

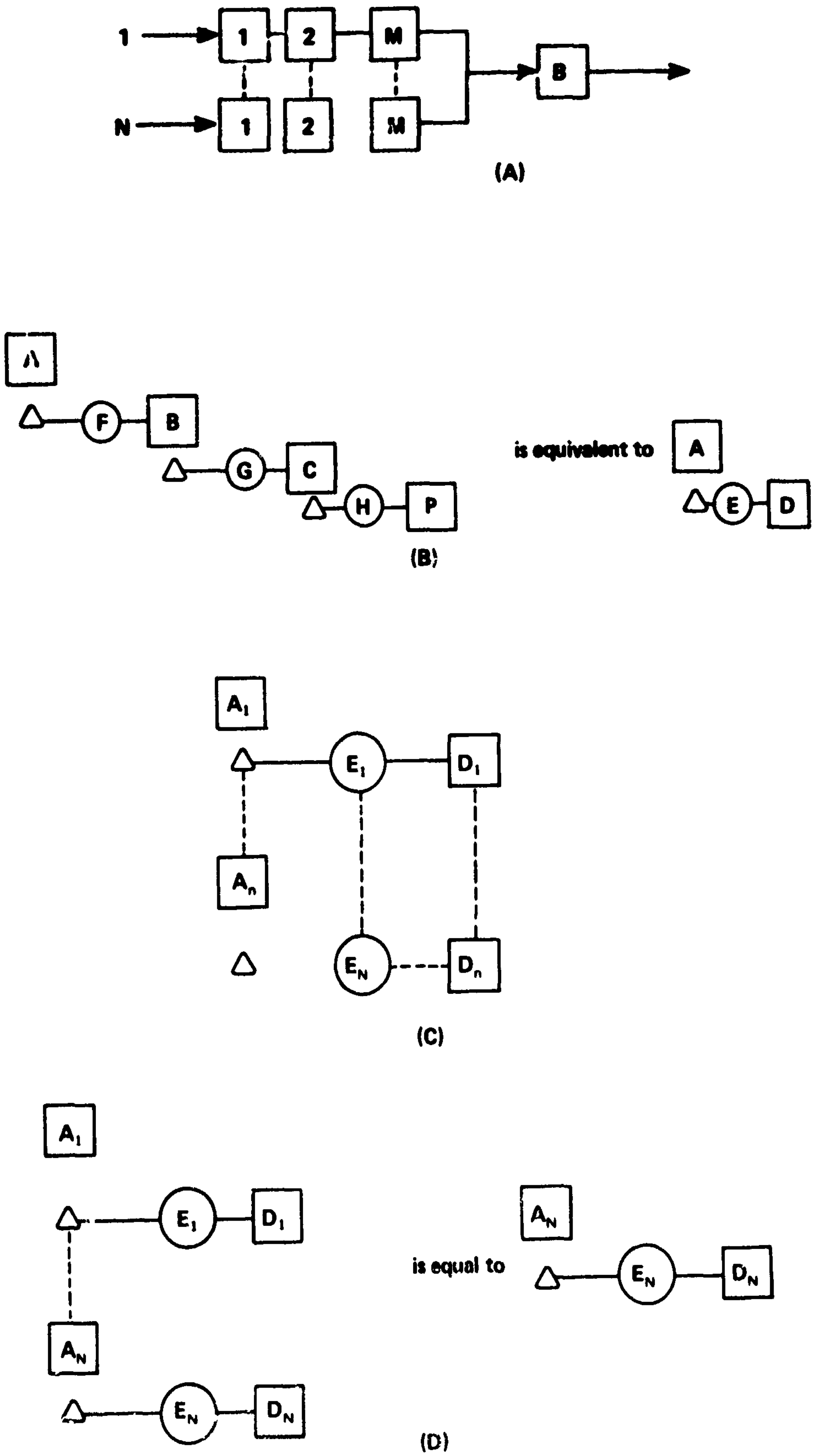


FIGURE 6-11. Large Numbers of Functional Branches in Parallel[2]

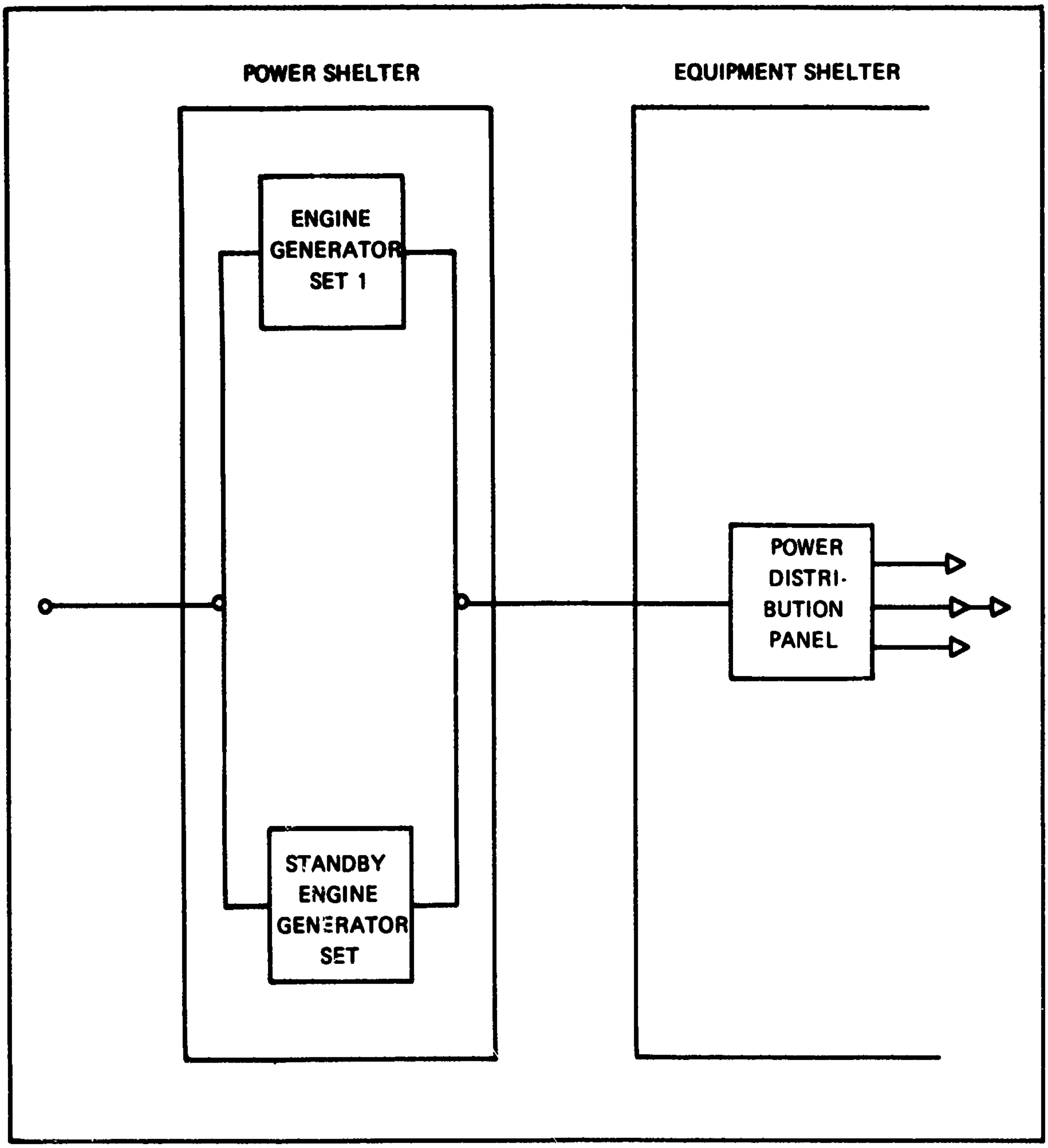


FIGURE 6-12. Power Supply Section of Tropospheric Scatter System Receive Function[2]

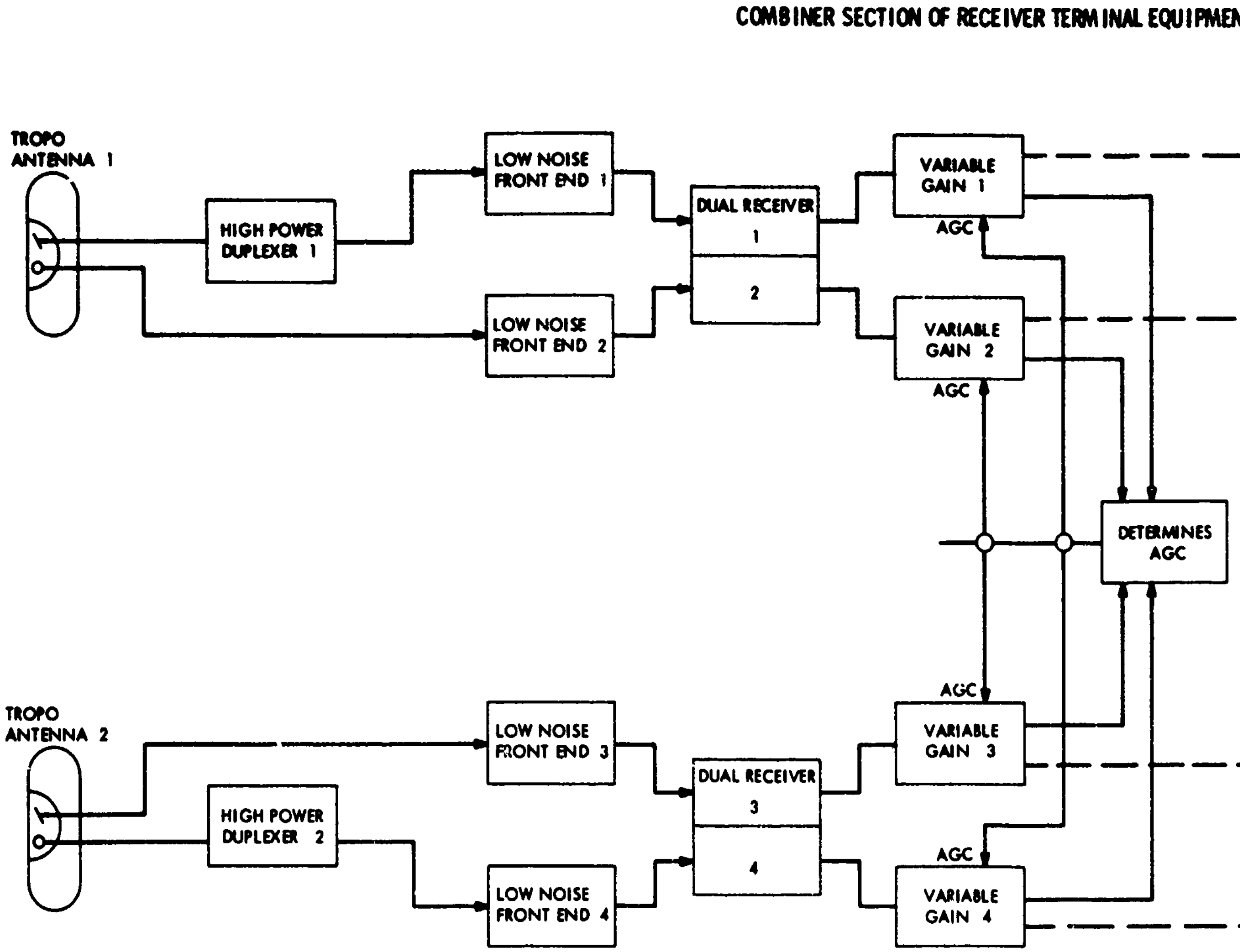
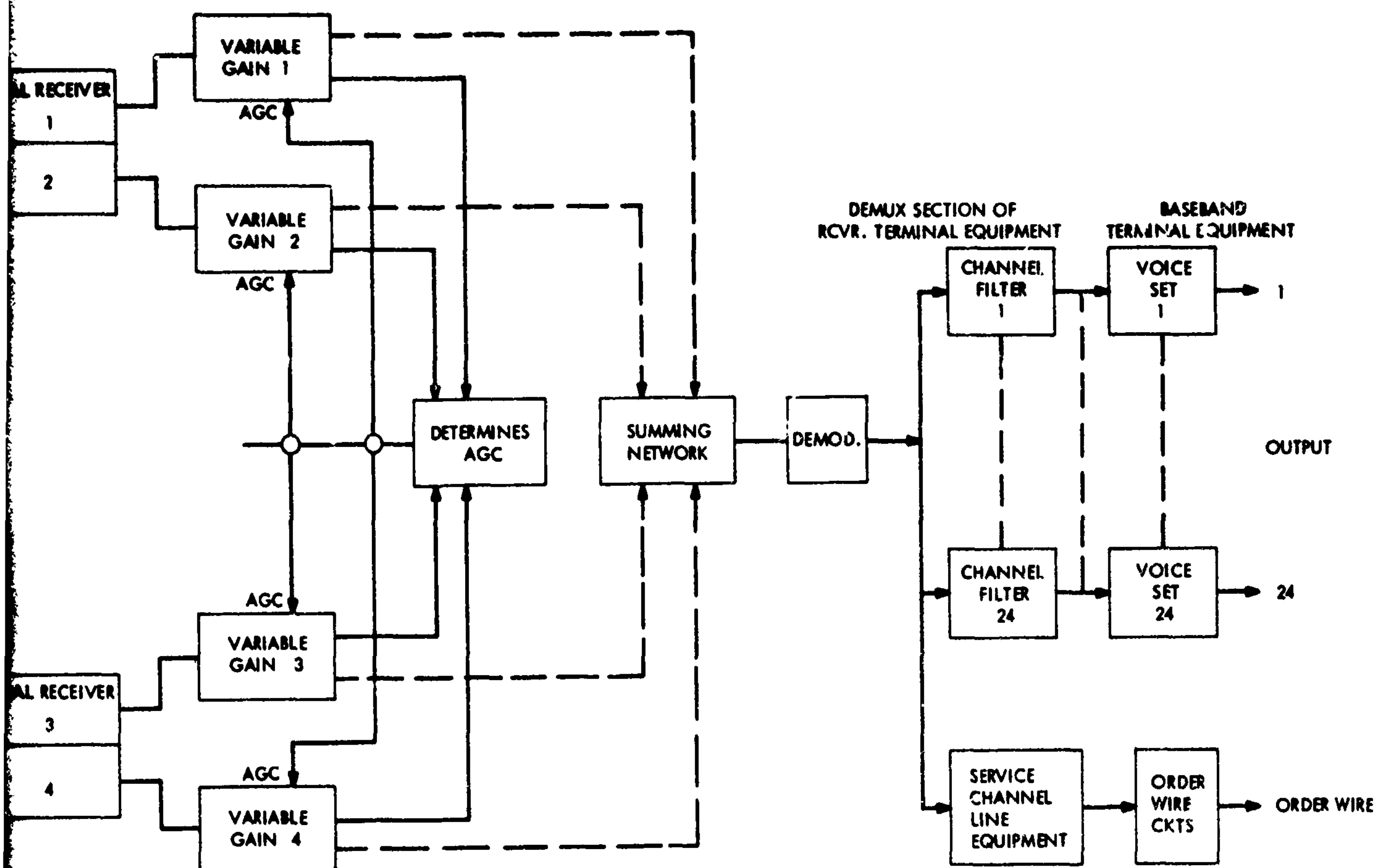


FIGURE 6-12. Block Diagram of Tropospheric Scatter System Receive Functions[2] (cont'd)

COMBINER SECTION OF RECEIVER TERMINAL EQUIPMENT



ive Functions[2] (cont'd)

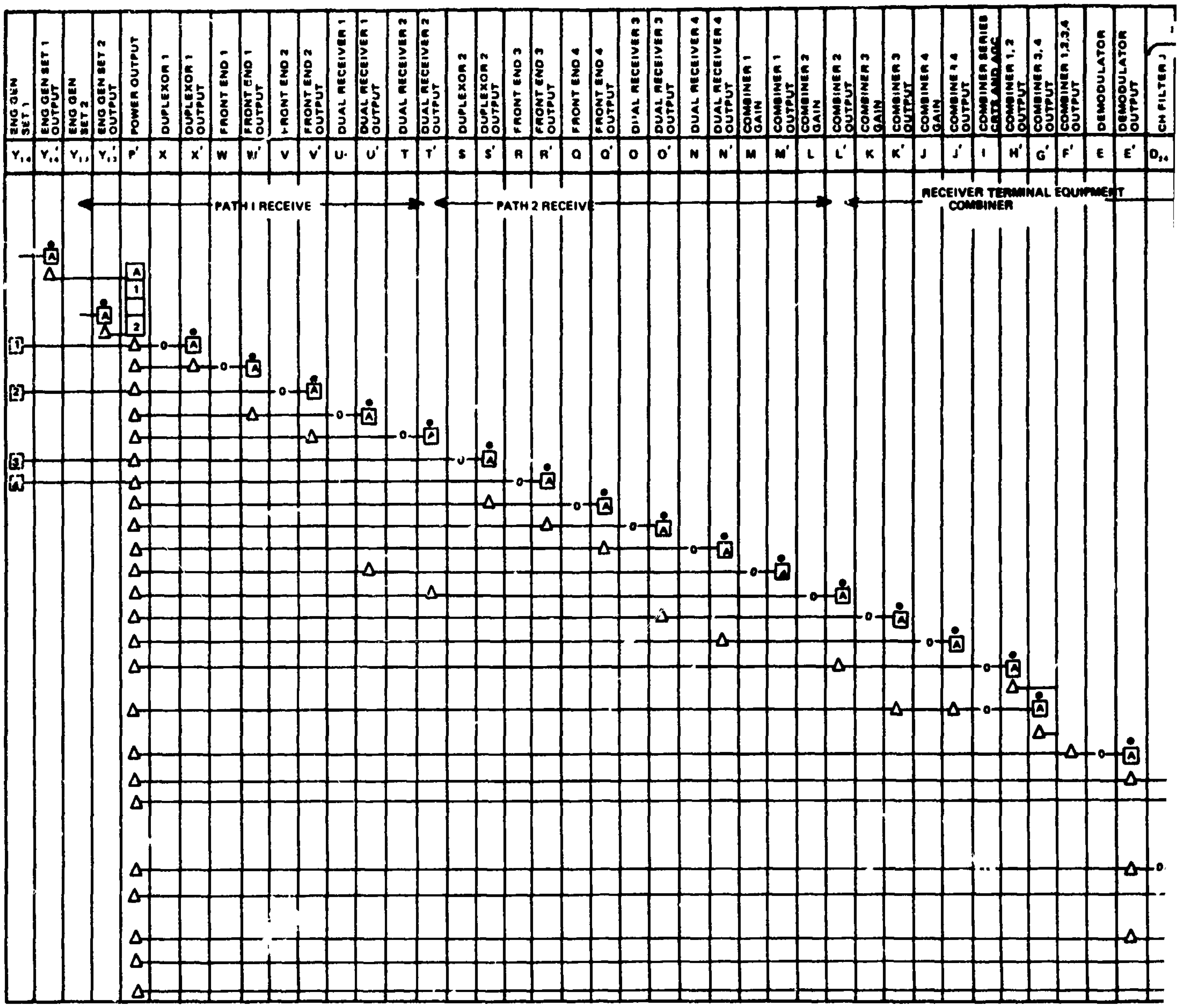
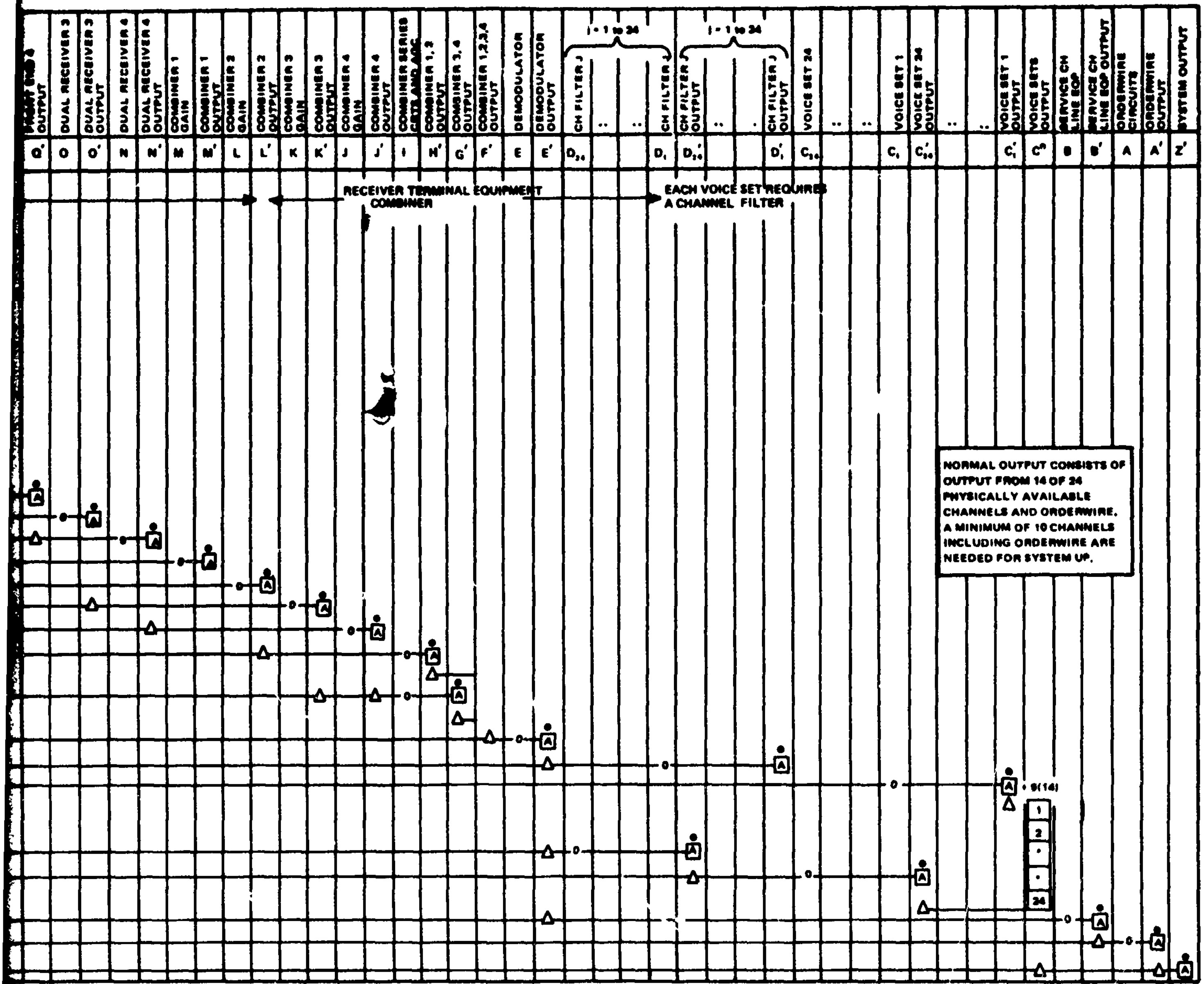


FIGURE 6-13. Dependency Chart for Tropospheric Scatter System[2]



...c Scatter System[2]

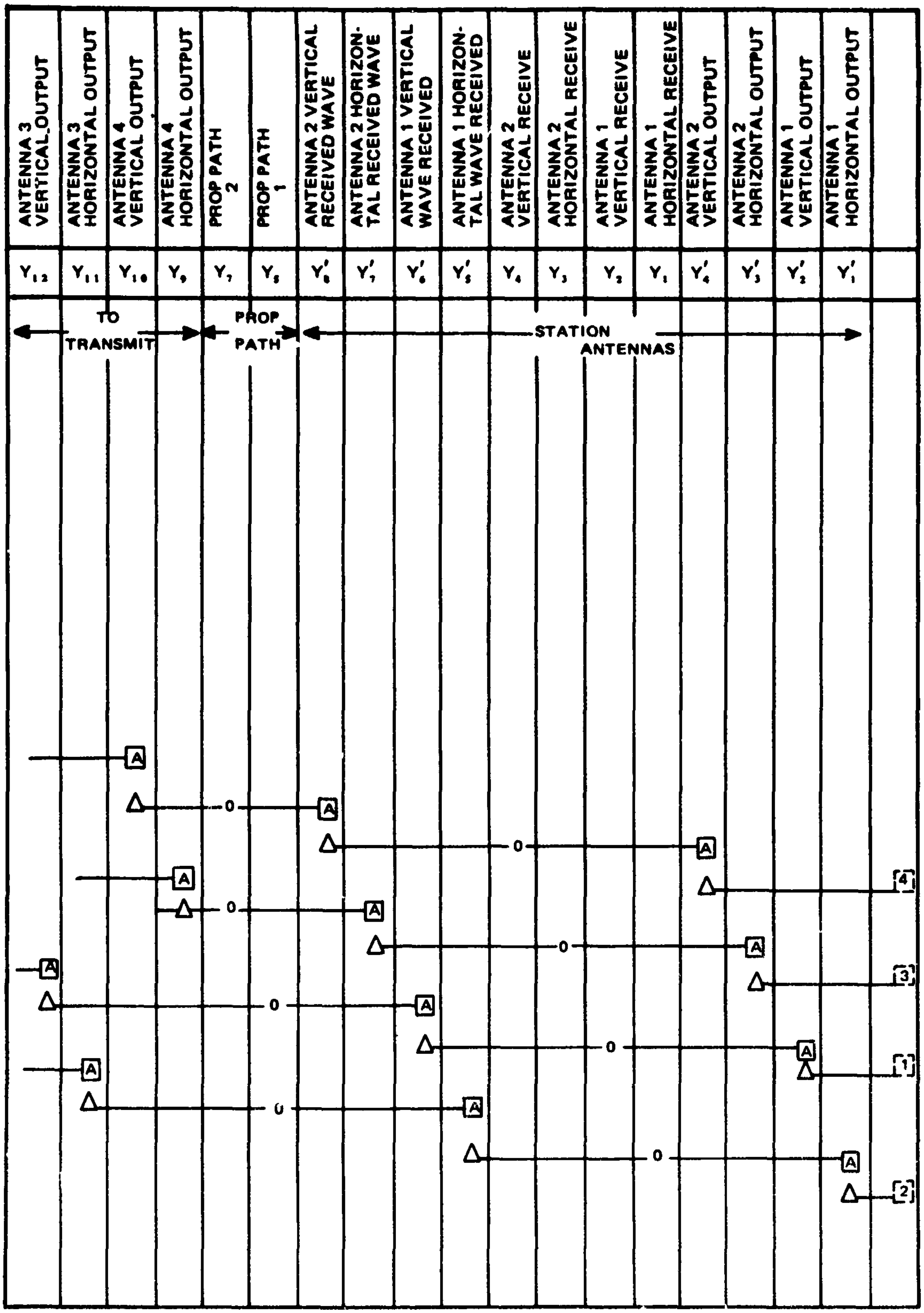


FIGURE 6-13. Dependency Chart for Tropospheric Scatter System[2] (cont'd)

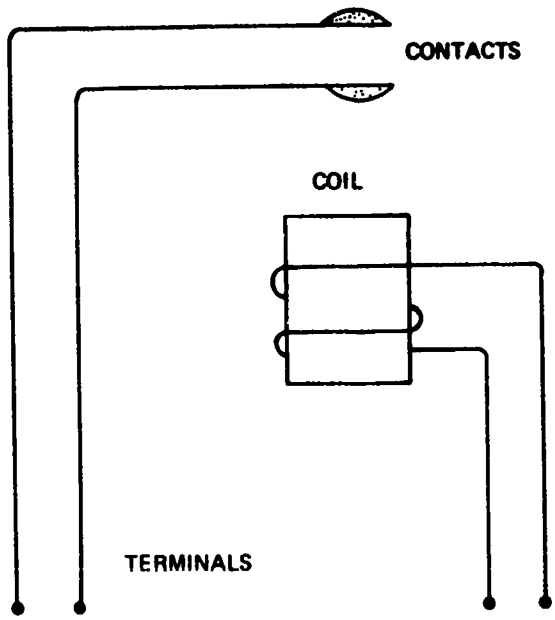


FIGURE 6-14. Functional Diagram of a Relay[9]

dependency diagram describes an action-at-a-distance force, the electromagnetic field, and the mechanical action of the contacts. The dependency structure readily can be used to represent mechanical and action-at-a-distance forces and can, therefore, be used to describe a wide variety of systems.

C. *A Packaged Speed Reducer (Mechanical).* A packaged speed reducer is an example of a mechanical system (Ref. 10). Packaged speed reducers are speed reduction gear trains that are assembled at the factory. Their use as off-the-shelf units results in considerable savings of time and money. The output, in this case, is a rotation of the output shaft. The output speed of rotation is related in an exact way to the speed of rotation of the input shaft by the gear arrangement. A packaged speed reducer is shown in Fig. 6-16 and its dependency diagram in Fig. 6-17.

6-3 DEVELOPMENT OF RELIABILITY MODELS

6-3.1 INTRODUCTION

The development of a reliability model is a complex process which involves the structure of the system, up-state rules, the parameter to be computed, the computation method, and the repair and spares strategies. As a result of these interactions, the reliability model is not a fixed entity, even for a specific system. Specifically, a reliability model consists of some or all of the following:

1. Reliability block diagram(s)
2. Definition of the up-state rules
3. Failure and repair rates of all functional elements
4. Definition of repair strategies
5. Definition of spares allocation and strategies.

The manner in which a reliability model can be structured is discussed in detail in the paragraphs that follow.

6-3.2 DEFINITIONS

Before proceeding with a detailed discussion of the derivation of reliability models, mathematical definitions of reliability without repair, reliability with repair, instantaneous availability, steady state availability, and mean time to failure (MTF) must be developed. These definitions are presented along with several other useful definitions, as adapted from Ref. 2.

1. *Reliability Without Repair.* The *s*-reliability without repair at time t is defined as the probability that the system will not fail (will perform satisfactorily) before time t, assuming that all components are good at $t = 0$ (the beginning of the mission). The *s*-reliability vs time curve has a value of 1 at $t = 0$ and monotonically decreases for increasing values of t.

2. *Reliability With Repair.* The *s*-reliability with repair of a system is defined as the probability that the system will not fail before time t, given that all components are good at $t = 0$, but with the provision that redundant items which fail are repaired. For a 1-unit system or a system made up of units in series, the *s*-reliability with repair is the same as reliability without repair, since the failure of one unit is considered as a system failure and, by definition of *s*-reliability, the system

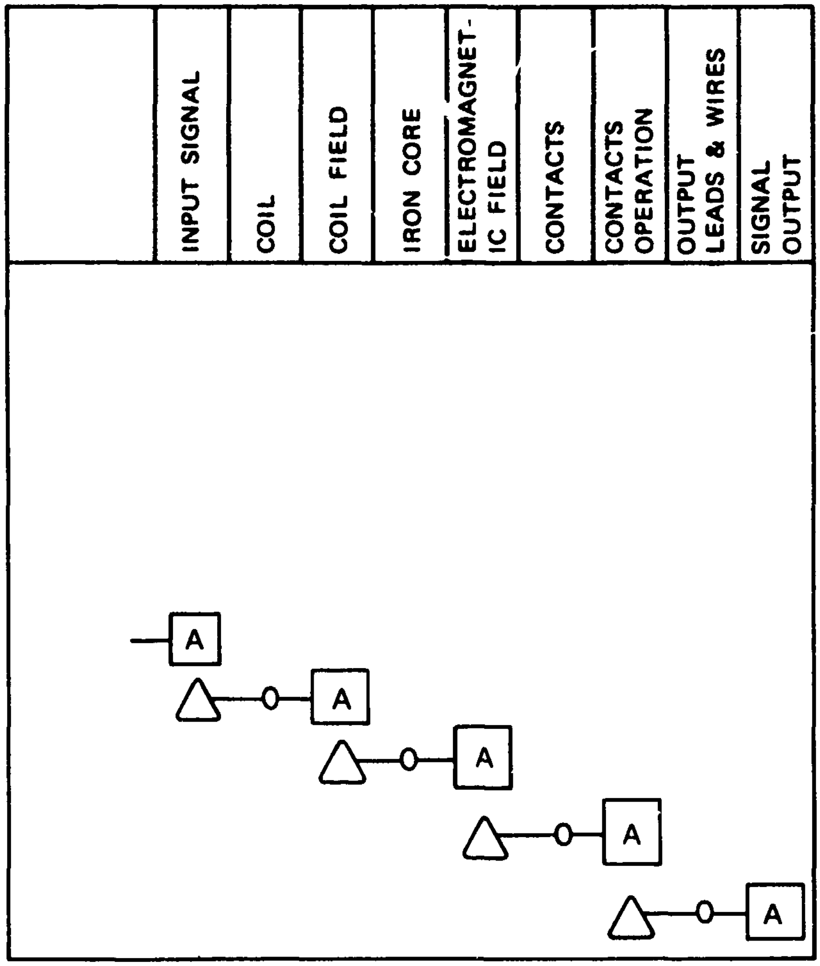


FIGURE 6-15. Relay Dependency Diagram

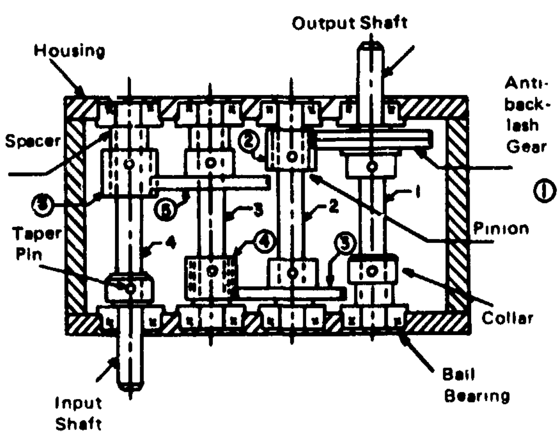


FIGURE 6-16. Packaged Speed Reducer[10]

is not permitted to go from a down-state to an up-state. The *s*-reliability with repair as a function of time begins at 1 for $t = 0$ and monotonically decreases. The shape of this curve is determined by the failure and repair distributions of the individual items as well as additional constraints on repairmen and/or spares.

3. *Instantaneous Availability*. The instantaneous availability of a system is defined as the probability that the system is up at the instant t, given that all components are good at $t = 0$. This means that the system could have failed and been restored many times during the interval from 0 to t. It also

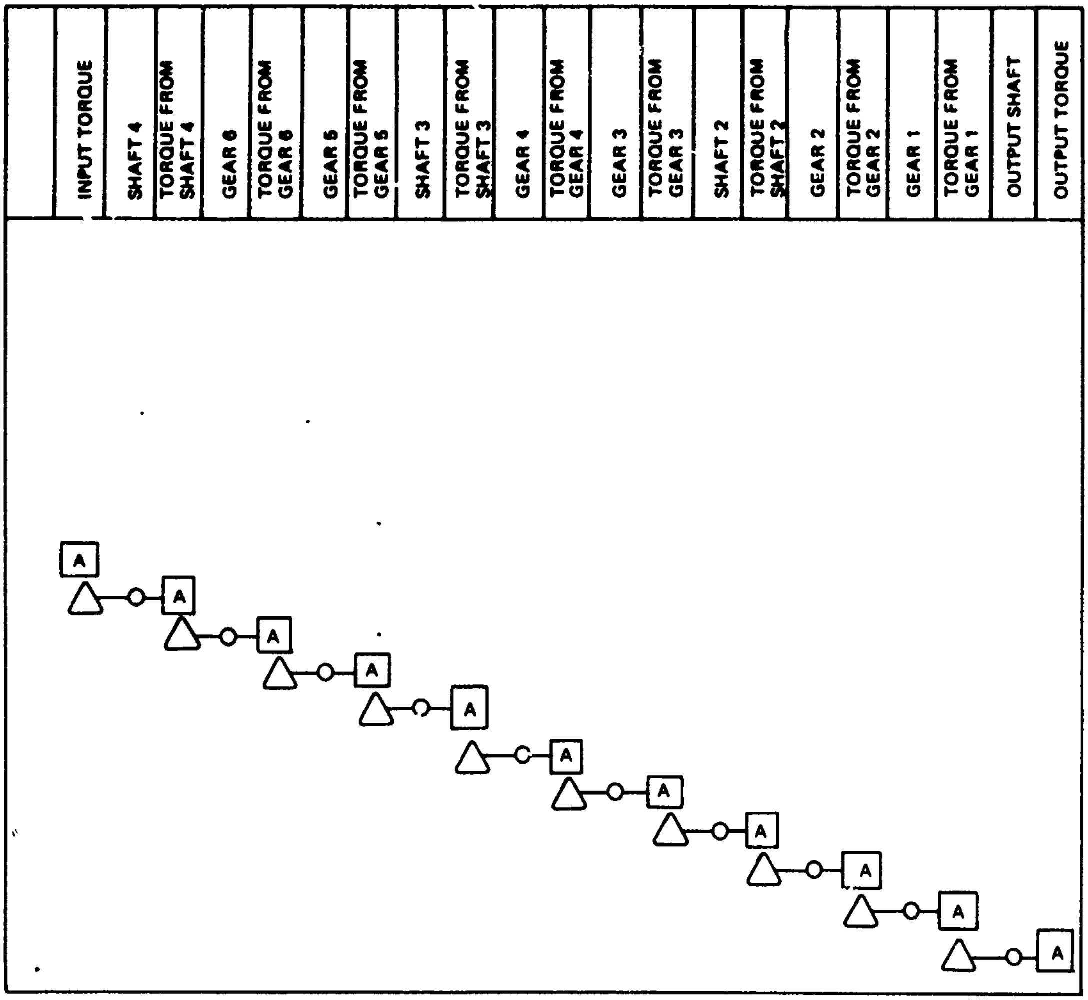


FIGURE 6-17. Packaged Speed Reducer Dependency Diagram

means that if repair is not allowed to take place on any of the items, the instantaneous availability is equal to the reliability without repair, because the only way the system could be up at the instant *t* under these circumstances is for the system to be up at $t = 0$ and remain up until *t*. The shape of the instantaneous availability curve depends on the types of failure and repair distributions the lowest level items are assumed to have.

4. *Steady-state Availability*. The steady-state availability of a system is the asymptotic value of the instantaneous availability and is defined as the probability that the system is up at *any* given point in time (but after a sufficiently long time so that steady-state is achieved). The steady-state availability is a constant and is not a function of time. Under the assumption of exponentially distributed times to failure and times to repair, the instantaneous availability monotonically decreases from a value of 1 to the steady-state availability and hence the steady-state availability under these circumstances is well defined and can be found readily.

5. *Mean Time to Failure (MTF)*. The MTF of a system is defined as the mean time to system failure. This definition is valid for nonrepairable systems and for repairable systems. The MTF can be obtained by integrating the reliability function (without repair or with repair) from 0 to ∞, assuming that the

integral exists. In this concept, once the system fails, it is dead and cannot be repaired.

It is important not to confuse the MTF with the MTBF (mean time between failures). MTBF may not be a workable concept for a particular system and may not be readily computed for complex repairable systems. For piece parts which are discarded after failure or for items that are restored to their original conditions and used as new spares, MTF is the appropriate concept.

6. *Equipment*. The term equipment will be used to designate an element of a system whose failure and repair characteristics are considered as those of a unit and not as a collection of smaller elements.

7. a. *Up*. An equipment or system is up if it is capable of performing its function.
b. *Degraded*. An equipment or system is degraded if it performs its function, but not well.

8. *Down*. An equipment or system is down if it is incapable of performing its function.

9. *Design Redundancy*. A system has design redundancy with respect to a given set of equipments if the system is up with only a part of the set in operation, i.e., the extra equipments are solely for the purpose of improving the reliability and availability characteristics of the system.

10. *On*. An equipment which is up and in operation is on.

11. *Idle*. An equipment which is up and not in operation, i.e., being held in standby is idle.

12. *Block*. A Block is a grouping of n identical equipments. The reliability of the grouping depends only on the number of equipments which are up in the block and not on which equipments in the block are up.

13. *Sections*. A Section is an s-independent grouping of equipments within a system. A system is divided into sections when the number of system up-states is so large that computer calculations are difficult. For example, calculation of system MTF with repair requires an inversion of the state matrix. If the computer available to the analyst cannot handle a matrix, the analyst must subdivide the system into two or more separate sections and compute s-reliability with repair for each. The system s-reliability with repair is the product of the section s-reliabilities; the MTF is computed by numerically integrating the system s-reliability.

14. A *k-out-of-n*:G-system has n components and is Good (up) if and only if at least k of them are Good (up).

15. A *k-out-of-n*:F-system has n components and is Failed (down) if and only if at least k of them are Failed (down).

6-3.3 DERIVATION OF A RELIABILITY DIAGRAM

The process of deriving a reliability block diagram (for s-reliability without repair) from a detailed system description is a complex process that involves many factors. This process must be analyzed to establish standardized procedures which form the basis of a formal mathematical technique. The analysis, using a part of a tropospheric scatter system, is described in the paragraphs that follow (Ref. 2).

Fig. 6-12 illustrates the equipment configuration for the receive function of a tropospheric station. Fig. 6-13 is the dependency diagram and Fig. 6-18 is the reliability diagram for the system in the particular mode being analyzed. The tropospheric system is complex and can operate in several modes. Each mode has a different reliability diagram. The possible modes are:

1. Voice Set Group output consists of outputs from 14 to 24 physically available channels of which nine or more must be up. (This statement on the dependency diagram implies two reliability diagrams.) If more than nine Voice Sets are up, the reliability diagram shows them in parallel. If nine Voice Sets are up, the reliability diagram shows them in series.

2. The output from *any* specific voice set functionally depends on that particular voice set AND on the output from *any* of the 24 Channel Filter outputs AND on the output from "Engine Generator Set 1 OR Engine Generator Set 2". (The parallel group of

Voice Sets is in series with the parallel group of Channel Filters and the parallel group of Engine Generator Sets.)

3. The Channel Filter outputs functionally depend on the corresponding Channel Filters AND on the Demodulator (via the Demodulator output) AND on the output from "Generator Set 1 OR Generator Set 2". (The parallel group of Channel Filters is in series with the Demodulator and the parallel group of Engine Generator Sets 1 and 2.)

4. The Demodulator output depends on the Demodulator Function AND on the Combiner series circuit output. The Combiner total output consists of an output via "Combiner Gain 1 AND 2" OR "Combiner Gain 3 AND 4". Both outputs via "Combiner Gain 1 AND 2" OR "Combiner Gain 3 AND 4" functionally depend on the Combiner series circuits (AGC and Summing Network) and Combiner Gain 1 AND 2 AND 3 AND 4, respectively. On the reliability diagram, the Demodulator is in series with the AGC and Summing Network which are in turn in series with Gain 1 AND 2 in parallel with Gain 3 AND 4.

5. Examination of the dependency diagram from this point to the system input reveals two chains of simple AND dependencies which are in parallel with each other. The first series chain consists of:

a. Received wave 1 (horizontal AND vertical component)

b. Antenna 1 (horizontal AND vertical feed)

c. Duplexer 1

d. Full polarization diversity, full space diversity[1]

e. Full polarization diversity and degraded space diversity

f. Degraded polarization diversity and full space diversity

g. Degraded polarization diversity and degraded space diversity.

Each of these modes can operate with or without Orderwire[2]. In this example, the case of full polarization diversity and degraded space diversity with up Orderwire is considered.

The reliability diagram can be derived from a simple set of logical statements implied directly by the dependency diagram. The set of logical statements follows and the effect on the reliability diagram is given in parentheses:

1. System output consists of output from Orderwire circuits and Voice Set Groups. (Orderwire circuits AND Voice Set Groups are in series.)

2. Orderwire output functionally depends on Orderwire circuits AND Service Channel Line Equipment output AND Demodulator circuit output AND ouptut from "Generator 1 OR Generator 2". (Orderwire circuits are in series with Service Channel Line Equipment and Demodulator and the parallel group of Generator Set 1 and 2.)

3. The Voice Set Group output depends on the outputs from any of the Channel Filters, the Demodulator, Summing Network, AGC Network, and the output from either of the Receive Channels. The Receive Channels each consist of a series grouping of functions. Receive Channel 1 consists of:

a. Received Wave 1 (horizontal AND vertical component)

b. Antenna 1 (horizontal AND vertical feed)

c. Duplexer 1

d. Front-end 1

e. Front-end 2

f. Receiver 1

g. Receiver 2

h. Combiner Gain 1 AND 2.

The second Receive Channel consists of:

a. Received Wave 2 (horizontal AND vertical component)

b. Antenna 2 (horizontal AND vertical feed)

[1] In polarization diversity, the transmitting and receiving antennas have dual feed horns. The wave is simultaneously transmitted with both horizontal and vertical polarization. In space diversity, the same wave is transmitted simultaneously over several physically distinct paths. Degraded diversity means that only one polarization direction or propagation path is operable.

[2] An Orderwire Channel allows station operators to communicate with each other.

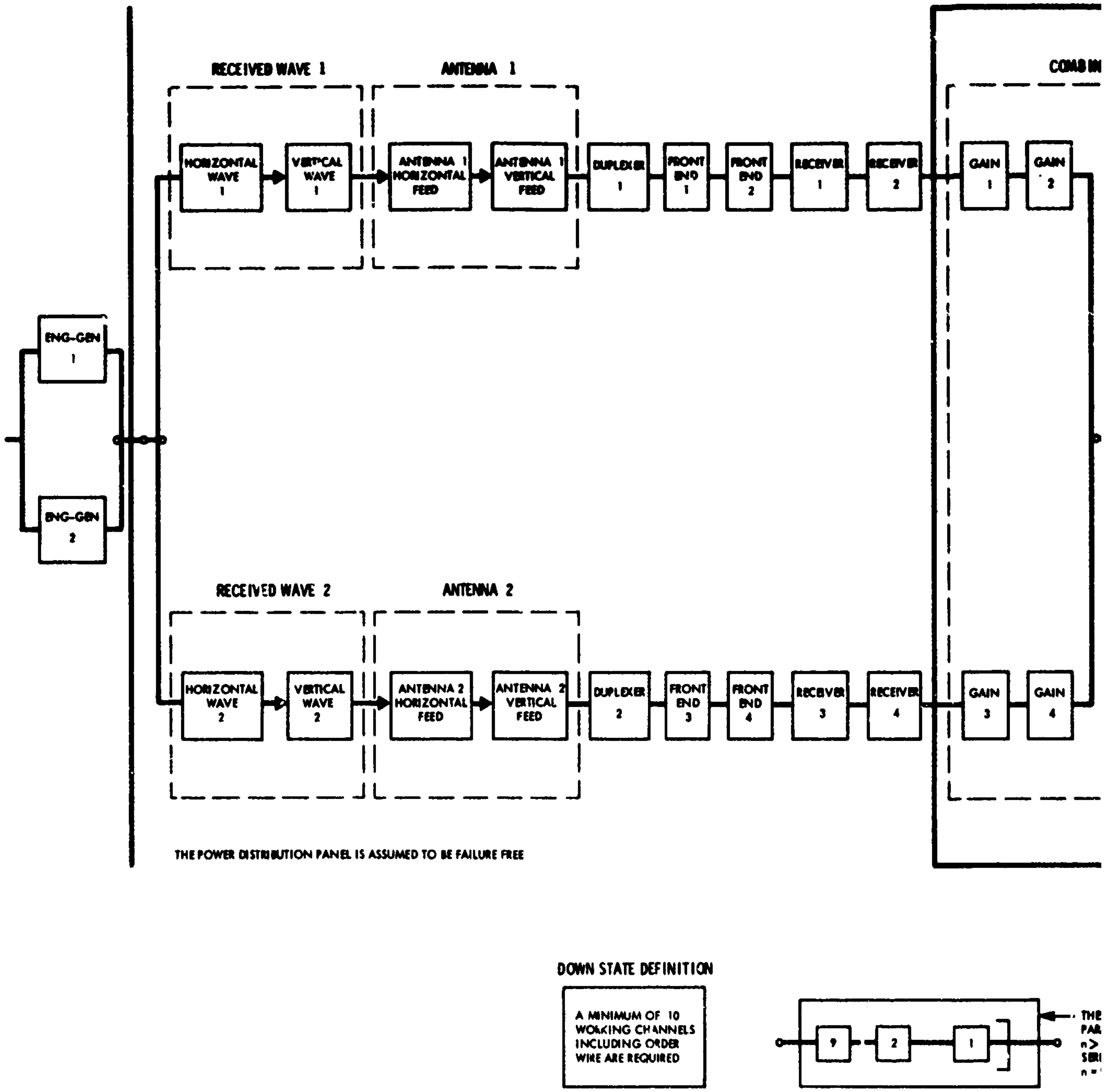
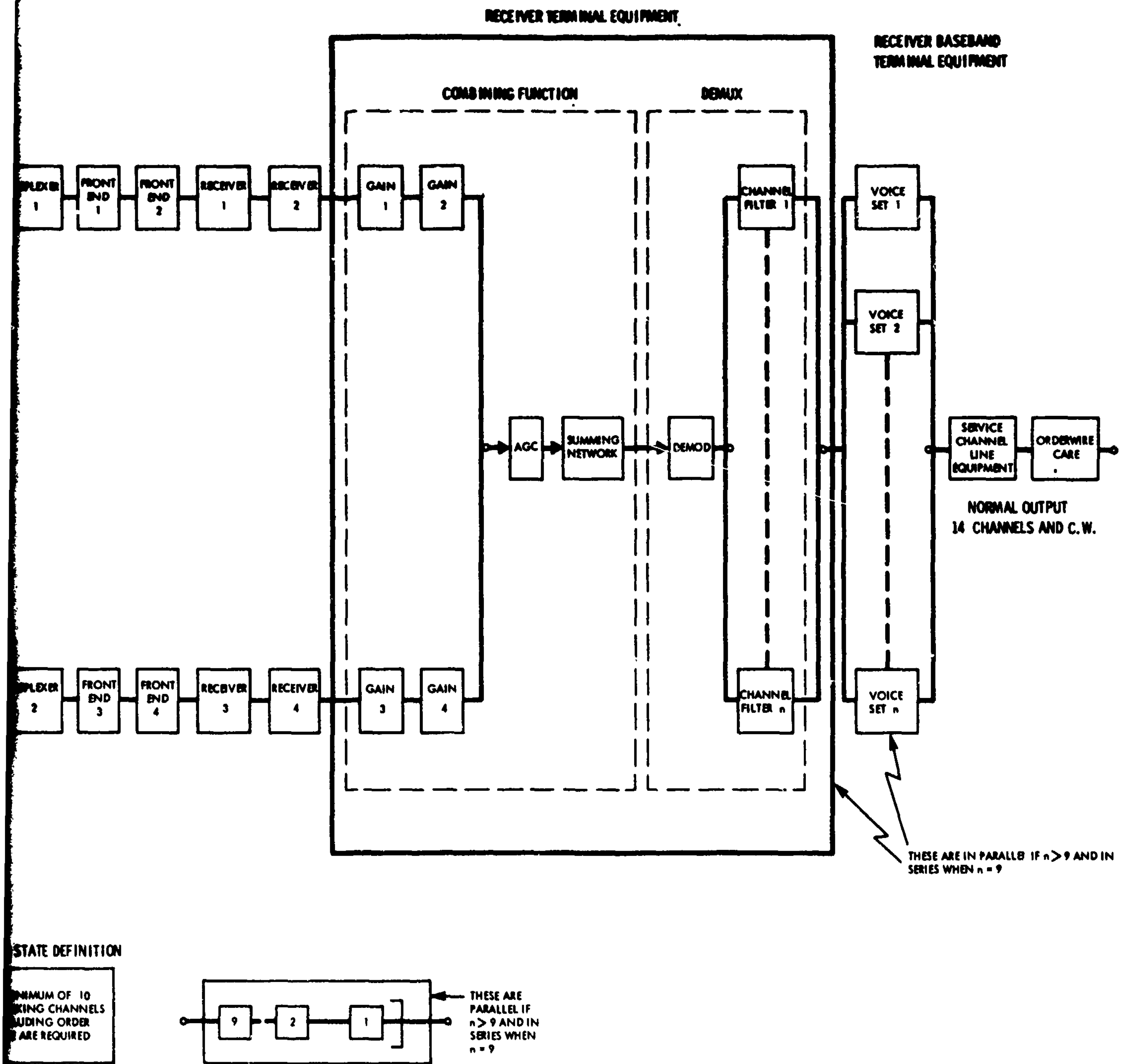


FIGURE 6-18. Reliability Diagram, Tropospheric Scatter System Receive Mode, Full Polarization and Degraded Space Diversity



tter System Receive Mode,
Diversity

c. Duplexer 2
d. Front-end 3
e. Front-end 4
f. Receiver 3
g. Receiver 4.

(The Channel Filters are in parallel. This parallel grouping is in series with the Demodulator, Summing Network, and AGC Network. These, in turn, are in series with the parallel combination of two Receive Channels.)

4. These two series chains of functions are in series with the output from Engine Generator Set 1 OR 2. (The parallel combination of Engine Generator Set 1 AND 2 is in series with the rest of the system.)

This analysis illustrates that the information contained in the dependency diagram can be used to derive a reliability block diagram for the case of s-reliability without repair. To summarize the previous discussions, the minimum information elements required for deriving a reliability block diagram are:

1. A dependency chart that clearly indicates the interdependencies between functional elements and events
2. A quantified definition of the system output
3. A statement of rules defining the system up-state.

6-3.4 MATHEMATICAL DERIVATION OF A RELIABILITY DIAGRAM

6-3.4.1 Basic Concepts

In this paragraph a simple example of how a reliability block diagram can be derived from a dependency diagram is presented.

Consider the dependency diagram in Fig. 6-19. A Boolean equation (Ref. 11) for each dependency line can be written as follows:

$$ANS = Z_1 + Z_2 \qquad (6\text{-}1)$$

$$\begin{aligned} Z_1 &= A \cdot K \\ Z_2 &= Z_3 \cdot B \\ Z_3 &= Z_4 \cdot A \\ Z_4 &= C + Z_5 \\ Z_5 &= P \cdot Q \end{aligned} \qquad (6\text{-}2)$$

By means of a series of substitutions, a Boolean function for the system can be generated in terms of its equipments. The steps are:

$$\begin{aligned} ANS &= A \cdot K + Z_3 \cdot B \\ &= A \cdot K + (Z_4 \cdot A) \cdot B \\ &= A \cdot K + [(C + Z_5) \cdot A] \cdot B \\ &= A \cdot K + [(C + P \cdot Q) \cdot A] \cdot B \\ &= A \cdot K + B \cdot [A \cdot (C + P \cdot Q)] \end{aligned} \qquad (6\text{-}3)$$

This function, when properly simplified and factored, forms the basis for the reliability block diagram. The factored form is:

$$ANS = A \cdot [K + B \cdot (C + P \cdot Q)] \qquad (6\text{-}4)$$

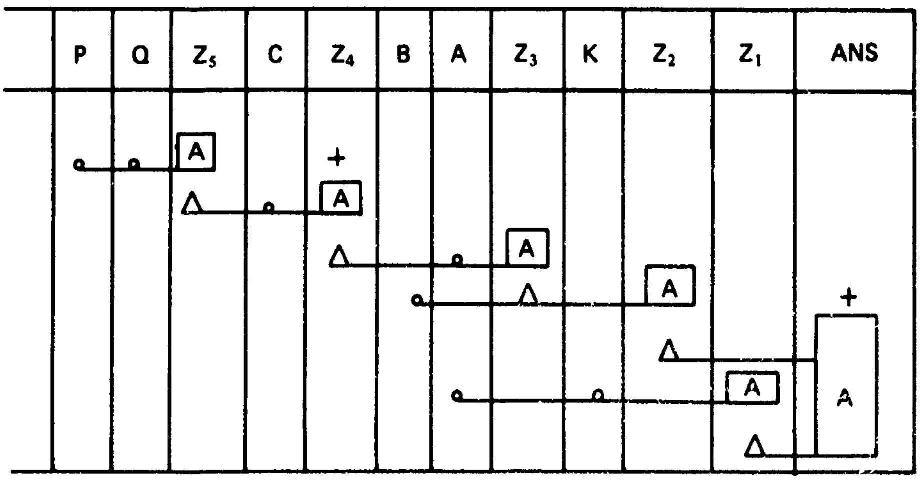


FIGURE 6-19. Simple Dependency Chart[2]

which is obtained by factoring A. The reliability block diagram corresponding to this tree is shown in Fig. 6-20.

6-3.4.2 A Complex Example

This paragraph explains in detail how a reliability model can be generated for a complex system for the case of s-reliability without repair; it is adapted from Ref. 2. The system to be considered is the tropospheric scatter communications system described previously (Fig. 6-12).

The reliability model can be generated by writing a Boolean expression for each dependency line. For example, if the dependency line shows that Z depends on A AND B, the Boolean equation is $Z = A \cdot B$; similarly, if Z depends on A OR B, then the Boolean equation is $Z = A + B$. This notation is used rather than $\cap$ (AND) and $\cup$ (OR) in deference to considerable custom in writing Boolean expressions.

In the tropo system, the parallel series structure shown in Fig. 6-21 occurs. The items $C_1, C_2, C_3, \ldots, C_{24}$ are identical and in parallel; normally only 14 out of the 24 items are in operation, the remaining 10 being in standby. This situation also applies to the $D_1, D_2, \ldots, D_{24}$ items. The output C'' is up when 9 out of 14 C items are up and D'' is up. The output D'' is up when 9 out of 14 D items are up.

The Boolean statements for the tropo system are listed. The symbol PS is a code for parallel-series function and the statement 9(14) represents the up-state definition for "9-out-of-14". The unprimed terms represent equipments, and the primed terms represent outputs, which will be eliminated as the expression for system output is developed. The following general equations can be written for the parallel grouping of C and D:

$$C'' = \mathrm{PS}(C'_{(j)}, \mathrm{j}=1,24), 9(14) \tag{6-5}$$

$$D'' = \mathrm{PS}(D'_{(j)}, \mathrm{j}=1,24), 9(14) \tag{6-6}$$

$$C'_{(j)} = C_{(j)} \cdot D'_{(j)} \cdot P' \tag{6-7}$$

$$D'_{(j)} = D_{(j)} \cdot E' \cdot P' \tag{6-8}$$

where E' represents the Demodulator output and P' represents the Power Supply output.

The Boolean equations for the tropo system (Fig. 6-13) are derived in the following manner:

$$Z' = A' \cdot C'' \tag{6-9}$$

$$A' = A \cdot B' \cdot P' \tag{6-10}$$

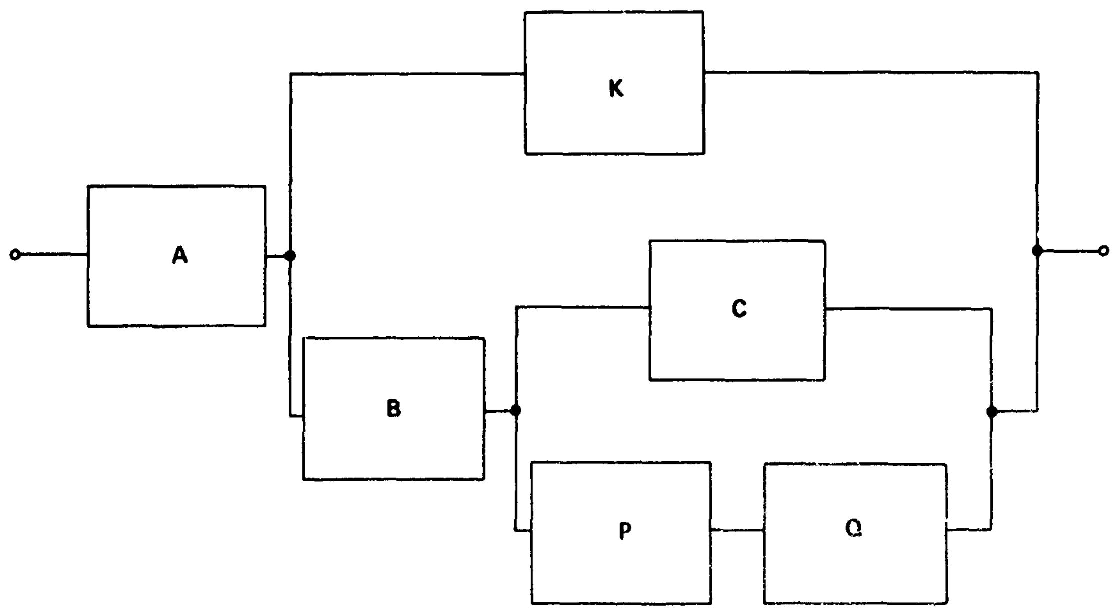


FIGURE 6-20. Simple Reliability Model

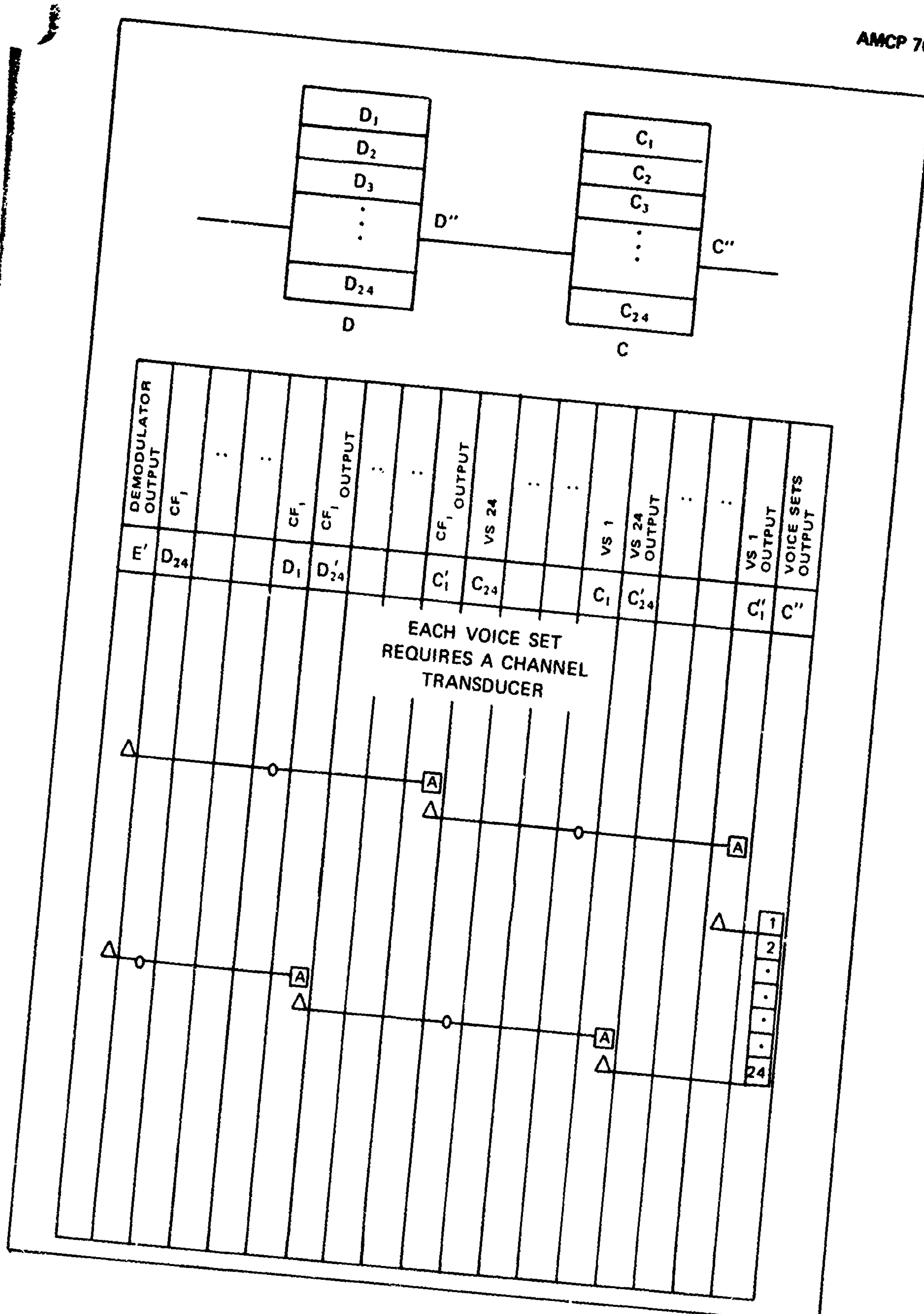


FIGURE 6-21. Tropospheric Scatter System Parallel Items[2]

$$B' = B \cdot E' \cdot P' \quad (6\text{-}11)$$
$$C'' = \mathrm{PS}(C'_{(j)}, j=1{,}24), 9(14) \quad (6\text{-}12)$$
$$C'_{(j)} = C_{(j)} \cdot D'_{(j)} \cdot P' \quad (6\text{-}13)$$
$$D'' = \mathrm{PS}(D'_{(j)}, j=1{,}24)\ 9(14) \quad (6\text{-}14)$$
$$D'_{(j)} = D_{(j)} \cdot E' \cdot P' \quad (6\text{-}15)$$
$$E' = E \cdot F' \quad (6\text{-}16)$$
$$F' = G' + H' \quad (6\text{-}17)$$
$$G' = I \cdot J' \cdot K' \cdot P' \quad (6\text{-}18)$$
$$H' = I \cdot L' \cdot M' \cdot P' \quad (6\text{-}19)$$
$$J' = J \cdot N' \cdot P' \quad (6\text{-}20)$$
$$K' = K \cdot O' \cdot P' \quad (6\text{-}21)$$
$$L' = L \cdot T' \cdot P' \quad (6\text{-}22)$$
$$M' = M \cdot U' \cdot P' \quad (6\text{-}23)$$
$$N' = N \cdot Q' \cdot P' \quad (6\text{-}24)$$
$$O' = O \cdot R' \cdot P' \quad (6\text{-}25)$$
$$Q' = Q \cdot S' \cdot P' \quad (6\text{-}26)$$
$$R' = R \cdot Y'_4 \cdot P' \quad (6\text{-}27)$$
$$S' = S \cdot Y'_3 \cdot P' \quad (6\text{-}28)$$
$$T' = T \cdot V' \cdot P' \quad (6\text{-}29)$$
$$U' = U \cdot W' \cdot P' \quad (6\text{-}30)$$
$$V' = V \cdot Y'_1 \cdot P' \quad (6\text{-}31)$$
$$W' = W \cdot X' \cdot P' \quad (6\text{-}32)$$
$$X' = X \cdot Y'_2 \cdot P' \quad (6\text{-}33)$$
$$Y'_1 = Y_1 \cdot Y'_5 \quad (6\text{-}34)$$
$$Y'_2 = Y_2 \cdot Y'_6 \quad (6\text{-}35)$$
$$Y'_3 = Y_3 \cdot Y'_7 \quad (6\text{-}36)$$
$$Y'_4 = Y_4 \cdot Y'_8 \quad (6\text{-}37)$$
$$Y'_5 = Y_5 \cdot Y_{11} \quad (6\text{-}38)$$
$$Y'_6 = Y_5 \cdot Y_{12} \quad (6\text{-}39)$$
$$Y'_7 = Y_7 \cdot Y_9 \quad (6\text{-}40)$$
$$Y'_8 = Y_7 \cdot Y_{10} \quad (6\text{-}41)$$
$$P' = Y'_{13} + Y'_{14} \quad (6\text{-}42)$$
$$Y'_{13} = Y_{13} \quad (6\text{-}43)$$
$$Y'_{14} = Y_{14} \quad (6\text{-}44)$$

These equations can be combined to generate a Boolean function for the system by a series of successive substitutions in the expression for Z'. Proceed as follows:

Given:

$$Z' = A' \cdot C'' \quad (6\text{-}9)$$

Substitute Eq. 6-10:

$$Z' = A \cdot B' \cdot P' \cdot C'' \quad (6\text{-}45)$$

Substitute Eq. 6-11:

$$Z' = A \cdot B \cdot E' \cdot P' \cdot P' \cdot C'' \quad (6\text{-}46)$$

Since $P' \cdot P' = P'$:

$$Z' = A \cdot B \cdot E' \cdot P' \cdot C'' \quad (6\text{-}47)$$

Eqs. 6-12, 6-13, 6-14, and 6-15 must be analyzed as a group. They are equivalent to the following equations:

$$C'' = \mathrm{PS}(C'_{(j)}, j=1{,}24), 9(14) \quad (6\text{-}12)$$
$$C'_{(j)} = C_{(j)} \cdot D_{(j)} \cdot E' \cdot P' \quad (6\text{-}48)$$

These can be further reduced.

$$C'' = \mathrm{PS}(C_{(j)} \cdot D_{(j)}, j=1{,}24),\ 9(14) \cdot E' \cdot P' \quad (6\text{-}49)$$
$$= \mathrm{PS}(C_{(j)}, j=1{,}24), 9(14) \cdot \mathrm{PS}(D_{(j)}, j=1{,}24), 9(14) \cdot E' \cdot P' \quad (6\text{-}50)$$
$$= C' \cdot D' \cdot E' \cdot P' \quad (6\text{-}51)$$

where

$$C' \equiv \mathrm{PS}(C_{(j)}, j=1{,}24), 9(14) \quad (6\text{-}52)$$
$$D' \equiv \mathrm{PS}(D_{(j)}, j=1{,}24), 9(14) \quad (6\text{-}53)$$

C' and D' are subsequently treated as elementary items.

Substitute Eq. 6-51:

$$Z' = A \cdot B \cdot E' \cdot P' \cdot C' \cdot D' \quad (6\text{-}54)$$

Substitute Eq. 6-16:

$$Z' = A \cdot B \cdot E \cdot F' \cdot P' \cdot C' \cdot D' \quad (6\text{-}55)$$

Substitute Eq. 6-17:

$$Z' = A \cdot B \cdot E \cdot (G' + H') \cdot P' \cdot C' \cdot D' \quad (6\text{-}56)$$

Substitute Eq. 6-18:

$$Z' = A \cdot B \cdot E \cdot (I \cdot J' \cdot K' \cdot P' + H') \cdot P' \cdot C' \cdot D' \quad (6\text{-}57)$$

Substitute Eq. 6-19:

$$Z' = A \cdot B \cdot E \cdot P' \cdot C' \cdot D' \cdot (I \cdot J' \quad K' \cdot P' + I \cdot L' \cdot M' \cdot P') \quad (6\text{-}58)$$

Substitute Eqs. 6-20 and 6-21:

$$Z' = \lambda P' \cdot (I \cdot J \cdot N' \cdot P' \cdot K \cdot O' \cdot I \cdot L' \cdot M' \cdot P') \quad (6\text{-}59)$$

where

$$\lambda \equiv A \cdot B \cdot E \cdot C' \cdot D' \quad (6\text{-}60)$$

Substitute Eqs. 6-22 and 6-23:

$$Z' = \lambda \cdot P' \cdot (I \cdot J \cdot N' \cdot P' \cdot K \cdot O + I' \cdot L \cdot T' \cdot M \cdot U' \cdot P') \quad (6\text{-}61)$$

Substitute Eqs. 6-24 and 6-25:

$$Z' = \lambda \cdot P' \cdot (I \cdot J \cdot N \cdot Q' \cdot P' \cdot K \cdot O \cdot R' + I \cdot L \cdot T' \cdot M \cdot U' \cdot P') \quad (6\text{-}62)$$

Substitute Eqs. 6-26 and 6-27:

$$Z' = \lambda \cdot P' \cdot (I \cdot J \cdot N \cdot Q \cdot S' \cdot P' \cdot K \cdot O \cdot R \cdot Y_4' + I \cdot L \cdot T' \cdot M \cdot U' \cdot P') \quad (6\text{-}63)$$

Substitute Eqs. 6-28 and 6-29:

$$Z' = \lambda \cdot P' \cdot (I \cdot J \cdot N \cdot Q \cdot S \cdot Y_3' \cdot P' \cdot K \cdot O \cdot R \cdot Y_4' + I \cdot L \cdot T \cdot V' \cdot M \cdot U' \cdot P') \quad (6\text{-}64)$$

Substitute Eqs. 6-30 and 6-31:

$$Z' = \lambda \cdot P' \cdot (\tau \cdot Y_3' \cdot P' \cdot Y_4' \cdot I + I \cdot L \cdot T \cdot V \cdot Y_1' \cdot M \cdot U \cdot W' \cdot P') \quad (6\text{-}65)$$

where

$$\tau \equiv J \cdot N \cdot Q \cdot S \cdot K \cdot O \cdot R \quad (6\text{-}66)$$

Substitute Eqs. 6-32 and 6-33:

$$Z = \lambda \cdot P' \cdot (\tau \cdot Y_3' \cdot P' \cdot Y_4' \cdot I + I \cdot L \cdot T \cdot V \cdot Y_1' \cdot M \cdot U \cdot W \cdot X \cdot Y_2' \cdot P') \quad (6\text{-}67)$$

Substitute Eqs. 6-34 and 6-35:

$$Z' = \lambda \cdot P' \cdot (\tau \cdot Y_3' \cdot P' \cdot Y_4' \cdot I + I \cdot \alpha \cdot Y_1 \cdot Y_5' \cdot Y_2 \cdot Y_6' \cdot P') \quad (6\text{-}68)$$

where

$$\alpha \equiv L \cdot T \cdot V \cdot M \cdot U \cdot W \cdot X \quad (6\text{-}69)$$

$$Z' = \lambda \cdot P' \cdot (\tau \cdot Y_3 \cdot Y_7' \cdot P' \cdot Y_4 \cdot Y_8' \cdot I + \alpha \cdot I \cdot Y_1 \cdot Y_5' \cdot Y_2 \cdot Y_6' \cdot P' \quad (6\text{-}70)$$

Substitute Eqs. 6-38, 6-39, 6-40, and 6-41:

$$Z' = \lambda \cdot P' \cdot (\tau \cdot I \cdot Y_3 \cdot Y_7 \cdot Y_9 \cdot P' \cdot Y_4 \cdot Y_{10} + \alpha \cdot I \cdot Y_1 \cdot Y_5 \cdot Y_{11} \cdot Y_2 \cdot Y_{12} \cdot P') \quad (6\text{-}71)$$

Substitute Eq. 6-42:

$$Z' = \lambda \cdot (Y_{13}' + Y_{14}') \, [\tau \cdot \epsilon_1 \cdot (Y_{13} + Y_{14}) + \alpha \cdot \epsilon_2 \cdot I \cdot (Y_{13} + Y_{14})] \quad (6\text{-}72)$$

where

$$\epsilon_1 \equiv Y_3 \cdot Y_4 \cdot Y_7 \cdot Y_9 \cdot Y_{10} \quad (6\text{-}73)$$

$$\epsilon_2 \equiv Y_1 \cdot Y_2 \cdot Y_5 \cdot Y_{11} \cdot Y_{12} \quad (6\text{-}74)$$

Substitute Eqs. 6-43 and 6-44:

$$Z' = \lambda \cdot (Y_{13} + Y_{14}) \cdot [\tau \cdot I \cdot \epsilon_1 \cdot (Y_{13} + Y_{14}) + \alpha \cdot \epsilon_2 \cdot I \cdot (Y_{13} + Y_{14})] \quad (6\text{-}75)$$

$$= \lambda \cdot (Y_{13} + Y_{14}) \cdot [KON1 \cdot I \cdot (Y_{13} + Y_{14}) + KON2 \cdot I \cdot (Y_{13} + Y_{14})] \quad (6\text{-}76)$$

where $KON1 \equiv \tau \cdot \epsilon_1$ (6-77)

$KON2 \equiv \alpha \cdot \epsilon_2$ (6-78)

Upon factoring out the term $(Y_{13} + Y_{14}) \cdot I$, one has

$$Z' = \lambda \cdot (Y_{13} + Y_{14}) \cdot I \cdot (KON1 + KON2) \quad (6\text{-}79)$$

The tree corresponding to Eq. 6-79 is shown in Fig. 6-22, where D_1 and D_2 are dummy variables. The Boolean symbols in Eq. 6-79 each represent an electrical function or a group of electrical functions, namely:

$\lambda = A \cdot B \cdot E \cdot C' \cdot D'$
= (Orderwire)
• (Service Channel Line Equip)
• (Demodulator)
• (Voice Sets)
• (Channel Filters)

I = (AGC)
• (Combiner Series Circuit)

Y_{13} = (Engine Gen Set 2)

Y_{14} = (Engine Gen Set 1)

$KON1 = Y_3 \cdot Y_4 \cdot Y_7 \cdot Y_9 \cdot Y_{10} \cdot J \cdot N \cdot Q \cdot S \cdot K \cdot O \cdot R$
= (Ant 2 Hor Receive)

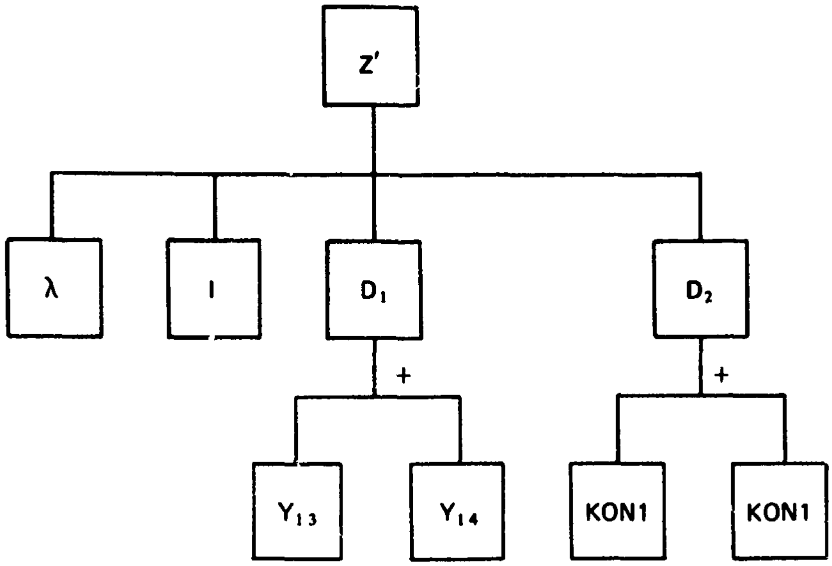


FIGURE 6-22. Boolean Tree

• (Ant 2 Vert Receive)
• (Propagation Path 2)
• (Ant 2 Hor Out)
• (Ant 2 Vert Out)
• (Combiner Gain 4)
• (Dual Rcvr 4)
• (Front End 4) • (Duplexer 2)
• (Combiner Gain 3)
• (Dual Rcvr 3)
• (Front End 3)

$$KON2 = Y_1 \cdot Y_2 \cdot Y_5 \cdot Y_{11} \cdot Y_{12} \cdot L \cdot T \cdot V \cdot M \cdot U \cdot W \cdot X$$

= (Ant 1 Hor Receive)
• (Ant 1 Vert Receive)
• (Propagation Path 1)
• (Ant 1 Hor Out)
• (Ant 1 Vert Out)
• (Combiner Gain 2)
• (Dual Rcvr 2)
• (Front End 2)
• (Combiner Gain 1)
• (Dual Rcvr 1)
• (Front End 1) • (Duplexer 1)

Refer back to Fig. 6-18 for the final reliability configuration.

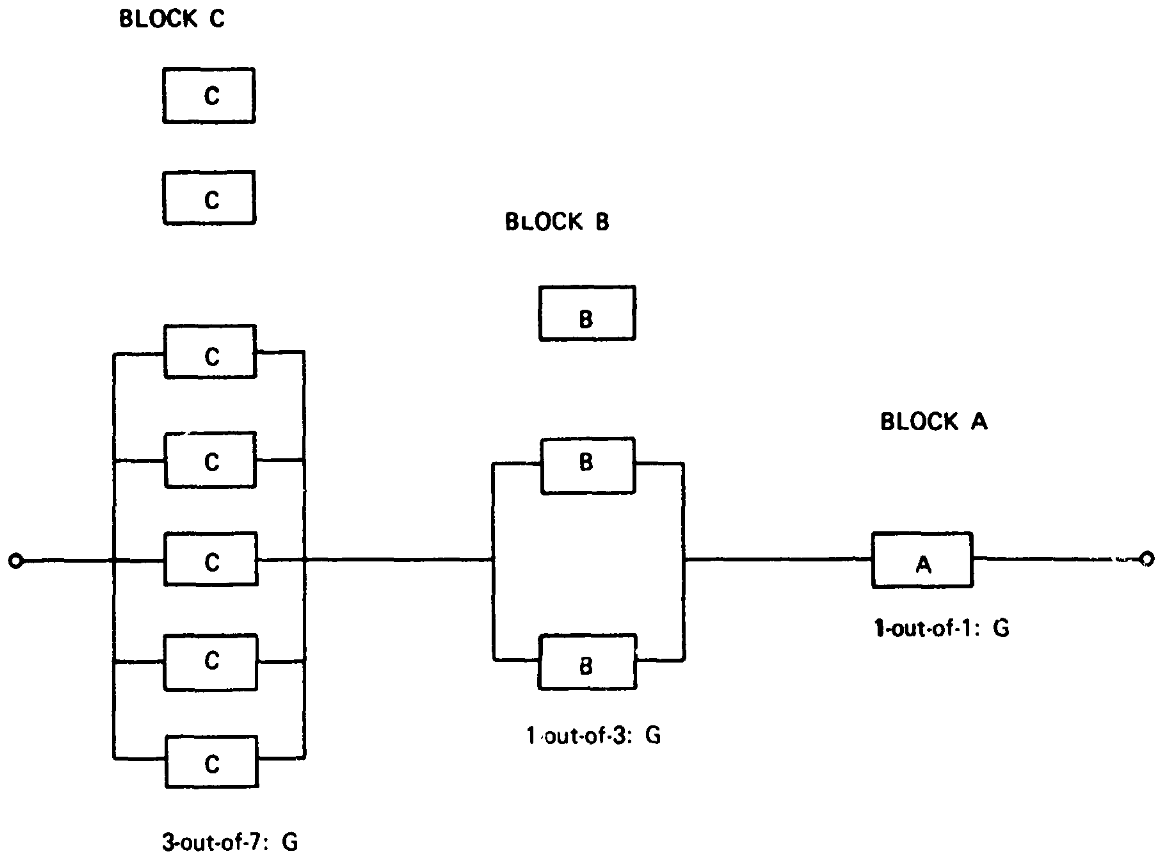
The relative ordering of equipment is not and need not be preserved in the reliability model.

6-3.4.3 Reliability Models for Maintained Systems

Reliability models for maintained systems require additional information above that derived from dependency diagrams and from the basic reliability block diagram. In practice (and in theoretical work) the distinction between redundancy and repair is often blurred. The names of some of the activities are sometimes different, but the activities themselves are very similar. We will use the term "replacement" to describe the activity of removing a nit that is presumed bad and inserting one that is presumed good. Whether it is the same unit after being repaired, or a different one, is irrelevant. Two examples are given.

6-3.4.3.1 Example No. 1 (Fig. 6-23)

All failure and replacement rates are constant. Blocks B and C have two kinds of spares, classified according to the ease of re-



1 repairman: strategy is defined in the text.
System is Good (up) if and only if A,B,C, are Good (up).

FIGURE 6-23. System For Example No. 1

placement; the kind shown separately in Fig. 6-23 are more difficult to replace.

The system consists of three blocks:

1. Block A is a 1-out-of-1:G-subsystem.
2. Block B is a 1-out-of-3:G-subsystem.
3. Block C is a 3 out-of-7:G-subsystem.

The system is up if and only if Blocks A, B, C are Good (up).

The optimum repair strategy can only be determined by choosing a figure-of-merit to optimize, and then solving the problem. A reasonable set of priorities (in the absence of the complete solution) for the repairman might be the following:

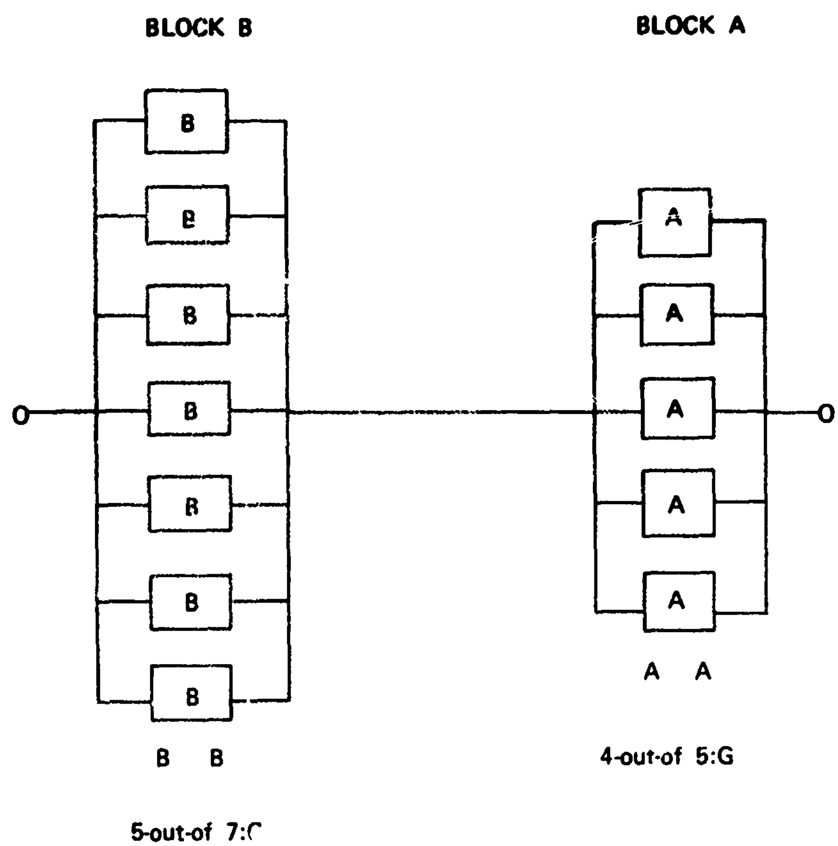
1. Finish replacing the unit being worked on, if any.
2. If more than 1 unit is failed, choose, in the following order, the one to be replaced:

a. A unit from a block that is down. If more than one block is down, it makes no difference which is chosen.

b. An easily-replaceable spare. If more than one block is down, choose the one from the block that has the fewest spares that are good.

3. If a rule is not given completely enough, choose one from the allowable failed units at random.

The rules can become quite complicated in a theoretical analysis. In practice, the repairman should not be required to make complicated calculations merely to find out which unit to work on. The rules also can be so complicated as to make theoretical analysis virtually impossible. If the replacement rate is much higher than the failure rate, the standard matrix techniques can be used.



2 repairmen for each Block.

System is Good (up) if and only if A and B are Good (up)

FIGURE 6-24. System For Example No. 2

6-3.4.3.2 Example No. 2 (Fig. 6-24)

All failure and replacement rates are constant. Block A is a 4-out-of-5:G-subsystem. Block B is a 5-out-of-7:G-subsystem. There is only 1 kind of spare in each block.

6-4 OTHER MODELS

The reliability block diagram has been used throughout this chapter to illustrate logic diagrams for a system. Other kinds of diagrams, e.g., fault tree, might be more appropriate in some cases. See *Part Two, Design for Reliability*, for a discussion of these other kinds of logic diagrams.

REFERENCES

1. S. Orbach, *The Generalized Effectiveness Methodology (GEM) Program*, Lab Project 920-72-1, Progress Report 1, US Naval Applied Science Laboratory, 8 May 1968.
2. *The Development of a Generalized Effectiveness Methodology*, Interim Report, US Naval Applied Science Laboratory, 30 September 1966.
3. P. Giordano, S. Seltzer, and C. Sontz, "General Effectiveness Methodology", *Operations Research Society of America Meeting*, Durham, North Carolina, 18 October 1966.

4. MIL-HDBK-226(NAVY), *Design Disclosure for Systems and Equipment*, 17 June 1968.
5. S. Seeley, *Radio Electronics*, McGraw-Hill Book Company, Inc., New York, 1956.
6. *Basic and Advanced Infrared Technology*, (AD-634 535), U S Army Missile Command, Redstone Arsenal, Alabama, 15 April 1965.
7. *Main Battle Tank-70, Reliability and Performance Status Report*, Main Battle Tank Engineering Agency, Detroit, Michigan, June 1970.
8. *Universal Combined Radio Relay and Troposcatter Equipment*, Vol. 1, RADC-TR-60-246, ITT Communications Systems, Inc., Prepared for Rome Air Development Center, Air Force Systems Command, U S Air Force, 15 July 1961.
9. G. Chernowitz, et al., *Electromechanical Component Reliability*, RADC-TDR-63-295, American Power Jet Company, May 1963.
10. N. P. Chironis, *Machine Devices and Instrumentation*, McGraw-Hill Book Company, Inc., New York, 1966.
11. I. M. Copi, *Introduction to Logic*, The Macmillan Company, New York, 1961.

CHAPTER 7 KINDS OF REDUNDANCY AND REPAIR

7-1 INTRODUCTION

Redundancy and repair are very similar concepts. In the general case where switching is not instantaneous it is easy to visualize two similar operations, one called redundancy and one called repair. In redundancy, the time used to replace a faulty unit is usually shorter than the time a repair is considered to take.

There are many important considerations in a redundancy/repair situation, i.e.,

1. In what state are all the units at $t = 0$? How does one know? Is checkout perfect?
2. In what state is a repaired unit? Is it good-as-new? How does one know? Is checkout perfect?
3. In what state is a repaired system? How does one know? Is checkout performed? Is it perfect?
4. What kinds of failures are being alleviated? If failures are due to the rare, random occurrence of severe conditions, redundancy might not be of much help.
5. How difficult is it to know that a unit has failed? How difficult is it to remove the faulty unit and replace it?
6. How much of an improvement in reliability is needed or expected? What reliability measure is important in your case? For example, mean time to failure is not a good reliability measure for short times.
7. How much does redundancy/repair cost in weight, dollars, volume, design effort, checkout, schedule time, heat dissipation, system complexity, extra connectors, etc.?
8. What about switching? Is information lost during switching?
9. What about the failure behavior of standby equipment?
10. Under what conditions are failures s-independent? When the correct calculations have been made, how much improvement in reliability will there be?

7-2 KNOWLEDGE OF SYSTEM STATE

In order to analyze a system, one needs to know the state (condition) of the system at several time instants. The two most important instants are "time = zero" and "just after repair".

If a system contains any redundancy, the question arises, "How does one know that each unit is good?" Just knowing that the system is up is not enough, since some units could be bad and the system would still be up. Therefore, there must be checkout of each unit in the system. This involves hardware, software, time, and money. Checkout is rarely perfect. Will the analysis take that into account? The knowledge of system state at "time = zero" is also important because in many analyses, a system or unit is presumed to be good-as-new viz., "time = zero" again after repair.

There are only two tractable choices in deciding the condition of a unit after repair: good-as-new and bad-as-old. Good-as-new often is taken to mean "perfect", but if checkout is involved all it means is that time reverts to zero for the unit that is good-as-new. The phrase bad-as-old was coined to contrast with good-as-new and to illustrate the condition where the failure rate of the system "immediately after a repair" is the same as it was "just before repair". An internal combustion engine after a minor tune-up is a good illustration of bad-as-old. The major components of the engine didn't change; perhaps all that was done was to clean and regap the spark plugs, and adjust the distributor gap and the timing. The engine certainly is not good-as-new. A Poisson process with nonconstant rate is an example of the bad-as-old behavior.

When the failure rate of each unit is constant, there is no difference between bad-as-old and good-as-new.

In theoretical analyses with complicated system-states a common assumption is that the repaired unit is good-as-new, but the other units are bad-as-old. Of course, because of tractability considerations, failure rates of units are assumed most commonly to be constant so that any time the system is known to be working, it is good-as-new. Many papers require that the assumptions be inferred from the mathematics; the authors have been remiss in stating assumptions.

Many systems use periodic checkout to ascertain the state of the system. Preventive maintenance is performed as required. But any time maintenance of any kind is performed, there is the real possibility and danger that some part of the system has been damaged unknowingly. There is a short period of "infant mortality" immediately after anyone fusses with any complicated system. One illustration of this fact is that, at least during World War II, the repair crew chief for aircraft was supposed to go along on the checkout flight after a repair.

The state of a complicated real system is not an easy thing to determine. Many analyses make the blithe assumption of perfection after repair, replacement, or checkout. Real equipment is rarely like that.

7-3 SYSTEM LEVEL FOR REDUNDANCY APPLICATION

In a system, at what level ought redundancy to be applied? In principle (in the mathematics anyway), one could make every piece-part redundant, or one could just have several systems. All of the factors listed in par. 7-1 apply to this decision. The question of switching is especially important, simply because so often it is assumed (in the mathematics) to be perfect: zero cost, instantaneous, no information lost, no size or weight, no design time, etc.

The lower the level at which redundancy is applied, the more likely are common-mode failures to be important. The question of conditional *s*-independence needs to be investigated very carefully. This question is allied closely with the level at which repair parts ought to be stocked. What about throw-away maintenance? At what level ought it be performed?

In practice, an analysis barely can hope to scratch the surface. Some rough guidelines can be developed, but pilot projects are the places where knowledge is really gained. It is easy for the proposed system to be intractable for anything but a Monte Carlo simulation. Therefore, the design engineer and his staff analysts need to know what simulation languages are available on their computer.

Many analyses are scattered in the literature. Rarely will the one be there that you want. They can, however, give an idea about what to analyze and what direction the results might take. See the chapters that follow and the Bibliography at the end of this chapter for some sources.

Roughly speaking, the lower the level at which redundancy is applied, the more effective it is (if switching is perfect and failures are *s*-independent) and the more it costs (in everything).

7-4 METHOD OF SWITCHING

In virtually all systems, some kind of "switching" is necessary for redundancy to be effective. A fluid flow system might require a check-valve on each redundant pump; an electronic system might have to be disconnected. The three main categories discussed here are automatic, manual, and repair.

In automatic switching, the operator need not do anything in case of a unit failure. He may not even be aware that anything has gone wrong. This is the easiest kind of redundancy to analyze, although it is difficult to implement in hardware. If periodic checkout is not performed, the failed unit might not be discovered until system failure.

Manual switching and repair/replacement are different degrees of the same thing. An operator might have only to turn a switch or valve handle; or he may merely release some catches or quick disconnects, pull out the faulty unit, and shove in a good one. The time it takes for removal/installation and the time for acquiring the spare are usually matters of degree, rather than of kind, in the analysis. In a fixed ground installation, the whole thing might be accomplished in a few minutes for a radio-receiver. The transmission in a tank might take hours to remove/install and days to fix or acquire another.

The method that the designer finally chooses depends on the system specifications and constraints, on what he is familiar with, and on what he thinks will really happen in the field. A lot depends on the kind of logistic system in use for that equipment.

Often, a Monte Carlo simulation of the system is the only practical way to analyze what will happen. In such an analysis it often pays to be aware of some of the "paths" a system takes during the failure/repair sequences. In complicated systems, the designer might be quite surprised at what happens; situations easily can arise that the designer never dreamed of.

Reconfiguration of the system to operate in a degraded mode after a failure and before a repair is effected is often a desirable situation. A computer for example might continue to operate but at a lower speed during the 5 min it takes to remove and replace a unit. A communication system might slow its message rate during switchover. The slew rate of a hydraulically powered system might drop to one-third its usual value while a redundant part of the pumping system is being replaced.

As a matter of practical fact, a designer will make many decisions without using much more than the engineering judgment of himself and his associates (staff or line). There is not enough time, money, or people to analyze everything.

7-5 FAILURE BEHAVIOR OF SPARES AND OTHER PARTS

The terminology in this field is very confusing because it has grown like Topsy. The best terminology seems to be cold-warm-hot spares; it is flexible and is not confused with other aspects of system design. The crux of the matter is the failure behavior of the units; but some of the terminology refers to the use of the unit and only indirectly implies the failure behavior. The remainder of this paragraph presumes constant failure rates. More-complicated failure distributions can be discussed, but the origin of time must always then be kept track-of for every unit--a difficult task indeed.

A cold unit has zero failure rate. This is not a likely situation because spares in storage, etc., do deteriorate. But it is very tractable in an analysis. This is the same as passive-redundancy. In many cases it is what an author means by standby-redundancy (unless he has otherwise specified the failure behavior).

A hot unit has the same failure rate as an operating unit, regardless of whether it is actually in operation or not. This is the same as active redundancy. It is sometimes implied (by some authors) just by the word redundancy.

A warm unit has a failure rate somewhere between a hot unit and a cold unit. Often it is taken to be the general case and includes hot and cold as limiting situations.

In some analyses where the units always are working, the individual failure rates depend on the number that are working. A conceptually simple example is several induction motors (tied firmly together so that their shafts are effectively in line). Suppose the failure mode is insulation failure due to temperature rise and there are six high-slip 5 hp motors driving a 20-hp load. The temperature rise of the operating motors will depend on the number of operating motors. Allow 10 percent for nonuniform distribution of load. Then the maximum load on each motor when six motors are operating is (20-hp/6) × 1.1 = 3.7-hp; for five motors it is 4.4-hp; for four motors, it is 5.5-hp; and for three motors, it is 7.3-hp. Obviously, the insulation will degrade much faster as the number of motors is reduced. At nominal 7.3-hp load, the current would probably be high enough to kick out the overloads. Another example is a communication system. If radio receivers are handling traffic in parallel, the failure rate of each receiver is probably independent of the number of units which are operating, unless heat dissipation is a critical factor.

It is best to use a term to describe redundancy which indicates the failure rate behavior, not the operating condition of a redundant/spare unit.

7-6 STYLES OF REDUNDANCY

There are at least three styles of creating redundancy:

(1) k-out-of-n systems
(2) Voting techniques
(3) Other.

The "Other" category includes combinations of the first two, and multiple units which do not easily reduce to k-out-of-n. Hammock (bridge) networks are in the latter category. It is most important to distinguish between the physical system and the logic chart used to describe the physical system. The description difficulty typically arises when there are two "opposite" failure modes: open - short, dud - premature, too soon - too late, high - low, etc.; then at least two logic charts are necessary for the one physical system. Very often a redundant feature for one mode turns out to be a series feature for the other mode. For example, features which decrease the probability of prematures, will usually increase the probability of duds. The Bibliography at the end of this chapter shows sources of further information.

7-6.1 *k*-OUT-OF-*n* SYSTEMS

A *k*-out-of-*n*:G-system has *n* units and is Good (up) if and only if at least *k* units are Good (up).

A *k*-out-of-*n*:F-system has *n* units and is Failed (down) if and only if at least *k* units are Failed (down).

A series system is a 1-out-of-*n*:F (*n*-out-of-*n*:G)-system—i.e., if 1 unit fails, the system fails—all units must be good for the system to be good.

A parallel system is usually taken to be a 1-out-of-*n*:G (*n*-out-of-*n*:F)-system—i.e., if 1 unit is good, the system is good—all units must be failed for the system to fail.

A *k*-out-of-*n*:F system is an $(n - k + 1)$-out-of-*n*:G-system; and a *k*-out-of-*n*:G-system is an $(n - k + 1)$-out-of-*n*:F-system. Sometimes the name parallel-system is used synonymously with a *k*-out-of-*n* system. Since the term *parallel* is ambiguous, it is best avoided when accurate description is needed. The *k*-out-of-*n*:G or *k*-out-of-*n*:F notations are much to be preferred.

A *k*-out-of-*n* system is also an ambiguous phrase and is used both ways in the literature. It is best to use the :G or :F notation when accurate description is needed, and to define it.

The *k*-out-of-*n* system is usually easy to analyze if the redundancy is either hot or cold and the switching is perfect. The general case for warm redundancy and imperfect switching has not been solved in general. Some results are available for small *n* and constant failure rates for each unit. Ref. 3 provides an extended summary and analysis of many *k*-out-of-*n* systems.

7-6.2 VOTING TECHNIQUES

Voting ordinarily is associated with digital electronic circuits, although some circuits for analog electronic systems have appeared in the literature. It does not appear to be applicable at all to mechanical systems.

A voter has *n* active inputs, the output corresponds to the inputs which are the same for more than $n/2$ of the inputs. In most hardware implementations, $n = 3$, and two inputs determine the output. If a unit fails (and the failure is somehow sensed), the failed unit can be removed and the voter can be restructured. If $n = 3$ and one unit fails without being removed, then $n = 2$ and all must agree, in order for a signal to be passed on. If those two disagree, then the designer has to decide what to do. Refs. 1, 2, and 4 discuss this situation and give some other references.

It is possible to have some spares for some voters, e.g., each element could be a *k*-out-of-*n* subsystem. The voters themselves can be arranged in a voting fashion. Refs. 1 and 4 describe many of the possibilities for redundancy in computers. Refs. 2 and 3 give many of the formulas that are useful in analyzing these redundancies.

7-6.3 OTHER SYSTEMS

Voting techniques can be combined with *k*-out-of-*n* systems to enhance hardware reliability along with masking of faults which need not be permanent. Very elaborate redundancy techniques are best avoided unless an extremely thorough investigation, both theoretical and practical, has been made of the proposed system. Coverage is a term used to describe the detection-switching-retention process in redundancy. In order for automatic redundancy to be effective, failed units must be detected accurately and without false alarms, then the spare unit (somehow known to be good) must be switched in, and the information that the system was processing cannot be mangled during the operation.

There are redundant (nonvoting) systems

that cannot be reduced to the *k*-out-of-*n* type. The logic diagrams for the irreducible networks often are called bridge or hammock networks (bridge because of the similarity to a Wheatstone bridge; hammock because the appearance can be like a rope hammock). The success or failure events for these networks usually are more complicated than simple series-parallel networks. Some analytic methods of reliability calculation do not handle bridge networks very well.

There are, of course, many kinds of redundancy which are not easily classified. For example, some auxiliary systems to be used only in emergencies are not equivalent to the systems they "replace". Another example is the restructuring kind of redundancy where, if a unit fails, other units are restructured to keep the system going, albeit at a reduced level.

REFERENCES

1. J. L. Bricker, "A Unified Method for Analyzing Mission Reliability for Fault Tolerant Computer Systems", IEEE Transactions on Reliability, R-22, pp. 72-77, June 1973.
2. N. G. Dennis, "Reliability Analyses of Combined Voting and Standby Redundancies", IEEE Transactions on Reliability, R-23, April 1974.
3. N. G. Dennis, "Insight Into Standby Redundancy via Unreliability", IEEE Transactions on Reliability, R-23, Dec. 1974.

   (The Dennis papers contain many further references.)
4. Mathur and deSousa, "Reliability Models of NMR Systems," IEEE Transactions on Reliability, R-24, June 1975.

BIBLIOGRAPHY

Gnedenko, Belyayev, and Solovyev, *Mathematical Methods of Reliability Theory*, Academic Press, N. Y., 1969.

IEEE Transactions on Reliability.

Proceedings of the Annual Symposia on Reliability.

Proceedings of the Annual Reliability and Maintainability Symposia.

M. L. Shooman, *Probabilistic Reliability*, McGraw-Hill Book Company, Inc., N. Y., 1968.

CHAPTER 8 RELIABILITY PREDICTION (PASSIVE REDUNDANCY, PERFECT SWITCHING)

8-0 LIST OF SYMBOLS

i_s, i_o, i_g = event of short, open, or good for capacitor i

F = event of failure

k-out-of-n:F = special kind of system

k-out-of-n:G = special kind of system

MTF_i = Mean Time to Failure for case i

n = number of logic elements

n_1 = greatest "integer $< n/2$"

$\mathcal{R}_i$ = s-reliability for case i

R_i, R = element s-reliabilities

R_v = s-reliability of the voter

$\bar{\mathcal{R}}_i$ = $1-\mathcal{R}_i$

$\bar{R}_i$ = $1-R_i$

s- = denotes statistical definitions

8-1 INTRODUCTION

This chapter deals with the simplest of formulas. The probability of failure of each element is not affected by its active/standby status nor by the condition of other elements. Switching is either (a) perfect, i.e., switching and all of its ramifications are not considered at all; or (b) can be represented adequately by a block in the logic diagram.

In analyzing a system by this method, the distinction between the physical situation and the logic chart always must be kept in mind. Elements that are physically in series can be logically in parallel (it depends on failure modes). If two centrifugal pumps are physically in tandem and one stops running, the other could possibly carry the load; they would be logically in parallel. Refs. 3-8 give many formulas for system reliability. Series and parallel are terms which are best avoided when precision is necessary.

All element behaviors are conditionally s-independent (the "conditional" is to emphasize that unconditional s-independence is rarely obtained).

8-2 k-OUT-OF-n SYSTEMS

A k-out-of-n:F-system has n elements and Fails if and only if at least k elements Fail.

A k-out-of-n:G-system has n elements and is Good if and only if at least k elements are Good.

Case 1. k-out-of-n:G, all $R_i = R$

$$\mathcal{R}_1 = \sum_{k}^{n} \binom{n}{i} R^i \bar{R}^{n-i}$$

$$= \sum_{0}^{n-k} \binom{n}{i} R^{n-i} \bar{R}^i \tag{8-1a}$$

$$\bar{\mathcal{R}}_1 = \sum_{0}^{k-1} \binom{n}{i} R^i \bar{R}^{n-i}$$

$$= \sum_{n-k+1}^{n} R^{n-i} \bar{R}^i \tag{8-1b}$$

Case 2. k-out-of-n:F, all $R_i = R$

$$\bar{\mathcal{R}}_2 = \sum_{k}^{n} \binom{n}{i} \bar{R}^i R^{n-i}$$

$$= \sum_{0}^{n-k} \binom{n}{i} \bar{R}^{n-i} R^i \tag{8-2a}$$

$$\mathcal{R}_2 = \sum_{0}^{k-1} \binom{n}{i} \bar{R}^i R^{n-i}$$

$$= \sum_{n-k+1}^{n} \bar{R}^{n-i} R^i \tag{8-2b}$$

Case 3. 1-out-of-n:G (parallel)

$$\bar{\mathcal{R}}_3 = \bar{R}_1 \; \bar{R}_2 \cdots \bar{R}_n \tag{8-3}$$

Case 4. 1-out-of-n:G (parallel), all $R_i = R$

$$\bar{\mathcal{R}}_4 = \bar{R}^n \tag{8-4}$$

Case 5. 1-out-of-n:F (series)

$$R_5 = R_1 \; R_2 \cdots R_n \tag{8-5}$$

Case 6. 1-out-of-n:F (series), all $R_i = R$

$$R_6 = R^n \tag{8-6}$$

The formulas for k-out-of-n systems when all $R_i \neq R$ are not tractable. They are derived generally as shown in par. 8-4.

8-3 COMBINATIONS OF SERIES-PARALLEL ELEMENTS

Many systems can be considered as made up of series-parallel combinations of elements. A convenient technique for reliability calculations is to reduce each simple combination of series or parallel elements to a single element with the reliability of the combination. Example No. 1 (Fig. 8-1) shows how the reduction is performed. Fig. 8-1(A) shows the original logic chart. Each block is an element and is numbered. Equivalent blocks are numbered further.

The first reduction takes place as follows (Fig. 8-1(A) to Fig. 8-1(B)):

$$\bar{R}_{14} = \bar{R}_7 \, \bar{R}_8 \, \bar{R}_9 \tag{8-7a}$$
$$R_{14} = 1 - \bar{R}_{14} \tag{8-7b}$$

$$R_{15} = R_{10} \, R_{11} \tag{8-8a}$$
$$\bar{R}_{15} = 1 - R_{15} \tag{8-8b}$$

$$R_{12} = R_2 \, R_3 \tag{8-9a}$$
$$\bar{R}_{12} = 1 - R_{12} \tag{8-9b}$$

$$\bar{R}_{13} = \bar{R}_4 \, \bar{R}_5 \tag{8-10a}$$
$$R_{13} = 1 - \bar{R}_{13} \tag{8-10b}$$

The second reduction is as follows (Fig. 8-1(B) to Fig. 8-1(C)):

$$R_{16} = R_6 \, R_{14} \tag{8-11a}$$
$$\bar{R}_{16} = 1 - R_{16} \tag{8-11b}$$

The third reduction is as follows (Fig. 8-1(C) to Fig. 8-1(D)):

$$\bar{R}_{17} = \bar{R}_{15} \, \bar{R}_{16} \tag{8-12a}$$
$$R_{17} = 1 - \bar{R}_{17} \tag{8-12b}$$

The fourth reduction is as follows (Fig. 8-1(D) to Fig. 8-1(E)):

$$R_{18} = R_{13} \, R_{17} \tag{8-13a}$$
$$\bar{R}_{18} = 1 - R_{18} \tag{8-13b}$$

The fifth reduction is as follows (Fig. 8-1(E) to Fig. 8-1(F)):

$$\bar{R}_{19} = \bar{R}_{12} \, \bar{R}_{18} \tag{8-14a}$$
$$R_{19} = 1 - \bar{R}_{19} \tag{8-14b}$$

The final reduction is as follows (Fig. 8-1(F) to Fig. 8-1(G)):

$$R_{20} = R_1 \, R_{19} \tag{8-15a}$$
$$\bar{R}_{20} = 1 - R_{20} \tag{8-15b}$$

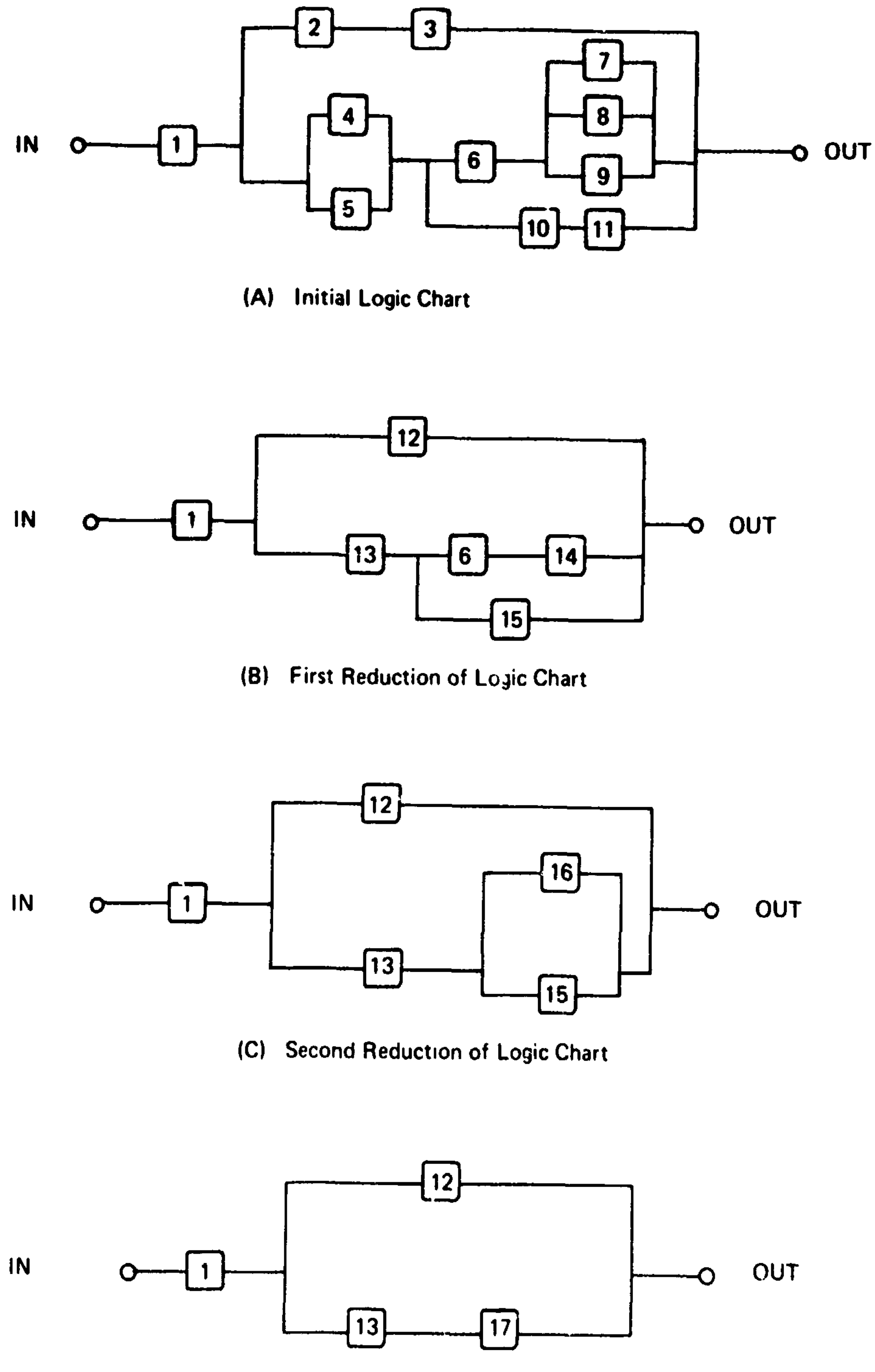
Thus a series of series-parallel reductions has solved the example problem in Fig. 8-1. There is no good reason to combine all the formulas into one expression; it would be tedious, long, and cumbersome.

Not all systems can be reduced by this technique, but a great many can. If the switching is not perfect, one of the other techniques is better—if for no other reason that not all failure events are likely to be s-independent.

8-4 EVENT ANALYSIS

When logic charts are not series-parallel arrangements, the analysis can proceed by looking at all possible events, classifying them into appropriate subsets (e.g., system-good, system-degraded, system-failure-type-1, system-failure-type-2). Then the probability of each subset is calculated by the rules for evaluating probabilities of combinations of events (Chapter 3).

Logic charts generally are drawn from a physical diagram and a knowledge of the requirements for success. In some cases, as in Example No. 2 (Fig. 8-2), it is too complicated to draw logic diagrams; instead the events are listed. There are three possible states of each capacitor and four capacitors;



(A) Initial Logic Chart

(B) First Reduction of Logic Chart

(C) Second Reduction of Logic Chart

(D) Third Reduction of Logic Chart

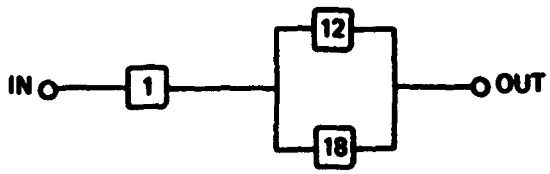
Series combinations are 1-out-of-*n* F, use Eq. 8-5.

Parallel combinations are 1-out-of-*n* G, use Eq. 8-3.

Find the system reliability and unreliability.

In this kind of diagram, success is a continuous path from input to output.

FIGURE 8-1. Logic Diagrams for Example No. 1.



(E) Fourth Reduction of Logic Chart

(F) Fifth Reduction of Logic Chart

(G) Final Reduction of Logic Chart

Series combinations are 1-out-of-n:F; use Eq. 8-5
Parallel combinations are 1-out-of-n:G, use Eq. 8-3.
Find the system reliability and unreliability.
In this kind of diagram, success is a continuous path from input to output.

FIGURE 8-1. Logic Diagrams for Example No. 1 (cont'd)

there are $3^4 = 81$ possible combinations. In order to simplify Table 8-1, the capacitor numbers are listed at the top of each column, and an "o", "s", or "g" put in the column for each event. An "f" indicates Failed for the network; a blank indicates Good. It is failed if (1 and 2 are short) ∪ (3 and 4 are short) ∪ (1 and 4 are short) ∪ (3 and 2 are short) ∪ (1 and 3 are open) ∪ (2 and 4 are open). Table 8-1 is long and tedious. The events can be put in more symbol notation and give the same results, i.e.,

$$F = (1_s \cap 2_s) \cup (3_s \cap 4_s) \cup (1_s \cap 4_s) \cup (2_s \cap 3_s) \cup (1_o \cap 3_o) \cup (2_o \cap 4_o) \; . \qquad (8\text{-}16)$$

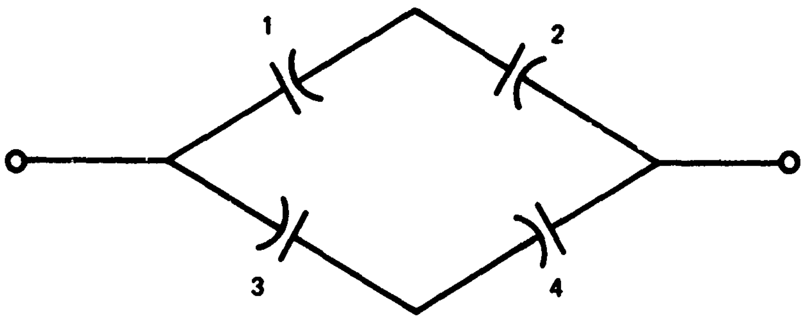
However, the events in the Table are all mutually exclusive whereas the events in parentheses in Eq. 8-16 are not.

It takes but little imagination to realize that this approach can get out of hand with very little complication of the network or system.

8-5 CUT SETS

A cut set is an event (subset of the sample space) such that when it occurs, the system fails in the indicated failure mode. A minimal cut set is a cut set such that the elimination of any element renders it no longer a cut set.

CAPACITOR BRIDGE



Capacitors can fail open or short. The network is good as long as it is neither open nor short.

i_o implies "open circuit of capacitor i"

i_s implies "short circuit of capacitor i"

i_g implies "good capacitor i"

FIGURE 8-2. Physical Diagram for Example No. 2

In the example from par. 8-4, in Fig. 8-2 and Eq. 8-16, each of the six events in parentheses in Eq. 8-16 is a minimal cut set. The $Pr\{F\}$ in Eq. 8-16 can be calculated by an iterative procedure using Eq. 2-20 which provides a series of upper and lower bounds on the $Pr\{F\}$.

The first upper bound is the sum of the probabilities of each of the six events in parentheses in Eq. 8-16. The first lower bound is found by subtracting (from the first upper bound) the sum of the probabilities of the 15 unions of each pair of the six events. The second upper bound is found by adding (to the first lower bound) the sum of the probabilities of the 20 unions of each triplet of the six events. As shown in Eq. 2-20, the unions are taken two, then three, then four, then five, and finally six at a time. The odd ones (one, three, five) are added, the even ones (two, four, six) are subtracted. An example of the procedure is shown in Ref. 1; a FORTRAN program for performing this calculation is shown in Ref. 2.

Even though the principles involved are straightforward, implementing them on any reasonably sized system can be very tedious and complicated.

Chapter 7 "Cause-Consequence Charts (and Fault Trees)" of *Part Two, Design for Reliability*, contains further information and references on finding minimal cut sets for systems; references are also made there to automated methods of finding all minimal cut sets for a fault tree.

8-6 MAJORITY VOTING

In majority-voting redundancy the proper output of the system is presumed to be the output of the majority of the individual logic elements which feed the voter (Ref. 3). The output is determined by the voter, which decides what the majority of the elements indicates. The system gives the correct output when less than half of the elements have failed and when the voter is good.

Case 7. Simple majority voting

$$\mathcal{R}_7 = R_v \sum_{n_1}^{n} \binom{n}{i} R_1^i \bar{R}_1^{n-i} \qquad (8\text{-}17)$$

where

n = number of logic elements

n_1 = greatest "integer $\leq n/2$"

R_v = s-reliability of the voter

R_1 = s-reliability of a logic element

$\bar{R}_1 = 1 - R_1$

TABLE 8-1

STATES OF CAPACITOR NETWORK IN FIG. 8-2

1 2 3 4	1 2 3 4	1 2 3 4
g g g g	s g g g	o g g g
g g g s	s g g s f	o g g s
g g g o	s g g o	o g g o
g g s g	s g s g	o g s g
g g s s f	s g s s f	o g s s f
g g s o	s g s o	o g s o
g g o g	s g o g	o g o g f
g g o s	s g o s f	o g o s f
g g o o	s g o o	o g o o f
g s g g	s s g g f	o s g g
g s g s	s s g s f	o s g s
g s g o	s s g o f	o s g o
g s s g f	s s s g f	o s s g f
g s s s f	s s s s f	o s s s f
g s s o f	s s s o f	o s s o f
g s o g	s s o g f	o s o g f
g s o s	s s o s f	o s o s f
g s o o	s s o o f	o s o o f
g o g g	s o g g	o o g g
g o g s	s o g s f	o o g s
g o g o f	s o g o f	o o g o f
g o s g	s o s g	o o s g
g o s s f	s o s s f	o o s s f
g o s o f	s o s o f	o o s o f
g o o g	s o o g	o o o g f
g o o s	s o o s f	o o o s f
g o o o f	s o o o f	o o o o f

Eq. 8-17 assumes that failure of any element is absolute (i.e., it cannot assist in giving the correct answer) and is *s*-independent. Other analyses are possible which make other more realistic assumptions about the failures.

The voter itself can be made into a majority element; the analysis of such a system becomes quite complicated.

REFERENCES

1. A. C. Nelson, J. R. Batts, R. L. Beadles, "A Computer Program for Approximating System Reliability", IEEE Transactions on Reliability, R-19, 61-65, May 1970.
2. J. R. Batts, "Computer Program for Approximating System Reliability-Part II", IEEE Transactions on Reliability, R-20, 88-90, May 1971.
3. *Handbook for Systems Application of Redundancy*, US Naval Applied Science Laboratory, 30 August 1966.
4. N. G. Dennis, "Reliability Analyses of Combined Voting and Standby Redundancies", IEEE Transactions on Reliability, R-23, April 1974.
5. N. G. Dennis, "Insight Into Standby Redundancy via Unreliability", IEEE Transactions on Reliability, R-23, December 1974.
6. M. L. Shooman, *Probabilistic Reliability*, McGraw-Hill, N.Y., 1968.
7. Gnedenko, Belyayev, and Soloveyv, *Mathematical Methods of Reliability Theory*, Academic Press, N.Y., 1969.
8. Mathur and deSousa, "Reliability Models of NMR Systems", IEEE Transactions on Reliability, R-24, June 1975.

CHAPTER 9 RELIABILITY PREDICTION (TIME DEPENDENT)

9-0 LIST OF SYMBOLS

$csqf(\chi^2, \nu)$ = chi square *Cdf* with ν degrees of freedom
$csqfc(\chi^2, \nu)$ = 1-$csqf(\chi^2, \nu)$
$f(t)$ = *pdf* of t
$f(t), g(t)$ = *pdf's* for elements in par. 9-6
f_α = *pdf* for element α in par. 9-7
F_α = *Cdf* for element α in par. 9-7
$\bar{F}(t), \bar{G}(t)$ = *Sf's* for elements in par. 9-6
$\bar{F}_\alpha$ = *Sf* for element α in par. 9-7
$gauf(\cdot)$ = *Cdf* for *s*-normal (Gaussian) distribution
$gaufc(\cdot)$ = 1-$gauf(\cdot)$
k = λ/λ'
MTF_i = Mean Time to Failure for case i
p_i, q_i = element *s*-reliability and *s*-unreliability, respectively, (Table 9-2)
pdf = probability density function
$R(t), R_i(t)$ = *s*-reliability during interval 0 to t
$\mathcal{R}_i$ = *s*-reliability for case i
$\bar{\mathcal{R}}_i$ = $1-\mathcal{R}_i$
s- = denotes statistical definition
Sf = Survivor function
t = time, time-to-failure
t_1 = a time $0 < t_1 < t$
z_α = standard *s*-normal variate
θ_i = an *MTF* for situation i
λ, λ_i = failure rates
λ', λ_α = failure rates
λt = dimensionless "parameter"
μ, σ = mean and standard deviation, respectively, for an *s*-normal distribution
τ = λt; time interval for par. 9-9

9-1 INTRODUCTION

There is a multitude of formulas for calculating reliability of redundant systems. Virtually all of them presume conditional *s*-independence of the elements. It is important in a practical analysis to list each set of conditions under which *s*-independence will hold.

In the vast majority of cases in analyses for redundancy, transition rates (e.g., failure and repair rates) are assumed to be constant. Any other assumption causes many complications in the analysis.

9-2 MEASURES OF RELIABILITY

The two measures most frequently used to compare the effectiveness of redundancy are:

1. Mean time to failure (*MTF*) of the system—useful when mission times are long compared to the lives of elements.
2. Probability of failure of the system—useful when mission times are short compared to the lives of elements.

In all cases in this volume, the proviso exists on all formulas that the indicated operation is "legal" and the result exists. The proviso is satisfied for practical reliability problems.

The *MTF* is defined as

$$MTF \equiv \int_0^\infty t\, f(t)\, dt = \int_0^\infty R(t)\, dt \tag{9-1}$$

where

$f(t)$ = *pdf* of time to failure
$R(t)$ = *Sf* of time to failure

9-3 THE EXPONENTIAL DISTRIBUTION

The time-to-failure *pdf* and the reliability function (survivor function *Sf*) of the exponential distribution are, respectively,

$$f(t) = \lambda e^{-\lambda t}$$
$$R(t) = e^{-\lambda t} \tag{9-2}$$

where λ is the constant failure (hazard) rate. All failures are *s*-independent and all standbys are hot (active).

Case 1. Two elements in parallel (1-out-of-2:G) have failure rates, λ_a and λ_b. The *s*-reliability $\mathcal{R}_1(t)$ is

$$\mathcal{R}_1(t) = 1 - (1 - e^{-\lambda_a t})(1 - e^{-\lambda_b t}) = e^{-\lambda_a t} + e^{-\lambda_b t} - e^{-(\lambda_a + \lambda_b)t} \tag{9-3a}$$

$$MTF_1 = \frac{1}{\lambda_a} + \frac{1}{\lambda_b} - \frac{1}{\lambda_a + \lambda_b} \tag{9-3b}$$

Case 2. Same as Case 1, except $\lambda_a = \lambda_b = \lambda$ (identical elements), then

$$R_2(t) = e^{-\lambda t}(2 - e^{-\lambda t}), \qquad (9\text{-}4a)$$

$$MTF_2 = \frac{3}{2\lambda}. \qquad (9\text{-}4b)$$

Case 3. *m* active-parallel elements (1-out-of-*m*:G, hot standby).

$$\bar{R}_3(t) = \prod_{i=1}^{m} (1 - e^{-\lambda_i t}) \qquad (9\text{-}5a)$$

$$MTF_3 = \sum_{i=1}^{m} \frac{1}{\lambda_i} - \sum_{\substack{i,j=1 \\ i<j}}^{m} \frac{1}{\lambda_i + \lambda_j} + \sum_{\substack{i,j,k=1 \\ i<j<k}}^{m} \frac{1}{\lambda_i + \lambda_j + \lambda_k} - \ldots \qquad (9\text{-}5b)$$

Case 4. Same as Case 3, except all elements are identical, $\lambda_i = \lambda$ for all *i*.

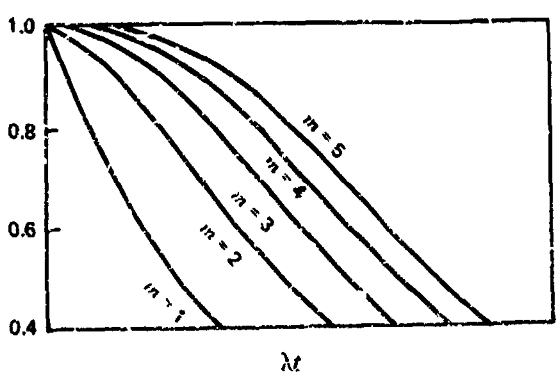
$$\bar{R}_{4(t)} = (1 - e^{-\lambda t})^m \qquad (9\text{-}6a)$$

$$MTF_4 = \frac{1}{\lambda} \sum_{i=1}^{m} \frac{1}{i}. \qquad (9\text{-}6b)$$

9-3.1 RELIABILITY IMPROVEMENT

The reliability functions for a system with *m* parallel (1-out-of-*m*:G, hot standby) elements (m = 1,2,3,4,5) and $\lambda = \lambda_i$ = constant are plotted in Fig. 9-1.

Another method of measuring reliability improvement is to calculate the ratios (or differences) in *MTF* of two systems. Table 9-1



William H. Von Alven, Ed., *Reliability Engineering*, © 1964 by ARINC Research Corporation. Reprinted by permission of Prentice-Hall, Inc., Englewood Cliffs, N.J.

FIGURE 9-1. Reliability Function for Systems With m Identical, Active-Parallel Elements, Each With Constant Failure Rate λ (1-out-of-m:G)

TABLE 9-1

RATIOS OF MTF'S FOR m ACTIVE-PARALLEL ELEMENTS[2]

m	θ_m/θ_1	θ_m/θ_{m-1}
1	1.00	–
2	1.50	1.50
3	1.83	1.22
4	2.08	1.14
5	2.28	1.10

William H. Von Alven, Ed., *Reliability Engineering*, © 1964 by ARINC Research Corporation. Reprinted by permission of Prentice-Hall, Inc., Englewood Cliffs, N.J.

gives the ratios of *MTF* for θ_m/θ_1 and θ_m/θ_{m-1}, for m = 1,2,3,4,5;

where θ_i = *MTF* for *i* elements as given by Eq. 9-6b.

From Table 9-1 it can be seen that the θ_m/θ_{m-1} maximum occurs when $m = 2$.

The improvements are, in most cases, the maximum that can be achieved. If the elements have more than one failure mode and/or if switching is imperfect, the effectiveness of the redundancy is reduced.

9-3.2 REDUNDANCY VERSUS IMPROVED ELEMENTS

A system designer may have the option of adding redundant elements or using improved elements in a nonredundant configuration to increase reliability (Refs. 1 and 2). The designer must consider effectiveness, cost, weight, maintenance, and other related considerations in making his choice.

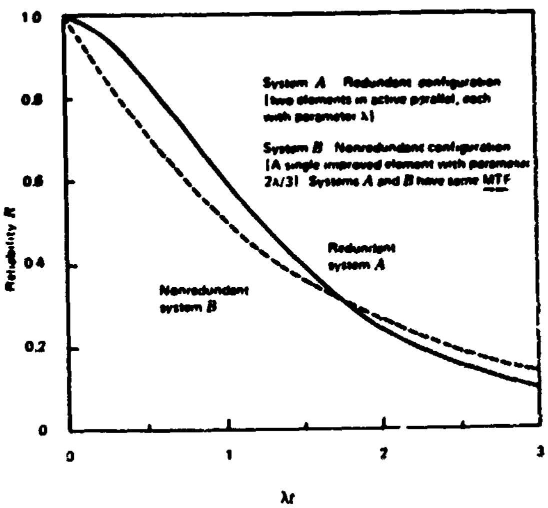
Case 5. Two alike elements are connected in active-parallel (Case 2); their *MTF* is $3/(2\lambda)$, from Eq. 9-4a. To obtain the same *MTF* with a single improved element, the improved element must have $\lambda' = 2\lambda/3$.

The s-reliability R_5 of the improved element is

$$R_5 = e^{-\lambda' t} = e^{-2\lambda t/3} \qquad (9\text{-}7a)$$

$$MTF_5 = \frac{1}{\lambda'} = \frac{1}{2\lambda} \qquad (9\text{-}7b)$$

The s-reliabilities R_2 and R_5 are plotted in



William H. Von Alven, Ed., *Reliability Engineering*, © 1964 by ARINC Research Corporation. Reprinted by permission of Prentice-Hall, Inc., Englewood Cliffs, N.J.

FIGURE 9-2. Survivor Functions for Two Particular Systems With the Same MTF[2]

Fig. 9-2. From the figure, the redundant system has the greater reliability up to $\lambda t \approx 1.75$. After that, the improved single-element system is the more reliable. The point of intersection of the two functions will change if more redundant elements are added, if the degree of element improvement varies, or if standby redundancy is used.

In redundancy applications, there is usually one time, say t', when the reliability of a nonredundant system with improved elements is equal to the reliability of a redundant system with less reliable elements. When $t < t'$, the redundant system has the greater reliability. When $t > t'$, the improved-element system is superior. The choice of the system configuration depends on the ratio of element life to mission time.

9-4 THE *s*-NORMAL DISTRIBUTION

The *s*-normal distribution is useful to describe many systems whose failure rate increases "to infinity". Its *pdf* is

$$f(t) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left[-\frac{1}{2}\left(\frac{t-\mu}{\sigma}\right)^2\right] \qquad (9\text{-}8)$$

where

μ = mean time to failure
σ = standard deviation

We also introduce the following notation.

gauf (z) = *Cdf* of the standard *s*-normal (Gaussian) distribution (μ=0, σ=1), the probability of failure)

gaufc (z) = *Sf* of the standard *s*-normal distribution (the reliability; it is the complement of the *gauf*(z). (the reliability)

Case 6. Two elements in active parallel redundancy (1-out-of-2:G, hot standby); each has an *s*-normal distribution of time to failure with parameters μ_a, σ_a and μ_b, σ_b. Define

$$z_a \equiv \frac{t-\mu_a}{\sigma_a},\ z_b \equiv \frac{t-\mu_b}{\sigma_b}. \qquad (9\text{-}9)$$

From Eq. 8-3, the probability of failure is

$$\bar{R}_6 = gauf\,(z_a)\,gauf\,(z_b) \qquad (9\text{-}10)$$

To illustrate Case 6, assume that the two components, A and B, have the following parameters:

$$\mu_a = 300 \text{ hr} \quad \mu_b = 400 \text{ hr}$$
$$\sigma_a = 40 \text{ hr} \quad \sigma_b = 60 \text{ hr} \qquad (9\text{-}11)$$

In order to evaluate the reliability of this redundant unit at, say 350 hr, the following computation is performed using Eq. 9-9:

$$z_a = \frac{350 \text{ hr} - 300 \text{ hr}}{40 \text{ hr}} = 1.25$$
$$z_b = \frac{350 \text{ hr} - 400 \text{ hr}}{60 \text{ hr}} = -0.833 \qquad (9\text{-}12)$$

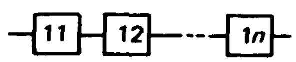
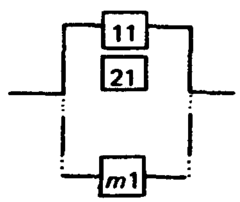
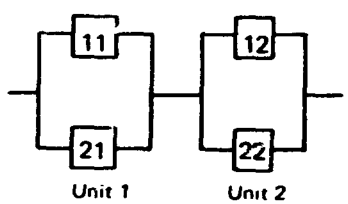
Now refer to the tables of the *s*-normal distribution.

Unreliability or probability of failure = *gauf*(1.25) *gauf*(−0.833) =
$0.8944 \times 0.2026 = 0.1812 \approx 0.18$ (9-13)

9-5 OTHER CONFIGURATIONS

Table 9-2 lists the reliability of several combinations of elements. The last column shows the *MTF* under the assumption that all elements have an identical constant failure rate.

TABLE 9-2. RELIABILITY FUNCTIONS FOR VARIOUS ACTIVE-PARALLEL (1-out-of-n: G) CONFIGURATIONS[2]

Reliability Block Diagram	Configuration	Reliability Function, $R(t)$	MTF
	1. Basic Series (a) General case (b) Identical elements	$p_{11}(t)p_{12}(t) \ldots p_{1n}(t)$ $p(t)^n$	$\frac{1}{n\lambda}$
	2. Basic Parallel (a) General case (b) Identical elements	$1 - q_{11}(t)q_{21}(t) \ldots q_{m1}(t)$ $1 - q(t)^m$	$\frac{1}{\lambda} \sum_{i=1}^{m} \frac{1}{i}$
	3. Series-Parallel 2 X 2 (a) General case (b) Identical elements in units (c) Identical elements	$[1 - q_{11}(t)q_{21}(t)][1 - q_{12}(t)q_{22}(t)]$ $[1 - q_1(t)^2][1 - q_2(t)^2]$ $[1 - q(t)^2]^2$	$\frac{11}{12\lambda}$

William H. Von Alven, Ed., ***Reliability Engineering***, © 1964 by ARINC Research Corporation. Reprinted by permission of Prentice-Hall, Inc., Englewood Cliffs, N.J.

TABLE 9-2. RELIABILITY FUNCTIONS FOR VARIOUS ACTIVE-PARALLEL (1-out-of-n: G) CONFIGURATIONS[2] (cont'd)

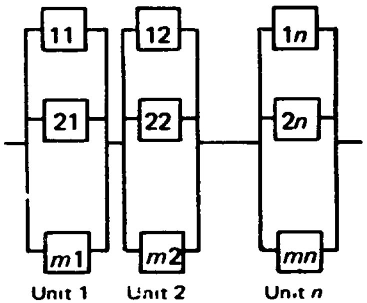
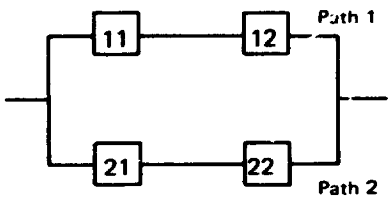
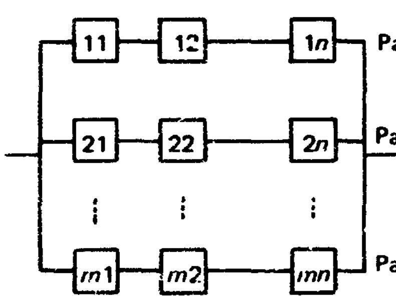
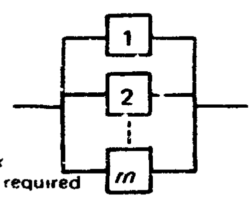
	4. Series-Parallel $m \times n$		
	(a) General case	$\prod_{j=1}^{n}[1-q_{1j}(t)q_{2j}(t)\ldots q_{mj}(t)]$	
	1 (b) Identical elements in units	$\prod_{j=1}^{n}[1-q_j(t)^m]$	
	(c) Identical elements	$[1-q(t)^m]^n$	$\frac{1}{\lambda}\sum_{i=1}^{n}(-1)^{i+1}\binom{n}{i}\sum_{j=1}^{im}\frac{1}{j}$
	5. Parallel-Series 2×2		
	(a) General case	$1-[1-p_{11}(t)p_{12}(t)][1-p_{21}(t)p_{22}(t)]$	
	(b) Identical elements in paths	$1-[1-p_1(t)^2][1-p_2(t)^2]$	
	(c) Identical elements	$1-[1-p(t)^2]^2$	$\frac{3}{4\lambda}$

TABLE 9-2. RELIABILITY FUNCTIONS FOR VARIOUS ACTIVE-PARALLEL (1-out-of-n: G) CONFIGURATIONS[2] (cont'd)

	6. Parallel-Series $m \times n$ (a) General case (b) Identical elements in paths (c) Identical elements	$1 - \prod_{i=1}^{m} [1 - p_{i1}(t)p_{i2}(t) \ldots p_{in}(t)]$ $1 - [1 - p_1(t)p_2(t) \ldots p_n(t)]^m$ $1 - [1 - p(t)^n]^m$	 $\frac{1}{n\lambda} \sum_{i=1}^{m} \frac{1}{i}$
At least k elements required	7. Partial Redundancy (require at least k satisfactory elements) (a) Identical elements	$\sum_{i=k}^{m} \binom{m}{k} p(t)^i [1 - p(t)]^{m-i}$	$\frac{1}{\lambda} \sum_{i=k}^{m} \frac{1}{i}$

Element ij refers to the element in the ith row and jth column.

$i = 1,2,\ldots, m$; $j = 1,2,\ldots, n$.

Notation

$p(t)$ = element reliability function j

$= \int_t^{\infty} f(t)dt$

$q(t) = 1 - p(t)$

= element unreliability function.

When elements have exponential failure *pdf* with failure rate λ,

$p(t) = e^{-\lambda t}, q(t) = 1 - e^{-\lambda t}.$

Notation for Table 9-2:

p_i = survival probability of element i

q_i = $1 - p_i$

λ = common constant failure rate for last column.

If the failure rates are neither common nor constant, the *MTF* is tedious and difficult to calculate. As an example, assume the redundant system in Fig. 9-3. System reliability can be determined from

$$R_s(t) = e^{-\lambda_a t}[e^{-\lambda_b t} + e^{-\lambda_c t} - e^{-(\lambda_b + \lambda_c)t}] \times [e^{-(\lambda_d + \lambda_e)t} + e^{-\lambda_f t} - e^{-(\lambda_d + \lambda_e + \lambda_f)t}]. \quad (9\text{-}14)$$

$$= \sum_{i=1}^{5} e^{-\lambda_i t} - \sum_{i=6}^{9} e^{-\lambda_i t}, \quad (9\text{-}15)$$

where

$$\lambda_1 \equiv \lambda_a + \lambda_b + \lambda_f = 0.020$$
$$\lambda_2 \equiv \lambda_a + \lambda_b + \lambda_d + \lambda_e = 0.022$$
$$\lambda_3 \equiv \lambda_a + \lambda_c + \lambda_d + \lambda_e = 0.027$$
$$\lambda_4 \equiv \lambda_a + \lambda_c + \lambda_f = 0.025$$
$$\lambda_5 \equiv \lambda_a + \lambda_b + \lambda_c + \lambda_d + \lambda_e + \lambda_f = 0.037$$
$$\lambda_6 \equiv \lambda_a + \lambda_b + \lambda_d + \lambda_e + \lambda_f = 0.027$$
$$\lambda_7 \equiv \lambda_a + \lambda_c + \lambda_d + \lambda_e + \lambda_f = 0.032$$
$$\lambda_8 \equiv \lambda_a + \lambda_b + \lambda_c + \lambda_d + \lambda_e = 0.032$$
$$\lambda_9 \equiv \lambda_a + \lambda_b + \lambda_c + \lambda_f = 0.030. \quad (9\text{-}16)$$

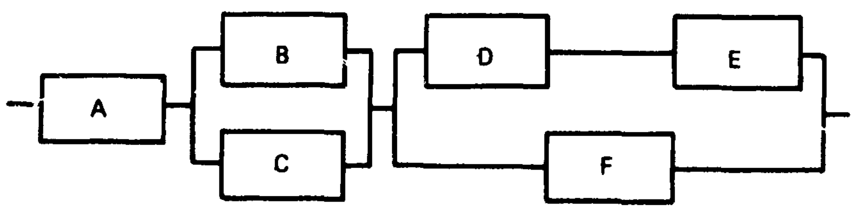
The *MTF* is computed by integrating the reliability function:

$$MTF = \sum_{i=1}^{5} \frac{1}{\lambda_i} - \sum_{i=6}^{9} \frac{1}{\lambda_i} = 199.5 - 132.9 = 66.6. \quad (9\text{-}17)$$

9-6 *s*-DEPENDENT FAILURE PROBABILITIES

Up to this point, it has been assumed that the failure of an active redundant element has no effect on the other active elements. However, the opposite condition often arises—the failure of one element does affect the others. For example, consider the block diagram in Fig. 9-4. A and B are both fully energized, and normally share or carry half the load—$L/2$. If either A or B fails, the survivor must then carry the full load. Hence, the probability that one (say B) fails depends on the state of the other if failure probability is related to load or stress. A simple example would be a 2-engine airplane which, if one engine fails, can still keep flying. However, the surviving engine now has to carry the full load and has a higher probability of failing.

For this relatively simple example, the reliability function can be derived by considering all possible ways of system success, as shown in Fig. 9-5. The bar above a letter represents failure of that element. The prime represents operation of that element



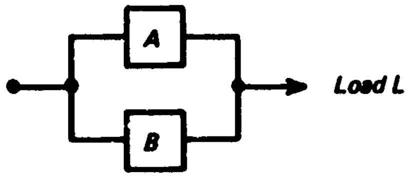
$\lambda_a = 0.0100$ $\quad$ $\lambda_c = 0.0100$ $\quad$ $\lambda_e = 0.0033$

$\lambda_b = 0.0050$ $\quad$ $\lambda_d = 0.0033$ $\quad$ $\lambda_f = 0.0050$

For convenience, the λ has been taken as dimensionless. Actually, the *MTF* will have the reciprocal dimension of the λ.

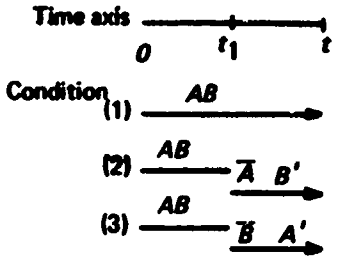
William H. Von Alven, Ed., *Reliability Engineering*, © 1964 by ARINC Research Corporation. Reprinted by permission of Prentice-Hall, Inc., Englewood Cliffs, N.J.

FIGURE 9-3. Illustrative System[2]



William H. Von Alven, Ed., *Reliability Engineering*, © 1964 by ARINC Research Corporation. Reprinted by permission of Prentice-Hall, Inc., Englewood Cliffs, N.J.

FIGURE 9-4. System With Load Dependent Failure[2]



Success = Conditions (1), (2), or (3)

FIGURE 9-5. Time Sequence Diagram[2]

under full load; absence of a prime represents operation under half load.

The derivation is as follows. Let

$f(t)$ = failure-time *pdf* of each element when both elements are operating;

$\bar{F}(t)$ = *Sf* correponding to $f(t)$

$g(t)$ = element failure-time *pdf* of the unfailed element when one element has failed;

$\bar{G}(t)$ = *Sf* corresponding to $g(t)$

$t_1 < t$ = some point in time

L = full load

The system operates satisfactorily at time t if either A or B both are operating successfully. Under the assumption that the elements are *s*-independent if both are operating, the probability that both will operate until time t is

$$[\bar{F}(t)]^2 \tag{9-18}$$

The *pdf* for one element failing at time t_1 and the other surviving to t_1 under $L/2$ and from t_1 to t under L is

$$f(t_1)\bar{F}(t_1)\bar{G}(t-t_1) \tag{9-19}$$

Since t_1 can range from 0 to t, this *pdf* is integrated over that range, and the resulting probability is doubled because the event can occur in either of two ways. Hence,

$$R(t) = [\bar{F}(t)]^2 + 2\int_0^t f(t_1)\bar{F}(t_1)\bar{G}(t-t_1)dt_1 \tag{9-20}$$

Special Case. The element failure times are exponentially distributed and each has a parameter λ under load $L/2$, and λ' under load L. Define

$$k \equiv \lambda'/\lambda. \tag{9-21}$$

The solution of Eq. 9-20 is

$$R(t) = [2\exp(-\lambda' t) - k\exp(-2\lambda t)]/(2-k), k \neq 2 \tag{9-22}$$

$$R(t) = (2\lambda t+1)\exp(-2\lambda t), k = 2 \tag{9-23}$$

The system *MTF* is

$$MTF = \frac{1}{\lambda}\left(\frac{1}{k} + \frac{1}{2}\right). \tag{9-24}$$

When $k = 1$, load-sharing is not present, i.e., increased load does not affect the element failure probability. This assumption was made in the previous discussions of active-parallel redundancy. If there were only one element, it would be operating under full load; therefore, the system *MTF* would be $1/\lambda' = 1/(k\lambda)$.

A single improved element can be used as an alternative to redundancy when this *s*-dependent model is assumed. The effects of using improved single elements or redundant standard elements can be illustrated as follows. Consider

A: Single standard element; $\lambda = 1/50$

B: Single improved element; $\lambda = 1/100$

C: *s*-Dependent model, standard elements; λ (half load) = 1/100, λ' (full load) = 1/50.

The *MTF*'s and s-reliability functions of these three configurations are

$$MTF_A = 50 \qquad (9\text{-}25a)$$

$$R_A(t) = e^{-t/50} \qquad (9\text{-}25b)$$

$$MTF_B = 100 \qquad (9\text{-}26a)$$

$$R_B(t) = e^{-t/100} \qquad (9\text{-}26b)$$

$$MTF_C = 100 \qquad (9\text{-}27a)$$

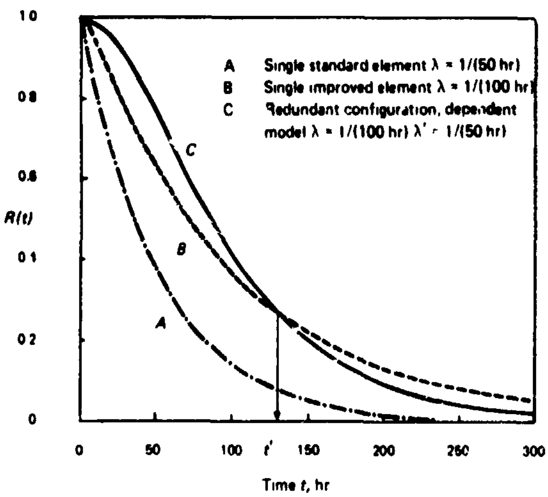
$$R_C(t) = e^{-t/50}(1 + t/50). \qquad (9\text{-}27b)$$

The s-reliability functions are shown in Fig. 9-6. Although systems B and C have the same *MTF*, the redundant system has greater reliability in early life. After approximately 125 hours, the improved single-element system is superior. If such factors as effectiveness, cost, weight, and complexity are approximately equivalent for systems B and C, the choice would depend on the Required Time of Operation for the system.

9-7 STANDBY REDUNDANCY

In a system of redundant elements that are completely on standby, the standby elements are cold (have zero failure rate) until the primary element fails (Ref. 2). The necessary switching is perfect.

Case 7. The system contains two elements, A and B; the reliability function can be found as indicated.



William H. Von Alven, Ed., *Reliability Engineering*, © 1964 by ARINC Research Corporation. Reprinted by permission of Prentice-Hall, Inc., Englewood Cliffs, N.J.

FIGURE 9-6. s-Reliability Functions for Redundant Configuration (Dependent Model) and Nonredundant Configurations[2]

The system will be successful at time t if either (letting A be the primary element):

1. A succeeds up to time t, or
2. A fails at time $t_1 < t$ and B operates from t_1 to t.

Fig. 9-7 shows these two conditions.

$$R(t) = F_a(t) + \int_0^t f_a(t_1) F_b(t-t_1)\, dt_1, \qquad (9\text{-}28)$$

The first term of Eq. 9-28 is the probability that element A will succeed until time t. The integrand is the *pdf* of A failing exactly at t_1 and B succeeding for the remaining $(t - t_1)$ hours. Since t_1 can range from 0 to t, t_1 is integrated over that range.

Case 8. Same as Case 7, but for the exponential case where the element failure rates are λ_a and λ_b,

$$R_8(t) = \frac{\lambda_b}{\lambda_b - \lambda_a} e^{-\lambda_a t} - \frac{\lambda_a}{\lambda_b - \lambda_a} e^{-\lambda_b t}, \lambda_a \neq \lambda_b \qquad (9\text{-}29a)$$

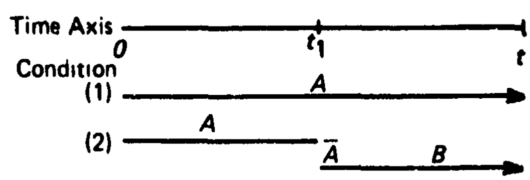
$$R(t) = e^{-\lambda t}(1 + \lambda t), \lambda_a = \lambda_b = \lambda. \qquad (9\text{-}29b)$$

It does not matter whether the more reliable element is used as the primary or the standby element.

Case 9. Same as Case 8 except there are n elements each with parameter λ.

$$R_9(t) = e^{-\lambda t} \sum_{r=0}^{n-1} \frac{(\lambda t)^r}{r!}. \qquad (9\text{-}30a)$$

$$MTF_9 = n/\lambda \qquad (9\text{-}30b)$$



William H. Von Alven, Ed., *Reliability Engineering*, © 1964 by ARINC Research Corporation. Reprinted by permission of Prentice-Hall, Inc., Englewood Cliffs, N.J.

FIGURE 9-7. Time Sequence Diagram for Standby Redundancy[2]

9-7.1 SWITCHING FAILURES

Case 10. The following notation will be used for a 2-element standby redundant unit requiring a decision-and-switching device that switches in one direction only (Ref. 2):

$f_a(t), f_b(t)$ = failure *pdf*'s of elements A and B

$f_{b'}(t)$ = failure *pdf* of element B when on standby

$f_x(t)$ = conditional contact failure *pdf* (failure of the contact to maintain a good connection, given that a good connection initially existed)

$f_y(t)$ = conditional dynamic failure *pdf* (failure to switch, given that A has failed)

$f_z(t)$ = conditional static failure *pdf* (switching when not required) required)

$F_\alpha, \bar{F}_\alpha$ = *Cdf* and *Sf* corresponding to f_α, $\alpha = a, b, b', x, y, z$

$f_x(t)$, $f_y(t)$, and $f_z(t)$ refer to decision-and-switching device failures which may not be time-dependent. If these failures are not time-dependent, the appropriate failure *pdf* is replaced by a constant probability of failure.

$$R_{10}(t) = \bar{F}_x(t)\Big\{\bar{F}_z(t)\bar{F}_a(t) + \int_0^t [\bar{F}_a(t_1) f_z(t_1)\bar{F}_y(t-t_1)\bar{F}_{b'}(t_1)\bar{F}_b(t-t_1)] dt_1 + \int_0^t [f_a(t_2)\bar{F}_z(t_2)\bar{F}_y(t_2) \bar{F}_{b'}(t_2)\bar{F}_z(t-t_2)\bar{F}_b(t-t_2)]dt_2\Big\}. \quad (9\text{-}31)$$

In Eq. 9-31, the first term inside the brackets represents the probability that A operates to t without premature switching. The second term represents the probability that a static failure occurs at time $t_1 < t$, but B operates to t. The last term represents the probability that A fails at time $t_2 < t$ and the decision-and-switching device switches to B (no dynamic failure), which operates to t. This equation represents a general case in that the following possibilities are included:

1. A and B can be different elements.

2. A static failure can occur if B is energized, resulting in no output or a false indication of system failure. If a static failure cannot occur when B is energized, then $\bar{F}_z(t) = 1$.

3. B can fail while on standby, and its failure *pdf* can be different from that when B is energized. If B is a "cold" rather than a "warm" or "hot" reserve, $f_{b'}(t) = 0, \bar{F}_b(t) = 1$.

Case 11. Same as Case 10, but identical elements (A and B) with constant failure rate $\lambda_A = \lambda_B = \lambda$ and cold standby. Eq. 9-31 becomes

$$R_{11}(t) = e^{-(\lambda+\lambda_z+\lambda_x)t}\left[1 + \lambda_z t + \frac{\lambda}{\lambda_y}(1 - e^{-\lambda_y t})\right]. \quad (9\text{-}32)$$

Case 12. Same as Case 11, but

$$\lambda_x = \lambda_y = \lambda_z = 0.$$

then, since

$$\lim_{\lambda_y \to 0} \left\{\frac{1 - e^{-\lambda_y t}}{\lambda_y}\right\} = t$$

$$R_{12}(t) = e^{-\lambda t}(1 + \lambda t).$$

which agrees with Eq. 9-29b, as it should. The effects of imperfect switching also are analyzed in Refs. 4,6,7.

9-7.2 OPTIMUM DESIGN: GENERAL MODEL

Case 13. There are n redundant paths with $(n - 1)$ in cold standby, and each path requires a switching device. In this model, the monitor represents the failure-detection and switching-control functions. These two functions can be considered as one for reliability purposes if it is assumed that the probability of compensating errors is negligible. All failure distributions have constant failure rates.

The following assumptions are made when

computing the reliability of these systems (Ref. 2):

1. Switching is in one direction only.
2. Standby (reserve) paths cannot fail if not energized.
3. Switching devices ought to respond only when directed to switch by the monitor; false switching operation (static failure) is detected by the monitor as a path failure, and switching is initiated.
4. Switching devices do not fail if not energized.
5. Monitor failure includes both dynamic and static failures. The monitor is a "series" element in the system.

Define terms as

λ = total (sum) failure rate of the series elements in a path

λ_s = failure rate of the switching device (includes contact failure)

λ_m = failure rate of the monitor

then, for n total paths,

$$R_{13}(t) = e^{-\lambda_m t}\left\{ e^{-(\lambda + \lambda_s)t} \times \sum_{i=0}^{n-1} \frac{[(\lambda+\lambda_s)t]^i}{i!} \right\}. \tag{9-33}$$

To illustrate the reliability gain provided by this model, assume that the system specification requires a high reliability for a mission of t hours. A nonredundant system therefore would have a reliability of

$$R_1(t) = e^{-\lambda t}, \tag{9-34a}$$

since no switching is required. The redundant system would have an s-reliability given by Eq. 9-33.

$$R_n(t) = R_{13}(t) \tag{9-34b}$$

Define $\tau \equiv \lambda t$ and substitute for t in Eq. 9-33, except in the λ_m term.

$$R_{13}(\tau) = \exp(-\lambda_m t) \exp\left[-(1 + \frac{\lambda_s}{\lambda})\tau\right] \sum_{i=0}^{n-1} \frac{[(1 + \frac{\lambda_s}{\lambda})\tau]^i}{i!} \tag{9-35a}$$

$$R_{13}(\tau) = \exp(-\lambda_m t)\, csqfc\,(2\tau(1 + \frac{\lambda_s}{\lambda}), 2n) \tag{9-35b}$$

where

$csqf(\chi^2, \nu)$ = chi square Cdf with ν degrees of freedom

$csqfc(\chi^2, \nu) = 1 - csqf(\chi^2, \nu)$ = complement of the $csqf$

(named in analogy with the error function)

The maximum reliability for a fixed τ that can be achieved, as $n \to \infty$, is $\exp(-\lambda_m t)$. Therefore, if $\lambda_m \geqslant \lambda$, (monitor is worse than an element) the optimum design has 1 element and no switching/monitoring.

Eq. 9-5 is a function of λ_s/λ, λ_m, and τ. The mission reliability of the redundant system can be calculated as a function of the parameters in Eq. 9-35. Table 9-3 and Figs. 9-8 and 9-9 show some of these calculations.

Table 9-3 shows how system reliability is influenced by the number of paths, if the switching device and the monitor have failure rates that are 1, 1/10, and 1/100 as great as the path failure rate.

In Fig. 9-8 the reliability of the redundant system is given as a function of the number of paths for various ratios of λ_m/λ when $R_{13}(t)$ = 0.80; arbitrarily, $\lambda_s/\lambda = 1/1000$. Fig. 9-9 is similar except that $\lambda_m/\lambda = 1/1000$, and λ_s/λ varies.

The following general conclusions can be drawn from this paragraph:

1. As the number of redundant paths increases, the mission reliability approaches the reliability of the monitor.
2. When the failure rates of the path, the switching devices, and the monitor are equal; standby redundancy with two paths results in a mission reliability considerably less than that of a single nonredundant path.
3. For systems where the switching-device and monitor failure rates are less than the path failure rate, the greatest increase in reliability occurs when one redundant path is added to a single path.

TABLE 9-3. EFFECT OF REDUNDANCY, CASE 13

1* 4.88 4.88 4.88 9.52 9.52 9.52 18.1 18.1 18.1
(1**) (13.9) (5.81) (18.4)
2

Cold standby; *n* elements total; imperfect switch and monitor; constant failure rates.

Failure probabilities listed in the body of the Table.

	τ = 0.05			τ = 0.10			τ = 0.20		
n γ	1	0.1	0.01	1	0.1	0.01	1	0.1	0.01
1*	4.88	4.88	4.88	9.52	9.52	9.52	18.1	18.1	18.1
(1**)	(13.9)	(5.81)	(4.97)	(25.9)	(11.3)	(9.66)	(39.3)	(21.3)	(18.4)
2	5.32	0.654	0.174	11.1	1.47	0.578	23.2	4.05	1.98
3	4.96	0.502	0.052	9.62	1.11	0.192	18.8	2.13	0.319
∞ (monitor only)	4.88	0.499	0.050	9.52	0.995	0.100	18.1	1.98	0.200

* No monitor or switch

** To show trends only, actually it is most impractical to have switch and monitor with only 1 unit

$\tau \equiv \lambda t$, λ = element failure rate

λ_s, λ_m = switch and monitor failure rates, respectively

$\gamma \equiv \lambda_s/\lambda = \lambda_s/\lambda$ for this Table

4. For a given path and switching-device failure rate, reliability improvement increases rapidly as the monitor failure rate decreases and the number of redundant paths increases. The same is true if the monitor failure rate is held constant and the switching-device failure rate decreases.

5. Important improvement in mission reliability through redundancy results from the use of switching devices and monitors that are much more reliable than the path being switched.

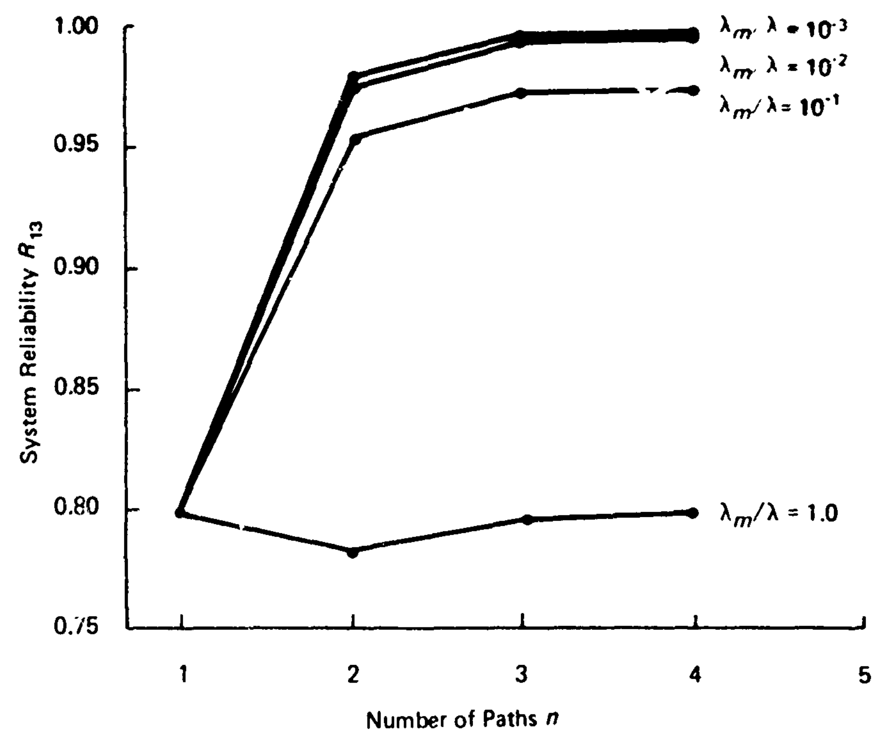
9-8 ACTIVE VERSUS STANDBY REDUNDANCY

For the basic models s-independent elements, perfect switching, and perfect reliability of de-energized elements), the reliability equations (along with intuition) indicate that standby redundancy is superior to active redundancy.

However, elements are not always s-independent; switching is rarely perfect; and certain parts and components can fail without being energized. Therefore, it is most unlikely that the simple standby system analyzed so far will be representative of practice.

9-9 MAINTENANCE CONSIDERATIONS

The previous analyses of redundancy were based on the assumption of unattended system operation. If maintenance is considered, even greater reliability improvements can be achieved. See also Refs. 4-7.



William H. Von Alven, Ed., *Reliability Engineering*, © 1964 by ARINC Research Corporation. Reprinted by permission of Prentice-Hall, Inc., Englewood Cliffs, N.J.

FIGURE 9-8. Mission Reliability for n Redundant Paths, Case 13, when $R_1(t) = 0.80$ ($\lambda t = 0.223$) $\lambda_s/\lambda = 0.001$ (Ref. 2).

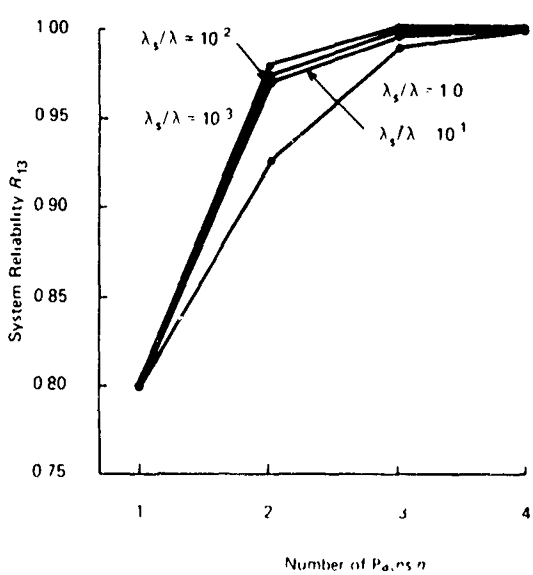


FIGURE 9-9 Mission Reliability for n Redundant Paths, Case 13, when $R_1(t) = 0.80$ ($\lambda t = 0.223$) $\lambda_m/\lambda = 0.001$ (Ref 2)

9-9.1 PERIODIC MAINTENANCE (Ref. 2)

Case 14. The following procedure will be assumed:

(1) Periodic maintenance is performed every T hours, starting at time 0. (2) Every element is checked, and any one which has failed is replaced by a like-new and statistically identical component.

Maintenance is perfect in that repaired/replaced units are good-as-new, no damage is done to the rest of the system, and the repaired system is good-as-new. In short, every T hours the system is restored to like-new.

Define

$$\tau = t - jT, \; 0 \leq \tau < T$$

where

τ = time since latest (number j) repair

j = 0, 1, 2, . . (repair number)

and

$R_{14} = R_T(t)$ the *s*-reliability function of a redundant system in which maintenance is performed every T hours

Let $R(t)$ be the *s*-reliability of the system during a period when no maintenance is done. Then for $j = 1$, $\tau = 0$,

$$R_T(T) = R(T). \tag{9-36}$$

If $j = 2$ and $\tau = 0$, the system has to operate the first T hours without failure of any redundant configuration. After replacement of all failed elements, another T hours of failure-free system operation are required; hence

$$R_T(2T) = [R(T)]^2. \tag{9-37}$$

If $0 < \tau < T$, then an additional τ hours of failure-free system operation are required, and

$$R_T(2T + \tau) = [R(T)]^2 R(\tau). \tag{9-38a}$$

In general,

$$R_{14} = R_T(jT + \tau) = [R(T)]^j R(\tau) \tag{9-38b}$$

where

$$j = 0, 1, 2, \ldots ; 0 \leq \tau < T.$$

$$MTF_{14} = \sum_{j=0}^{\infty} \int_{jT}^{(j+1)T} R_T(t)dt.$$

$$(t = jT + \tau)$$

$$= \left\{ \sum_{j=0}^{\infty} [R(T)]^j \right\} \int_0^T R(\tau)d\tau$$

$$= \frac{\int_0^T R(\tau)d\tau}{1 - R(T)}. \tag{9-39}$$

The effect of periodic maintenance can be illustrated in the example that follows. Two identical elements with constant failure rates of 1/(100 hr) are placed in an active-parallel configuration (1-out-of-2:G, hot standby). Compare the reliability functions and MTF's for $T = \infty$, 150, 100, 50, and 10 hr (Ref. 2). Use Eq. 9-4a for $R(t)$.

Reliability functions follow:

1. No maintenance: $(T = \infty)$

$$R_T(t) = R(t) = 2e^{-t/100} - e^{-t/50}. \tag{9-40}$$

2. With maintenance: $(T = jT + \tau; 0 \leq \tau < T)$

For $T = 150$ hr:

$$R_T(t) = [2e^{-1.5} - e^{-3}]^j [2e^{-\tau/100} - e^{-\tau/50}]. \tag{9-41}$$

For $T = 100$ hr:

$$R_T(t) = [2e^{-1} - e^{-2}]^j [2e^{-\tau/100} - e^{-\tau/50}]. \tag{9-42}$$

For $T = 50$ hr:

$$R_T(t) = [2e^{-0.5} - e^{-1}]^j [2e^{-\tau/100} - e^{-\tau/50}]. \tag{9-43}$$

For $T = 10$ hr:

$$R_T(t) = [2e^{-0.1} - e^{-0.2}]^j [2e^{-\tau/100} - e^{-\tau/50}]. \tag{9-44}$$

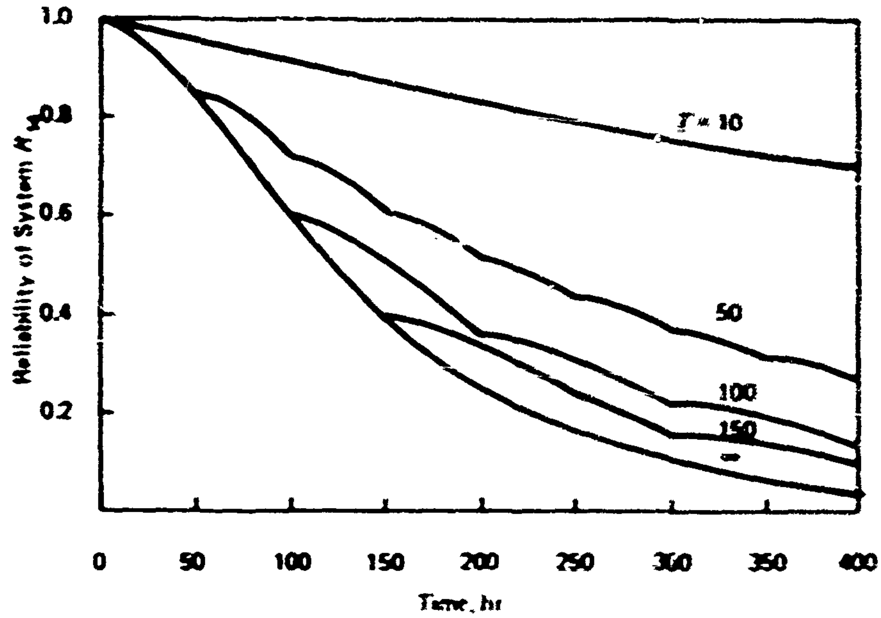
The reliability functions are plotted in Fig. 9-10. From 0 to 10 hr, all five functions are identical since $j = 0$ over this period for each system.

MTF is calculated using Eq. 9-39.

$$MTF_{14} = \frac{\int_0^T R(\tau)d\tau}{1 - R(T)}$$

$$= \frac{\int_0^T [2e^{-\tau/100} - e^{-\tau/50}]d\tau}{1 - 2e^{-T/100} + e^{-T/50}}$$

$$= \frac{150 + 50e^{-T/50} - 200e^{-T/100}}{1 - 2e^{-T/100} + e^{-T/50}} \tag{9-45}$$



William H. Von Alven, Ed., *Reliability Engineering*, © 1964 by ARINC Research Corporation. Reprinted by permission of Prentice-Hall, Inc., Englewood Cliffs, N.J.

FIGURE 9-10. s-Reliability Functions for Active-parallel Configuration Case 14 on Which Maintenance Restored to Like-new is Performed Every T hours (Ref. 2).

The *MTF*'s for the various *T*'s follow:

T, hr	MTF_{14}, hr
∞	150
150	179
100	208
50	304
10	1097

Considerable increase in *MTF* (and reliability) can be achieved by a perfect preventive maintenance policy.

9-9.2 CORRECTIVE MAINTENANCE

Reliability functions for some simple 2-unit redundant designs, for which repair of a failed unit is possible, were developed by Epstein and Hosford, and are summarized in this paragraph (Refs. 2 and 3).

At $t = 0$, all elements are good. Repair starts immediately upon failure of a unit and is perfect. The failure and repair rates are constant (independent of time). Three designs will be considered — Cases 15, 16, and 17.

Case 15. Two units in active redundancy. The constant failure rate of each unit is λ and the constant repair rate is μ.

$$R_{15}(t) = \frac{s_1 e^{s_2 t} - s_2 e^{s_1 t}}{s_1 - s_2}, s_1 \neq s_2 \qquad (9\text{-}46)$$

$$s_1 \equiv \tfrac{1}{2}\left[3(\lambda+\mu) - \sqrt{\lambda^2 + 6\lambda\mu + \mu^2}\right] \qquad (9\text{-}47a)$$

$$s_2 \equiv \tfrac{1}{2}\left[3(\lambda+\mu) + \sqrt{\lambda^2 + 6\lambda\mu + \mu^2}\right] \qquad (9\text{-}47b)$$

$$MTF_{15} = \frac{3\lambda + \mu}{2\lambda^2} \qquad (9\text{-}48)$$

Case 16. Two units in standby

redundancy. Constant unit failure rate is λ; constant unit repair rate is μ.

$$\mathcal{R}_{16}(t) = \frac{s_3 e^{s_4 t} - s_4 e^{s_3 t}}{s_3 - s_4} \tag{9-49}$$

$$s_3 \equiv \tfrac{1}{2}(2\lambda + \mu - \sqrt{\mu^2 + 4\lambda\mu}) \tag{9-50a}$$

$$s_4 \equiv \tfrac{1}{2}(2\lambda + \mu + \sqrt{\mu^2 + 4\lambda\mu}). \tag{9-50b}$$

$$MTF_{16} = \frac{2\lambda + \mu}{\lambda^2} \tag{9-51}$$

Case 17. Two units in standby redundancy. It takes exactly τ hours to repair a failed unit. Constant failure rate is λ.

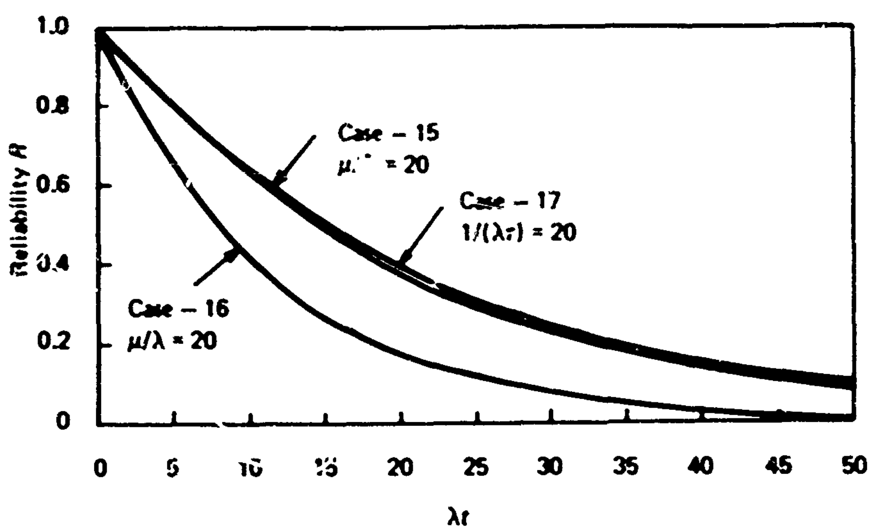
$$\mathcal{R}_{17}(t) = \sum_{i=0}^{[t/\tau]+1} \frac{e^{-\lambda t}(\lambda t)^i}{i!} [1 - (i-1)\tau/t]^i. \tag{9-52}$$

where

$[t/\tau]$ = greatest "integer $< t/\tau$".

i = exact number of failures

A plot of the reliability functions for these circuits is given in Fig. 9-11.



William H. Von Alven, Ed., *Reliability Engineering*, © 1964 by ARINC Research Corporation. Reprinted by permission of Prentice-Hall, Inc., Englewood Cliffs, N.J.

FIGURE 9-11. Comparison of s-Reliability Functions for Three Maintenance Situations Cases 15, 16, and 17 (ref. 2).

REFERENCES

1. *Handbook for Systems Application of Redundancy*, US Naval Applied Science Laboratory, 30 August 1966.
2. W. H. Alven, Ed., *Reliability Engineering*, Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1964.
3. B. Epstein and J. Hosford, "Reliability of Some Two-unit Redundant Systems", Proceedings of the Sixth National Symposium on Reliability and Quality Control, 6, 469-88 (January 1960).
4. S. Osaki, "On a 2-unit Standby-redundant System With Imperfect Switchover", IEEE Transactions on Reliability R-21, 20-24 Feb. 1972 (Corrections, *ibid*. p. 195, Aug. 1972).
5. S. Osaki, "Reliability Analysis of a 2-unit Standby-redundant System With Preventive Maintenance", IEEE Transactions on Reliability R-21 24-29, Feb. 1972 (Corrections, *ibid*. p. 195, Aug. 1972).
6. D. S. Taylor, "A Reliability and Comparative Analysis of Two Standby System Configurations", IEEE Transactions on Reliability R-22 13-19, April 1973.
7. D. K. Chow, "Reliability of Some Redundant Systems With Repair", IEEE Transactions on Reliability R-22, 223-228, Oct. 1973.

CHAPTER 10 RELIABILITY PREDICTION (GENERAL)

10-0 LIST OF SYMBOLS

e = par. 10-2.4, element *s*-reliability
A,B,S = elements (par. 10-4)
A_n,B_n = coefficients
C,O = event of contact Closure or contact Open
$D_a,D_b,$
S_a,S_b,S_c = events (par. 10-4.2)
F = $O \cup C$
f,g = *s*-unreliability of binary unit or gate (par. 10-3.3)
F_c = $1 - R_c$
F_{ir},F_i = failure probabilities (par. 10-3.3)
K = $2M$
m = number of chains (par. 10-4.2)
M = number of parts in system (par. 10-3.3)
MVT = Majority Vote Taker
p = element *s*-reliability; par. 10-2.4, proportion of open failures
P = number of units
$P(\dot{\psi})$ = probability of $\dot{\psi}$; $\dot{\psi}$ is any event (par. 10-4.2)
p_a = Pr{contact fails to close}
p_b = Pr{contact fails to open}
p_r = *s*-reliability of MVT
p_{ij} = *s*-reliability of element i,j
$Pr\{\ \}$ = probability of ...
q = $1 - p$
q_i = failure probabilities (par. 10-4.1)
q_{ij} = $1 - p_{ij}$
q_s,q_o = probabilities of failing short or open
q_v = failure probability of voter
q_α = $1 - p_\alpha$
Q_i = failure probability for event i (par. 10-4.1)
Q_i = *s*-unreliability of circuit i
R_c = *s*-reliability of chain (par. 10-4.2)
R_o = *s*-reliability of nonredundant device (par. 10-4.2)
R,R_s = network *s*-reliability
R_e,P_R = *s*-reliability of elements and/or systems (par. 10-3.3)
s- = denotes statistical definition
S_j = set for minimal-cut j
$\alpha,\alpha_i,$
$e,e_i,x^c,H,F,F_i,$
$d,d_i,P(i/j)$ = notation (par. 10-3.4)
λ,λ_α = failure rate; failure rate of element α
$\dot{\psi}$ = not ψ; ψ is any event (par. 10-4.2)
ϕ_α = failure *pdf* for α (par. 10-4.1)
$\bar{\Phi}_\alpha$ = *Cdf* for α (par. 10-4.1)
Φ_α = *Sf* for α (par. 10-4.1)
$2n + 1$ = number of identical circuits feeding an MVT
$\in$ = ... is a member of ...
$\cup$ = union

10-1 INTRODUCTION

Three main forms of redundancy (Fig. 10-1) will be discussed in this chapter, namely

1. Nondecision redundancy
2. Decision redundancy without switching
3. Decision redundancy with switching.

Nondecision redundant structures do not require external components to perform the functions of detecting, decision, and switching when an element or path in the structure fails. Examples are Moore-Shannon, single mode series-parallel, single mode binomial, and bimodal series-parallel.

Decision redundant structures without switching require an external element to detect and make a decision when an element or path in the structure fails, but do not need an external element to perform the switching function. Examples are majority logic, multiple line networks, gate connector, and coding.

Decision redundant structures with switching are those in which external elements are required to detect, make a decision, and switch to another element or path to

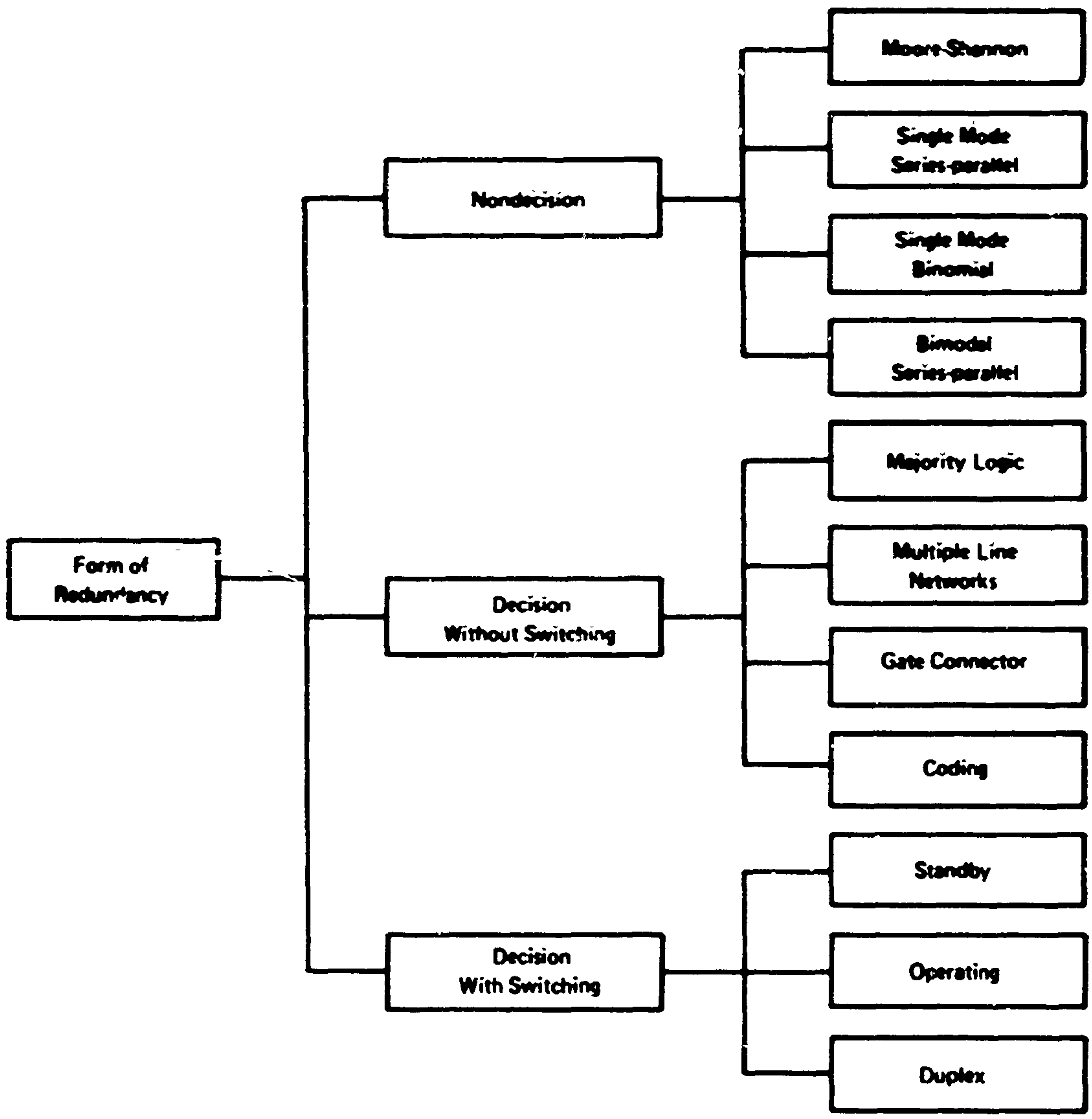


FIGURE 10-1. Redundancy Tree Structure[1]

replace a failed element or path. Examples are standby, operating, and duplex.

For each of the redundancy forms, several major characteristics will be covered to permit uniformity of comparison. Each of the forms will be defined and illustrated. Where feasible, reliability block diagrams and the mathematical model for each form will be given. All of the time-dependent models assume that all components are good at time zero. In general, the redundancy models will yield increasing failure rate (IFR) functions for similar elements. See also Chapters 8 and 9.

10-2 NONDECISION REDUNDANCY

10-2.1 MOORE-SHANNON REDUNDANCY

Moore and Shannon (Refs. 2 and 3) proposed connecting the contacts of relays, with their coils connected in parallel, in physical series-parallel circuits in such a manner that the resulting circuit acts exactly like a single relay.

An idealized switch is defined as a 4-terminal element where complete isolation exists between the control signal and switching path and which presents to the logic signal an infinite impedance ratio between desired and undesired transmission states. The analysis is not generally applicable, however, to 2- and 3-terminal devices such as transistors and tunnel diodes. Furthermore, the Moore-Shannon theory assumes only catastrophic failures; hence, drift failures and aging effects are excluded. Time is not considered at all.

Three assumptions are made in developing the mathematical model.

1. The failure of any element is *s*-independent of the failure of any other element.
2. Only intermittent, complete failures are considered.
3. The probability of failure of an element is defined for each operation and is the same for every element; time never appears explicitly.

Fig. 10-2 illustrates three elementary, redundant, relay-contact networks considered by Moore and Shannon. If p is the probability that a single contact will operate properly, then the probability that two contacts will operate properly is p^2. The probability that neither contact operates properly is $1 - p^2$. Consequently, if two relay-contacts, physically in series, are used to connect a path, and both are operated simultaneously, the redundancy improves the reliability for opening the path, but reduces the reliability for closing the path. If four relay-contacts are connected in a physical series-parallel arrangement, as shown in Fig. 10-2(A), the probability of opening the path is $(1 - p^2)^2$, and the probability of closing the path is

$$R = 1 - (1 - p^2)^2 = 2p^2 - p^4 \quad (10\text{-}1)$$

The network illustrated in Fig. 10-2(B) is the dual of the one shown in Fig. 10-2(A); the probability of closing the path is

$$R = [1 - (1 - p)^2]^2 = 4p^2 - 4p^3 + p^4 \quad (10\text{-}2)$$

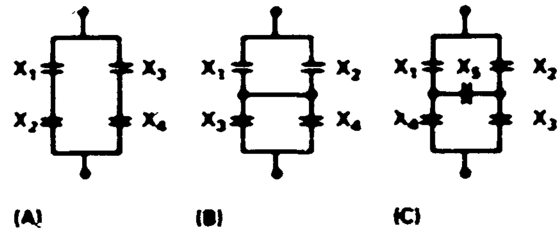


FIGURE 10-2. Relay Networks Illustrating Moore-Shannon Redundancy[1]

The network illustrated in Fig. 10-2(C) is slightly more complex because of the additional contact X_5; the probability of closing the path is

$$R = 2p^2 + 2p^3 - 5p^4 + 2p^5 \quad (10\text{-}3)$$

These results may be generalized to include any complex redundant network between two points. If m contacts are used in a switching array between two points and if n of them constitute a subset of closed contacts, the probability of closing the path is

$$R = \sum_{n=0}^{m} A_n p^n (1 - p)^{m-n} \quad (10\text{-}4)$$

where A_n is the number of combinations of the subsets which correspond to a closed path. Similarly, the probability of opening the path is

$$1 - R = \sum_{n=0}^{m} B_n (1 - p)^n p^{m-n} \quad (10\text{-}5)$$

where B_n is the number of subsets of n contacts such that if all contacts in a subset are open and all others closed, the path is open.

By using this approach, arbitrarily reliable relay networks can be built from arbitrarily poor (low reliability) relays, provided enough of the poor ones are used.

Time can be introduced explicitly if the following are assumed:

1. The failure of any element is *s*-independent of the failure of any other element.
2. All failures are permanent; i.e., when an element fails, it remains in the failed condition.

3. The reliability of the elements is known (as a function of time) and is the same for every element. Two failure distributions are defined:

$q_a(t)$ = probability that a contact will fail to close during the interval 0 to t.

$p_b(t)$ = probability that a contact will fail to open during the interval 0 to t.

It follows that:

1. The probability that a contact will be closed whenever it should be closed during the interval 0 to t is

$$p_a(t) = 1 - q_a(t) \qquad (10\text{-}6)$$

This is the reliability of being closed, defined for this interval.

2. The probability that a contact will be open whenever it should be open during the interval 0 to t is

$$q_b(t) = 1 - p_b(t) \qquad (10\text{-}7)$$

This is the reliability of being open, defined for this interval.

The total probability of failure of the circuit in the interval 0 to t is the sum of the disjoint probabilities of failure to close and failure to open. The probability that the circuit will fail to close at some time during the interval 0 to t is

$$Pr\{\bar{C}\} = \sum_{n=0}^{4} B_n (1-p_a)^n p_a^{4-n}$$

$$= \sum_{n=0}^{4} B_n q_a^n (1-q_a)^{4-n} \qquad (10\text{-}8)$$

where

C = event of closure ($\bar{C}$ = event of not-closure).

Since

$$B_0 = B_1 = 0, B_2 = 2, B_3 = 4, B_4 = 1,$$

$$Pr\{\bar{C}\} = 2q_a^2(1-q_a)^2 + 4q_a^3(1-q_a) + q_a^4$$
$$= 2q_a^2 - q_a^4. \qquad (10\text{-}9)$$

The probability of the circuit failing to open at some time during the interval 0 to t is

$$P(\bar{O}) = \sum_{n=0}^{4} A_n p_b^n (1-p_b)^{4-n}$$
$$= 4p_b^2(1-p_b)^2 + 4p_b^3(1-p_b) + p_b^4$$
$$= 4p_b^2 - 4p_b^3 + p_b^4 \qquad (10\text{-}10)$$

where

O = event of opening ($\bar{O}$ = event of not-opening).

Then, the total probability of circuit failure in the interval 0 to t is

$$Pr\{F\} = Pr\{\bar{O} \cup \bar{C}\} = Pr\{\bar{O}\} + Pr\{\bar{C}\}$$
$$= 2q_a^2 - q_a^4 + 4p_b^2 - 4p_b^3 + p_b^4 \qquad (10\text{-}11)$$

where

$$F = O \cup C .$$

The straight-line in each figure is the nonredundant case.

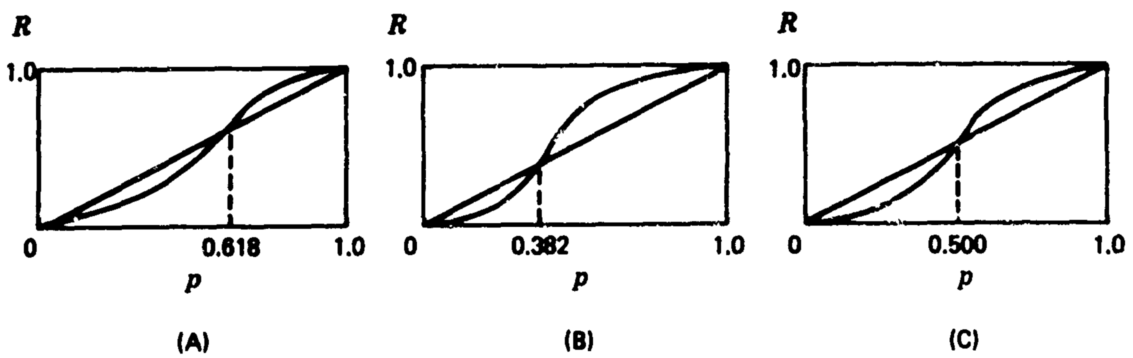


FIGURE 10-3. s-Reliability Functions for Redundant Relay Networks[1]

The reliability functions for the circuits in Fig. 10-2 are given in Fig. 10-3. The figure describes the reliability of the circuits as a function of the reliability of the individual relay.

For the network illustrated in Fig. 10-2(A), the reliability function (Fig. 10-3(A)) lies above the diagonal $R = p$ for values of p greater than 0.618. Therefore, the redundant circuit represents an improvement over a single contact if the reliability of each contact closing is better than 0.618.

For the second network (Fig. 10-2(B)), R crosses the diagonal at 0.382, as shown in Fig. 10-3(B). The bridge network illustrated in Fig. 10-2(C) has a symmetrical probability curve which crosses the diagonal at 0.5 (Fig. 10-3(C)).

As shown in the discussion of reliability gain, the reliability of a Moore-Shannon type circuit can be degraded below some specified value, depending on the topography of the circuit. The use of these circuits in situations where the performance characteristics of the parts must be considered also may degrade the reliability of the redundant structure as compared with the single part.

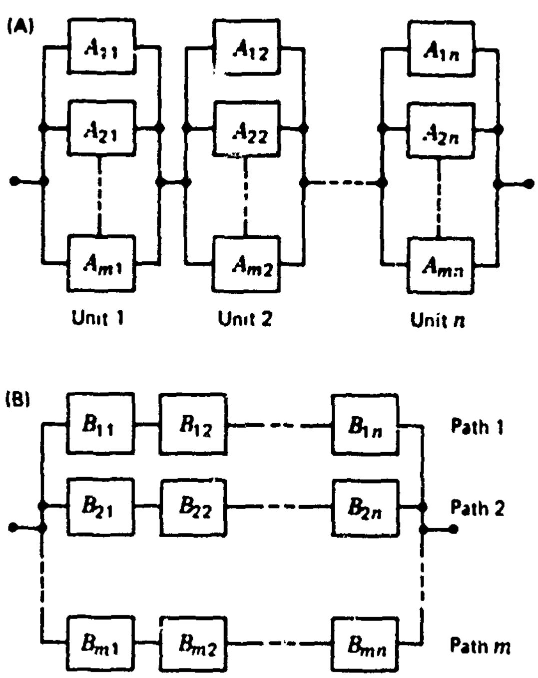


FIGURE 10-4. Single Mode Series-parallel Redundant Structures[1]

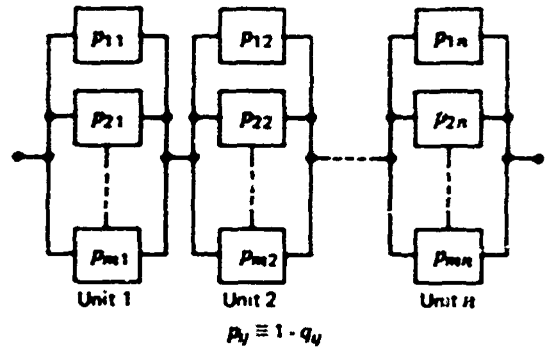


FIGURE 10-5. Reliability Block Diagram for a Single Mode Series-parallel Redundant Structure[1]

10-2.2 SINGLE MODE SERIES-PARALLEL REDUNDANCY

The single mode series-parallel structure is a group of n units in series; there are m parallel elements in each unit in which only one mode of failure can occur (Ref. 1).

In the circuits of Fig. 10-4, A_{ij} elements are subject only to open-type failures while B_{ij} elements are subject only to short-type failures. Both of these circuits have the reliability block diagram shown in Fig. 10-5. Each of the elements is s-independent of each other, with failure probability q_{ij} for the element i in the unit j, so that

$$R = \sum_{j=1}^{n} (1 - q_{ij} q_{2j} \cdots q_{mj}) \qquad (10\text{-}12)$$

10-2.3 SINGLE MODE BINOMIAL REDUNDANCY (k-out-of-n)

The reliability of a k-out-of-n:G system is, from Eq. 8-1a,

$$R = \sum_{k}^{n} \binom{n}{i} p^i (1 - p)^{n-i} \qquad (10\text{-}13)$$

where

p = reliability of a single unit (see par. 8-2).

10-2.4 BIMODAL SERIES-PARALLEL REDUNDANCY

A bimodal series-parallel redundant

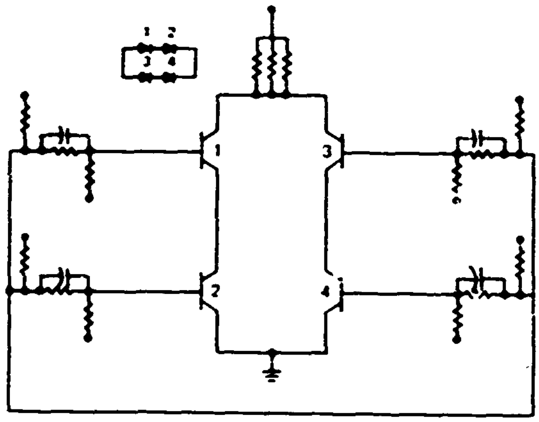


FIGURE 10-6. Schematic Diagram of a Diode and Transistor Quad Bridge Network Illustrating Bimodal Series-parallel Redundancy[1]

structure is one in which elements are connected in a series-parallel configuration, and which is susceptible to two modes of failure, such as opens and shorts (Ref. 1). The reliabilities are all conditional on the set of events which are required for the elements to be conditionally *s*-independent. For example, if four transistors are on the same chip, they will not be *s*-independent for many failure modes. Included in this form are what are commonly known as Quad configurations. A typical circuit is shown in Fig. 10-6 and the *reliability block diagram* in Fig. 10-7. The elements are *s*-independent of each other. They can fail either open or short.

The conditional reliability of the transistor Quad, where a is the probability of nonfailure of a transistor and p is the proportion of transistor failures due to opens, is (Ref. 4)

$$R = \underbrace{a^4}_{\mathrm{I}} + \underbrace{4a^3(1-a)}_{\mathrm{II}} + \underbrace{4a^2(1-a)^2[1+p(1-2p)]}_{\mathrm{III}} + \underbrace{8a(1-a)^3p(1-p)^2}_{\mathrm{IV}} \tag{10-14}$$

where

I = probability that all four transistors survive t hours of operation without failure.

II = probability that three of the four transistors survive t hours of operation without failure while the other transistor fails.

III = probability that two of the four transistors survive while the other two transistors fail prior to time t in a favorable manner; i.e., failure of the two transistors does not cause configuration failure. This probability represents the sum of:

1. $4a^2(1-a)^2(1-p)^2$, the probability that two transistors short prior to time t (however, both failures are not in the same leg of the Quad); and
2. $12a^2p(1-p)(1-a)^2$, the probability that two transistors fail prior to time t where one is a short and the other an open.

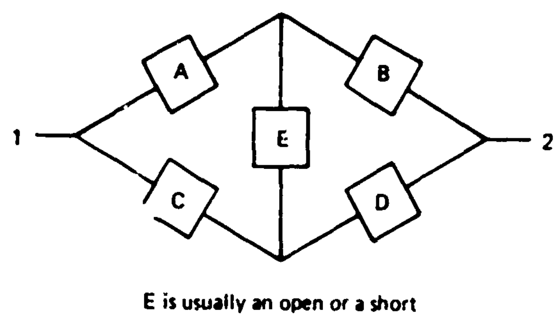


FIGURE 10-7. Reliability Block Diagram of a Diode and Transistor Quad Bridge Network[1]

IV = probability that three transistors fail prior to time t: two of the transistors short and the other opens (however, the two shorts are not in the same leg of the Quad).

In general, for a network of identical elements in m paths, where success is neither an open nor short network,

$$R = [1 - q_s^n]^m - [1 - (1-q_o)^n]^m \tag{10-15}$$

where

q_s = probability of failing short, for an element

q_o = probability of failing open, for an element.

The reliability equation for the bridge network is a function of whether or not the elements are polarized. Polarized elements allow current to flow in one direction only.

For identical nonpolarized elements which allow current to flow in either direction (Ref. 2)

$$R = (1 - q_o - q_s)[q_o^4 - 2q_o^2 + (1 - q_s^2)^2]$$
$$+ q_o\{(1 - q_s) - [1 - (1 - q_o)^2]^2\}$$
$$+ q_s\{(1 - q_o)^2 - [1 - (1 - q_s)^2]^2\}. \quad (10\text{-}16)$$

For identical polarized elements which allow current to flow in one direction only,

$$R = (1 - q_o - q_s)[2q_o^3 - 3q_o^2 + (1 - q_s^2)^2]$$
$$+ q_o\{(1 - q_s)^2 - [1 - (1 - q_o)^2]^2\}$$
$$+ q_s\{(1 - q_o)^2 - [1 - (1 - q_s)^2]^2\}.$$

(10-17)

Although conditional reliability increases as a result of using a Quad, several important design factors must be considered, namely:

1. Using transistors in a Quad configuration subjects them to more vigorous and demanding parameter requirements.
2. The redundant configuration can drive but one fourth the load of the nonredundant circuit.
3. The Quadding approach is inherently a slower one, increasing signal propagation time by at least 2:1.
4. The redundant design will dissipate up to, and possibly more than, four times the power of a single transistor, if maximum speed is desired.
5. The Quadding layout usually will demand a greater supply voltage and, therefore, cause the minimum power ratio to be about 2:1, redundant to nonredundant.
6. Failure of any unit of a Quad can increase semiconductor heat dissipation per unit up to four times. A direct consequence of this is requiring the lowering of ambient operating temperature to keep semiconductor junction temperatures below the danger point.

10-2.5 SUMMARY TABLE

Table 10-1 summarizes the important characteristics of component redundancy for different combinations of short to open failure when the elements are susceptible to both. The failure conditions, reliability equation, approximate probability of failure, and impedance variation due to redundancy are presented.

10-3 DECISION-WITHOUT-SWITCHING REDUNDANCY

10-3.1 MAJORITY LOGIC REDUNDANCY

Majority logic is a form of decision redundancy for which the correct output is assumed to be the one found in a majority of the channels. The concept of majority logic was first proposed by von Neumann and has since been enlarged upon by many authors. Von Neumann's original concept required extremely high redundancy to achieve high reliabilities, but later techniques give high reliability with a rather low degree of redundancy. Typical structures are shown in Figs. 10-8 and 10-9.

The probability of success for the majority group is, from Eq. 8-17,

$$p_n = p_v \sum_{i=n}^{2n+1} \binom{2n+1}{i} p^i q^{2n+1-i} \quad (10\text{-}18)$$

where

p = probability that a circuit is operating properly
q = $(1 - p)$ = probability that the circuit has failed
p_v = probability of success of Majority Vote Taker MVT
$2n + 1$ = number of units.

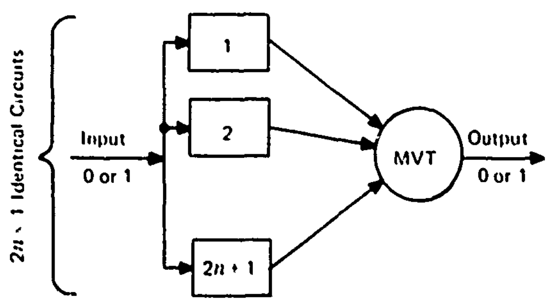
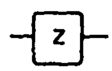
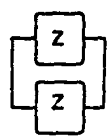
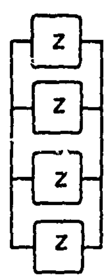
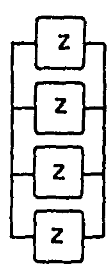
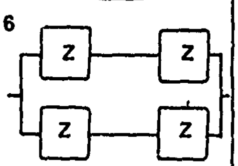
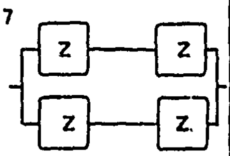


FIGURE 10-8. Basic Majority Vote Redundant Circuit[1]

TABLE 10-1. COMPONENT REDUNDANCY[1]

Component Configuration	Failure Conditions	Reliability ($\rho_o \equiv P_o/P_f$, $\rho_s \equiv P_s/P_f$)*	Approximate Probability of Failure $P_f << 1$	Maximum Impedance Variation Due to Redundancy
1 	Short or open	R	P_s+P_o**	0%
2 	Single open or two shorts. Used where ρ_o << 0.5	R^2+2RP_s	sP_o	−50%
3 	Single short or two opens. Used where ρ_o >> 0.5	R^2+2RP_o	$2P_s$	+100%
4 	Single short or two opens. Used where ρ_o >> 0.5	$R^4+4R^3P_o$	$6P_o{}^2+4P_s$	+33 1/3%
5 	Single short or three opens. Used where ρ_o >> 0.5	$R^4+4R^3P_o+6R^2P_o$	$4P_o{}^2+4P_s$	+100%
6 	Two shorts in same leg or one open in each leg. Used where $\rho_o < 0.5$	$R^4+4R^3P_o+4R^3P_s$ $+12R^2P_oP_s+2R^2P_o$ $+4R^2P_s{}^2+4RP_o{}^2P_s$ $+8RP_s{}^2P_o$	$4P_o{}^2+2P_s^2$	+100%
7 	Three opens or opens in both elements connected to either input or output nodes. Two shorts in same leg or shorts at alternate ends of two legs. Used where $\rho_o > 0.5$	$R^4+4R^3P_o+4R^3P_o$ $+4R^2P_o+2R^2P_s{}^2$ $+12R^2P_oP_s+8RP_o{}^2P_s$ $4RP_s{}^2P_o$	$4P_s{}^2+2P_o{}^2$	+100%

*$P_o(P_s)$ is the conditional probability of the component opening (shorting) given that the component fails.
**$R + P_o + P_s = 1$ for single element.

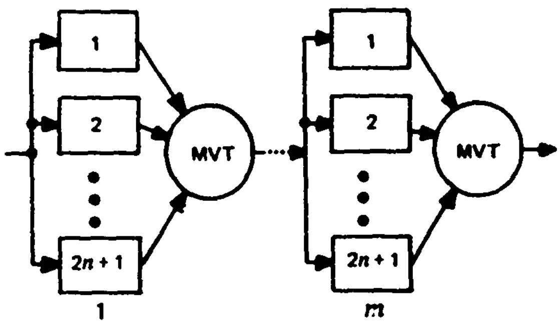


FIGURE 10-9. Majority Vote Redundant Circuit With Multiple Majority Vote Taker[1]

The lower degrees of redundancy give the approximate failure probabilities listed in Table 10-2.

TABLE 10-2

APPROXIMATE FAILURE PROBABILITIES FOR MAJORITY LOGIC REDUNDANCY[1]

$2n + 1$ (Degree of Redundancy)	Approximate Failure Probability of Circuit
3	$q_v + 3q^2 - 2q^3$
5	$q_v + 10q^3 - 15q^4 + 6q^5$
7	$q_v + 35q^4 - 84q^5 + \cdots$
9	$q_v + 126q^5 - 420q^6 + \cdots$

q_v = failure probability of MVT
q = failure probability of logic element

Using higher degrees of redundancy will not substantially improve overall reliability, since the majority vote taker (MVT) reliability soon becomes the limiting factor. Even for threefold redundancy ($2n + 1 = 3$), q_v is the major cause of failure if q is reasonably small.

When majority logic is applied to each block, and every MVT is triplicated except the last one, the resultant failure probability for the general case, using a $(2n + 1)$—fold majority logic and m blocks, is as follows:

$$1 - R = (1 - q_{fv}) \times \left[1 - \sum_{i=n+1}^{2n+1} \binom{2n+1}{i} q_b^i p_b^{2n+1-i}\right] \times \left[1 - \sum_{i=n+1}^{2n+1} \binom{2n+1}{i} (q_b + q_v - q_b q_v)^i (1 - q_b - q_v + q_b q_v)^{2n+1-i}\right]^{m-1} \tag{10-19}$$

where the notation is shown on Fig. 10-10.

Assuming that all the failure probabilities are reasonably small, this becomes

$$1 - R \approx q_{fv} + \binom{2n+1}{n+1} (q/m)^{n+1} + (m-1)\binom{2n+1}{n+1} (q_v + q/m)^{n+1} \tag{10-20}$$

where q is the probability of failure for the nonredundant system.

For threefold majority logic ($n = 1$), the probability is

$$1 - R \approx q_{fv} + 3(q/m)^2 + 3(m-1)(q_v + q/m)^2 \tag{10-21}$$

The MVT is considered ideal if $\lambda_v m_o t \ll 1$ where λ_v is the failure rate of the MVT and t is the mission time. If the MVT is ideal, rather than infallible, and if the number of MVT failures in a given length of time obeys the Poisson distribution, then

$$p_v(t) = e^{-\lambda_v t} \tag{10-22}$$

where p_v is the probability that a vote taker is working properly.

It is assumed that the output of a nonfunctioning vote taker is the complement of the correct output.

If the failure rate of the MVT's is too large to be neglected, redundant MVT's can be used. In this case, the failure rate of an individual circuit can be considered to include the circuit and the vote taker feeding that circuit. The overall system then becomes equivalent to a system using nonredundant ideal MVT's. If the probability of survival for an individual circuit is

$$p = p_v p_o = (e^{-\lambda_v t})(e^{-\lambda_o t}) = e^{-(\lambda_v + \lambda_o)t}; \tag{10-23}$$

then

$$R = \left[\sum_{i=0}^{n} \binom{2n+i}{i} q^i p^{2n+1-i} \right]^m, \tag{10-24}$$

which is equivalent to the probability of success for m majority groups.

It can be shown that the maximum reliability is achieved with nonideal vote takers if (Ref. 5)

$$\lambda_o / \lambda_v = 1/(2n+1) \tag{10-25}$$

where

λ_o = failure rate of the circuit
λ_v = failure rate of the vote taker

$$(2n+1) = \text{number of identical circuits.} \tag{10-26}$$

It is usually necessary to carry system output on a single line, in which case the redundancy scheme proposed by Moore and Shannon could be used to improve the reliability of system output, thus eliminating the final vote taker from the analytic expression. This form of redundancy is usually associated with binary inputs and outputs. It can be applied in situations that call for either intermittent or continuous operation in time. Some conclusions which can be drawn from all this are:

1. Assuming ideal vote takers, a digital system will be most reliable if majority logic is applied at as low a level as possible, i.e., when the system is divided into as many digital subsystems, each followed by a majority vote taker, as possible.

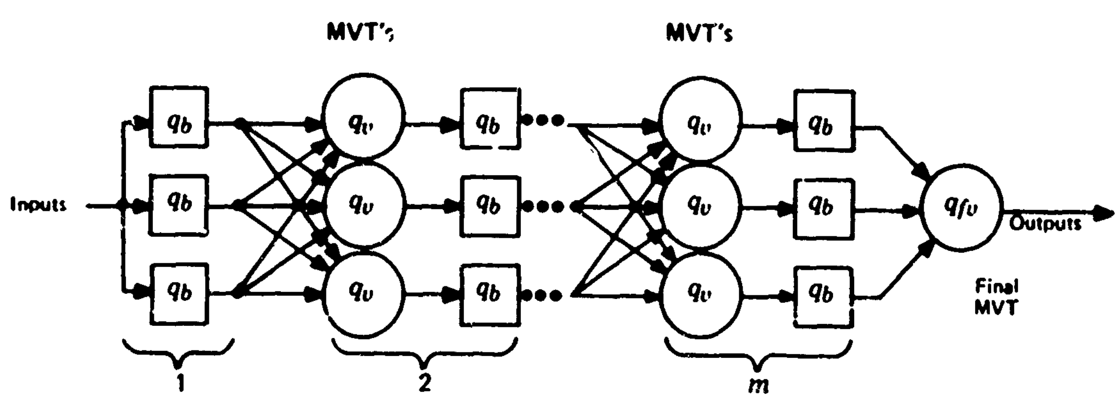
2. On the other hand, it is clear that the MTF for the system will always be less than the MTF for the individual circuit. In the limit as $n \to \infty$, the system MTF can be 0.69 times the MTF for the individual circuit.

3. The use of redundancy and majority logic gives the greatest improvement in reliability in the case of large systems, i.e., in systems for which it is possible to achieve large values of m.

4. The full reliability improvement can be realized only if all circuits are working properly at time $t = 0$. This causes a checkout and repair problem.

5. Unless the nonredundant fault probability q is small, very high degrees of redundancy are required to reduce system failure probability. For $q > 0.5$, any degree of majority logic redundancy will actually degrade reliability, although $q > 0.5$ is not very realistic for anything but deep-space probes.

6. If nonredundant MVT's of limited reliability are used anywhere in a redundant



MVT = Majority Vote Taker
Failure probability is shown for each element.

FIGURE 10-10. Reliability Block Diagram for Circuit With Threefold Majority Logic[1] system, they will constitute for some period of time the most likely source of system failure.

10-3.2 MULTIPLE LINE REDUNDANCY

Multiple line redundancy has been studied extensively by Westinghouse and is one of the most efficient types of circuit redundancy (Refs. 6 and 7). It is applied by replacing the single circuit of a nonredundant network by nonidentical circuits operating in parallel, where m is called the order of the redundancy.

The reliability improvement achieved by these redundant circuits depends on the ability of the network to experience circuit failures without degradation of the network operation. The use of restorers within the network provides this characteristic. The restorer consists of m restoring circuits which, when operating correctly, can derive the correct output from k of m correct inputs. A typical circuit is shown in Fig. 10-11.

A reliability model can be developed based on the following assumptions:

1. The circuits in the network are s-independent.
2. Only an approximation to the exact reliability will be given, and it is based on techniques described in Refs. 8 and 9. The approximation is good if the reliabilities of the circuits in the network are close enough to one.
3. The approximation is based on the concepts of minimal cuts, discussed previously, and coherent systems. A system is coherent if it meets the following four conditions:

a. If a group of circuits in the system is failed, causing the system to fail, the occurrence of any additional failure or failures will not return the system to a successful condition.

b. If a group of circuits in the system is successful and the system is successful, the system will not fail if some of the failed components are returned to the successful condition.

c. When all the circuits in the system are successful, the system is successful.

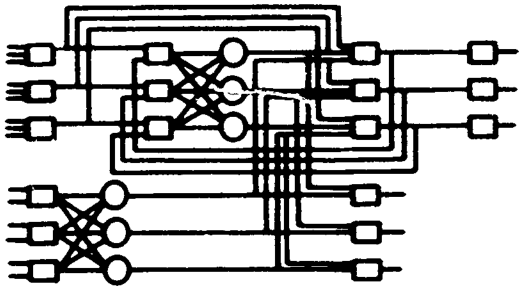


FIGURE 10-11. Order-three Multiple Line Redundant Network[1]

d. When all the circuits in the system fail, the system fails. The system shown in Fig. 10-12 is an example of a coherent system.

The lower bound to system reliability is the probability that none of the system minimal cuts fail: for the sample in Fig. 10-12, it is

$$R_s \approx (1 - Q_1 Q_2)(1 - Q_4 Q_5)(1 - Q_2 Q_3 Q_4) \tag{10-27}$$

where

R_s = system reliability

Q_i = the probability of failure for circuit i.

This equation is approximate because the failures of minimal cuts are assumed to be s-independent which is generally not true, since one component may appear in several minimal cuts.

If minimal cut j is denoted by set S_j then $\prod_{i \in S_j} Q_i$ is the probability of failure for minimal

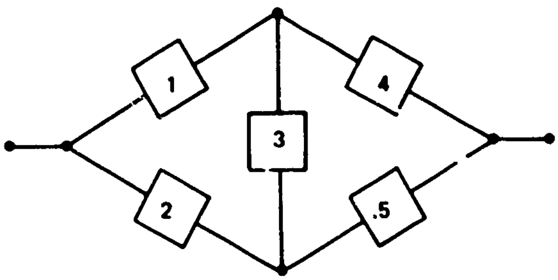


FIGURE 10-12. A Coherent System[1]

cut j. The lower bound of the system reliability is

$$R_S = \prod_{j=1}^{c} \{1 - \prod_{i \in S_j} Q_i\} \tag{10-28}$$

where

c = number of minimal cut sets

$\in$ = "is a member of".

Thus, the determination of the lower bound on reliability requires that the minimal cuts of the network be identified. In a multiple line network with restorers, a cut is any group of circuits whose failure causes the outputs of at least one restored function to have $(m - k + 1)$ or more failed lines. This would constitute a network failure.

The minimal cuts of a multiple line redundant network have three characteristics that are sufficient to establish their identity:

1. All the members of the minimal cut are circuits in a restored function or restorers that are the input sources of that restored function.
2. The failure of each member of the minimal cut will cause one output line of the restored function to be in error, and each member will be in a different position.
3. The failure of a minimal cut will cause exactly $(m - k + 1)$-output lines of the restored function to be in error; hence, a minimal cut will have $(m - k + 1)$-members.

If all the sets of circuits that fulfill these characteristics are listed for each of the restored functions in the network, all of the minimal cuts of the network and the lower bound for the network reliability can be found.

The improvement in system reliability is comparable to the improvement in the reliability of a circuit when a particular element is made redundant. The improvement will not be of the same magnitude, because of the addition of restorers in the multiple line network.

Multiple line redundancy results in improved reliability of the system unless the individual circuit reliabilities are very low. Low circuit reliabilities cause the restorers to choose the wrong value if k of the m circuits have failed.

The lower limit approximation given for the multiple line network is not good if the circuit reliabilities are not close enough to one. If the order of the redundancy exceeds three, the determination of the input sources becomes quite difficult. Boolean techniques can be used for determining the input sources of a function.

10-3.3 GATE-CONNECTOR REDUNDANCY

Gate-connector, or gate-connected, redundancy is a combination of several binary circuits connected in parallel along with a circuit of switch-like gates which serves as the connecting majority organ (Refs. 1 and 10). The gates contain no components whose failure would cause the redundant circuit to fail, and any component failures in the gate connector act as though the binary circuits were at fault.

Gate-connector redundancy applied to four units in parallel and a 4-element network for the gate connector is shown in Fig. 10-13.

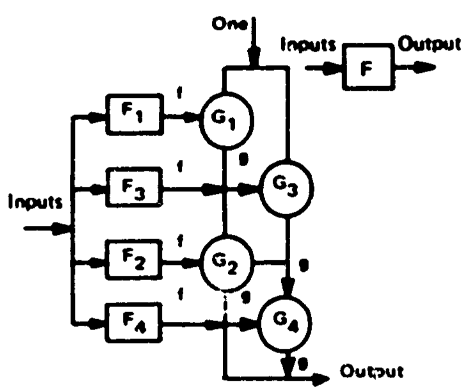


FIGURE 10-13. Circuit Illustrating Gate-connector Redundancy[1]

For this circuit, the following assumptions and nomenclature are used.

1. f = probability of failure for each binary unit
2. g = probability of failure for each gate
3. Failures are s-independent
4. If the circuit is closed when it should be open, it is a Type I failure
5. If the circuit is open when it should be closed it is a Type II failure.

The output with Type I failures should be zero, but may be, mistakenly, one. The output of G_1 will be one if unit 1 fails, G_1 fails, or both fail. The probability of this event taking place is

$$F_{1a} = 1 - (1 - f_1)(1 - g_1) \quad (10\text{-}29)$$

where the subscript 1 designates Type I failures. When a one is received from G_1, a one will be transmitted to the output if unit 2 fails, G_2 fails, or both fail. Therefore, the probability of getting a one in the left channel is

$$F_{1b} = 1 - (1 - f_1)(1 - g_1)^2 . \quad (10\text{-}30)$$

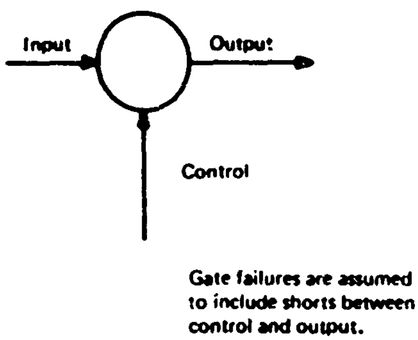


FIGURE 10-14. Gate Unit[1]

Now we must investigate what happens when a zero is at the output of G_1 and both unit 2 and G_2 fail. Whether a failure occurs or not depends on how the gate circuit fails. Fig. 10-14 shows a gate unit with leads labeled control, input, and output. In the gate-connector circuit, the control is connected to the output of the binary unit, and the input and output connections are used in the connector circuit. The gate input is connected electrically to the output only if a one is present on the control. Now, if it is assumed that only a one can be obtained from the output when a one is present in the input, the circuit will not fail when G_2 has a zero on the input and unit 2 and G_2 fail. However, if it is assumed that the gate unit fails in a shorted condition in such a way that a one is obtained at the output when a zero is on the input and a one is on the control element, the circuit will fail if unit 2 and G_2 fail. This latter case will be assumed and, when this is taken into account, the probability of failure for one channel becomes

$$F_{1c} = [1 - (1 - f_1)(1 - g_1)]^2 + (1 - f_1)(1 - g_1)f_1 g_1 , \quad (10\text{-}31)$$

and the probability of failure of the circuit of Fig. 10-13 due to Type I failures is

$$F_1 = 1 - \{1 - [1 - (1 - f_1)(1 - g_1)]^2 - (1 - f_1)(1 - g_1)f_1 g_1\}^2 , \quad (10\text{-}32)$$

The failure probability for Type II failures will be simpler. When the output should be one and the failures make it zero, the extra term does not appear and the equation for Type II failures is simply

$$F_2 = \{1 - [(1 - f_2)(1 - g_2)]^2\}^2 . \quad (10\text{-}33)$$

If it is assumed that $f_1 = f_2$ and $g_1 = g_2$, it cannot be shown that one of the expressions is greater than the other for all values of f and g; but in the region of values of f and g where reliability improvement is obtained, $F_2 > F_1$. Let F be the upper bound of failure probability for the redundant circuit, and let f and g be the greater of the Type I or Type II failure probabilities. Then, in the region where reliability improvement is obtained,

$$F = \{1 - [(1 - f)(1 - g)]^2\}^2 . \quad (10\text{-}34)$$

If a nonredundant system with reliability R_o is divided into M s-independent parts of equal reliability, part M of the system would have a reliability equal to the Mth root of R_o. The reliability of part M of the nonredundant portion of the system corresponds to $(1 - f)$ in the equations. Thus,

$$R_o^{\frac{1}{M}} = (1 - p) . \quad (10\text{-}35)$$

The reliability of the redundant system is the reliability of one redundant unit raised to the power M. This gives the following equation for reliability:

$$P_R = \{1 - [1 - (1 - g)^2 R_o^{\frac{2}{M}}]^2\}^M . \quad (10\text{-}36)$$

There is an optimum value of M. In the region of g and R_o, where reliability improvement is obtained, the maximum value of M should be used. In practice, it is difficult to use single active element circuits as s-independent circuits. A reasonable s-independent block in a system would consist of two active elements. Such circuits would include flip-flops, clock generators, two-way logic circuits, and so forth.

If a machine consists of K active element circuits, $M = K/2$. A gate is assumed to be equivalent to one active element circuit. When this is substituted into the preceding equation, the result is

$$P_R = \left[2R_o^{\frac{6}{K}} - R_o^{\frac{12}{K}}\right]^K \qquad (10\text{-}37)$$

Since the gate-connector redundancy can be applied at a low component organization level, it is suitable for use in conjunction with the Moore-Shannon redundancy.

Critical components that require better than ±50 percent component-value tolerances can be made redundant by the gate-connector redundancy in a machine that is made redundant by Moore-Shannon redundancy.

A factor which should not be overlooked when designing with gate-connector redundancy is that the switch-like gate connector must contain no components whose failure would cause the redundant circuit to fail.

10-3.4 CODING REDUNDANCY

Coding redundancy is a method of incorporating passive self-repair in order to improve reliability (Refs. 1 and 11). It is used for processing unreliable information in logical networks such as computers. Binary signals that are to be used as inputs can be checked using coding redundancy.

Under certain restrictions, the type of coding redundancy proposed by Tooley (Ref. 11) avoids the usual complexity requirements for redundancy.

A model for an AND gate is shown in Fig. 10-15 in two equivalent forms with noise, denoted by $P(0|1)$ and $P(1|0)$, added. The restrictions assumed in the model by Tooley are:

1. The errors for each of the logical devices must be s-independent.
2. The logical function of a device cannot be changed by some condition in one of its inputs.

The method for increasing the reliability of combinational logic networks can be summarized as follows. A given network designed to compute a function $F(x^m)$ is replaced by one that is designed to compute a new function $H(x^c)$. $H(x^c)$ is defined as that function which is equivalent to successive applications of a decoding function $d(x^c)$, a desired com-

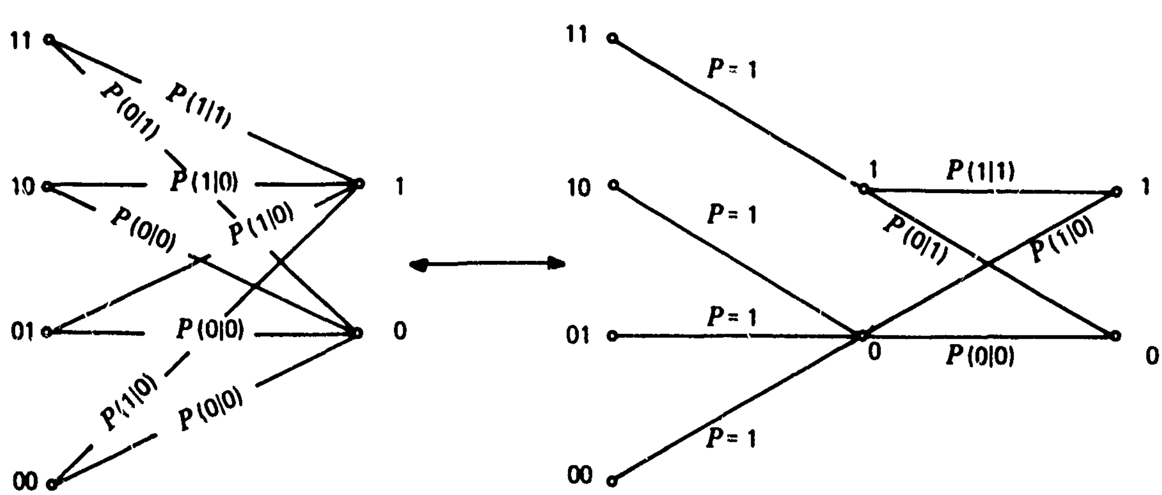


FIGURE 10-15. Two Models for a Noise AND Gate[1]

putation $F(x^m)$, and an encoding function e such that

$$H(x^\ell) = e\{F[d(x^\ell)]\} \qquad (10\text{-}38)$$

where

$$H(x^\ell) = \{H_1(x^\ell), H_2(x^\ell), \ldots, H_n(x^\ell)\}$$

$$e\{F[d(x^\ell)]\} = [e_1\{F[d(x^\ell)]\}, e_2\{F[d(x^\ell)]\}, \ldots, e_n\{F[d(x^\ell)]\}]$$

$$e_i\{F[d(x^\ell)]\} = e_i\{F_1[d(x^\ell)], F_2[d(x^\ell)], \ldots, F_m[d(x^\ell)]\}$$

$$F_i[d(x^\ell)] = F_i[d_1(x^\ell), d_2(x^\ell), \ldots, d_m(x^\ell)]$$

$$d_i(x^\ell) = d_i(x_1^\ell, x_2^\ell, \ldots, x_\ell^\ell) \ . \qquad (10\text{-}39)$$

Here, $d(x^\ell)$ is the decoding function corresponding to some error-correcting code that is assumed to have been used on the output of the preceding network, and e is the encoding function corresponding to some code which, of course, also must be accommodated on the input of the following network. The net result is to replace one network by another where the two networks are related through two error-correcting codes, such that, in the absence of errors, a given input and output state of the second is the encoded form of the corresponding input and output states of the first.

The performance of devices using coding redundancy can improve the correctness of output signals and also the engineering confidence of the individuals using the equipment. If the decoding function becomes complex, the usefulness of coding redundancy is minimized, and this appears to be the major drawback of coding redundancy.

To estimate reliability improvement, consider first a system model that will be used to estimate a system error probability. In this model, a system consists of N combinatorial networks arranged in an arbitrary order (any combination of series and parallel). Network j has n_j outputs being generated by devices having a fan-in of ℓ_j, each of which has an error probability of $p(\ell_j)$. Let α_j be defined as the probability that more than t_j of the n_j outputs are in error, where t_j is the maximum number of errors that can be corrected by the code used in the output of network j. Assume that a system error is obtained if one or more networks generate an output having more than t_j errors, then P, the probability of a system error, can be calculated as

$$P = 1 - \prod_{i=1}^{N} (1 - \alpha_i) \qquad (10\text{-}40)$$

where

$$\alpha_i = \sum_{i=t_j+1}^{n_j} \binom{n_j}{i} p^i(\ell_j)[1 - p^i(\ell_j)]^{n_j - i}$$

$$\approx \binom{n_j}{t_j + 1} p^{t_j+1}(\ell_j)$$

$$n_j p(\ell_j) \ll 1 \ . \qquad (10\text{-}41)$$

Let a measure of improvement I, the improvement factor, be defined as the ratio of the system error probability before and after coding,

$$I = p_B / p_A \qquad (10\text{-}42)$$

where

B,A are subscripts referring to Before and After coding, respectively.

If, for simplicity, it is assumed that the system is sufficiently homogeneous that all the networks have the same number of inputs and outputs, and the same error-correction capacity($n_i = n_j, \ell_i = \ell_j$, and $t_i = t_j$, for all i and j), then

$$\alpha_i = \alpha_j = \alpha \qquad (10\text{-}43)$$

for all i, j, and

$$p = 1 - (1 - \alpha)^N \approx N\alpha, \qquad (10\text{-}44)$$

$N\alpha \ll 1$.

Thus,

$$I \approx \alpha_B / \alpha_A \ . \qquad (10\text{-}45)$$

A detailed explanation of the practical problems associated with this type of design is presented in Ref. 12.

10-4 DECISION-WITH-SWITCHING REDUNDANCY

10-4.1 STANDBY REDUNDANCY

A system in which a component or unit is standing by idly (cold standby) and operates only when the preceding unit fails is said to be using standby or sequential redundancy (Refs. 1 and 13). A standby system usually requires failure-sensing and/or switching net-

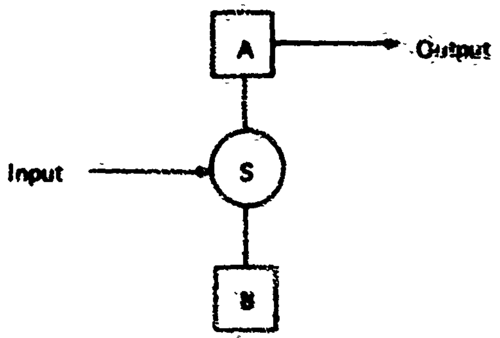


FIGURE 10-16. System Illustrating Standby Redundancy[1]

works or devices to put the next unit into operation.

Fig. 10-16 shows two elements where A is operating and B is in standby redundancy, waiting until A fails, and S is the sensing and switching mechanism. The device operates in the following four mutually exclusive ways:

1. S is operating properly. It monitors A, and if A fails, it turns B on, and the device operates until B fails (Case 1).
2. S fails by not going able to sense and/or switch, and when it fails, A is operative and the device fails when A fails (Case 2).
3. S fails and in failing it switches to B. A is still operating when S fails, but the device fails when B fails (Case 3).
4. A is operating and S fails. The signal path through S becomes open or short and the entire device fails at the time S fails (Case 4).

The notation for Eqs. 10-46 through 10-49 follows:

ϕ_α = failure *pdf* for α, $\alpha = A,B,S$
Φ_α = failure *Cdf* for α, $\alpha = A,B,S$
$\bar{\Phi}_\alpha$ = $1 - \Phi_\alpha$
q_1 = probability that S fails and the switch stays on A
q_2 = probability that S fails and the switch goes to B
q_3 = probability that S fails in such a way that the signal path is shorted or open

$$q_1 + q_2 + q_3 = 1 \ .$$

For Case 1:

$$Q_1(t) = \int_{t_2=0}^{t} \int_{t_1=0}^{t-t_2} \bar{\Phi}_s(t_1)\phi_A(t_1)\phi_B(t_2)dt_1 dt_2 \tag{10-46}$$

For Case 2:

$$Q_2(t) = q_1 \int_{t_1=0}^{t} \phi_S(t_1)\Phi_A(t_1)dt_1 \ . \tag{10-47}$$

For Case 3:

$$Q_3(t) = q_2 \int_{t_2=0}^{t} \phi_B(t_2) \int_{t_1=0}^{t_2} \bar{\Phi}_A(t_1)\phi_S(t_1)dt_1 dt_2 \ . \tag{10-48}$$

For Case 4:

$$Q_4(t) = q_3 \int_{t=0}^{t} \bar{\Phi}_A(t_1)\phi_S(t_1)dt_1 \ . \tag{10-49}$$

For the entire device

$$Q(t) = Q_1(t) + Q_2(t) + Q_3(t) + Q_4(t) \ . \tag{10-50}$$

For the special case of the exponential failure law where λ_S is the failure rate of the switching mechanism, and $\lambda = \lambda_A = \lambda_B$ is the failure rate of the two systems A and B, standby redundancy is better than two systems in parallel if $\lambda > \lambda_S$. If $\lambda = \lambda_S$, the two types of redundancy are equal; and if $\lambda < \lambda_S$, parallel redundancy is superior.

The gain for a specified mission can be measured in terms of the ratio of the reliability of the structure with standby redundancy to the reliability of alternate structures.

10-4.2 OPERATING REDUNDANCY

In operating redundancy, *s*-independent identical units operate simultaneously with a common input (Refs. 1 and 14). A failure detector is associated with each unit, and a switch is connected to the outputs. All units are operating initially, and the output of one unit is used until that unit fails. The switch then steps to the next operating unit and remains there until that unit fails.

Fig. 10-17 shows a typical switching circuit in which C represents the redundant components and D the individual detectors. The reliability block diagram has the same form as Fig. 10-17.

The following assumptions are made:

1. There are m chains ordered 1, ..., m and all m operate from the initial time until each fails.
2. The stepping switch is connected so that its inputs are the outputs of the m chains; the output of the switch is the output of the system. The switch operates sequentially, starting with chain 1. The switch indicates when all m chains have failed.
3. A failure-detecting device operates in conjunction with each chain and performs the following functions:

   a. If failure occurs in the chain to which the switch is connected, a signal is sent immediately to the switch, causing it to step.

   b. If a failure occurs in a chain to which the switch is not connected, a signal is stored; and if the switch steps to that chain, it is signaled to step once more.
4. No time is consumed by the failure-detecting and switching operations.
5. The reliability of a chain is the product of the reliabilities of its components.

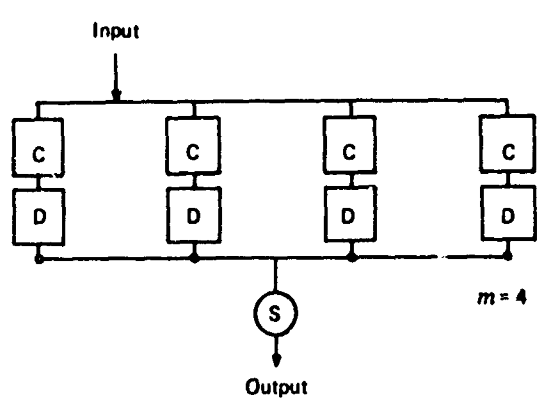


FIGURE 10-17. System of m Redundant Chains Illustrating Operating Redundancy[1]

The reliability of the system depends on the reliabilities of the chains, the failure detectors, and the switches. For the detectors and switches, there are two modes of behavior with which reliabilities are associated, i.e.,

1. D_a and S_a (Fig. 10-18): the device operates when failure occurs. This function can be performed only once for each chain, and the probability is defined for a single operation that takes place in negligible time.
2. D_b and S_b: the device does not spontaneously operate during a period of time in which no failure occurs. This type of failure, like a chain, is defined for the length of time required for the machine to complete the assigned task.

Therefore, the following probabilities are defined:

1. R_c = s-reliability of the chain, i.e., the probability that it performs its functions adequately for the duration of the assigned task.
2. $P(D_a)$ = conditional probability that when a failure occurs in a chain, the failure is detected and a signal is sent to the switch under conditions a or b. A consideration in $P(D_a)$ is the probability that the switch control is connected to the error detector for the chain at which the switch is positioned.
3. $P(D_b)$ = conditional probability that when no failure occurs in a chain for the duration of the task, no signal is transmitted to the switch when it is positioned at that chain.
4. $P(S_a)$ = conditional probability that when the switch receives a failure signal, the connection at which it stands is broken and a good connection is made to the next chain.
5. $P(S_b)$ = conditional probability that if the switch does not receive a failure signal for the duration of the task, it does not step at any time during the run. If it does step, it makes contact on the next chain.
6. $P(S_c)$ = conditional probability that if a good connection is made every time the switch steps, a good connection exists between some chain (or the device indicating system failure) and the system output at all times during the run. Switching occurs in zero time.

The reliability of the system of m redundant chains is defined as the probability that it performs the assigned task successfully. This occurs if, for the duration of the task,

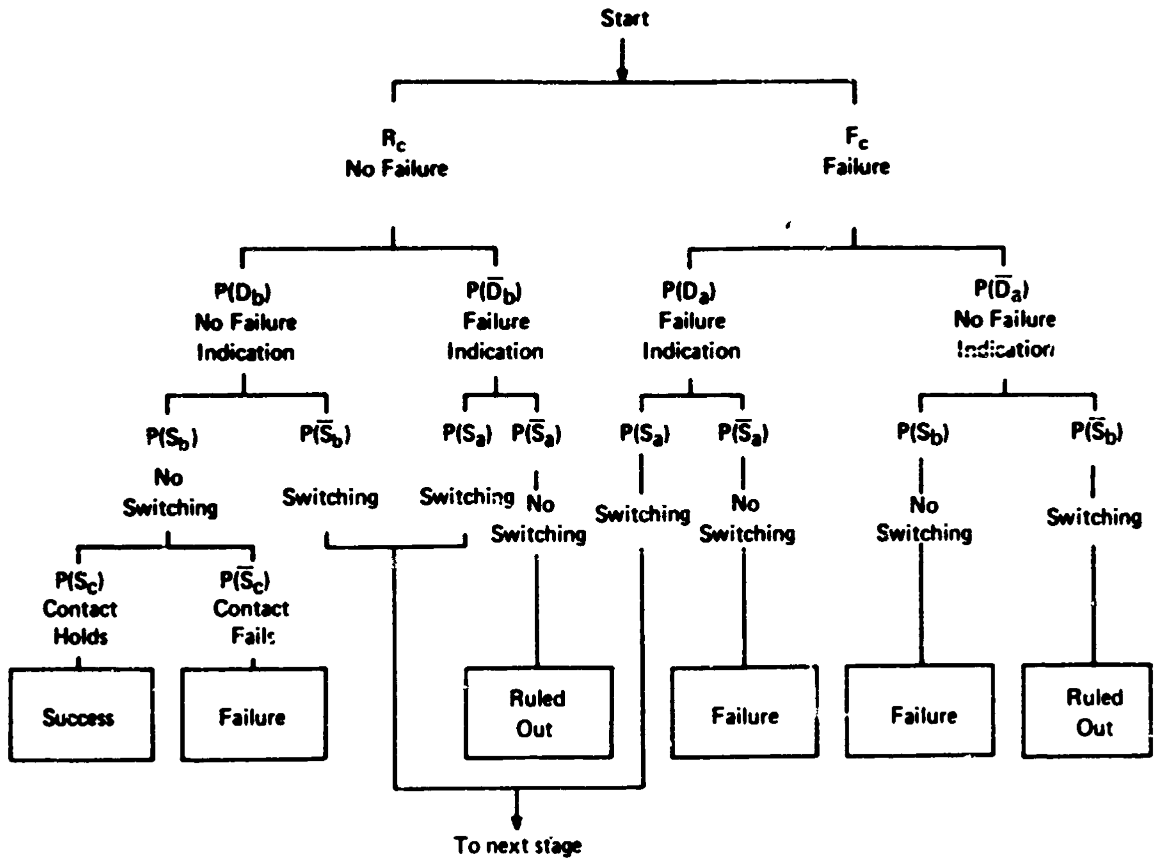


FIGURE 10-18. Failure Diagram of a Chain[1]

the switch constantly makes a good connection to a chain that is functioning adequately. This can take place in m mutually exclusive ways, corresponding to the final connection to the m switch contact.

The possible modes of behavior of a chain are diagrammed in Figure 10-18. Successful operation through a given chain requires that the chain function adequately, R_c; that the failure detector not signal an error, $P(D_b)$; that the switch not step simultaneously while connected to this chain, $P(S_b)$; and that the switch contact remain good, $P(S_c)$. The probability of successful operation is

$$R_1 = R_c P(D_b) P(S_b) P(S_c). \quad (10\text{-}51)$$

The use of one value of $P(S_b)$ for the probability of no spontaneous stepping of the switch from any position is an approximation. A precise analysis would use $P(S_b)$ as previously defined only for the first chain with successively larger values for this probability for chains 2, ..., m. The final computed reliability is actually somewhat lower than the correct result. However, since the probability of spontaneous switching in all practical applications is very small, the more precise analysis does not appear to be warranted.

A stepping of the switch can occur in three ways (the symbols are for probabilities rather than for events):

1. The chain fails ($F_c = 1 - R_c$); the detector signals failure, $P(D_a)$; and the switch steps, $P(S_a)$.
2. The chain does not fail, R_c; but the detector erroneously signals failure, $P(\bar{D}_b) = 1 - P(D_b)$; and the switch steps, $P(S_a)$.
3. The chain does not fail, R_c; the detector does not signal failure, $P(D_b)$; but the switch steps spontaneously, $P(\bar{S}_b) = 1 - P(S_b)$.

Thus, the probability of one stepping of the switch is

$$\alpha \equiv (F_c) P(D_a) P(S_a) + R_c P(\bar{D}_b) P(S_a) + R_c P(\bar{S}_b) P(D_b) \tag{10-52}$$

There are several modes of behavior of one chain that lead immediately to system failure without any failure indication, due to a bad switch contact $P(\bar{S}_c)$, to failure of the switch to respond to an error signal $P(\bar{S}_a)$, or to failure of the detector to indicate failure $P(\bar{D}_a)$. In addition, there are modes of behavior in which the detector and switch both make errors that cancel each other. These second-order effects will be arbitrarily ruled out.

The probability of successful operation with the final connection to switch-contact i is equal to the probability of $(i-1)$-steppings of the switch times the probability of successful operation through one chain, or $\alpha^{(i-1)} R_1$.

Then, the reliability of the system is the sum of the probabilities for the m switch contacts:

$$R = \sum_{i=1}^{m} \alpha^{(i-1)} R_1 = R_1 \left(\frac{1-\alpha^m}{1-\alpha} \right) \tag{10-53}$$

where

$$R_1 = R_a P(D_b) P(S_b) P(S_c)$$

$$R_c = \prod_{i=1}^{n} R_i \ . \tag{10-54}$$

Because all $P(\cdot) \leqslant 1$,

$$R < P(S_c) \tag{10-55}$$

$$R \leqslant 1 - (1 - R_c)^m \ . \tag{10-56}$$

In the present application, the device, with no redundancy, is considered to have a reliability R_0. It is assumed that it is possible to break the device up into p groups of equal reliability, $R_0^{1/p}$. It is further assumed that the failure detector for the complete device consists of p units, each associated with a group, such that indications of failure originating from any of these units are equally probable. Then, if $P(D_a)$ and $P(D_b)$ are probabilities associated with the failure detector for one complete device, the corresponding probabilities for the units associated with a group will be $P(D_a)^{1/p}$ and $P(D_b)^{1/p}$. If each chain is made n times redundant, the system reliability, for perfect failure detection and switching, is

$$R_s = [1 - (1 - R_0^{1/p})^n]^p \ . \tag{10-57}$$

The exact equations are complicated and are given in Refs. 1 and 14.

Operating redundancy is used in continuous time applications primarily, but it can be used in intermittent situations if the failure-detecting device is capable of signaling the switching mechanism at the proper time.

The performance of these systems in many instances will be limited by the reliability of the failure-detecting and switching assemblies.

Tables and charts given in Ref. 14 can be used in designing systems with operating redundancy: Given an estimate of the initial unreliability for a nonredundant system and the tolerable unreliability permitted in the final system, the degree of redundancy and the number of chains that will meet the specifications can be estimated from the appropriate curves in the reference.

For initially unreliable systems and a moderate degree of redundancy, high reliability can be achieved only by applying the redundancy to relatively small units. Imperfect switching limits the reliability attainable in all cases such that the unreliability is not a steadily decreasing function of n, but has a definite minimum beyond which it increases.

10-4.3 DUPLEX REDUNDANCY

Duplex redundancy uses duplicated logic circuits operating in parallel (Refs. 1, 13, and 15). It has an error detector at the output of each circuit which detects any noncoincident outputs and starts a diagnostic procedure. This procedure may last from a few microseconds to a few milliseconds, depending on the diagnostic process chosen in the design. Figure 10-19 illustrates the duplex scheme.

If the exponential failure law is assumed, the reliability of the system when duplex redundancy and error detection is used is:

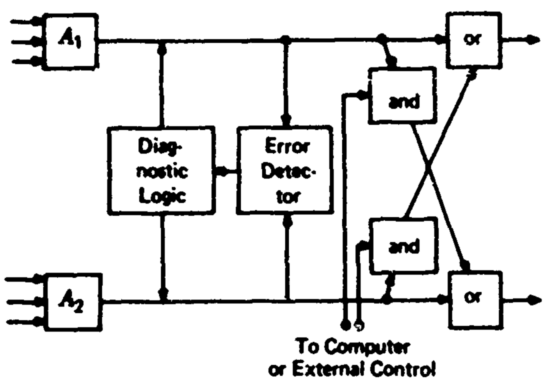


FIGURE 10-19. Illustration of Duplex Redundancy[1]

$$R = e^{-\tau}(1 + e^{-\alpha\tau} - e^{-(1+\alpha)\tau}) \quad (10\text{-}57)$$

where

$\tau = \frac{1}{n} \times$ (failure rate of individual circuit)

n = the number of circuits in sequence

$\alpha = \dfrac{\text{failure rate of error detector}}{\text{failure rate of individual circuits.}}$

Duplex redundancy can be used in digital computer logic circuits to protect against faulty outputs from basic logic elements. Duplex redundancy should improve digital system reliability. However, the system will not automatically correct intermittent errors or two simultaneous failures.

Features of a duplex logic redundancy system are:

1. Basic logic circuitry is fully redundant.
2. All errors are detected and the faulty logic unit is disabled, thus correcting the error. Faulty logic units can be repaired without interrupting system operation. If both A_1 and A_2 fail at the same time, there is no error detection; however, this situation is very unlikely to occur.
3. The system is disabled only when both logic units fail.
4. The error detector is not in series with the output signals; hence, its failure does not affect the output.
5. Maintenance problems are simplified since the faulty logic unit can be identified automatically. Rapid identification of faults permits rapid replacement of failed units.

The main disadvantage of duplex redundancy is the need for a short diagnostic procedure in the event of failure. Also, in order to avoid losing essential information, it may be necessary to record the contents of important registers and the input data. In this way, after an error is corrected, the original situation can be restored.

REFERENCES

1. *Handbook for Systems Application of Redundancy*, U S Naval Applied Science Laboratory, 30 August 1966.
2. J. J. Suran, "Use of Passive Redundancy in Electronic Systems", *IRE Transactions on Military Electronics*, Moore-Shannon Discussion, November 3, 1961.
3. W. E. Dickinson and R. M. Walker, "Reliability Improvement by the Use of Multiple-Element Switching Circuits", *IBM Journal*, April 1958.
4. R. M. Fasano and A. G. Lemack, "A QUAD Configuration - Reliability and Design Aspects", Proceedings Eighth National Symposium on Reliability and Quality Control, 8, 394-407, January 1962.
5. J. K. Knox-Seith, *A Redundancy Technique for Improving the Reliability of Digital Systems*, Technical Report No. 4816-1, Stanford Electronics Laboratories.
6. P. A. Jensen, "The Reliability of Redundant Multiple-Line Networks", IEEE Transactions on Reliability, R-13 23-33 (March 1964).
7. M. K. Cosgrove, et al., *The Synthesis of Redundant Multiple-Line Networks*, AD-602 749, 1 May 1964.
8. J. D. Esary and F. Proschan, "The Reliability of Coherent Systems", *Redundancy Techniques for Computing Systems*, R. H. Wilcox and W. C. Mann, Eds., Spartan Books, Washington, D.C., 47-61 (1962).
9. J. D. Esary and F. Proschan, "Coherent Structures of Non-Identical Components", Technometrics 5, 191-209 (May 1963).
10. R. Teoste, "Digital Circuit Redundancy", IEEE Transactions on Reliability, R-13, 42-61 (June 1964).

11. J. Tooley, "Network Coding for Reliability", *AIEE Transactions*, Part 1, 407-14 (1962).
12. W. H. Pierce, *Failure Tolerant Computer Design*, Academic Press, New York and London, 1965.
13. L. A. Aroian, "The Reliability of Items in Sequence with Sensing and Switching", *Redundancy Techniques for Computing Systems*, R. H. Wilcox and W. C. Mann, Eds., Spartan Books, Washington, D.C., 318-27 (1962).
14. B. J. Flehinger, "Reliability Improvement through Redundancy at Various System Levels", *IBM Journal*, 148-58 (April 1958).
15. R. W. Lowrie, "High-Reliability Computers Using Duplex Reliability", *Electronic Industries*, August 1963.

CHAPTER 11 MONTE CARLO SIMULATION

11-0 LIST OF SYMBOLS

Cdf = Cumulative distribution function

pdf = probability density function

s- = denotes statistical definition

t = time for r failures

Var{ } = Variance of

λ_i = true failure rate for i

$\hat{\lambda}_i$ = estimated failure rate for i, a random variable

χ^2 = chi-square, a special random variable

11-1 INTRODUCTION

In formal terms, Monte Carlo simulation (often just called simulation) is a method of mathematically simulating a physical experiment to determine some probabilistic property of a population of events by the use of random sampling applied to the components of the events; see Refs. 1-4 for more information. Less formally, simulation involves determining the probability distributions of the components of the system, and selecting a random sample from each component distribution. The resultant component sample values then are combined in a model to estimate the system reliability measure. This process is repeated many times until enough data have been obtained to estimate the system probability distribution with the required precision. The measure can be s-reliability or mean time to failure, or it can be a performance parameter such as bandwidth, gain, noise, or power output.

Simulation can be applied at various phases of a program. For example, if actual performance or failure data are available on some of the components, the distribution of these values can be determined. Then by random sampling of these distributions and by combining the sample values into a model describing the system in terms of its components, the distribution of system performance can be derived. These methods also can be used as a prediction and analysis tool. For example, during the system conceptual phase, a system model can be developed in terms of its components and, through use of various assumed component distributions, the performance of the system can be evaluated. Simulation also can be used as a comparative tool. Through simulation of various systems and their component distributions, the different types of systems can be compared, and an optimum approach can be selected with a high degree of assurance that, if the models used to describe the system are realistic, the selection truly will be optimum.

Simulation is based on several principles of probability and on the techniques of probability transformations. One of the underlying principles is the law of large numbers, which states that the larger the sample, the more certainly the sample mean will be a good estimate of the population mean. The central-limit theorem gives a more precise statement of the law of large numbers (there are several theorems under this heading, all relating to the same topic--see Ref. 5 or Bibliography at end of Chapter 1): If a population has a finite variance σ^2 and mean μ, then the distribution of the sample (size n) mean approaches the s-normal distribution with variance σ^2/n and mean μ as the sample size n increases.

An interesting thing about the central-limit theorem is that nothing is implied about the form of the population distribution function. Whatever the distribution function, within reasonable limits, the sample mean will have approximately the s-normal distribution for large samples.

11-2 PROPERTIES OF DISTRIBUTIONS

Chapters 2 and 3 introduced the concept of probability density functions (pdf) for continuous random variables, the probability mass function (pmf) for discrete random variables, and the cumulative distribution function (Cdf) for any random variable. Textbooks, such as Ref. 5 and the Bibliography at the end of Chapter 1, give an adequate introduction to probability theory.

The s-expectaction of the average of N s-independent trials of a function of $g(x_j)$ is the *s*-expectation of $g(x)$, where X is a random variable.

A generalization of the law of large numbers comes into play during the repeated Monte Carlo trials:

$$\lim_{N\to\infty} \left\{ Pr \left\{ \left| \int_{-\infty}^{\infty} g(x)f(x)dx - \frac{1}{N}\sum_{1}^{N} g(x_j) \right| > \epsilon \right\} \right\} = 0 \tag{11-1}$$

where

ϵ = any positive number
$f(x)$ = *pdf* of x
$g(x)$ = any function of x; usually, the one being simulated
N = sample size
x_j = sample value of X

Eq. 11-1 shows that the chance of departure from the true value of $g(x)$, weighted according to the frequencies of the x's, becomes less as N increases.

The reasoning can be extended to a function of many variables.

11-3 THE SIMULATION METHOD

The simulation method is a way to determine the distribution of a function of one or more variables from the distributions of the individual variables. The method involves random sampling from the distributions of all variables and inserting the values so obtained in the equation for the function of interest. Suppose the function whose distribution is to be estimated is $g(x_1, x_2, \ldots, x_n)$ and that the $X_1, X_2, \ldots, X_n$ are *s*-independent random variables whose distributions are presumed to be known. The procedure is to pick a set of x's randomly from the distributions of the X's, calculate g for that set, and store that value of g. The procedure is repeated many times until enough values of g are obtained. From this sample of g values, its distribution and parameters can be estimated. Very often, one settles for estimating the mean and standard deviation of g.

Simulation is a well developed art/science. It is virtually always done on a computer because a tremendous number of calculations are involved. Special simulation languages have been developed. Check with your computer installation to find out what simulation facilities are available, and what programming assistance that installation can offer.

11-4 MEASURES OF UNCERTAINTY

Several methods are available for establishing *s*-confidence intervals and estimating uncertainties in the results of a simulation. They are essentially the same as in any sampling technique. Chapter 4 reviews some of the statistical concepts and gives references for further reading. The procedures are all quite standard and well-known (to mathematicians).

The required sample size for a given minimum uncertainty is a handy number to have. It is useful for getting an idea of how much computer time is likely to be involved. For simulations of equipments, the programming and analytic effort to get ready to simulate will far outweigh the cost of actually running the simulations. Table 11-1 shows typical sample sizes for various *s*-confidence levels and goodness-of-fit (to the *Cdf*).

TABLE 11-1

MINIMUM SAMPLE SIZE REQUIRED FOR MONTE CARLO SIMULATION[6]

δ	$\gamma = 0.90$	$\gamma = 0.95$	$\gamma = 0.99$
0.01	6800	9600	16500
0.02	1700	2400	4125
0.03	750	1066	1833
0.05	272	384	660

δ = maximum deviation of sample *Cdf* from true *Cdf*
γ = *s*-confidence level
This table is derived from the Kolmogorov-Smirnov test of goodness-of-fit. It does not depend on the form of the distribution.

Since theory shows that the Monte Carlo technique gives a true random sample of the population (function) to be estimated, there is no need to go into special discussions about the statistical theory.

All random distributions used for digital computers are pseudo-random. Since a pattern is used to generate the pseudo-random numbers, modest attention ought to be devoted to being assured that the numbers will behave well enough for your particular simulation. Rarely in reliability work will difficulties from this source arise, but it can happen.

11-5 APPLICATIONS

In principle, the demonstration of the reliability of a system is a fairly straightforward procedure. Take several systems, operate them for a sufficient length of time, record the number of failures which occur, and evaluate the results by one of a number of available statistical techniques. Unfortunately, this is not practical — particularly for dealing with complex, costly systems. Even an optimum mix of time, available systems, manpower, and test facilities is often economically prohibitive.

Because of the complexity of many systems, extensive tests at the system level often are limited because of time, facilities, cost, and schedules. Instead, extensive testing generally is done at the subsystem level. This permits testing to be conducted earlier in a program, and reveals potential difficulties at the earliest possible time. Two management and statistical difficulties arise if the test results are to be used to assess the reliability potential of the system. Such tests may be part of the design-development program, and the reliability data obtained may be a byproduct rather than the end result of the test. Therefore, there is no longer a controlled condition in the statistical sense, and the analyst is forced to work with the information that becomes available.

The synthesis of system reliability from the results of subsystem tests is not a simple problem. As a rule, each subsystem type will be run a different number of total operating hours, and different numbers of failures will be observed.

To illustrate the second point, consider a simple series (1-out-of-3:F) system consisting of 3 *s*-Independent subsystems, with the operating times and observed failures indicated in Table 11-2.

The subsystems have constant failure rates. The failure rate of the system is just the sum of the subsystem failure rates, and we could try the same formula using the estimated failure rates from Table 11-2, viz. $\hat{\lambda}_s$ = (0.40 + 0.25 + 0.20) per 1000 hr = 0.85 per 1000 hr; $\hat{\lambda}$ is an estimate of the failure rate λ. We have an estimate of λ_s; but, (as mentioned in Chapter 4 "Review of Statistical Theory") the trick is, not to get an estimate

TABLE 11-2

SUMMARY OF SUBSYSTEM OPERATING TIMES, FAILURES, FAILURE-RATE ESTIMATES AND *s*-CONFIDENCE INTERVALS FOR FAILURE RATES

Subsystem	Total operating time t, hr	Test stopped after r failures	$\hat{\lambda}_i = r/t$, per 1000 hr	*s*-Confidence interval for λ_i lower 5%	*s*-Confidence interval for λ_i upper 5%
1	5000	2	0.40	0.071	0.95
2	8000	2	0.25	0.044	0.59
3	10000	2	0.20	0.036	0.47
System	---	·	0.85	?	?

$\hat{\lambda}_i$ is an estimate of the true failure rate λ_i.

The *s*-confidence intervals were obtained from a table of the chi-square distribution; $2\lambda_i t$ has a chi-square distribution with $2r$ degrees of freedom. From tables such as those in Part Six, Mathematical Appendix and Glossary, for 4 degrees of freedom, the lower 5% point is χ^2 = 0.711 and the upper 5% point is χ^2 = 9.49. λ bound = $\chi^2/(2t)$.

(anyone can do that), but to know its statistical properties. Unfortunately, the statistical properties of the estimate we just used are not known. The statistical properties of estimates of system reliability from a knowledge of subsystem sample data is an unsolved problem (except for a few special cases).

For each subsystem, it is known that $2\lambda t$ has a chi-square distribution with γ degrees of freedom; $\gamma = 2r$ if the test is stopped after r failures, and $\gamma = 2(r+1)$ if the test is stopped after a fixed time t; λ is the true failure rate.

We will solve our particular problem by Monte Carlo simulation. The equation for $\hat{\lambda}_s$, whose distribution we want to estimate is

$$\hat{\lambda}_s = \hat{\lambda}_1 + \hat{\lambda}_2 + \hat{\lambda}_3 \qquad (11\text{-}2)$$

In each subsystem, the procedure is to run until 2 failures occur. We cannot simulate unless we know the distributions from which the $\hat{\lambda}_i$ come. So we cannot solve the problem in Table 11-2 by a short simulation; we can, however, solve a similar one, as given in Table 11-3. We have to know all the parameters in a problem in order to solve it by Monte Carlo simulation. It is neither correct nor meaningful to use the random times in Table 11-2 to find a "distribution for λ"; in classical statistics, λ does not have a distribution, it is fixed. See Ref. 7 for an advanced discussion of s-confidence.

One of the big difficulties with Monte Carlo simulation is that it is so restricted. Like other numerical techniques, it does not answer general questions; it only treats the specific numbers used in it.

Let χ_4^2 be a random value from a chi-square distribution with 4 degrees of freedom.

TABLE 11-3

SYSTEM FAILURE BEHAVIOR

Subsystem	True failure rate λ_i, per 1000 hr	Test stopped after r failures, r
1	0.80	2
2	0.50	2
3	0.10	2
System	1.40	.

Then, for this example

$$\hat{\lambda}_i = r_i/t_i = 2/t_i \qquad \text{(defines } \hat{\lambda}_i\text{)} \qquad (11\text{-}3)$$

$$2\lambda_i t_i = n_{2r}^2 = \chi_4^2 \qquad (2\lambda t \text{ has a } \chi_{2r}^2 \text{ distribution}) \qquad (11\text{-}4)$$

$$\hat{\lambda}_i = 4\lambda_i/\chi_4^2 \qquad (11\text{-}5)$$

Eq. 11-5 is used to calculate $\hat{\lambda}_i$ from a randomly generated value of chi-square (with 4 degrees of freedom). Table 11-4 is a collection of pseudo-random numbers from the chi-

TABLE 11-4

RANDOM NUMBERS FROM THE CHI-SQUARE DISTRIBUTION WITH 4 DEGREES OF FREEDOM

No. 1	No. 2	No. 3
11.73	4.959	6.134
0.6107	3.858	4.721
2.628	1.566	7.891
6.040	6.393	3.485
2.106	2.590	1.867
4.994	4.870	3.040
2.135	14.47	4.920
2.977	3.897	4.376
3.172	7.499	1.331
9.594	1.331	2.262
5.751	3.487	3.083
0.1846	0.5026	2.660
9.423	6.447	2.254
4.967	0.3100	2.194
6.093	3.182	5.509
5.074	7.010	5.559
4.347	9.706	1.177
1.094	1.498	3.107
3.696	8.131	4.455
0.3131	7.743	2.267
0.4130	4.379	4.907
3.559	7.291	1.333
2.523	1.311	6.511
6.946	10.32	4.688
1.571	3.098	0.9772
18.71	1.456	3.709
13.02	2.405	5.368
7.036	9.338	4.619
2.787	1.767	7.469
6.049	3.203	2.261

TABLE 11-5

MONTE CARLO ANALYSIS OF EXAMPLE SYSTEM

		Subsystem No. 1		Subsystem No. 2		Subsystem No. 3		System	
		n	$\hat{\lambda}_1$	n	$\hat{\lambda}_2$	n	$\hat{\lambda}_3$	n	$\hat{\lambda}_s$
		3	0.273	12	1.403	4	0.063	1	0.739
		27	5.240	16	1.518	10	0.085	26	5.843
		21	1.218	24	1.277	1	0.051	22	2.546
		10	0.530	11	0.313	16	0.115	5	0.957
		24	1.519	21	0.772	26	0.214	21	2.506
		13	0.641	13	0.411	19	0.132	8	1.183
		23	1.499	1	0.138	8	0.081	18	1.718
		19	1.075	15	0.513	14	0.091	17	1.680
		18	1.009	7	0.267	28	0.301	15	1.576
		4	0.334	27	1.503	22	0.177	19	2.013
		11	0.556	17	0.574	18	0.130	11	1.260
		30	17.335	29	3.979	20	0.150	30	21.464
		5	0.340	10	0.310	23	0.177	4	0.827
		14	0.644	30	6.451	25	0.182	27	7.278
		8	0.525	19	0.629	6	0.073	10	1.226
		12	0.631	9	0.285	5	0.072	6	0.988
		15	0.736	3	0.206	29	0.340	12	1.282
		26	2.925	25	1.335	17	0.129	25	4.389
		16	0.866	5	0.246	13	0.090	9	1.202
		29	10.219	6	0.258	21	0.176	29	10.654
		28	7.748	14	0.457	9	0.082	28	8.286
		17	0.899	8	0.274	27	0.300	14	1.474
		22	1.268	28	1.526	3	0.061	23	2.856
		7	0.461	2	0.194	11	0.085	2	0.740
		25	2.732	20	0.646	30	0.409	24	3.787
		1	0.171	26	1.374	15	0.108	16	1.653
		2	0.246	22	0.832	7	0.075	7	1.152
		6	0.455	4	0.214	12	0.087	3	0.756
		20	1.148	23	1.132	2	0.054	20	2.333
		9	0.529	18	0.624	24	0.177	13	1.330
sample mean	$\bar{x}$		2.126		0.922		0.142		3.190
sample standard deviation	s		3.664		1.282		0.091		4.221
	$s/\bar{x}$		1.72		1.39		0.64		1.32

n is the order number in the sample.

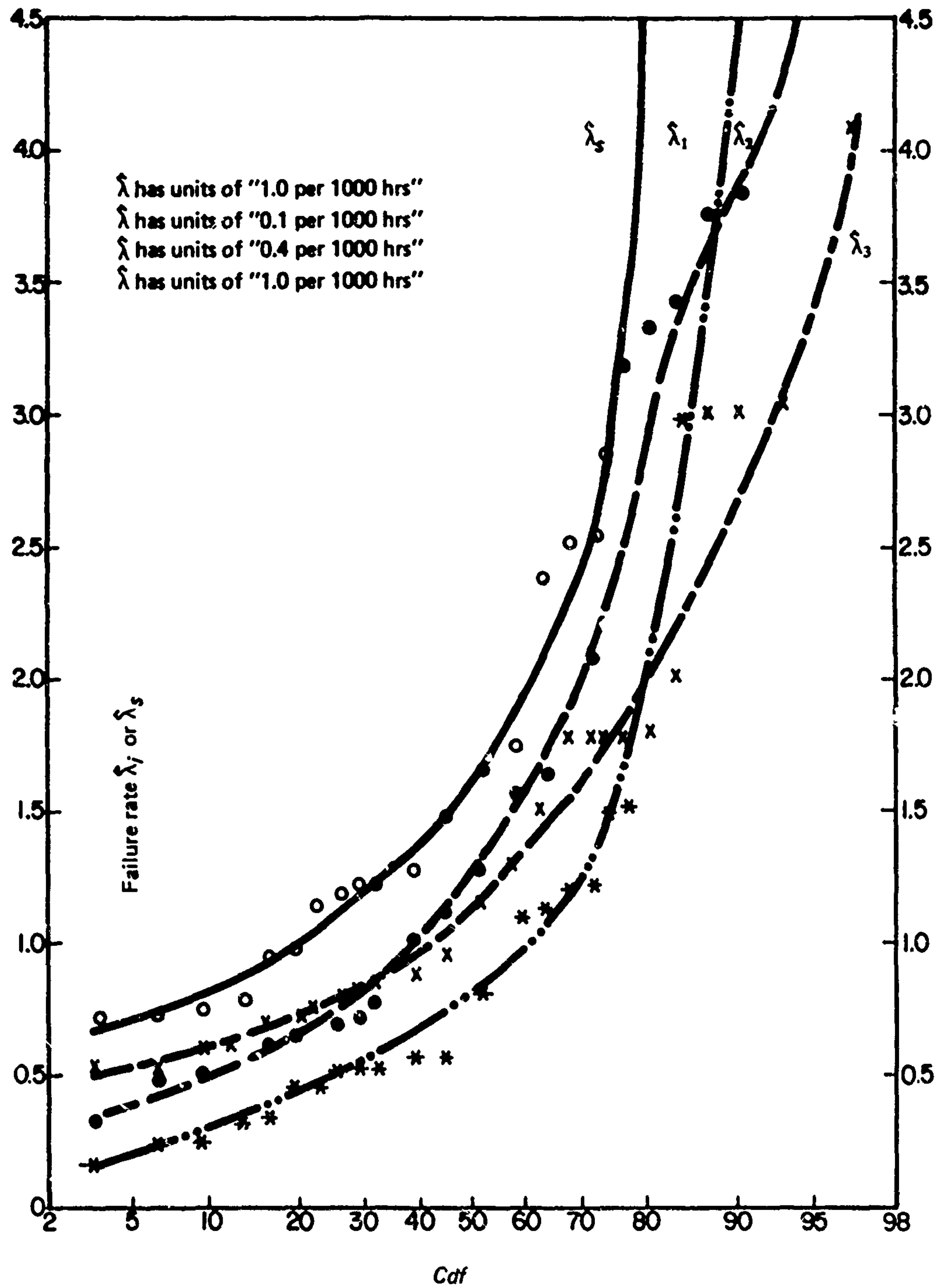


FIGURE 11-1. Sample CDF's for the Example (s-Normal distribution paper)

square distribution with 4 degrees of freedom, as required in Eq. 11-5. They are pseudo-random because they exist beforehand (for us) on a sheet of paper. Since the numbers are pseudo-random, a choice must be made on how to use them. Arbitrarily pick column i for $\hat{\lambda}_i$, $i = 1, 2, 3$; and begin at the top and go down in sequence. We will hope that the method of generating these numbers did not have a "cycle" such that the rows contain highly correlated numbers.

Table 11-5 contains the calculations for the $\hat{\lambda}_i$ and for $\hat{\lambda}_s$; $\hat{\lambda}_s$ is derived from Eq. 11-2. The estimate of $\hat{\lambda}_i$ in Table 11-5 occupies the same relative position that the random number does in Table 11-4. Column 4 in Table 11-5 contains the estimates of the system failure rate. At the bottom of each column, there is the sample mean $\bar{x}$, sample standard deviation s, and the ratio $s/\bar{x}$.

As to be expected, the mean of $\hat{\lambda}_s$ is the sum of the means of the $\hat{\lambda}_i$. But the variance of $\hat{\lambda}_s$ is more than the sum of the variances of the $\hat{\lambda}_i$. This means that there was some correlation along the rows. A statistical test showed that the ratio $17.82/15.08 = 1.18$ of the Var $\{\hat{\lambda}_s\}$ /(Var $\{\hat{\lambda}_1\}$+ Var $\{\hat{\lambda}_2\}$ + Var $\{\hat{\lambda}_3\}$) would be exceeded by chance about 25% of the time; probably not too bad.

The sample *Cdf*'s are plotted (smoothed somewhat) in Fig. 11-1, on *s*-normal distribution paper (an *s*-normal *Cdf* would appear as a straight line). Needless to say, none of the distributions are *s*-normal. The *pdf*'s are all skewed to the right; there are some very large sample values. The coefficient of variation ($s/\bar{x}$) is more than 1, which also shows the skewness of the distributions.

The curves for $\hat{\lambda}_i$ would all be the same (except for scale) if very large samples were used.

The tests were all terminated at the second failure. Obviously, there is a great deal of scatter in the test results.

This Monte Carlo trial, by hand, has shown the shape of the central portion (say, 5% to 95%) of the distributions. More trials would extend that range. The example was set up to use only one probability distribution for the trials; this was for convenience in doing hand calculations. In practice the distributions need not be the same for all elements.

REFERENCES

1. A. H. Cronshagen, *Application of Monte Carlo Techniques in Reliability Evaluation*, Aerojet-General Corp., Azusa, Calif., 5 June 1962.
2. C. W. Churchman, R. L. Ackoff, and E. L. Arnoff, *Introduction to Operations Research*, John Wiley and Sons, Inc., N.Y., p. 75.
3. DA Pam 70-5, *Mathematics of Military Action, Operations and Systems.*
4. M. L. Shooman, *Probabilistic Reliability*, McGraw-Hill Book Company, Inc., N.Y., 1968.
5. E. Parzen, *Modern Probability Theory and Its Applications*, John Wiley & Sons, Inc., N.Y., 1960.
6. *Mathematical Simulation for Reliability Predictions*, RADC Report, Sylvania Electric Products, Waltham, Mass., October 1961.
7. M. G. Kendall and A. Stuart, *The Advanced Theory of Statistics, Vol. II, Statistical Inference and Statistical Relationships*, Hafner, 3rd ed., 1971.

CHAPTER 12 RELIABILITY OPTIMIZATION

12-0 LIST OF SYMBOLS

a,b,A = matrices in par. 12-2.2
$f(\mathbf{x})$ = some function of x
$g(\alpha)$ = see Eq. 12-3
$g_i(\mathbf{x})$ = inequality-type constraint function
$h_i(\mathbf{x})$ = equality-type constraint function
r = number of constraints
R, R_i = constraint sets
s- = denotes statistical definition
s = gradient of $f(\mathbf{x})$, subscript i means value at iteration i
$\mathbf{x}, \mathbf{x}_i$ = vector with several components, value at iteration i
$\mathbf{x}^*$ = x for global minimum of f
x_i, x_α = individual dimensions (components of x)
$\mathbf{x}_0$ = some particular x; the starting point of x for an iterative solution for $f(\mathbf{x})$
α, α_i = scalar parameter, for iteration i
ϵ = some positive number (usually small)
λ, λ_i = scalar parameter between 0 and 1
ϕ_0, ϕ_1, ψ = special functions (par. 12-3.6)
ψ^T = implies transpose of ψ; x is any vector or matrix
∇ = gradient operator

12-1 INTRODUCTION

Seldom is it feasible to optimize a reliability function of a complicated system without using a computer. Thus, most of this chapter is written with computers in mind. Computer-aided design techniques offer the engineer relief from complicated calculations. Optimization programs can apply prespecified constraints and determine the most desirable component values. To accomplish these tasks, the computer must be provided with a method for generating alternate values for the design variables and some measure for comparing the resulting designs. This measure is usually a single function such as reliability, and the design goal is to optimize its value. A design which does this is called optimal. Methods for generating alternate solutions that account for constraints and that converge to an optimal solution generally are called mathematical programming techniques.

Mathematical programming techniques optimize a given objective function $f(\mathbf{x})$ by proper choice of a vector of design variables x. If x is restricted to certain allowable values, then the problem is constrained; if not, the problem is unconstrained.

The branch of mathematical programming that deals with linear constraints and linear objective functions is called linear programming. Since it is widely used and well described elsewhere (Refs. 1 and 2), linear programming will not be discussed here. Instead, nonlinear programming problems, i.e., those which have at least one nonlinear constraint or a nonlinear objective function, or both, will be discussed. Multistage problems which fall under the heading of dynamic programming will also be considered.

In engineering problems, the designer often wants to maximize or minimize a function of n variables, $f(\mathbf{x})$, in a situation where the design constraints do not restrict the values of the variables x. Many problems in which the constraints are binding can be converted to unconstrained problems or sequences of such problems. Since the problem of maximizing $f(\mathbf{x})$ is equivalent to that of minimizing $-f(\mathbf{x})$, we need consider only the minimization problem.

A point $\mathbf{x}^*$ is said to be a global minimum of $f(\mathbf{x})$ if, for all values of x,

$$f(\mathbf{x}^*) < f(\mathbf{x}) \ . \tag{12-1}$$

If the strict inequality holds, the minimum is said to be unique. If Eq. 12-1 holds only for all x in some neighborhood of $\mathbf{x}^*$, then $\mathbf{x}^*$ is said to be a local minimum of $f(\mathbf{x})$, since in this case $\mathbf{x}^*$ is the best point in the immediate vicinity but not necessarily the best point in the whole region of interest.

If $f(\mathbf{x})$ is continuous and has continuous first and second partial derivatives for all x, the first necessary condition for a relative minimum at $\mathbf{x}^*$ is that all the partial derivatives of $f(\mathbf{x})$ be zero, when evaluated at $\mathbf{x}^*$ (Ref. 3).

$$\left.\frac{\partial f(\mathbf{x})}{\partial x_i}\right|_{\mathbf{x}^*} = 0, \text{ for all } i \tag{12-2}$$

The second necessary condition is that the matrix of second partial derivatives evaluated at x^* be positive semidefinite. Any point x^* that satisfies Eq. 12-2 is called a stationary point of $f(x)$. Sufficient conditions for a relative minimum are that the matrix of second partial derivatives of $f(x)$ be positive definite and that Eq. 12-2 must hold.

12-2 NUMERICAL METHODS FOR FINDING UNCONSTRAINED MINIMA

The most obvious approach to finding the minimum of $f(x)$ is to solve Eq. 12-2. If $f(x)$ is not quadratic, Eq. 12-2—the set of n equations in n unknowns—is nonlinear, and solving large sets of nonlinear equations is usually a very difficult task. The function $f(x)$ may be so complicated that it is difficult even to write Eq. 12-2 in closed form. Further, even if the equations could be solved, there would be no guarantee that a given solution represented an actual minimum rather than some saddle point or maximum. We will, therefore, consider other methods of locating unconstrained minima.

12-2.1 GRADIENT METHODS

If $f(x)$ is continuous and differentiable, a number of minimization techniques using the gradient of $f(x)$ are available. The gradient $\nabla f(x)$ is a vector pointing in the direction of greatest increase of $f(x)$. At any point x_0, the vector $f(x)$ is normal to the contour of constant function value which passes through x_0. Two methods are presented.

12-2.1.1 Steepest Descent

The method of steepest descent for minimizing $f(x)$ is detailed in Table 12-1. In Step 2, the gradient can be found either by analytic formulas or by computing differences. Step 3 uses the direction of search determined in Step 2 and decides how far to move in this direction. The computer spends most of its time computing the gradient in this method, so the step length, α_i for Step i is selected to get the largest possible decrease in $f(x)$ for each gradient computation. Therefore, α_i is selected to minimize the function

$$g(\alpha) = f(x_i + \alpha s_i) \quad . \qquad (12\text{-}3)$$

Define also,

$$s_i = -\nabla f(x_i), \qquad (12\text{-}4)$$

the gradient of f. Both x_i and s_i are known vectors; α is the only unknown variable in Eq. 12-3.

The method of steepest descent converges to at least a local minimum of $f(x)$, providing certain mild restrictions are met (Ref. 5). The computations in Steps 2, 3, and 4 of the steepest descent method are repeated until a satisfactory value for x is found.

Several tests for determining when the computation should be stopped are also listed in Table 12-1. Stop Criteria 1 and 2 are based on the fact that the gradient vanishes at a minimum. When Criteria 3 and 4 are used, the computation will stop if the function value or current point changes by less than some small value ϵ. It has been found that Criterion 3 is the most dependable, providing it is met for several successive values of i. In all criteria, ϵ is a small positive number which the user selects. As ϵ decreases, the location of the minimum is more accurate, but more iterations are required to achieve this accuracy.

12-2.1.2 Cubic and Quadratic Interpolation

Finding a value α^* to minimize Eq. 12-3 can be thought of as a problem of 1-dimensional minimization in the direction of s_i. The cubic interpolation procedure outlined in Table 12-1 solves this problem for any given direction of s_i in which the function $f(x)$ initially decreases.

For the cubic interpolation procedure and the quadratic interpolation which follows, the components of x are scaled so that a unit change in any variable is an important (but not too large) fractional change in that variable. For example, if a capacitor is expected to have a value near 100μF, then a 1μF change would be important, but a 10μF change would be too large.

Steps 1 and 2 of the cubic interpolation procedure normalize s so that its components are less than or equal to 1 in magnitude. This, along with scaling, insures that s is a reasonable change in x. Step 3 moves along the direction s to place the desired minimum value α^* in the interval $a < \alpha^* < b$. Steps 4 through

TABLE 12-1

OPTIMIZING UNCONSTRAINED PROBLEMS

Method of Steepest Descent

1. Start the computation at some initial point $\bar{x}_0$, usually the best available estimate of the minimum. The ith iteration ($i = 0, 1, 2, \ldots$) proceeds as follows.
2. Compute the gradient $\nabla f(x_i)$ and let the current direction of search be $\bar{s}_i = -\nabla f(x_i)$.
3. Compute a step length α_i by choosing α_i to minimize $f(\bar{x}_i + \alpha\bar{s}_i)$. Cubic and quadratic interpolation procedures are detailed below.
4. Compute a successor vector for $\bar{x}_i$:

$$\bar{x}_{i+1} = \bar{x}_i + \alpha_i\bar{s}_i$$

5. Check a stop criterion (see below). If it is satisfied, stop. Otherwise, return to step 2 and replace i by $i + 1$.

Possible Stop Criteria for Terminating Computation

1. $\text{Max}_j \left|\dfrac{\partial f}{\partial x_j}\right| < \epsilon$
2. $\displaystyle\sum_{j=1}^{n} \left(\frac{\partial f}{\partial x_j}\right)^2 < \epsilon$
3. $f(\bar{x}_i) - f(\bar{x}_{i+1}) < \epsilon$
4. $\text{Max}_j(|\bar{x}_{i+1} - \bar{x}_i|)_j < \epsilon$

Cubic Interpolation

1. Calculate Δ, the maximum value of $|s_j|$.
2. Divide each component of the vector $\bar{s}$ by Δ.
3. Compute $g(\alpha)$ and $g'(\alpha) = \bar{s}'\nabla f(\bar{x} + \alpha\bar{s})$ for $\alpha = 0, 1, 2, 4, \ldots a, b$, where b is the first of these values at which either g' is nonnegative or g has not decreased. If $g(1) \gg g(0)$, divide the components of $\bar{s}$ by some factor (2 or 3) and repeat this step.
4. Compute

$$z = 3\,\frac{g(a) - g(b)}{b - a} + g'(a) + g'(b)$$

5. Compute

$$w = [z^2 - g'(a)g'(b)]^{1/2}$$

6. Compute

$$\alpha_e = b - \frac{g'(b) + w - z}{g'(b) - g'(a) + 2w}\,(b - a)$$

7. If $g(\alpha_e) < g(a)$ and $g(\alpha_e) < g(b)$, accept α_e as the desired minimum value α^*.
8. If $g(\alpha_e) \geqslant g(a)$ or $g'(\alpha_e) \geqslant 0$, repeat steps 4 through 6 using $b = \alpha_e$.
9. Otherwise, repeat steps 4 through 6 using $a = \alpha_e$.

Quadratic Interpolation

1. Calculate Δ, the maximum value of $|\bar{s}_j|$.
2. Divide each component of the vector $\bar{s}$ by Δ.
3. If $g(1) > g(0)$, compute $g(\alpha)$ for $\alpha = \frac{1}{2}, \frac{1}{4}, \ldots$ until $g(\alpha) < g(0)$. Set $a = 0$, $b = \alpha$, and $c = 2\alpha$ and go to step 5.
4. Compute $g(\alpha)$ for $\alpha = 0, 1, 2, 4, 8, \ldots, a, b, c$. Stop the computation at $\alpha = c$ when the present value of $g(\alpha)$ is greater than the last computed value.
5. Compute

$$\alpha_e = \tfrac{1}{2}[g(a)(c^2 - b^2) + g(b)(a^2 - c^2) + g(c)(b^2 - a^2)] \div [g(a)(c - b) + g(b)(a - c) + g(c)(b - a)]$$

6. If $g(\alpha_e) < g(b)$, accept α_e as the desired minimum value α^*; otherwise accept b as the desired value α^*.

The Fletcher-Powell Method

1. Start with a positive definite matrix H_0 (usually chosen as the identity matrix) and an initial point $\bar{x}_0$. The ith step, $i = 0, 1, \ldots$ proceeds as follows.
2. Compute the gradient, $\nabla f(\bar{x}_i)$.
3. Compute the direction:

$$\bar{s}_i = -H_i \nabla f(\bar{x}_i)$$

4. Choose a step length α_i to minimize $g(\alpha) = f(\bar{x}_i + \alpha\bar{s}_i)$. See cubic or quadratic interpolation procedure above.
5. Compute $\bar{\sigma}_i = \alpha_i\bar{s}_i$
6. Compute a new value $\bar{x}_{i+1}$ from the relationship

$$\bar{x}_{i+1} = \bar{x}_i + \alpha_i\bar{s}_i$$

7. Compute

$$\bar{y}_i = \nabla f(\bar{x}_{i+1}) - \nabla f(\bar{x}_i)$$

8. Compute the matrix

$$A_i = \frac{\bar{\sigma}_i\bar{\sigma}_i'}{\bar{\sigma}_i'\bar{y}_i}$$

9. Compute the matrix

$$B_i = \frac{-H_i\bar{y}_i\bar{y}_i'H_i}{\bar{y}_i'H_i\bar{y}_i}$$

10. Compute the successor matrix

$$H_{i+1} = H_i + A_i + B_i$$

11. Check the stop criterion. If it is satisfied, stop. Otherwise, return to step 2, using the successor matrix as the new H_i, and replace i by $i + 1$.

The Conjugate Gradient Method

1. Start with an initial vector of variables $\bar{x}_0$ and an initial direction $\bar{s}_0 = -\nabla f(\bar{x}_0)$. The ith step ($i = 0, 1, 2, \ldots$) proceeds as follows.
2. Choose a step length α_i to minimize

$$g(\alpha) = f(\bar{x}_i + \alpha\bar{s}_i)$$

See cubic or quadratic interpolation procedure above.

TABLE 12-1 (cont'd)

OPTIMIZING UNCONSTRAINED PROBLEMS

3. Compute a new vector of variables, $\bar{x}_{i+1} = \bar{x}_i + \alpha_i \bar{s}_i$ 4. Compute $\nabla f(\bar{x}_{i+1})$. 5. Compute $B_i = \dfrac{\nabla f'(\bar{x}_{i+1})\, \nabla f(\bar{x}_{i+1})}{\nabla f'(\bar{x}_i)\, \nabla f(\bar{x}_i)}$ 6. Compute a successor direction, $s_{i+1} = -\nabla f(x_{i+1}) + \beta_i s_i$ 7. Check a stop criterion. If it is satisfied, stop. Otherwise, return to step 2 and replace i by $i + 1$.	**Powell's Method** 1. For $r = 1, 2, ..., n$, calculate α_r so that $f(\bar{x}_{r-1} + \alpha \bar{s}_r)$ is a minimum (see cubic or quadratic interpolation procedure) and define $\bar{x}_r = \bar{x}_{r-1} + \alpha_r \bar{s}_r$ 2. Find the integer m, $1 \leq m \leq n$, so that $[f(\bar{x}_{m-1}) - f(\bar{x}_m)]$ is a maximum, and define $\Delta = f(\bar{x}_{m-1}) - f(\bar{x}_m)$ 3. Calculate $f_3 = f(2\bar{x}_n - \bar{x}_0)$ and define $f_1 = f(\bar{x}_0)$ and $f_2 = f(\bar{x}_n)$.	4. If $f_3 \geq f_1$ or if $(f_1 - 2f_2 + f_3) \cdot (f_1 - f_2 - \Delta)^2 \geq \frac{1}{2} \Delta \cdot (f_1 - f_3)^2$, or both, use the old directions $\bar{s}_1 \bar{s}_2, ..., \bar{s}_n$ for the next iteration and use $\bar{x}_n$ as the next $\bar{x}_0$. 5. If neither condition in step 4 holds, define $\bar{s} = (\bar{x}_n - \bar{x}_0)$, and calculate α so that $f(\bar{x}_n + \alpha \bar{s})$ is a minimum (see cubic or quadratic interpolation procedure). 6. Use $\bar{s}_1, \bar{s}_2, ..., \bar{s}_{m+1}, \bar{s}_{m+1}, ..., \bar{s}$ as the directions for the next iteration and $\bar{x}_n = \alpha \bar{s}$ for the next $\bar{x}_0$.

6 fit a cubic polynomial to the computed values $g_{(a)}, g'_{(a)}, g_{(b)}$, and $g'_{(b)}$. This polynomial has a unique minimum located at α_e in the interval between a and b. In Step 7, α_e is taken as the desired value of α^* if α_e is a better choice than either a or b. If not, the interpolation is repeated over a smaller interval in Steps 8 and 9.

If derivatives are not available or are difficult to compute, the quadratic interpolation procedure can be used for 1-dimensional minimization. Step 5 of this procedure fits a quadratic polynomial to the three values $g_{(a)}, g_{(b)}$, and $g_{(c)}$. The minimum of this polynomial is located at α_e.

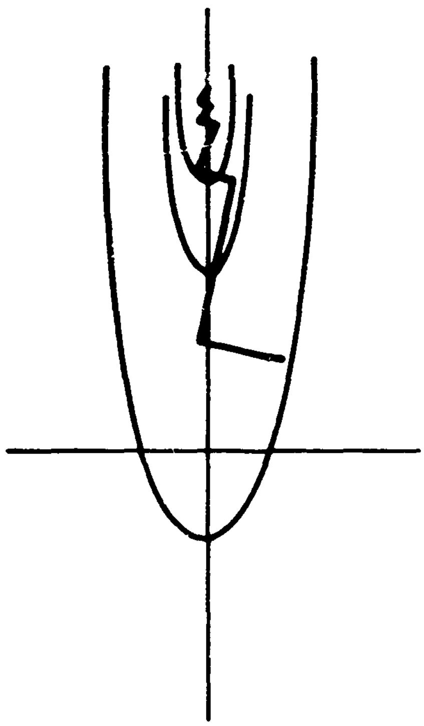
The most that can be guaranteed by the steepest descent method, or any other iterative minimization technique, is that it will find a local minimum, usually the one "nearest" to the starting point x_0. To attempt to find all local minima (and thus the global minimum), the usual approach is to repeat the minimization from many different initial points.

12-2.1.3 Numerical Difficulties

Since successive steps of the method of steepest descent are orthogonal, some functions converge very slowly. If the function contours are circles (or, in the n-dimensional case, hyperspheres), the method finds the minimum in one step. However, for other contours, the gradient direction is generally quite different from the direction to the minimum, and the method produces the inefficient zig-zag behavior shown in Fig. 12-1. Since many, if not most, of the functions occurring in practical applications have eccentric or nonspherical contours, we often must turn to more efficient methods than steepest descent.

12-2.2 SECOND-ORDER GRADIENT METHODS

A number of minimization techniques have been developed to overcome the difficulties of the method of steepest descent. The general notion behind these techniques is that methods which quickly and efficiently minimize a general function must fulfill two criteria. They must work well on a quadratic function, and they must be guaranteed to converge (eventually) for any general function. These criteria are based on the observation that, since the first partial derivatives of a function vanish at the minimum, a Taylor series expansion about the minimum x^* yields



The parabolas are equi-value contours of the objective function $y = 16x_1^2 + (x_2 - 4)^2$. The heavy zig-zag line shows the path taken by a steepest descent procedure seeking the minimum value of this function.

FIGURE 12-1. Finding the Minimum Using the Steepest Descent Method[4]

$$f(\mathbf{x}) = f(\mathbf{x}^*) + \tfrac{1}{2}(\mathbf{x} - \mathbf{x}^*)^T \mathbf{H}_f(\mathbf{x}^*)(\mathbf{x} - \mathbf{x}^*), \qquad (12\text{-}5)$$

where

T indicates the transpose of a matrix

and

$\mathbf{H}_f(\mathbf{x}^*)$ = matrix of second partials of f evaluated at $\mathbf{x}^*$.

$\mathbf{H}_f$ is assumed to be positive definite; thus, the function behaves like a pure quadratic in the vicinity of $\mathbf{x}^*$.

12-2.2.1 Conjugate Directions

Most, if not all, of the newer, more efficient unconstrained minimization procedures are based on the idea of conjugate directions (Refs. 6-8).

The general (positive definite) quadratic function can be written as

$$q(\mathbf{x}) = a + \mathbf{b}^T\mathbf{x} + \mathbf{x}^T \mathbf{A}\mathbf{x} \qquad (12\text{-}6)$$

where the matrix $\mathbf{A}$ is positive definite and symmetric. The procedure for finding the minimum value $q(\mathbf{x}^*)$ consists of starting at some initial point $\mathbf{x}_0$ and taking successive steps along the directions $\mathbf{s}_0, \mathbf{s}_1, \cdots, \mathbf{s}_{n-1}$. All these directions are chosen to be A-conjugate; i.e., for all $i \neq j$, $i, j = 0, 1, \cdots, n-1$, these directions satisfy the relationship

$$\mathbf{s}_i^T \mathbf{A} \mathbf{s}_j = 0 \ . \qquad (12\text{-}7)$$

Successive points in the minimization procedure are computed from

$$\mathbf{x}_{i+1} = \mathbf{x}_i + \alpha_i \mathbf{s}_i. \qquad (12\text{-}8)$$

As in the steepest descent method, the value of the step size α_i is found by minimizing $f(\mathbf{x}_i + \alpha \mathbf{s}_i)$.

It can be shown that, regardless of the starting point, this sequential process leads to the desired minimum value of $q(\mathbf{x}^*)$ in n steps or less (where n is the number of variables in the vector $\mathbf{x}$) (Ref. 8). Thus, conjugate directions minimize a quadratic very efficiently.

12-2.2.2 The Fletcher-Powell Method

The method presented by Fletcher and Powell (outlined in Table 12-1) is probably the most powerful general procedure now known for finding a local minimum of a general function $f(\mathbf{x})$ (Refs. 8 and 9).

Central to the method is a symmetric, positive definite matrix $\mathbf{H}_i$, which is updated at each iteration, and which supplies the current direction of motion $\mathbf{s}_i$ when multiplied by the gradient vector. The numerators $\mathbf{A}_i$ and $\mathbf{B}_i$ in Steps 8 and 9 of the Fletcher-Powell method are both matrices, while the denominators are scalars. Fletcher and Powell have demonstrated that their method will always converge, since the objective function f is initially decreasing along the direction $\mathbf{s}_i$. When the method is applied to a quadratic (Eq. 12-5), the directions $\mathbf{s}_i$ are A-conjugate, and the process converges to a minimum in n steps. The matrix $\mathbf{H}_i$ converges to the inverse matrix $\mathbf{A}^{-1}$ after n steps. When applied to a general function, $\mathbf{H}_i$ tends to become the inverse of the matrix of second partial derivatives of $f(\mathbf{x})$ evaluated at the optimum.

Numerical tests bear out the rapid convergence of this method. Consider, for example, the function

$$f(x_a,x_b) = 100(x_b - x_a^2)^2 + (1 - x_b)^2 \quad (12\text{-}9)$$

This is called the Rosenbrock function (Ref. 10). Its contours are shown in Fig. 12-2. The minimum is at (1,1), and the steep curving valley along $x_b = x_a^2$ makes minimization difficult. The paths taken by the optimum gradient technique and by the Fletcher-Powell method are also in Fig. 12-2. Notice that the Fletcher-Powell technique follows the curved valley and minimizes very efficiently.

Another conjugate direction minimization technique is the conjugate gradient method, outlined in Table 12-1. It requires computation of the gradient of $f(\mathbf{x})$ and storage of only one additional vector, the actual direction of search (Ref. 9). This method is not quite as efficient as the Fletcher-Powell technique but requires much less storage, a significant advantage when the number of variables n is large (Ref. 9).

There are a number of minimization techniques that do not require derivatives. Powell's method seems to be the most efficient of these (Refs. 8 and 9). In this method, outlined in Table 12-1, each iteration requires n 1-dimensional minimizations down n linearly independent directions, $\mathbf{s}_1, \mathbf{s}_2, \cdots, \mathbf{s}_n$. As a result of these minimizations a new direction $\mathbf{s}$ is defined. If a specified test is passed, $\mathbf{s}$ replaces one of the original directions. The process usually is started from the best estimate of the minimizing $\mathbf{x}$ using the initial $\mathbf{s}_i$'s as the reference coordinate directions.

12-3 CONSTRAINED OPTIMIZATION PROBLEMS

In constrained minimization problems, the variables $\mathbf{x}$ may take on only certain allowable values. In Fig. 12-3, for instance, the unshaded area is the set of allowable values of variables x_a and x_b, called the constraint set. This is the set of all points satisfying the inequalities $x_a \geqslant 0$, $x_b \geqslant 0$, $g_1(\mathbf{x}) \geqslant 0$, and $g_2(\mathbf{x}) \geqslant 0$.

A general programming problem may have equality constraints as well as inequality constraints. Equalities often describe the operation of a system, while inequalities define limits within which certain physical variables must lie. Thus the general problem of constrained minimization can be posed as one of minimizing the objective function $f(\mathbf{x})$ subject to inequality and equality constraints:

$$\left.\begin{array}{ll} g_i(\mathbf{x}) \leqslant 0 & i = 1, 2, \cdots, s \\ h_j(\mathbf{x}) = 0 & j = 1, 2, \cdots, r \end{array}\right\} (12\text{-}10)$$

When the functions f, g_i, and h_j are all linear, the problem is one of linear programming; if any of the functions are nonlinear, the programming problem is nonlinear.

Constrained optimization problems are generally more difficult to solve than those without constraints. However, it is sometimes possible to eliminate inequality constraints by appropriate transformations. A number of transformations, as well as sequences of transformation, have been found useful (Ref. 10).

12-3.1 NONLINEAR CONSTRAINTS

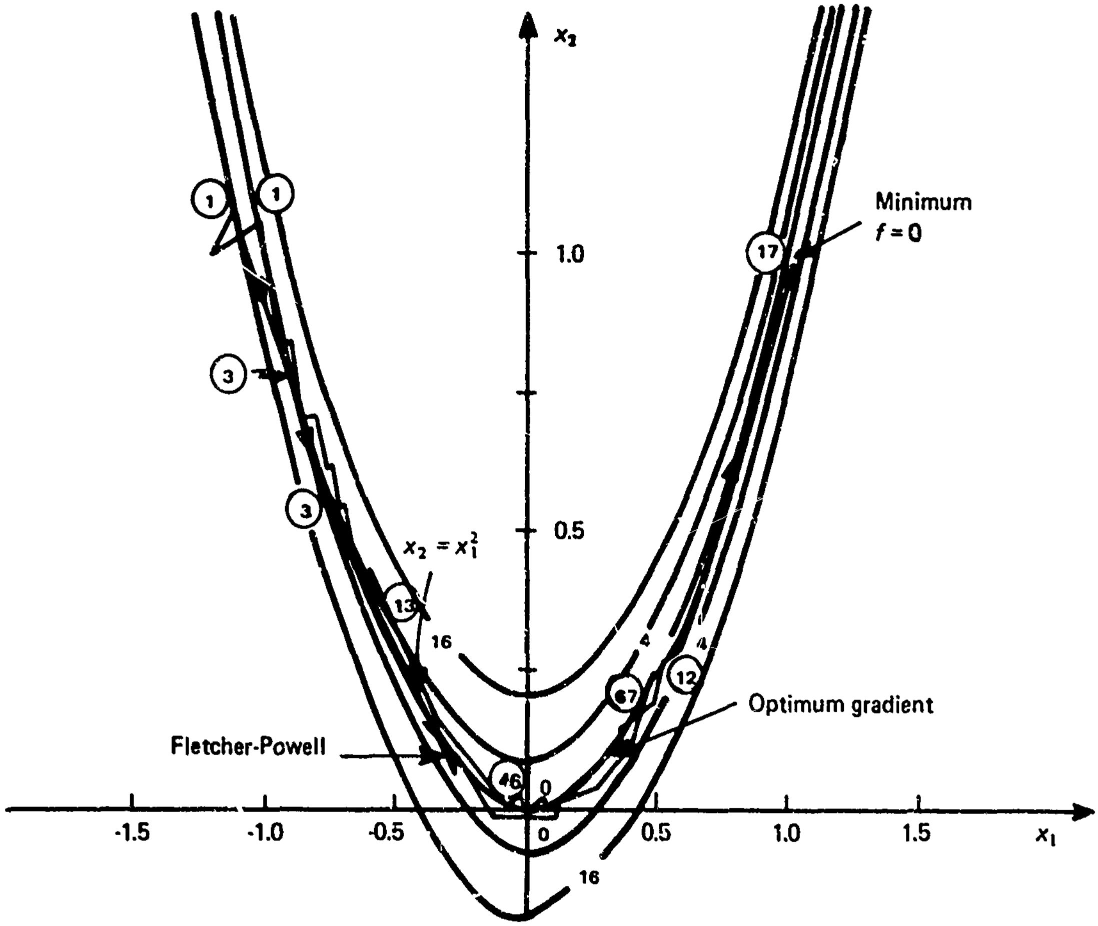
A specific nonlinear programming problem is shown in Fig. 12-4. The constraints are all linear inequalities ($x_1 \geqslant 0$, $x_2 \geqslant 0$, $5 - x_1 - x_2 \geqslant 0$, $-2.5 + x_1 - x_2 \leqslant 0$) which form a constraint set with four corners. The nonlinear objective function, represented by a set of concentric circles, is

$$f(\mathbf{x}) = (x_1 - 3)^2 + (x_2 - 4)^2. \quad (12\text{-}11)$$

The minimum value of $f(\mathbf{x})$ corresponds to the contour of lowest value having at least one point in common with the constraint set. This is the contour labeled $f(\mathbf{x}) = 2$, and the desired solution is at its point of tangency with the constraint set ($x_1^* = 2$, $x_2^* = 3$); this is not a corner point of the set, although it is a boundary point (for linear programs, the minimum is always at a corner point). Fig. 12-5 shows what happens to the problem when the objective function is changed to

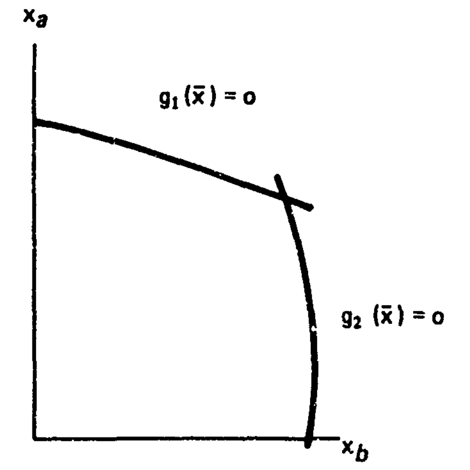
$$f(\mathbf{x}) = (x_1 - 2)^2 + (x_2 - 2)^2. \quad (12\text{-}12)$$

The minimum is now at $x_1^* = 2$, $x_2^* = 2$, which is not even a boundary point of the constraint set. Therefore, this problem could have been solved as an unconstrained minimization of $f(\mathbf{x})$.



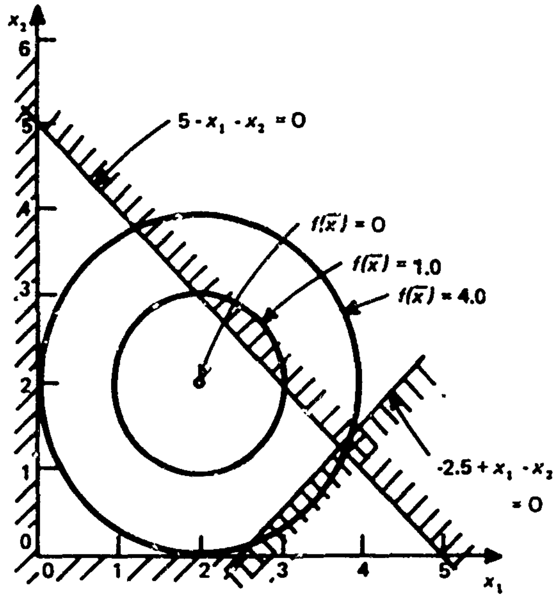
The Fletcher-Powell procedure found the minimum in 17 computational iterations. The optimum gradient technique required 67 iterations.

FIGURE 12-2. Comparison of Fletcher-Powell and Optimum Gradient Techniques for Minimizing a Difficult Function[8]



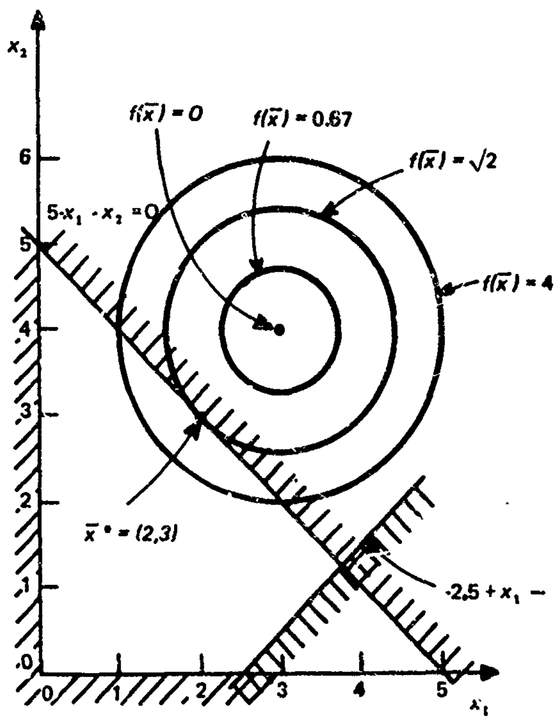
Allowable values for the variables in a problem may be limited or constrained. The area within four boundary curves is called the constraint set.

FIGURE 12-3. Constraint Set



When the minimum value of the objective function is inside the constraint set, the constraint does not affect the solution. Here the point $f(x) = 0$ is the desired minimum value.

FIGURE 12-5. Nonlinear Programming Problem With Objective Function Inside the Constraint Set[4]



Values of the nonlinear objective function, which is to be minimized, are shown as concentric circles. The constrained minimum is one of these lines.

FIGURE 12-4. Nonlinear Programming Problem With Constrained Minimum[4]

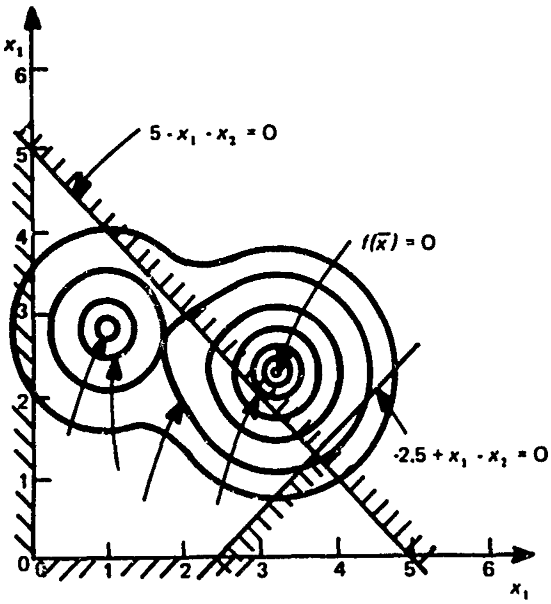
As an example of a nonlinear problem in which local optima occur, consider an objective function with two minima, both of which fall within the constraint set so that there are two local minima. Contours of such a function are like those shown in Fig. 12-6.

The chief nonlinearity in a programming problem often appears in the constraints rather than in the objective function. The constraint set will then have curved boundaries. A problem with nonlinear constraints can very easily have local optima, even if the objective function has only one unconstrained minimum. This is demonstrated in Fig. 12-7, where there is a nonlinear objective function with a nonlinear constraint set that gives local optima at the two points a and b. No point of the constraint set in the immediate vicinity of either point yields a smaller value of $f(\mathbf{x})$.

From these examples we can see that the optimum of a nonlinear programming problem will not necessarily be at a corner point of the constraint set and may not even be on the boundary. In addition, there may be local optima distinct from the global optimum. These properties are direct consequences of the nonlinearity. However, a class of nonlinear problems can be defined which are guaranteed to be free of distinct local optima. These are called convex programming problems. Before some of the specific methods of solving constrained minimization problems are described, the concept of convexity and its implications for nonlinear programming will be discussed.

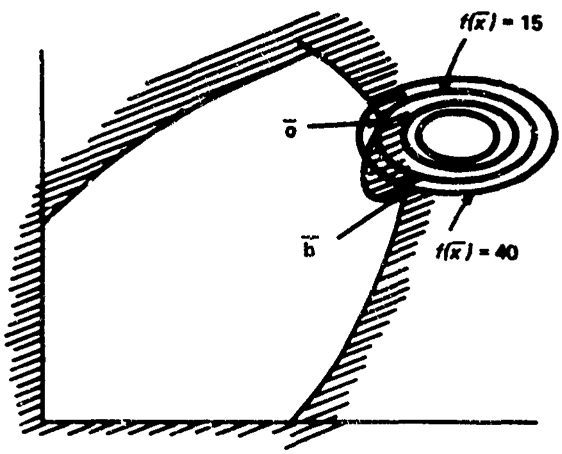
12-3.2 CONVEXITY

There are several reasons why the concepts of convexity and convex functions (which will be defined in this paragraph) are important in nonlinear programming. It is usually impossible to prove that a given procedure will find the global minimum of a nonlinear programming problem unless the problem is convex. Even though there are many real-world problems that are not convex, results obtained under convexity assumptions often can give insight into the properties of more general problems. Sometimes, such results even can be carried over to problems that are not convex, but in a weaker form. In fact, few important mathematical results have been derived in the programming field without assuming convexity.



There may be more than one minimum point within the constraint set. Here, f(x) = 4 and f(x) = 3 are both constrained minima, but f(x) = 4 is only local.

FIGURE 12-6. Local Minimum[4]



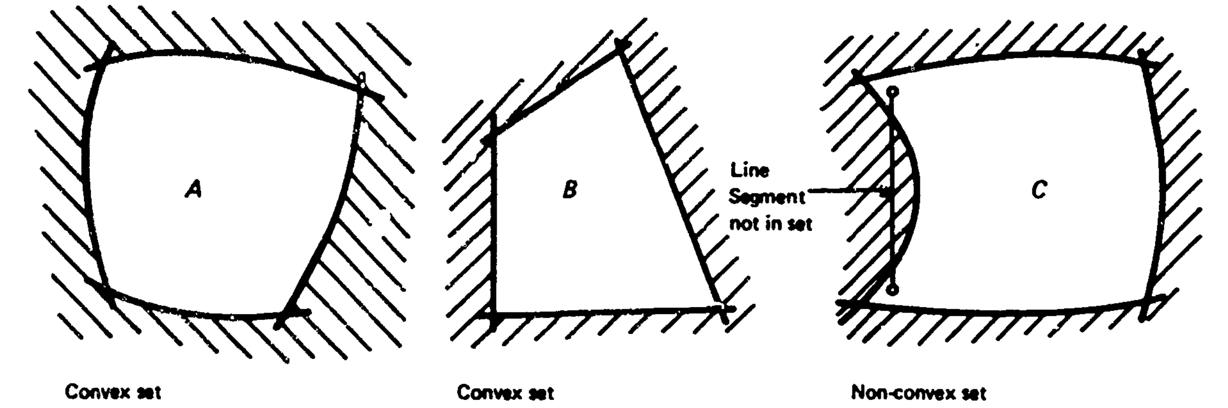
Here the constraint set has curved boundaries which cause the local minimum $f(x)$ to be 40; the global minimum $f(x)$ in this case is 15.

FIGURE 12-7. Local Minima Due to Curved Constraints[4]

Convexity thus plays a role in mathematical programming which is similar to the role of linearity in the study of dynamic systems, where many results derived from linear theory are used in the design of nonlinear control systems.

The main theorem of convex programming is that any local minimum of a convex programming problem is a global minimum. If the problem has a number of points at which the global minimum exists, the set of all such points is convex, and no distinct, separate, local minima with different functional values can exist. This is a very convenient property since it greatly simplifies the task of locating the global minimum.

A set of points is convex if the line segment joining any two of these points remains in the set. In Fig. 12-8, sets A and B are convex, while C is not. A convex set can be



A linear constraint set is always convex.

FIGURE 12-8. Convex and Nonconvex Sets[4]

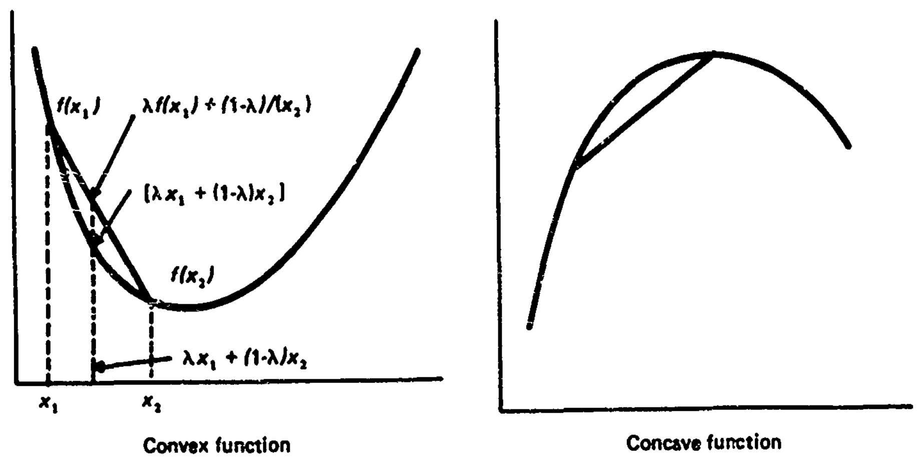
thought of as one whose walls do not bulge inwards. The constraint set of a linear programming problem is always convex.

In the multidimensional case, these geometrical ideas must be formulated in algebraic terms. In particular, the line segment between two points must be defined. If the two points are $\mathbf{x}_1$ and $\mathbf{x}_2$, the segment between them is the set

$$S = \{\mathbf{x} | \mathbf{x} = \lambda \mathbf{x}_1 + (1 - \lambda)\mathbf{x}_2, 0 < \lambda < 1\} . \tag{12-13}$$

If $\lambda = 0$, $\mathbf{x} = \mathbf{x}_2$; if $\lambda = 1$, $\mathbf{x} = \mathbf{x}_1$; as λ varies between these extreme values, $\mathbf{x}$ moves along the line joining $\mathbf{x}_1$ and $\mathbf{x}_2$. This can easily be verified in two or three dimensions.

A function $f(\mathbf{x})$ is convex if the line segment drawn between any two points on the graph of the function never lies below the graph. If the line segment never lies above the graph, the function is concave. Examples of concave and convex functions are shown in Fig. 12-9. The left function is strictly convex, since the line segment is always above the function; the right function is strictly concave. A linear function is both convex and concave, but neither strictly convex nor strictly concave.



A linear function is both convex and concave.

FIGURE 12-9. Concave and Convex Functions[4]

Algebraically, a function $f(\mathbf{x})$ is convex if

$$f(\lambda \mathbf{x}_1 + (1-\lambda)\mathbf{x}_2) < \lambda f(\mathbf{x}_1) + (1-\lambda) f(\mathbf{x}_2) \qquad (12\text{-}14)$$

for all $\mathbf{x}_1, \mathbf{x}_2$ in the (convex) domain of definition of f. The function is strictly convex if the strict inequality holds.

A convex programming problem is one of minimizing a convex function over a convex constraint set. As we mentioned earlier, the main theorem regarding such programs is that any local minimum of a convex programming problem is a global minimum. Furthermore, if there are a number of points at which the global minimum is attained, the set of all such points is convex. Thus, there can be no separated local minima with different functional values. Since most procedures can locate only local minima, these properties are very advantageous. The theorems of convexity (Refs. 11 and 12) listed in Table 12-2 allow this to be done in some cases.

As a consequence of convexity theorems 1 and 2, the problem of minimizing a convex function $f(\mathbf{x})$, subject to r constraints $g_i(\mathbf{x}) > b_i$, $i=1,\cdots,r$ with all g_i convex, is always a convex programming problem. This is true because, from theorem 1, each of the sets

$$\mathbf{R}_i = \{\mathbf{x} | g_i(\mathbf{x}) > b_i\} \qquad (12\text{-}15)$$

is convex. The constraint set R, which is the intersection of all the sets $\mathbf{R}_i$ is also convex by convexity theorem 2.

Since all linear functions are convex, a linear programming problem is always a convex programming problem. This establishes more firmly the geometrically evident fact that a linear program cannot have local optima distinct from the global optimum.

Since convex programs can be identified by determining whether the objective and constraint functions of the problem are convex, it is important to characterize convex functions closely. This can be done by using convexity theorems 3 through 6. Statement b in theorem 3 says that the function, evaluated at any point $\mathbf{x}_1$, never lies below its tangent plane passed through any other point $\mathbf{x}_2$. Theorem 4 is a direct consequence of statement c in theorem 3.

Since $f(\mathbf{x} + \alpha \mathbf{s})$ is the function evaluated at points along the line s passing through the point x, theorem 6 implies that a convex function is convex along any line. This allows us to test to see whether a given function of n variables is not convex, for if any line in n-dimensional space can be found along which $g(\alpha)$ is not convex, then $f(\mathbf{x})$ is not convex either.

12-3.3 MIXED PROBLEMS

Many problems involve both equality and inequality constraints. In such problems, it has been found that the linear function $g(\mathbf{x}) = \mathbf{a}^T \mathbf{x}$ is the only function for which the set

$$\mathbf{R} = \{\mathbf{x} | g(\mathbf{x}) = 0\} \qquad (12\text{-}16)$$

is convex.

Nonlinear functions in two dimensions have graphs that are curved surfaces. If $\mathbf{x}_1$ and $\mathbf{x}_2$ are on the graph and are, therefore, in the constraint set R, then points on the line segment joining $\mathbf{x}_1$ and $\mathbf{x}_2$ will, in general, not lie on the graph (will not be in R). A hyperplane, being "flat", is an obvious exception.

Consider the problem of minimizing $f(\mathbf{x})$ subject to the constraints $g_i(\mathbf{x}) > 0$, $i = 1,\cdots,r$ and $h_j(\mathbf{x}) = 0$, $j = 1,\cdots,s$. From the preceding statements, this may not be a convex programming problem if any of the functions $h_j(\mathbf{x})$ are nonlinear. This, of course, does not preclude efficient solution of such problems, but it does make it more difficult to guarantee the absence of local optima.

In many cases, the equality constraints can be used to eliminate some of the variables, leaving a problem with only inequality constraints and fewer variables. Even if the equalities are difficult to solve analytically (for example, if they are highly nonlinear), it may still be worthwhile to solve them numerically. Such an approach has been used successfully for structural design (Refs. 13 and 14).

12-3.4 THE KUHN-TUCKER CONDITIONS

The most important theoretical results in the field of nonlinear programming are the conditions of Kuhn and Tucker, which must be satisfied at any constrained optimum, local or global, of any linear and of most nonlinear programming problems (Ref. 15). These con-

TABLE 12-2

OPTIMIZING CONSTRAINED PROBLEMS

Convexity Theorems

Theorem 1. If $f(\bar{x})$ is convex, the set

$$R = \{\bar{x}|f(\bar{x}) \leq k\}$$

is convex for all scalars (k).

Theorem 2. The intersection of any number of convex sets is convex.

Theorem 3. If $f(\bar{x})$ has continuous first and second derivatives, the following three statements are all equivalent:

a. $f(\bar{x})$ is convex;
b. $f(\bar{x}_1) \geq f(\bar{x}_2) + \nabla f'(\bar{x}_2)\cdot(\bar{x}_1 - \bar{x}_2)$ for any two points $\bar{x}_1, \bar{x}_2$;
c. the matrix of second partial derivatives of $f(\bar{x})$ is positive semidefinite for all points $\bar{x}$.

Theorem 4. A positive semidefinite quadratic form is convex.

Theorem 5. A positive linear combination of convex functions is convex.

Theorem 6. A function $f(\bar{x})$ is convex if and only if the one-dimensional function $g(\alpha) = f(\bar{x} + \alpha\bar{s})$ is convex for all fixed $\bar{x}$ and $\bar{s}$.

Zoutendijk's Method of Feasible Directions

1. Start with an initial point x_0 which satisfies all constraints. For $i = 0, 1, \ldots$, do the following steps.
2. At the current point, x_i, determine which constraints are binding (or almost binding) and form the set I containing their indices.
3. Choose a set of $\theta_i (0 < \theta < 1)$ used to steer away from nonlinear constraint boundaries.
4. Compute a new usable feasible direction, $\bar{s}_i$, by solving the direction-finding problem of minimizing ξ subject to the conditions

$$\nabla g_j'(\bar{x}_i)\bar{s} + \theta_j\xi \geq 0$$
$$\nabla f'(\bar{x}_i)\bar{s} - \xi \geq 0$$
$$\bar{s}'\bar{s} = 1$$

If the minimum value of $\xi \geq 0$, no such direction exists and the computation is terminated. The current point is generally a local constrained minimum. If $\xi < 0$ proceed to step 5.

5. Compute a step length α_i by minimizing $f(\bar{x}_i + \alpha\bar{s}_i)$ subject to the condition that $\bar{x}_i + \alpha\bar{s}_i$ violates no constraints.
6. Using α_i, compute a successor point $\bar{x}_{i+1} = x_i + \alpha_i\bar{s}_i$ and return to step 2 with i replaced by $i + 1$.

Rosen's Gradient-Projection Method

1. Start at a point x_0 that satisfies the constraints. The ith iteration, $i = 0, 1, \ldots$ proceeds as follows:
2. Compute $\nabla f(\bar{x}_i)$.
3. Determine which constraints are binding at $\bar{x}_i$ and call these the constraints associated with $\bar{x}_i$.
4. Compute $\bar{s}_i$, the projection of $-\nabla f(\bar{x}_i)$, on the intersection of the constraints associated with the point $\bar{x}_i$.
5. If $\bar{s}_i$ is not the zero vector, compute a step length α_i by minimizing $g(\alpha) = f(\bar{x}_i + \alpha\bar{s}_i)$ subject to the condition that $\bar{x}_i + \alpha\bar{s}_i$ violates no constraints. This determines a new point $\bar{x}_{i+1} = \bar{x}_i + \alpha_i\bar{s}_i$. Return to step 2 and replace i with $i + 1$.
6. If $\bar{s}_i$ is zero, then

$$\nabla f(\bar{x}_i) = \sum_j u_j\bar{\alpha}_j$$

which is a linear combination of normals $\bar{\alpha}_j$ to the binding constraint planes.

7. If all $u_j \geq 0$, then $\bar{x}_i$ is the solution of the problem, for the Kuhn-Tucker conditions are satisfied.
8. Otherwise, define a new set of planes to be associated with $\bar{x}_i$ by deleting from the present set one plane for which $u_j < 0$, and return to step 4.

The Fiacco-McCormick Conditions

1. The interior of the constraint set is non-empty.
2. The functions f and g_i are twice continuously differentiable.
3. The set of points in the constraint set for which $f(\bar{x}) \leq k$ is bounded for all $k < \infty$.
4. The function $f(\bar{x})$ is bounded below for $\bar{x}$ in the constraint set.

- If conditions 1 through 4 hold, at least one finite local minimum of $P(\bar{x},r)$ [see Eq (24)] exists within the constraint set for any $r > 0$. Furthermore, f is monotonically nonincreasing as r is reduced (Ref. 25).

5. $f(\bar{x})$ is convex.
6. The $g_i(\bar{x})$ are concave functions.
7. $P(\bar{x},r)$ is strictly convex in the interior of the constraint set for any $r > 0$.

- If conditions 5 though 7 also hold, there is a convex programming problem; any local minimum is global, and the procedure converges to the global minimum as $r \to 0$.

The Fiacco-McCormick Method

1. Start with $\bar{x}_0$, which must be strictly inside the constraint set, and $r_1 > 0$. Let $i = 1, 2, \ldots$.
2. Minimize $P(\bar{x}, r_i)$, starting from $\bar{x}_{i-1}$, and subject to no constraints.
3. Reduce r by choosing $r_{i+1} < r_i$, and return to step 2 with i replaced by $i + 1$.
4. Stop if the change in the objective function fails to exceed a specified value for some predetermined number of iterations.

ditions form the basis for the development of many computational procedures. In addition, the criteria for stopping many procedures (i.e., for recognizing when a local constrained optimum has been achieved) are derived directly from these conditions.

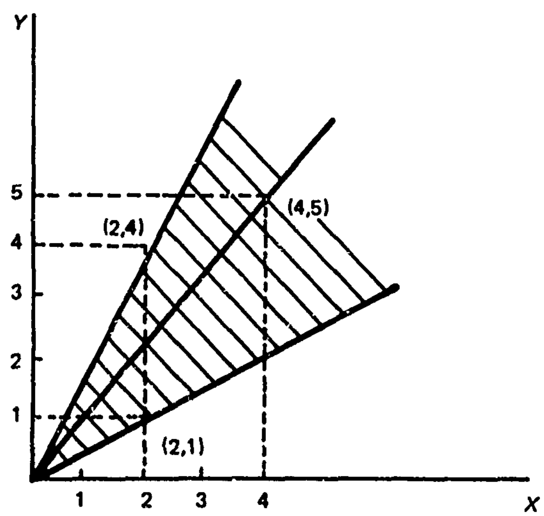
The concept of a cone can be used to help visualize the Kuhn-Tucker conditions. A cone is defined as a set of points R such that, if x is in R, λx is also in R for $\lambda \geqslant 0$. A convex cone R has the additional property that if x and y are in R, x + y is also in R. The set of all non-negative linear combinations of a finite set of vectors forms a convex cone; i.e., the set R is a convex cone, where

$$R = \{x | x = \lambda_1 x_1 + \cdots + \lambda_m x_m ; \quad \lambda_i \geqslant 0; i=1,\ldots, m\}. \tag{12-17}$$

The vectors $x_1, x_2, \cdots, x_m$ are called the generators of the cone. For example, the convex cone of Fig. 12-10 is generated by the vectors (2,1) and (2,4). Any vector that can be expressed as a non-negative linear combination of these vectors lies in this cone. In Fig. 12-10 the vector (4,5) in the cone is given by

$$(4,5) = 1 \cdot (2,1) + 1 \cdot (2,4). \tag{12-18}$$

The Kuhn-Tucker conditions are predicated on the fact that at any constrained optimum, no small, allowable change in the problem variables can improve the objective function. To illustrate this, consider the nonlinear programming problem shown in Fig. 12-11. It is evident that the optimum is at the intersection of the two constraints. At (1,1) in Fig. 12-11 the set of all feasible directions lies between the line $-x - y + 2 = 0$ and the tangent line $y = 2x - 1$. In other words, this set is the cone generated by these two lines. The vector $-\nabla f$ points in the direction of the maximum rate of decrease of the objective function $f(x,y)$. A move along any direction making an angle of less than 90 deg with $-\nabla f$ will decrease $f(x,y)$. Thus, at the optimum, there can be no feasible direction with an angle of less than 90 deg between it and $-\nabla f$.



The shaded area represents a cone generated by vectors (2,1) and (2,4).

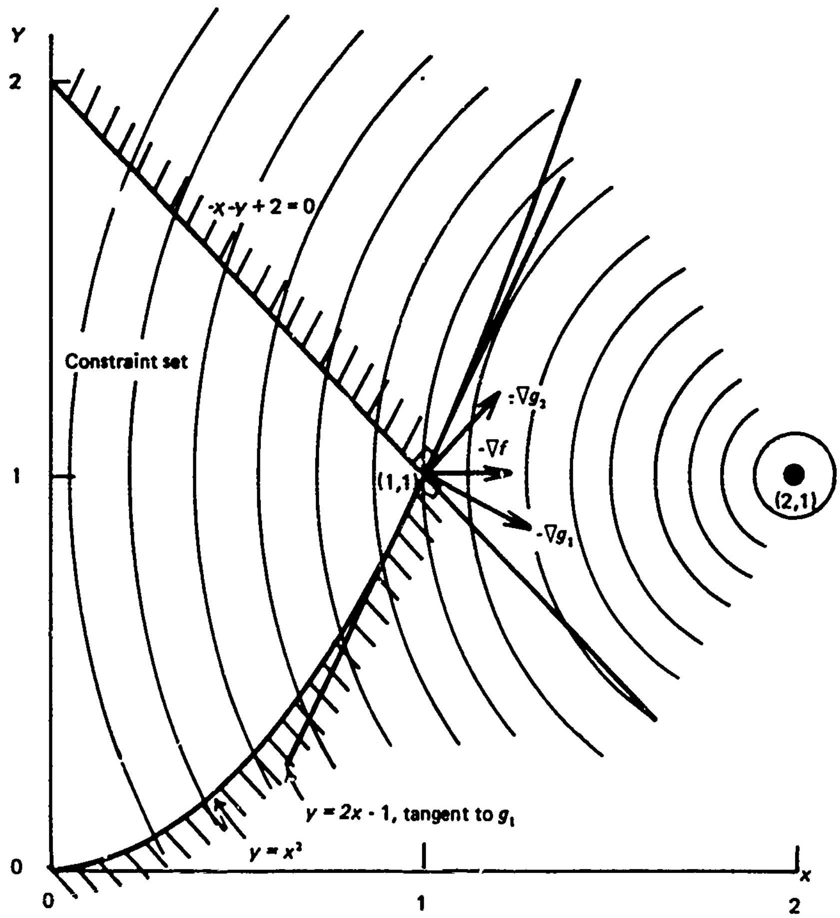
FIGURE 12-10. Convex Cone

The negative gradients $-\nabla g_1$ and $-\nabla g_2$ are also shown in Fig. 12-11; and $-\nabla f$ is contained in the cone generated by these negative gradients. If $-\nabla f$ were not contained in the cone, but slightly above $-\nabla g_2$, it would make an angle of less than 90 deg with a feasible direction just below the line $-x - y + 2 = 0$. Similarly, if $-\nabla f$ were slightly below $-\nabla g_1$, it would make an angle of less than 90 deg with a feasible direction just above the line $y = 2x - 1$. Neither of these cases can occur at an optimum point, and both cases are excluded if and only if $-\nabla f$ lies within the cone generated by $-\nabla g_1$ and $-\nabla g_2$. This is the geometric statement of the Kuhn-Tucker conditions; a necessary condition for x to minimize $f(x)$, subject to the constraints $g_i(x) > 0$ where $i=1,\cdots,r$, is that the gradient ∇f lie within the cone generated by the gradients of the binding constraints.

In an algebraic statement of the Kuhn-Tucker conditions, since ∇f lies within the cone described, it must be a nonnegative linear combination of the gradients of the binding constraints. In other words, there must exist numbers $u_i \geqslant 0$ such that

$$\nabla f(x^*) = \sum_{i=1}^{p} u_i \nabla g_i(x^*) \tag{12-19}$$

where the binding constraints are assumed to be $g_i, \cdots, g_p, (p \leqslant r)$. This relationship can be extended to include all constraints by defining the coefficient u_i to be zero if $g_i(x^*) > 0$.



The objective function is shown by concentric circles, and the constrained minimum is clearly at the point (1,1). All feasible directions at this point are obtained in the cone generated by the gradients $-\nabla g_1$ and $-\nabla g_2$, which are normal to the constraint boundaries.

FIGURE 12-11. Nonlinear Program Illustrating the Use of a Cone[4]

If this is done, the product $u_i g_i(\mathbf{x}^*)$ is zero for all i. Eq. 12-19 is the form in which the Kuhn-Tucker conditions usually are stated.

If a minimization problem with inequality constraints is a convex programming problem whose constraint set has a nonempty interior, the Kuhn-Tucker conditions are both necessary and sufficient for a point $\mathbf{x}$ to be a constrained minimum (Ref. 15).

Most existing nonlinear programming methods can be classified either as methods of feasible direction (such as Zoutendijk's procedure and Rosen's gradient projection method) or as penalty function techniques (such as the Fiacco-McCormick method).

12-3.5 METHODS OF FEASIBLE DIRECTIONS

Methods of feasible directions use the same general approach as the techniques of unconstrained minimization, but they are constructed to deal with inequality constraints. The idea is to pick a starting point that satisfies the constraints, and then to find a direction along which a small move violates no constraint and, at the same time, improves the objective function. We then move some distance in the selected direction, obtaining a new and better point, and repeat the procedure until we reach a point from which the objective function cannot be improved without violating at least one constraint. In general, such a point is a constrained local minimum of the problem, not necessarily a global minimum for the entire region of interest.

A direction along which a small move can be made without violating any constraints is called a feasible direction, while a direction which is feasible and at the same time improves the objective function is called a usable, feasible direction. Since there are many ways of choosing such directions, there are many different methods-of-feasible-directions.

An iterative procedure of this type is illustrated in Fig. 12-12. The starting point is $\mathbf{x}_0$, and the usable, feasible direction chosen is

$$\mathbf{s}_0 = -\nabla f(\mathbf{x}_0) \quad . \qquad (12\text{-}20)$$

The procedure is to choose the distance moved along $\mathbf{s}_0$ so as to minimize f, and the first improved point is $\mathbf{x}_1$. Here, a problem arises: proceeding in the negative gradient direction at $\mathbf{x}_1$ would violate the constraints. There are many feasible directions in which we could move from $\mathbf{x}_1$; any direction pointing into the constraint set or along a constraint boundary would do. The "best" direction we can choose, however, is that feasible direction along which $f(\mathbf{x}_1)$ decreases most rapidly, i.e., along which $-\mathbf{s}_1^T\nabla f(\mathbf{x}_1)$ is minimized. This is the feasible direction that makes the smallest angle with $-\nabla f(\mathbf{x}_1)$, and is the projection of $-\nabla f(\mathbf{x}_1)$ on the constraint boundary.

The farthest we can move along $\mathbf{s}_1$ without crossing the constraint boundary is to the point $\mathbf{x}_2$. Repeating the smallest angle procedure leads us to $\mathbf{x}_3$ with negative gradient $-\nabla f(\mathbf{x}_3)$. At this point there is no usable feasible direction, since no feasible direction at $\mathbf{x}_3$ makes an angle of less than 90 deg with $-\nabla f(\mathbf{x}_3)$. In this case, $\mathbf{x}_3$ happens to be at the global minimum of $f(\mathbf{x})$ over the constraint set.

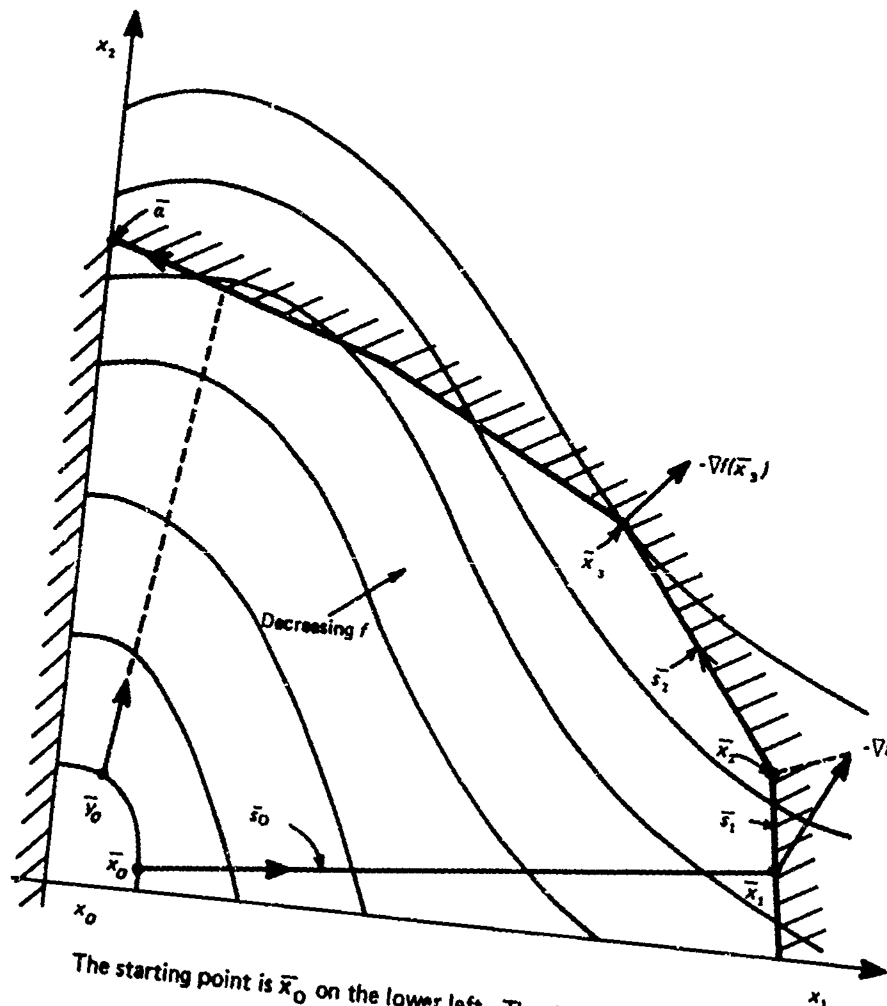
The global minimum is not, however, always reached by this procedure. In this example, the same procedure, starting with $\mathbf{y}_0$ in Fig. 12-12, leads to a local minimum at the point a, which is distinct from the global minimum at $\mathbf{x}_3$. This example illustrates the difficulties such procedure may encounter with local optima. These difficulties are common to all methods, and one can be sure of avoiding them only for a convex programming problem.

12-3.5.1 Zoutendijk's Procedure

Consider the problem of minimizing $f(\mathbf{x})$, subject to the inequality constraints $g_i(\mathbf{x}) > 0$; $i=1,\cdots,m$. If a starting point $\mathbf{x}_0$ that satisfies the constraints is assumed, the problem is to choose a vector $\mathbf{s}$ which is both usable and feasible. Let $\mathbf{I}$ be a set of indices i, for which $g_i(\mathbf{x}_0) = 0$. For all feasible vectors $\mathbf{s}$, a small move along the vector from $\mathbf{x}_0$ makes no binding constraint negative; i.e., for all i in the set $\mathbf{I}$,

$$\frac{d}{d\alpha}\left[g_i(\mathbf{x}_0 + \alpha\mathbf{s})\right]\Bigg|_{\alpha=0} = \nabla g_i^T(\mathbf{x}_0)\mathbf{s} \geqslant 0 \qquad (12\text{-}21)$$

where α is the scalar parameter that determines how far along $\mathbf{s}$ one might go. A usable,



The starting point is $\bar{x}_0$ on the lower left. The desired global minimim is at $\bar{x}_3$.

FIGURE 12-12. Constrained Minimization With Usable, Feasible Directions[4]

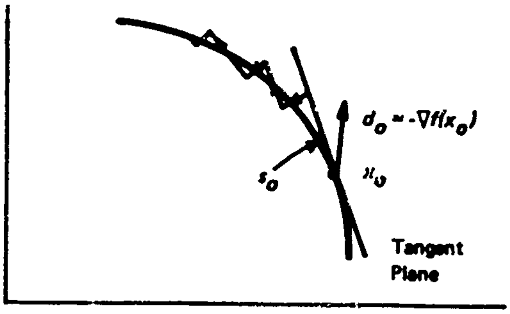
feasible vector has the additional property that

$$\frac{d}{d\alpha}\left[f(\mathbf{x}_0 + \alpha \mathbf{s})\right]\Big|_{\alpha=0} = \nabla f^T(\mathbf{x}_0)\mathbf{s} < 0 \quad (12\text{-}22)$$

Therefore, the function initially decreases along such a vector.

In searching for a "best" vector $\mathbf{s}$ along which to move, we could choose the feasible vector that minimizes $\nabla f^T(\mathbf{x}_0)\mathbf{s}$. However, if some of the binding constraints were nonlinear, this could lead to the difficulty shown in Fig. 12-13. Here, the feasible direction $\mathbf{s}_0$ that minimizes $\nabla f^T(\mathbf{x}_0)\mathbf{s}$ is the projection of $-\nabla f(\mathbf{x}_0)$ on the tangent plane through the starting point $\mathbf{x}_0$. Since the constraint surface is curved, movement along $\mathbf{s}_0$ for any finite distance violates the constraint. Thus, a recovery move must be made to come back inside the constraint set. Repetitions of this procedure lead to inefficient zig-zagging. Therefore, to avoid zig-zagging, it is wise to choose a locally "best" direction that moves away from the boundaries of the nonlinear constraints as it decreases the objective function.

An algorithm using Zoutendijk's direction finding procedure is given in Table 12-2. Step 5 is almost the same as in the unconstrained case. It is still desirable to minimize the objective function along the vector $\mathbf{s}$, but now no constraint may be violated. The cubic or quadratic interpolation procedures of Table 12-1, modified to account for constraints, may be used to compute α_i. For convex programs, Zoutendijk's method converges to the global minimum (Ref. 12).



The zig-zag motion shown here is time-consuming and can be avoided by using Zoutendijk's minimization procedure.

FIGURE 12-13. An Inefficient Search Procedure[4]

12-3.5.2 Rosen's Gradient Projection Method

At each iteration of Zoutendijk's procedure, an optimization problem must be solved to find a direction in which to move. Although this direction is in some sense "best", the procedure can be time-consuming. An alternative is provided by Rosen's gradient projection method, where a usable, feasible direction is found without solving an optimization problem (Ref. 16). This direction, however, may not be locally "best" in any sense. Rosen's method, probably most efficient when all constraints are linear, uses the Kuhn-Tucker conditions both to generate new directions and as a stop criterion.

12-3.6 PENALTY FUNCTION TECHNIQUES

12-3.6.1 General

Since powerful methods are available for unconstrained minimization, it would seem convenient to solve constrained problems using unconstrained methods. This is exactly what a "penalty function" allows us to do.

Instead of dealing with the constraints directly, penalty function techniques find the unconstrained minimum of the function

$$\psi(\mathbf{x}) = f(\mathbf{x}) + \sum_{i=1}^{m} \phi[g_i(\mathbf{x})] \tag{12-23}$$

where $\phi[\cdot]$ is the penalty function, yet to be determined. For example, suppose that the penalty function is $\phi_0(y)$, where $\phi_0(y) = 0$ for $y \geqslant 0$, and $\phi(y) \to \infty$ for $y < 0$. If all constraints $g_i(\mathbf{x}) \geqslant 0$ in Eq. 12-23 are satisfied, the summation term contributes nothing and minimizing ψ is equivalent to minimizing f. If any g_i is less than zero, $\phi_0(g_i) \to \infty$ which is certainly not anywhere near the minimum of $\psi(\mathbf{x})$; thus, the summation term "penalizes" any violation of the constraints. Any procedure which minimizes ψ will never select a point outside the constraint set and will, in fact, select that point of the constraint set that minimizes $f(\mathbf{x})$.

Unfortunately, there are certain difficulties that must be overcome in order to use this powerful technique. To illustrate them, consider the problem of minimizing $x_a^2 + x_b^2$ subject to the constraint $x_a \geqslant 3$; $\mathbf{x} = (x_a, x_b)$. We know in advance that the solution to this problem is $x_a = 3, x_b = 0$. For this example,

$$\psi(\mathbf{x}) = x_a^2 + x_b^2 + \phi_0(x_a - 3) \tag{12-24}$$

Contours of ψ in the feasible region (to the right of the line $x_a = 3$) are circles with center at the origin, and the penalty term $\phi_0(x_a - 3)$ has no effect. Just to the left of $(x_a - 3)$, ψ becomes unbounded, so that as soon as we move to the left from $x_a = 3$, we immediately cross all the contours of constant value. A gradient minimization procedure starting at $\mathbf{x}_0$ would move to the boundary at $\mathbf{x}_1$ and could proceed no further. In fact, since the function ψ is discontinuous and has no derivative along $x_a = 3$, minimization is almost hopeless.

These difficulties may be relieved by defining other, less "harsh" penalty functions. For example, the function $\phi_1(y)$, where $\phi_1(y) = 0$ for $y \geqslant 0$ and $\phi_1(y) = ky^2$ for $y < 0$, is continuous and has continuous first derivatives for all values of y, $(k > 0)$. If ϕ_1 is used, the penalty for constraint violations is no longer infinite and some violations are possible.

Consider applying this new penalty function to the previous problem by minimizing

$$\psi(\mathbf{x}) = x_a^2 + x_b^2 + \phi_1(x_a - 3). \tag{12-25}$$

The contours of this function to the right of $x_e = 3$ are circular but to the left they are elongated ellipses, showing the same bunching effect as before. This effect gets worse as k increases.

A gradual approach is more practical. Rather than solve only one unconstrained problem, we solve a sequence of such problems, each one bringing us closer to the final solution. For example, we can solve the problem with a small value of k. Then, using that solution as a starting point, choose a larger value of k and re-solve the problem. Repeat the procedure several times. In general, the sequence of unconstrained minima approaches the solution of the original constrained problem.

When the penalty function ϕ_1 is used, intermediate solutions usually violate the constraints. Thus, the method approaches the constrained minimum from outside the constraint set. In many cases, this may be unsatisfactory. If small violations of the constraints are not permitted, intermediate solutions often cannot be used. The method is inefficient if the objective or constraint functions are ill-behaved exterior to the constraint set. Moreover, the approach cannot be used at all when any of these functions is not defined outside of the constraint set.

12-3.6.2 The Fiacco-McCormick Method

The Fiacco-McCormick method avoids the difficulties we just described by approaching the optimum from inside the constraint set (Refs. 17 and 18). To use this method, we first define the function

$$\Psi(\mathbf{x}, r) = f(\mathbf{x}) + r \sum_{i=1}^{m} \frac{1}{g_i(\mathbf{x})} \qquad (12\text{-}26)$$

where $r > 0$. Let $r_1 \to 0$ and choose $\mathbf{x}_0$ inside the constraint set. In the problem of minimizing $\Psi(x, r_1)$ starting from $\mathbf{x}_0$ and subject to no constraints, a minimum must exist inside the constraint set, since $\Psi(\mathbf{x}, r_1) \to \infty$ on the boundary of this set (because some $g_i(x) = 0$). Thus, the path of steepest descent leading from the point $\mathbf{x}_0$ (a path on which $\Psi(\mathbf{x}, r_1)$ is strictly decreasing) cannot penetrate the boundary of the constraint set. The minimizing point depends, of course, on the choice of r_1, and is denoted by $\mathbf{x}(r_1)$. By this reasoning, $\mathbf{x}(r_1)$ will always be inside the constraint set.

If this minimization process is repeated for a sequence of values $r_1 > r_2 > \cdots r_k > 0$, each minimizing point $\mathbf{x}(r_i)$ also will be strictly inside the constraint set. Furthermore, as the value of r is reduced, the influence of the term which "penalizes" closeness to the constraint boundaries (the last term in Eq. 12-26) also is reduced and, in minimizing $\Psi(\mathbf{x}, r)$, more effort is concentrated on reducing the $f(\mathbf{x})$ term. Thus, the sequence of points $\mathbf{x}(r_1), \mathbf{x}(r_2), \cdots$ can come as close as necessary to the boundary of the constraint set. We would expect that as r approaches zero, the minimizing point $\mathbf{x}(r)$ approaches the solution of the original problem of minimizing $f(\mathbf{x})$ subject to the constraints $g_i \geqslant 0$.

This method is particularly attractive in dealing with problems that have markedly nonlinear constraints, since it approaches the solution value from inside the constraint set. Motion along the boundaries of this set, which can be very cumbersome when the boundaries have large curvature, is completely avoided.

Fiacco and McCormick have shown that all the previous conjectures are true under certain conditions (see Table 12-2). Condition 7 is not implied by conditions 5 and 6, but only small additional requirements on f and g_i are needed for it to hold (Ref. 16).

The Fiacco-McCormick procedure is given in Table 12-2. Step 2 may be accomplished by any of the unconstrained minimization procedures in this paragraph. In Step 3, r ought to be reduced by dividing each time by the same factor.

12-4 DYNAMIC PROGRAMMING

Dynamic programming is a general approach for solving a sequential decision process. Optimization is merely one kind of sequential decision process. This topic is not grasped easily from a short exposition, nor is it often practical for reliability problems, except when the problems can as easily be solved another way. Therefore, several references (Refs. 19-22, 34) are given for further study, should the need arise.

Dynamic programming suffers from a major drawback–dimensionality. Problems with two or three state variables may be solved with increasing difficulty, and solution with more than three state variables is very difficult. This is because the functions $f_i(h)$, where h is the state vector of dimension k, must be tabulated over a k-dimensional grid. If each dimension has 10 subdivisions, this requires the storage of 10^k numbers, which generally exceeds the fast memory space of most computers for $k > 4$. Any increase in k is then quite difficult and can be accomplished only by trading memory space for computation time.

12-5 LUUS-JAAKOLA METHOD

Luus and Jaakola developed a very simple method for optimization by direct-search and interval-reduction, Refs. 35 and 36. It is extremely simple to program, evaluates no derivatives, does not invert any matrices and can handle inequality constraints. Equality constraints are presumed to have been eliminated by usual methods.

For integer problems, e.g., parallel redundancy, Luus has extended the method, again in a very simple way (both programming and conceptually), see Ref. 36. Especially for the novice, but even for the high-powered theorists, this method has a great deal of appeal and utility. Ref. 36 is reproduced as Appendix A.

12-6 APPLICATIONS

It is difficult to find good nontrivial applications of complicated reliability optimization in the literature. Generally, in the literature, the analyst has to make too many unrealistic assumptions, or picks a problem no one in practice is really going to care about. For example, cost and weight are usually major real constraints; but there is not a continuum of equipments available with reliability tabulated as functions of cost and weight. Solving for optimum parallel redundancy in the presence of constraints is another favorite problem. But rarely are there more than a few redundant units; so the calculations could easily be carried out for all feasible combinations.

One ought to be concerned with the region around the optimum point. If it is very flat, then it makes little difference where, in the flat region, one chooses a solution. There are usually many important variables, mostly qualitative, that are left out of the formal analysis. These may well determine where in the flat region one chooses the solution.

If there are a great many independent variables, it is difficult to visualize the "space" in which the problem is to be solved. The ramifications of assumptions and solutions are difficult to grasp. Therefore, most big problems ought to be reduced to a series of little ones whose meaning can be comprehended. If necessary, one can go back after the first trial solutions and modify the way the little problems were formulated.

Perhaps the biggest difficulty of all with optimizing a very large problem is that when it is finished, people tend to be extremely pleased and impressed. They tend to believe that they now know the answer to some real-world problem. But they don't. What they do know is the answer to a mathematical problem which contains gross approximations (to be tractable) and which was solved with guessed-at data. Since "no one" can understand the whole problem at once, there is a tendency to grasp the computerized solution like a drowning man grasping at straws.

Obviously, some very complicated problems have been solved by optimization techniques. These tend to be problems where plant process operation is quite well known, but where the magnitude of the calculation is just too much. The models themselves tend to be rather simple in concept; their complexity comes from their scope.

Some journal articles which apply optimization techniques are Refs. 22-33; Ref. 33 is a relatively new approach. Anyone who wishes to apply optimization techniques to complicated reliability engineering problems ought to find professional assistance from people who are skilled in using the available computer programs. To begin from scratch is usually to waste inordinate amounts of time and money, except that the Luus-Jaakola method (par. 12-5) can be used by almost anyone--conceptually and practically it's so simple.

REFERENCES

1. S. I. Gass, *Linear Programming: Methods and Applications*, McGraw-Hill Book Co., Inc., N.Y., 1958.
2. G. D. Dantzig, *Linear Programming and Extensions*, Princeton University Press, Princeton, N.J., 1963.
3. E. B. Hildebrand, *Methods of Applied Mathematics*, Prentice-Hall, Inc., Englewood Cliffs, N.J., 1952.
4. L. S. Lasdon and A. D. Warren, "Mathematical Programming for Optimal Design", *Electro-Technology*, 55-70 (November 1967).
5. H. Curry, "Methods of Steepest Descent for Non-Linear Minimization Problems", *Quart. Appl. Math.* No. 2, 258-61 (1954).
6. "Function Minimization by Conjugate Gradients", Brit. Computer J., 7, 149-54 (1964).
7. R. Fletcher and M. J. D. Powell, "A Rapidly Convergent Descent Method for Minimization", Brit. Computer J., 6, 163-68 (1963).
8. M. J. D. Powell, "An Efficient Method for Finding the Minimum of a Function of Several Variables Without Calculating Derivatives", Brit. Computer J., 7, 155-62 (1964).
9. M. J. Box, "A Comparison of Several Current Optimization Methods and the Use of Transformations in Constrained Problems", Brit. Computer J., 9, 67-68 (1966).
10. H. H. Rosenbrock, "An Automatic Method for Finding the Greatest or Least Value of a Function", Brit. Computer J., 3, 175 (1960).
11. C. Hadley, *Nonlinear and Dynamic Programming*, Addison-Wesley Publishing Co., Reading, Mass., 1964.
12. G. Zoutendijk, *Methods of Feasible Directions*, American Elsevier Publishing Co., Inc., N.Y., 1960.
13. R. Fox and K. Wilmert, "Optimum Design of Curve-Generating Linkages with Inequality Constraints", *Trans. ASME, J. Engineering, for Industry* (February 1967).
14. R. Fox and L. Schmit, "Advances in the Integrated Approach to Structural Synthesis", Spacecraft and Rockets, 3, No. 6, 858 (June 1966).
15. H. W. Kuhn and A. W. Tucker, "Nonlinear Programming", *Proc. Second Berkeley Symp. on Mathematical Statistics and Probability*, Berkeley, Calif., 1950, pp. 481-92.
16. J. B. Rosen, "The Gradient Projection Method for Nonlinear Programming. Part I – Linear Constraints", *J. Soc. Industrial and Applied Mathematics*, No. 8, 181-217 (1960).

17A. A. V. Fiacco and G. P. McCormick, "The Sequential Unconstrained Minimization Technique for Nonlinear Programming, A Primal Dual Method", Management Science, 10, No. 2, 360-66 (1964).

17B. A. V. Fiacco and G. P. McCormick, "Computational Algorithm for the Sequential Unconstrained Minimization Technique for Nonlinear Programming", Management Science, 10, 601-17 (July 1964).

17C. A. V. Fiacco and G. P. McCormick, "Extension of SUMT for Nonlinear Programming: Equality Constraints and Extrapolation", Management Science, 12, No. 11, 816-29 (July 1966).

18A. A. V. Fiacco and G. P. McCormick, *Programming Under Nonlinear Constraints by Unconstrained Minimization. A Primal-Dual Method*, RAC Tp-96, The Research Analysis Corp., Bethesda, Md., September 1963.

18B. A. V. Fiacco and G. P. McCormick, *Nonlinear Programming Sequential Unconstrained Minimization Techniques*, Wiley, N.Y., 1968.

19. R. Bellman, *Adaptive Control Processes*, Princeton University Press, Princeton, N.J., 1961.
20. R. E. Bellman, *Dynamic Programming* Princeton University Press, Princeton, N.J., 1957.
21. R. E. Bellman, "Dynamic Programming and Language Multiplier", Proceedings National Academy of Sciences, 42, 767-69 (1956).
22. F. A. Tillman, C. L. Huang, L. T. Fan, K. C. Lai, "Optimal Reliability of a Complex System", IEEE Transactions

on Reliability, **R-19**, No. 3 (August 1970).

23. D. E. Fyfee, W. W. Hines, N. K. Lee, "System Reliability Allocation and a Computational Algorithm", IEEE Transactions on Reliability, **R-17**, No. 2 (June 1968).
24. R. M. Burton, G. T. Howard, "Optimal Design for System Reliability and Maintainability", IEEE Transactions on Reliability, **R-20**, No. 2 (May 1971).
25. A. S. Cici, V. O. Muglia, "Computer Reliability Optimization System", IEEE Transactions on Reliability, **R-20**, 110-116 (August 1971).
26. K. B. Misra, "A Method of Solving Redundancy Optimization Problems", IEEE Transactions on Reliability, **R-20**, 117-120 (August 1971).
27. B. K. Lambert, A. G. Walvekar, J. P. Hirmas, "Optimal Redundancy and Availability Allocation in Multistage Systems", IEEE Transactions on Reliability, **R-20**, 182-185 (August 1971).
28. J. Sharma, K. V. Venkateswaran, "A Direct Method for Maximizing the System Reliability", IEEE Transactions on Reliability, **R-20**, 256-259 (November 1971).
29. K. B. Misra, "A Simple Approach for Constrained Redundancy Optimization Problems", IEEE Transactions on Reliability, **R-21**, 30-34 (February 1972).
30. K. B. Misra, "Reliability Optimization of a Series-Parallel System", IEEE Transactions on Reliability, **R-21**, 230-238 (November 1972).
31. K. Inoue, S. L. Gandhi, E. J. Henley, "Optimal Reliability Design of Process Systems", IEEE Transactions on Reliability, **R-23** (April 1974).
32. T. Nakagawa, S. Osaki, "Optimum Preventive Maintenance Policies for a 2-Unit Redundant System", IEEE Transactions on Reliability, **R-23** (June 1974).
33. B. P. Lientz, "Allocation of Components to Maximize Reliability Using An Implicit Method", IEEE Transactions on Reliability, **R-23** (June 1974).
34. DA Pam 70-5, *Mathematics of Military Action, Operations and Systems.*
35. R. Luus and T. H. I. Jaakola, "Optimization by Direct Search and Systematic Reduction of the Size of Search Region", AIChE Journal, **19**, 760-766 (July 1973).
36. Rein Luus, "Optimization of System Reliability by a New Nonlinear Integer Programming Procedure", IEEE Transactions on Reliability, **R-24**, 14-16 (April 1975).

APPENDIX A*

Optimization of System Reliability by a New Nonlinear Integer Programming Procedure[36]

Abstract—This paper presents a useful procedure of solving nonlinear integer programming problems. It finds, first, a pseudo-solution to the problem, as if the variables were continuous. Then it uses direct search in the neighbourhood of the pseudo-solution to find the optimum. The effectiveness of the method is shown with a 15-variable problem, which requires about 1 day's FORTRAN programming effort and 8 seconds of computer time for its solution on an IBM 370/165 digital computer.

Reader Aids:
Purpose: Widen state-of-the art.
Special math needed for explanations: None
Special math needed for results: None
Results useful to: Design and reliability engineers, programmers.

INTRODUCTION

INCREASING reliability by the introduction of redundancy is well known. However, the problem of how to optimize the reliability through the selection of redundancy has not yet been adequately solved. Tillman and Liittschwager [1] presented an integer programming formulation for the solution of reliability problems. The method requires transformation of the objective function and introduction of auxiliary variables. Misra [2] discusses the overall applicability of integer programing approach to solving reliability problems; later Misra [3] introduces the use of Lagrange multipliers and the Maximum principle to solve reliability optimization problems. Sharma and Venkateswaran [4] presented a simpler method with no assurance of obtaining the true optimum. Banerjee and Rajamani [5] use the Lagrange multiplier approach to solve the reliability problem to yield optimum or near optimum results. Misra and Sharma [6] classified the methods into two groups, one which includes methods which require simple formulation and yield approximate results and the other which includes methods which are complicated but yield an exact integer solution to the problem. These authors then provide a geometric programming formulation for the reliability problem which gives an approximate answer.

The purpose of this paper is to present a method which is easy to formulate and which gives an optimum for the reliability optimization problem. Although there is no assurance of obtaining the global optimum, in practical problems the method will come very close to finding the global optimum.

PROBLEM FORMULATION

Maximize a nonlinear function of n variables denoted by

$$f(x_1, x_2, ..., x_n)$$

subject to the constraints

$$g_j(x_1, x_2, ..., x_n) \leq b_j, j = 1, 2, ..., m \qquad (1)$$

$$x_i, i = 1, 2 ..., n \text{ must be positive integers} \qquad (2)$$

The constraint functions g_j need not be linear and the number of inequality constraints m need not be less than n. A procedure involving three steps is proposed.

SOLUTION TO THE GENERAL PROBLEM

Step 1: Solution to the Pseudo-Problem

Relax the condition of requiring each x_i to be integer and solve the maximization problem as if the variables were continuous. Only an approximate solution is necessary to this pseudo-problem.

Step 2: Filling in the slack by steepest ascent

Take the values of x_i obtained in Step 1 and convert them to integers by truncation (toward zero) so that the inequality constraints (1) are satisfied.

There may now be adequate slack in (1) to allow an increase in at least one of the x_i. Therefore, attempt to increment each x_i by 1, check to see if (1) is satisfied, and increment only the x_i which gives the greatest contribution to the maximization of f. Continue this filling of slack until no x_i can be incremented without violating at least one of the constraints.

Step 3: Systematic exchange of variables

Carry out $n(n-1)$ tests whereby one variable is incremented by 1 and the others are decremented by 1 in turn. For example, suppose x_1 is incremented to $x_1 + 1$. Now decrement x_2 to $x_2 - 1$ and check whether inequalities are satisfied. If so, then calculate the corresponding value of f and compare that value to the maximum f in Step 2. If the most recently calculated f is greater, then retain in the memory the fact that x_1 incremented by 1 and x_2 decremented by 1 gives a better value. However, before making a change in this variable, continue through the entire cycle up to x_n. Then choose the set x_i which has given the greatest value for f. Perform the cycle by incrementing x_2 and continue with x_3, x_4, etc. up to x_n. In total, there are thus a maximum of $n(n-1)$ tests to be done. The set giving the largest value of f is retained as the optimum.

TABLE 1

Reliability, Cost and Weight Factors for Example 1

Stage number i	Reliability r_i	Cost c_i	Weight w_i	Allocation (x_i) Step 1	Step 3
1	0.80	1.2	1.0	5.3	6
2	0.70	2.3	1.0	6.3	6
3	0.75	3.4	1.0	5.2	5
4	0.85	4.5	1.0	3.8	4
System reliability				0.9979	0.9977
System cost (56 max)				56.0	56.0
System weight (30 max)				20.7	21.0

TABLE 2

Reliability, Cost and Weight Factors for Example 2

Stage Number i	Reliability r_i	Cost c_i	Weight w_i	Allocation (x_i) Step 1	Step 3
1	0.90	5	8	2.9	3
2	0.75	4	9	4.2	4
3	0.65	9	6	4.9	5
4	0.80	7	7	3.7	3
5	0.85	7	8	3.0	3
6	0.93	5	8	2.3	2
7	0.78	6	9	3.4	4
8	0.66	9	6	5.0	5
9	0.78	4	7	4.0	4
10	0.91	5	8	2.7	3
11	0.79	6	9	3.5	3
12	0.77	7	7	3.7	4
13	0.67	9	6	5.1	5
14	0.79	8	5	4.3	5
15	0.67	6	7	5.0	5
System reliability				0.952	0.945
System cost (400 max)				386.0	389.0
System weight (414 max)				413.7	414.0

EXAMPLES

Since there is no assurance that the global optimum is reached, it is instructive to test this method by applying it to a class of reliability problems which have been handled by other methods.

Example 1

The reliability problem [6] maximizes the reliability function

$$f = \prod_{i=1}^{4} \{1 - (1 - r_i)^{x_i}\} \tag{3}$$

subject to the constraints

$$\sum_{i=1}^{4} c_i x_i \leq 56 \tag{4}$$

$$\sum_{i=1}^{4} w_i x_i \leq 30 \tag{5}$$

There are 4 stages and the reliability, cost, and weight factors are given in Table 1.

For Step 1, it is easiest to use the optimization method of Luus and Jaakola [7]; see the Appendix for the simple algorithm. The initial value for each x_i, $i = 1, 2, ..., 4$ was chosen as 2.0, the initial region for the random numbers as 5.0, the reduction factor for the regions after each iteration was chosen to be 0.02, and 100 iterations were specified. The algorithm for step 1 is given in the Appendix.

At the end of Step 1 the results are as shown in Table 1. These values of x_i were then truncated and Steps 2 and 3 were performed to yield the results shown in Table 1. The answer is better than that obtained by Misra and Sharma [6].

The total computation time by the 3-step procedure was 3 seconds on IBM 370/165 digital computer, during which the reliability function was evaluated 5384 times.

Example 2

To provide a more rigorous test of the proposed procedure, consider a 15 stage reliability problem where the constraints of (4) and (5) are 400 and 414 respectively; the reliability, cost and weight factors are in Table 2.

Exactly the same computational procedure as in Example 1 was used. The results after Steps 1 and 3 are given in Table 2. The total number of function evaluations was 5362 and the computation time was 7.8 seconds.

DISCUSSION

The negligible computation time for the 15 stage reliability problem shows that the proposed method is very useful for solving reliability problems where discrete units are specified. To emphasize that the recommended procedure does not involve exhaustive enumeration requires only a very simple calculation. Suppose we look at the possibility of having either 1, 2, 3, 4 or 5 units at each of the 15 stages. To evaluate all possibilities would require $5^{15} = 3 \times 10^{10}$ calculations, which is an immense, completely impractical number.

ACKNOWLEDGMENTS

This work was performed with the assistance of a grant from the National Research Council of Canada, A-3515. Computations were performed with the facilities of the University of Toronto Computer Centre.

REFERENCES

[1] F.A. Tillman and J.M Liittschwager, "Integer Programming Formulation of Constrained Reliability Problems", *Management Science*, Vol. 13, pp. 887-899, July 1967.

[2] K.B. Misra, "A Method of Solving Redundancy Optimization Problems", *IEEE Trans. Rel.*, Vol. R-20, pp. 117-120, August 1971.

[3] K.B. Misra, "Reliability Optimization of a Series–Parallel System", *IEEE Trans. Rel.*, Vol. R-21, pp. 230-238, November 1972.

[4] J. Sharma and K.V. Venkateswaran, "A Direct Method for Maximizing the System Reliability", *IEEE Trans. Rel.*, Vol. R-20, pp. 256-259, November 1971.

[5] S.K. Banerjee and K. Rajamani, "Optimization of System Reliability Using a Parametric Approach", *IEEE Trans. Rel.*, Vol. R-22, pp. 35-39, April 1973.

[6] K.B. Misra and J. Sharma, "A New Geometric Programming Formulation for a Reliability Problem", *Int. J. Control*, Vol. 18, pp. 497-503, September 1973.

[7] R. Luus and T.H.I. Jaakola, "Optimization by Direct Search and Systematic Reduction of the Size of Search Region", *AIChE J.*, Vol. 19, pp. 760-766, July 1973.

APPENDIX

Algorithm for Direct Random Search and Interval Reduction
[Equality constraints are presumed to have been eliminated] [7]

Notation:

x the set of x_i which are the unknowns
$x^{*(j)}$ the center value of x at iteration j which corresponds to the best value of x at iteration $j-1$.
$r^{(j)}$ the set of r_i which are the ranges for direct search at iteration j; the direct search for x_i is over the range.

$$x_i^{*(j)} - 0.5\, r_i^{(j)} < x_i < x_i^{*(j)} + 0.5\, r_i^{(j)}$$

y a pseudo random number, uniform over the range −0.5 to 0.5
n total number of iterations, e.g. $n = 100$
p number of random trials for each iteration, e.g. $p = 100$
ϵ the small number by which the range is reduced for each iteration, e.g. $\epsilon = 0.02$

Algorithm:

0. Choose initial values $x^{*(j)}$ and $r^{(j)}$, set $j = 1$.
1. Calculate p sets $x_i^{(j)} = x_i^{*(j)} + y\, r_i^{(j)}$; y is a new pseudo random number for each calculation.
2. Test the inequality constraints, retain only those $x^{(j)}$ that satisfy the constraints. Calculate the objective function for each retained $x^{(j)}$.
3. Find the $x^{(j)}$ which maximizes the objective function. Call it $x^{*(j)}$, the center value for next iteration. If the maximum number of iterations is reached, stop.
4. Calculate $r^{(j+1)} = (1-\epsilon)r^{(j)}$. Increment j and go to Step 1.

CHAPTER 13 COMPUTER PROGRAMS

13-1 INTRODUCTION

Modern computers are powerful tools that can be used by the engineer to compute the reliability characteristics of complex systems. A variety of mathematical methods have been developed which can be applied to solving many different types of reliability problems. Programs are available for computing parameters such as reliability, availability, and MTF for repairable and unrepairable systems.

Some of the programs can handle very large systems of hundreds of elementary units for which failure and repair information must be provided. Other programs permit cost-effective systems to be designed by computing optimum allocations of redundant units which obey constraints on weight, size, cost, and other factors. Simulation techniques have been developed for systems that are too complex to be evaluated by other methods.

A large number of computer programs have been developed for predicting the reliability parameters of systems. These programs have been written by many companies for a number of governmental agencies. Some of the programs were developed for a specific system, and some are more general and can be applied to many system configurations.

13-2 MATHEMATICA'S AUTOMATED RELIABILITY AND SAFETY EVALUATION PROGRAM (MARSEP)

MATHEMATICA, Inc., developed a program that automates the evaluation of the reliability and unreliability of electromechanical systems (Ref. 1). MATHEMATICA'S AUTOMATED RELIABILITY AND SAFETY EVALUATION PROGRAM (MARSEP), was originally developed for the SANDIA Corporation for use in evaluating nuclear weapon systems. It can be used for both reliability and unreliability calculations. The unreliability calculations are used in system safety analyses where unreliability terms of very small magnitude may be very important.

MARSEP provides a means of computing an exhaustive Boolean expression that includes all possible success and failure events. MARSEP has been programmed for computers at the Picatinny Arsenal and the Harry Diamond Laboratories.

MARSEP accepts as input a description of the system and a definition of system success. The computer determines which combinations of component events are required for system operation and system failure.

The system description contains a list of individual system components and their operating and failure modes. A set of two events, success and failure, must be defined for each component. Failure of any individual component does not cause failure in any other component.

A simple circuit consisting of a battery, switch, relay, light, and squib is shown in Fig. 13-1. The circuit description includes all terminals and wires, including the ground terminal. In using MARSEP, it is assumed that possible failure in connections and wire leads are important and must be considered.

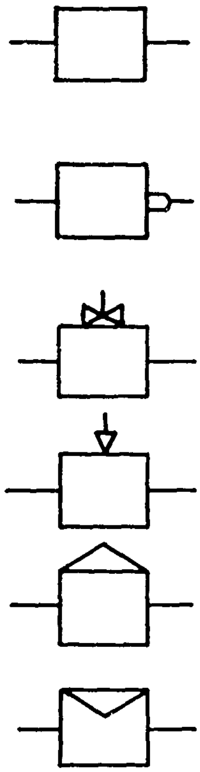
A model must be prepared from the circuit diagram. The MARSEP model is a block diagram whose elements represent the individual system components, their possible failure modes, and operating conditions. Some of the symbols used to prepare a MARSEP model are shown in Table 13-1. The MARSEP model for the sample circuit is shown in Fig. 13-2.

MARSEP provides a modeling language that is used to describe the elements in the MARSEP model and their interconnections. Each element in the MARSEP model must be given a name for use in the system description part of the input data. For example, in Fig. 13-2, the battery is defined as BATTRY, and the short mode of failure is called SHORT.

A set of symbols is also required, each symbol representing the probability of occurrence of the usual (most likely) event(s) for each element in the MARSEP model. The prefix P is used to identify events which correspond to a component functioning successfully, or transmitting a signal, or both. The prefix Q identifies events associated with a component failing to function, or opening the circuit, or both. Both types of symbols are referred to as P Names. Table 13-2 shows the

TABLE 13-1 MARSEP MODELING SYMBOLS[1]

MODELING BOXES. with electrical interpretations



BASIC MODELING. Passes signal from input (1) to output (2). Has success and failure events associated with it.

SIGNAL SOURCE. This box produces a signal at (2). It can be affected by shorts to ground and connections to ground.

AND BOXES. These boxes usually need both a usual input (1) and a second input (2) in order to provide an output at (3). There is a second event set defined for the situation when the input at (2) is missing.

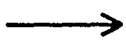
SHORT-TO-GROUND. If this box fails the circuit is shorted to ground.

FUSE. This box indicates a point in the circuit which should open when a signal passes through it.

BOX OR TERMINAL MODIFIERS

QUALITY SENSITIVE. Indicates that the box on which this appears is sensitive to the type of input received. A different event set is defined for each type of input. Signal types are defined at their source.

ENVIRONMENT. An externally determined input that provides for conditional event sets in the model.



MODELING DIODE. Indicates that a high resistance to ground exists within the box to which it is attached. This is interpreted as preventing a ground connection from draining a signal source.

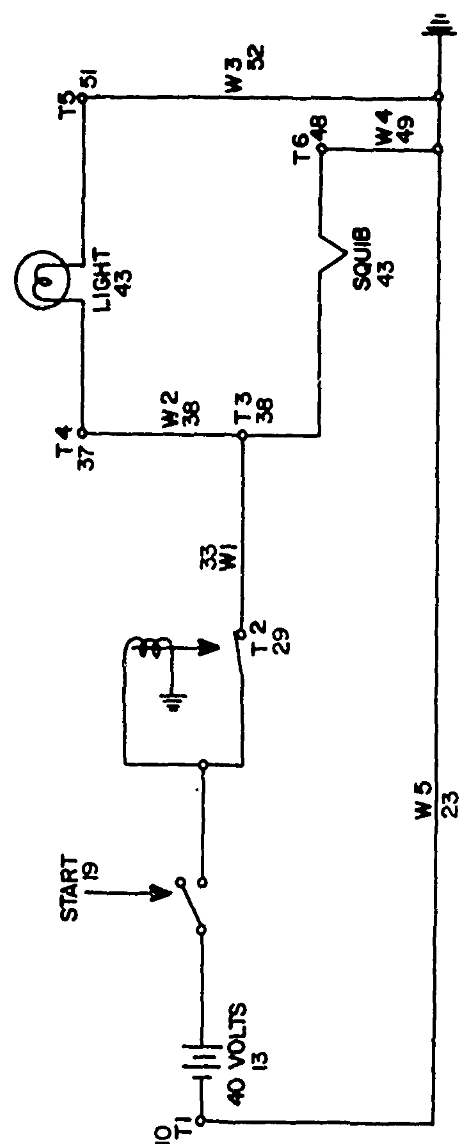
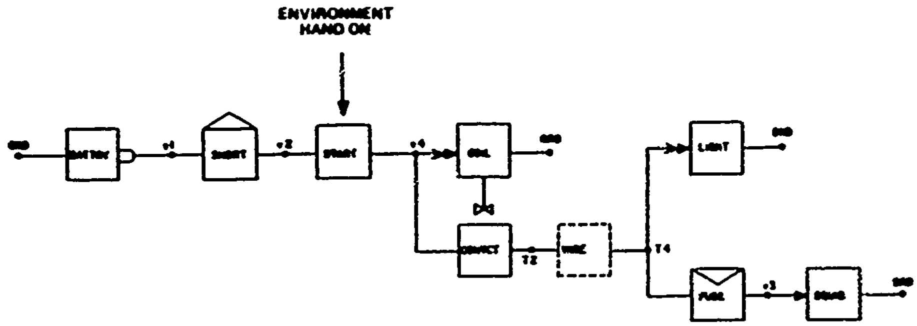


FIGURE 13-1. Simple Circuit for MARSEP Analysis[1]

TABLE 13-2

ASSIGNMENT OF P NAMES TO SIMPLE CIRCUIT MODEL[1]

ELEMENT NAME	P NAME	EVENT
BATTRY	PVØLT	battery delivers proper voltage
SHORT	PSTG	short to ground does not occur at this point
START	PCLØS	switch closes when pressure applied
	QØFF	switch remains open before pressure is applied
COIL	PPICK	symbolizes the event that coil picks contact when proper input is applied
CONTCT	PCØNT	contacts provide continuity when picked
	QERLY	contacts remain open before relay is picked
LIGHT	PLITE	light burns when proper voltage applied
FUSE	PØPEN	squib open when proper input applied
SQUIB	PBLØW	squib fires when proper voltage is applied
WIRE	PGØØD	wire carries signal applied



MARSEP Model Element	Element Name	P Name
Battery	BATTRY	PVOLT
Short in battery	SHORT	PSTG
Start switch	START	PCLOS, QOFF
Relay coil	COIL	PPICK
Relay contacts	CONTCT	PCONT QERLY
Light	LIGHT	PLITE
Fuse action of squib	FUSE	POPEN
Squib	SQUIB	PBLOW
Wire	WIRE	PGOOD

FIGURE 13-2. MARSEP Model of Simple Circuit[1]

P Names assigned to elements in the sample system and the events which they define.

Special component properties and environmental or outside factors can be included in the MARSEP model. For example, in Fig. 13-2, the effect of the human operator who turns the system on and off is shown as START with the corresponding P Names PCLOS and QOFF. The effects of pressure and temperature, as well as enabling procedures, also can be included.

By use of the MARSEP modeling language, the element names, and the P Names, the MARSEP model is converted into a series of statements which become the input to the MARSEP program. Table 13-3 shows some elements of the MARSEP modeling language.

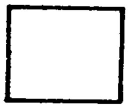
The MARSEP program consists of three subprograms: (1) the preprocessor, (2) the analyzer, and (3) the postprocessor. The system to be analyzed is represented in the computer by lists of components and a list structure for each component and terminal. The preprocessor accepts as input a description of the system which is converted into the required format for the analyzer. Then, the analyzer, written in Information Processing Language V (IPLV), generates the success and failure expressions for the system. The postprocessor substitutes the external names provided in the input for the internal symbols used by the analyzer. The equations generated by the analyzer are not altered by the postprocessor. The MARSEP program also edits and applies set theory to the success and failure expressions.

For the system in Fig. 13-1, the MARSEP program would perform an analysis of the effects of shorts-to-ground and spurious electrical connections (shorts) on the operation of the system. In the shorts-to-ground analysis it is assumed that components transmit a signal that must be maintained at some level other than the level associated with ground. All ground terminals or possible connections to ground are, therefore, examined to determine if they can possibly nullify a useful signal in the system. Special messages are printed in the program output which indicate when useful signals are nullified at their source by a connection to ground. Shorts between terminals in the system are checked to determine if they can cause undesirable operation. The user can designate in his input statements where shorts are likely to occur, and the program also will search automatically for shorts.

Outputs prepared by MARSEP for the sample circuit are presented in Table 13-4. Two expressions are developed for system success. The first expression is for system success when the environment EHAND is applied in such a way that the switch is open. The

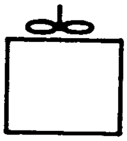
TABLE 13-3

MARSEP MODELING LANGUAGE[1]



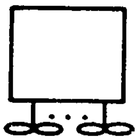
A2 (P name)

This attribute states the probability that the given element works, given all proper inputs, is P name.



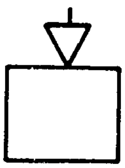
A3 (B name, P name)

Denotes the element receives an enabling input from element B name. In the absence of that input, the element will give an output with probability P name. (The probability of nonoperation given a proper enabling input is given by A2.)



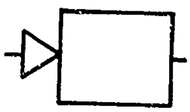
A4 (B name 1,............B name n)

Denotes the element has enabling ouputs to the elements B name 1.............. B name n.



A14 (E name, P name)

E name is the name of some environment. It is any item such as HEAT, PRES 6, RAD 2, etc., that is listed as an environment. P name is the probability that the element functions in the absence of named environment.



A6 (T name, A name, N name, P name*, N name, P name*···*N name P name)

Thus, A6 is followed by a compound list:

T name (or V name) is the name of an input terminal to the element which is dependent upon the value of the input signal (quality input).

A name is either A10 if the input is voltage-sensitive, one of the attributes, or A50 through A90 for nonvoltage sensitive sources. The A number may be left out of subsets after the first subset. In this case it will be interpreted to be the same as the last one listed.

N name is either a value of the input signal at the terminal in question (i.e., an integer) or an item with head N which will symbolically indicate a signal level.

P name is the probability of operation given N name. The probability of non-operation given the usual value of N name is given in A2.

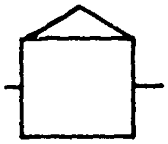
Only one A6 is allowed per element.

TABLE 13-3 (Cont'd)

MARSEP MODELING LANGUAGE

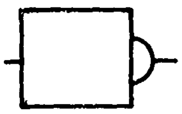
A8

Fuse behavior



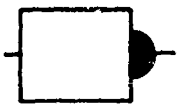
A9

Short to ground



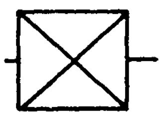
A10 (T name, N name)

Indicates that the named terminal is voltage source whose value is given by N name. N name is defined as for A6.



Ann (T name, N name)

nn can range from 50 through 90. This set is used to identify a power source other than a voltage source.



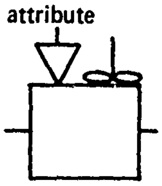
A12

Indicates Q name of A2 is very near to one.

Any box that has an A6 attribute

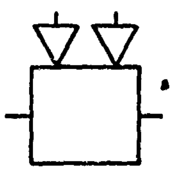
A99

Terminal device



A2 (see format discussed above)
A3 (see format discussed above)
A14 (see format discussed above)
A16 (see format discussed above)

P name for A16 attribute is probability of operation given no environment and activation.



A17 (E name 1, P name 1, E name 2, P name 2, P name 3)

A2 (same format as is discussed above)

where:

P name 1 is probability that box operates given E name 1 is present.
P name 2 is similar to P name 1.
P name 3 is probability of operation given E name 1 and E name 2 are absent.

A7 (T name,............T name)
Indicates that the named terminals will not propagate a ground.

TABLE 13-4
MARSEP DEVELOPED SUCCESS EXPRESSIONS FOR SIMPLE CIRCUIT[1]

EHAND SET OFF

(Complete Expression)

PVOLT * PSTG * POPEN * 'QOFF * PPICK * PCONT * PGOOD
* PLITE * PBLOW + PVOLT * PSTAG * 'POPEN * 'QOFF
* 'PPICK * 'QERLY * PGOOD * PLITE * PBLOW

(After Editing)

'QOFF + 'QOFF * 'PPICK * 'QERLY

(After Set Theory)

'QOFF

<u>Second Expression</u>

EHAND SET ON

(Complete Expression)

'PVOLT + 'PSTAG + 'POPEN + 'PCLOS + 'PPICK * QERLY + 'PCONT
+ 'PGOOD + 'PLITE + 'PBLOW

(After Editing and Set Theory)

'PVOLT + 'PSTG + 'POPEN + 'PCLOS + 'PPICK + 'PCONT + 'PGOOD
+ 'PLITE + 'PBLOW

second expression is for system success when the switch is closed by the hand.

13-3 GENERAL EFFECTIVENESS METHODOLOGY (GEM)

The GEM system was developed by the Naval Applied Sciences Laboratory in order to provide engineers with a user oriented reliability evaluation technique (Refs. 2-5). The user interacts with GEM by means of a language especially developed for use in reliability problems.

The GEM system consists of the GEM language, a System Library, a Formula Library, and a program system containing a processor and update programs.

The GEM processor is designed to accept descriptions of reliability block diagrams together with associated data and to calculate one or more reliability measures. The description and computed results can be stored in the System Library which can later be retrieved, modified, and re-evaluated.

The Formula Library contains a set of mathematical subroutines for computing various reliability parameters, relieving the engineer of the burden of constructing a new program for each new system evaluation.

The GEM program system was developed using a modular approach that facilitates the modification of existing programs and addition of new routines as needed. The general organization of the GEM program system is shown in Fig. 13-3.

GEM can be used to support systems development, trade-off analyses, evaluation, and optimization. The processor is structured to evaluate variables such as reliability with or without repair, instantaneous availability, and interval reliability for systems that include such hardware interdependencies as bridge networks, shared elements, standby equipment, and environmental strategies and priorities including repairmen and spare parts pools (see Fig. 13-4).

13-3.1 STRUCTURE OF GEM

The engineer using GEM provides a system description consisting of a reliability model; failure, repair, and replacement rates; the up-state rules; replacement and repair strategies; and support constraints for the system. The support constraints are the number of repairmen and their specific assignment, the number of spares pools, the spares in each pool and identification of the items that share each pool, allocation strategies to be used in cases of conflicting demands on repairmen and/or spares, and identification of items to be held in standby. The user also specifies which reliability parameters are to be calculated by the GEM processor.

The system description is written in the GEM System Definition Language, and the parameters to be calculated are stated in the GEM Command Language (Ref. 4). The Command Language also is used to make modifications to previously defined system descriptions.

The System Library is a magnetic tape containing system descriptions, calculation requests, and calculated results for previously evaluated systems (Ref. 5). The Formula Library is a magnetic tape containing the formulas and computer routines for calculating the reliability parameters that are part of the GEM system (Ref. 5).

The GEM processor refers to the System Library (if the system has been previously evaluated) and the Formula Library, while it first translates the system description and calculation requests into a mathematical model for computing the parameters requested, then performs the calculations, and finally, prints the results.

Error-checking routines are built into the processor to detect omissions, inconsistencies in the description or data, wrong parameters, impossible values of parameters, and other errors. When errors are detected, the processor prints error messages that define the nature of the errors and their location.

The GEM system also contains a set of Library Update Programs for generating, maintaining, and updating the System Library and the Formula Library.

The GEM system provides a printed output in the form of a tabulation of computed results or a plot output. The user's original system description is presented as part of the output.

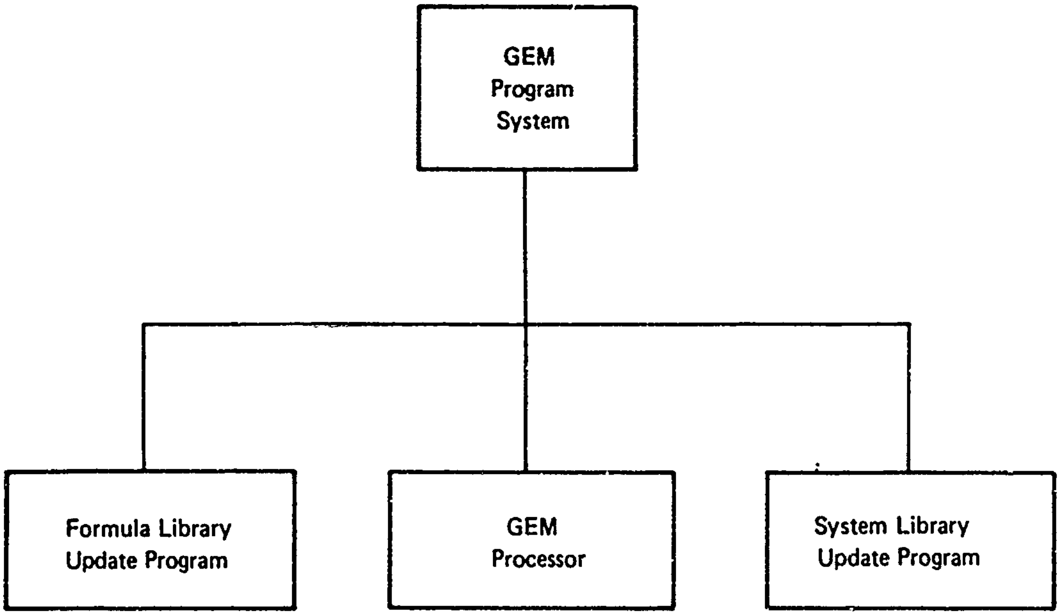


FIGURE 13-3. GEM Program System Organization[4]

The GEM program was implemented on a CDC 6600 computer located at the Courant Institute of New York University. Minimum requirements for running the program are 135,000 words of memory for most problems and 300,000 words for calculating reliability with repair and availability of systems with nonexponential failure and/or repair distributions. The GEM processor was designed using the Chippewa Operating System.

13-3.2 THE GEM SYSTEM

The computer equipment configuration required by the GEM processor is:

1. CDC 6600
2. Five magnetic tape drives
3. Disc file
4. Card reader
5. Printer.

All possible GEM inputs and outputs are illustrated by the GEM flow diagram, Fig. 13-5. The GEM processor requires formula input and system definition input. Formula input takes one of the three following forms:

1. Previously created formula library tape.
2. A new formula library tape, created from a set of cards, containing variables, formulas, and update commands.
3. A revised formula library tape created from a combination of the two preceding forms—i.e., a previously created formula library tape, plus a set of cards containing additional variables, formulas, update commands, etc., which would result in a revised formula library tape.

System definition input takes one of the three following forms:

1. A set of cards containing system definitions, evaluation verbs, and (if desired) modification verbs.
2. A previously created system library tape plus a set of cards containing evaluation and modification verbs (and, if additional systems are required, a set of cards containing new system definitions).
3. A previously created print file tape (containing system definitions) plus a set of cards containing evaluation and modification verbs (and, if additional systems are required, a set of cards containing new system definitions).

Output consists of a printout sheet (printed output listings) and a magnetic tape (Print File).

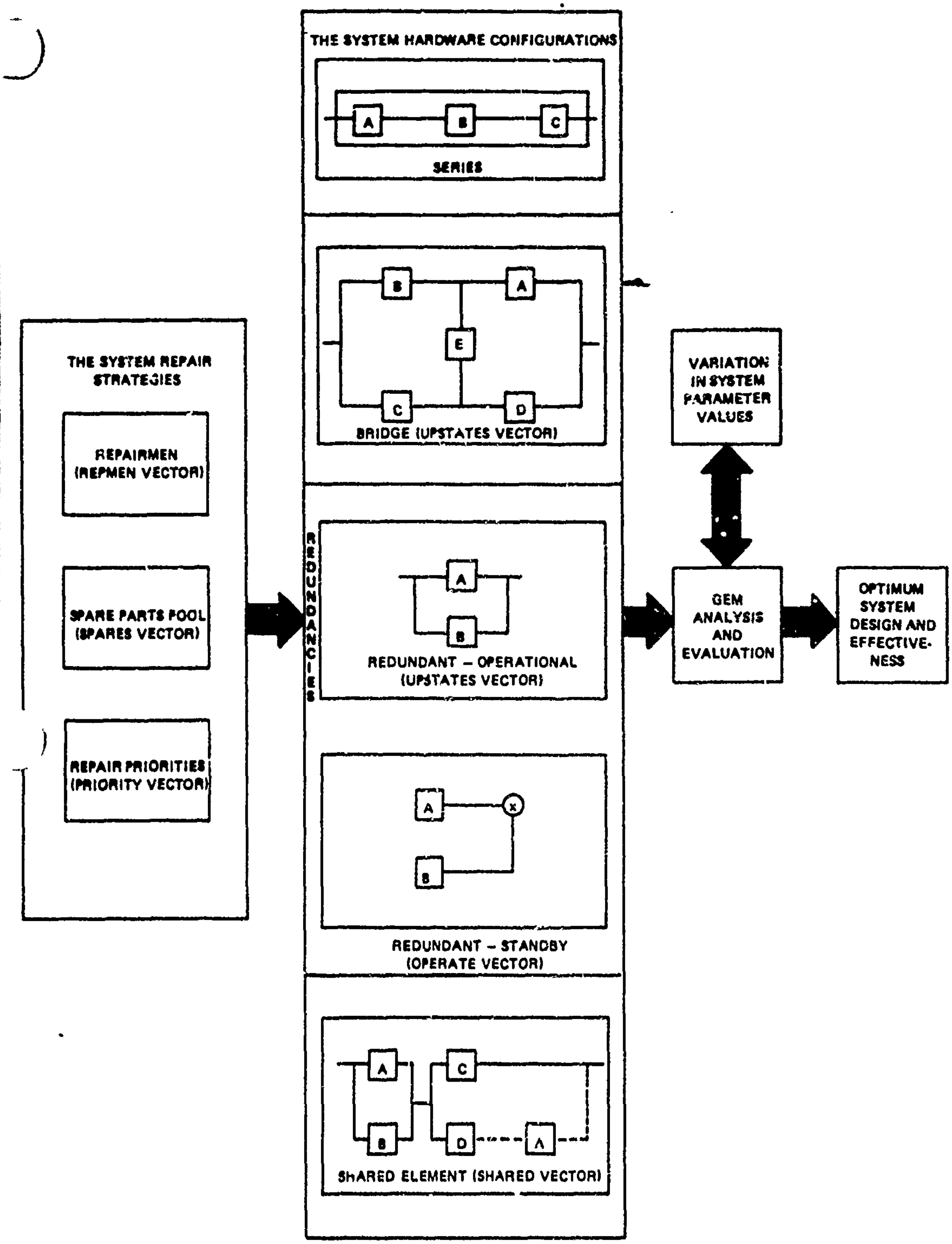


FIGURE 13-4. Interrelation of GEM Environmental Vector Definitions and Overall System Effectiveness

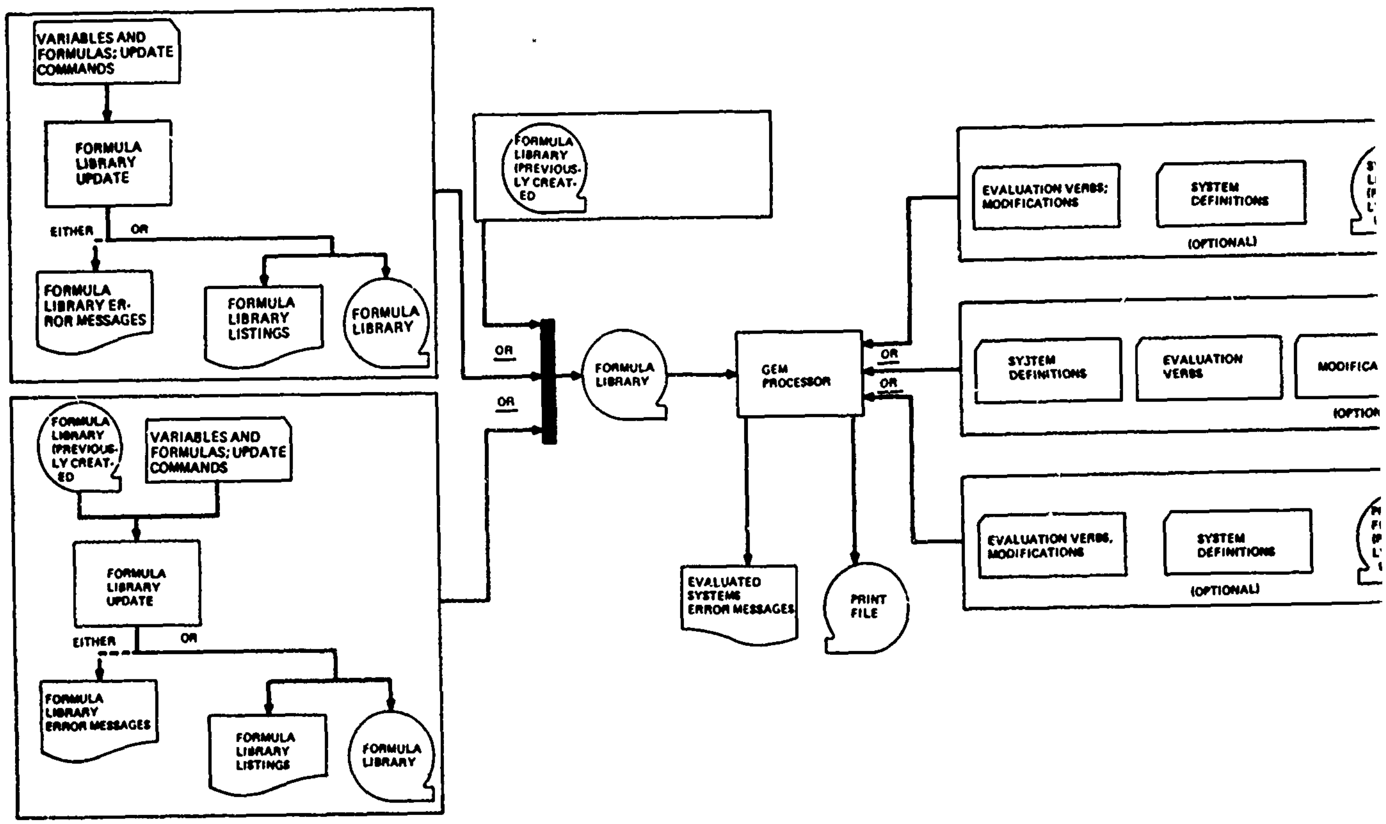
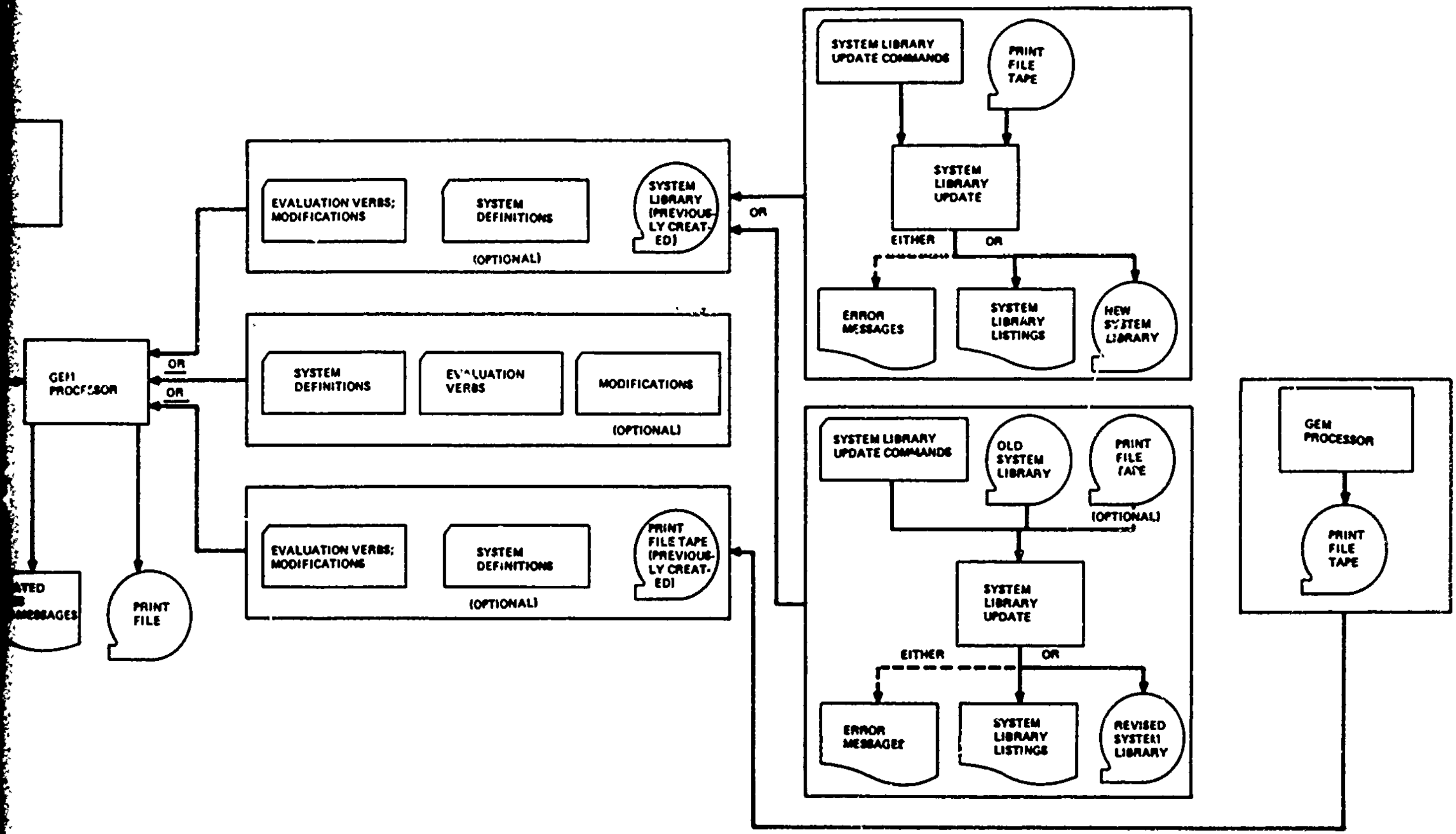


FIGURE 13-5. GEM Input/Output Diagram[5]


SYSTEM LIBRARY UPDATE COMMANDS
PRINT FILE TAPE
SYSTEM LIBRARY UPDATE
EITHER
OR
ERROR MESSAGES
SYSTEM LIBRARY LISTINGS
NEW SYSTEM LIBRARY
EVALUATION VERBS; MODIFICATIONS
SYSTEM DEFINITIONS
(OPTIONAL)
SYSTEM LIBRARY (PREVIOUSLY CREATED)
SYSTEM DEFINITIONS
MODIFICATIONS
PRINT FILE TAPE (PREVIOUSLY CREATED)
OLD SYSTEM LIBRARY
REVISED SYSTEM LIBRARY
GEM PROCESSOR
PRINT FILE

iagram[5]

There are three phases to GEM. During Phase 1, information is read (transferred) into the computer, error checked, and stored in files within the computer in a compact form. Phase 2 processing involves making the modifications indicated by the original modification commands. In Phase 3, the newly created system is used to generate a FORTRAN source program, to permit the calculation of the systems effectiveness measures. The FORTRAN program is compiled and executed, the answer tables are created, and, subsequently, the output (a printout of the evaluated systems and error messages and a print file tape) is generated.

13-3.3 THE GEM LANGUAGE

13-3.3.1 The System Definition Language

Some of the basic elements (vocabulary) of the System Definition Language are (Ref. 4):

1. Level Number
2. Duplicate Number
3. Item Name
4. Formula Name
5. Parameters
6. Environmental Vectors (E Vectors).

The Level of an item is its level of comprehensiveness or its position in a hierarchy that represents the manner in which the user views the system.

The Duplicate number of an item states the number of identical items in a system and is used to avoid having to describe identical items more than once.

The Item Name is used for identification and is arbitrarily chosen by the user. Names need not be unique except for items of the same level if they are not identical.

The Formula Name is either a statement of the relationship that items in a lower level bear to one another, or it identifies the name of a failure and/or repair distribution associated with a lowest level item.

The Parameters serve as either further clarification of the relationship stated in the formula or they give the parameters of the failure and/or repair distributions associated with the lowest level items.

Environmental Vectors serve two basic functions. They enable the user to describe complex configuration or upstate rules which cannot be stated in terms of series-parallel statements. They also enable one to specify constraints with respect to repairmen and/or spares as well as their deployment and the order of priority to be followed when there are not enough repairmen and/or spares for every item that is in a downstate.

13-3.3.2 Illustration of the System Definition Language

The concept used in describing a system configuration in GEM permits the connectivity of the items in a block diagram to be defined in stages (levels of comprehensiveness) so that more detail is stated at each level until the lowest level item is reached. In effect, the block diagram consists of a hierarchy of levels and, at each level, the appropriate relationship of the items just one level below is defined. To illustrate this procedure, consider the block diagram in Fig. 13-6.

The system in Fig. 13-6 is made up of two subsystems connected in series. The first subsystem consists of four identical items and the upstate rule is that at least two must be up (2-out-of-4:G). The second subsystem is a parallel-series configuration. The breakup of a system in terms of its levels can be portrayed by a GEM diagram. For the example in Fig. 13-6 this would have the form shown in Fig. 13-7. The description of this system in the GEM Definition Language would be as in Table 13-5.

In Table 13-5, the entry in the Formula column designates that at the 01 level, the rule of combination for the two 02 level items (SBSYS1 and SBSYS2) is the statement that these items are connected in series (SER). It is not necessary to state the mathematical formula for a series connection, only its code. At the first 02 level, the Formula entry is PAR to designate that the four 03 level items are connected in parallel. The entry in the Parameter column, M = 2, states that at least two of the 03 items must be up in order for the 02 item to be up. The entry of 4 in the Dup. column for item A states that there are four identical items, each called A, and the FENO entry in the Formula column states that the times to

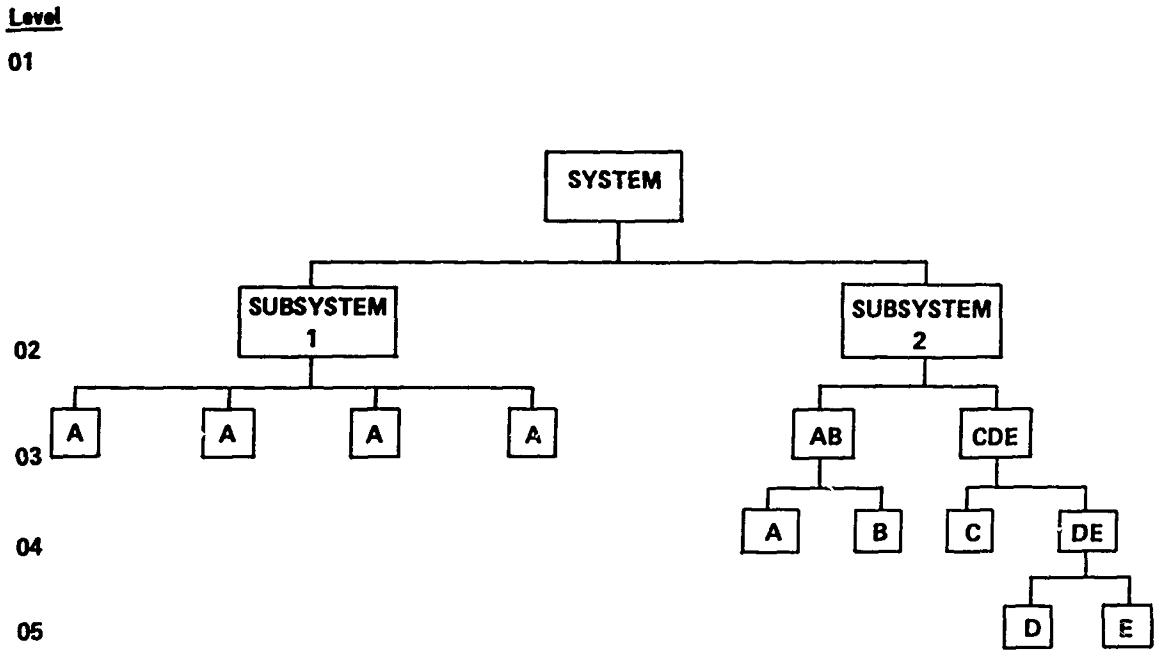


FIGURE 13-6. Sample System for GEM Analysis[4]

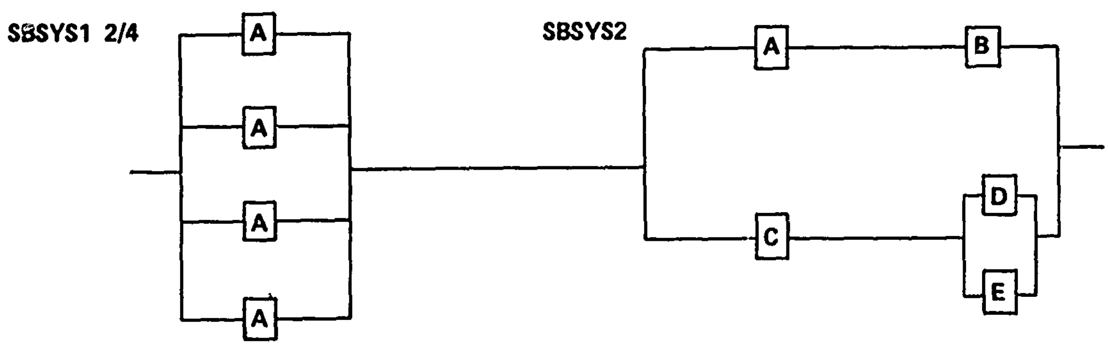


FIGURE 13-7. GEM Diagram for Sample System[4]

TABLE 13-5
SYSTEM DESCRIPTION IN GEM SYSTEM DEFINITION LANGUAGE[4]

Level	Dup.	Name	Formula	Parameters
01		SYSTM	SER	
02		SBYS1	PAR	M = 2
03	4	A	FENO	λ =
02		SBSYS2	PAR	M = 1
03		AB	SER	
04		A	FLNO	μ = , σ =
04		B	FWNO	α = , β =
03		CDE	SER	
04		C	FGNO	α = , β =
04		DE	PAR	M = 1
05		D	FLNO	μ = , σ =
05		E	FTNO	μ = , σ =

TABLE 13-6

GEM SYSTEM DEFINITION LANGUAGE FORMULA SYMBOLS[4]

FORMULA NAME	MEANING	PARAMETERS
FENO	One piece of equipment with exponential failure and no repair	RLAM - Failure Rate
FWNO	One piece of equipment with Weibull failure and no repair.	ALPH - TIME PER FAILURE BETA -
FGNO	One piece of equipment with gamma failure and no repair.	ALPH - TIME PER FAILURE BETA -
FLNO	One piece of equipment with log-normal failure and no repair.	XMU - SIG -
FTNO	One piece of equipment with truncated s-normal failure and no repair.	XMU - SIG -
SER	The subsystems are in series.	All the resultant names of the subsystems must be X.
PAR	The subsystems are redundant (parallel) of which M must be working.	M - The number that must be working. All the resultant names of the subsystems must be X.
LIN	The subsystems are identical and layed out in a linear array. M must be working and no two adjacent subsystems may have failed.	M - The number which must be working. The resultant names of the subsystems must be X.
GIR	The subsystems are identical and layed out in a circular array. M must be working and no two adjacent subsystems may have failed.	M - The number which must be working. The resultant names of the subsystems must be X.

failure of item A are exponentially distributed. The parameter of the distribution (the failure rate λ) is given in the Parameter column. The other entries are made in a similar manner. Table 13-6 explains the other formula symbols and gives the parameter notations to be used.

13-3.3.3 Additional Characteristics of the System Definition Language

The preceding description of the system is valid only for the computation of a variable that can be calculated by purely combinatorial means, starting from the lowest level item results and continuously passing these up to a higher level until the top level (01) or system answer is obtained. This procedure can be used to calculate reliability without repair (R) and/or availability in the absence of repairmen and/or spares constraints (provided the repair distribution for each item is given).

This procedure cannot be used to calculate reliability with repair (RR) since RR for SBSYS2 cannot be obtained from the values of RR for items A-E by somehow combining these results. (As a matter of fact, the RR's for A-E are equal to the R's for these items.) The reason for this is that Items A-E are *s*-dependent for the purpose of calculating RR for SBSYS2, although they are not *s*-dependent for the purpose of calculating R. However, since SBSYS1 and SBSYS2 are connected in series, it is permissible to calculate RR for SBSYS1 and SBSYS2 separately and then obtain RR for SYSTM by combining these results in series, i.e., by multiplying them.

Whenever items have to be handled as a group due to their *s*-dependence, either because of the variable to be computed or because they share spares and/or repairmen, then the preceding description of the system is not adequate, and a different one has to be used. Also, if any part of the block diagram contains items that are connected in a manner that cannot be expressed as combinations of series-parallel groups, i.e., the upstate rules cannot be given in terms of series-parallel statements, then another means of describing the configuration is required—even for the purpose of calculating R.

To permit system descriptions of a more general nature and to provide the user with a capability to impose repairmen and/or spares constraints, the System Definition Language of GEM introduces the concept of a section (Ref. 4). A section is a group of items to which the user can apply any of the six environmental vectors.

Some elements of the System Definition Language were not discussed before, because they were not central to the basic concepts employed in the description and to avoid confusion. The additional elements of the System Definition Language are:

1. Resultant Name
2. Formula Modification Code (MOD)
3. Variable Code.

The Resultant Name is the name chosen by the user for either the answer for the variable of an item after it has been evaluated, or the name that is chosen for use in an E Vector. All references to items in that E Vector must use the Resultant Names and it is, therefore, important that these be unique within a section unless items are identical.

The Formula Modification Code (MOD) for Duplicate Items was introduced for future capabilities in GEM which might evaluate a variable for which one might want to ignore the fact that there are duplicates of an item.

The Variable Code designates the type of computation that will be used in evaluating the variable—e.g., purely algebraic, a state calculation involving differential equations, or some combination of these. The code TE is a generalized code which can be used to calculate all variables provided the necessary conditions are met.

The names of the combinatorial formulas in the Formula Library and the notations used for their associated parameters are presented in Table 13-7. This table presents the names of the formulas associated with sections and the notation to be used for their associated parameters. A GEM System Definition Coding Form is shown in Fig. 13-8 for guidance regarding the columns to be used for entering the information resulting from the description of a system by the System Definition Language. The columns for the placement of the command verbs to be described also are shown.

TABLE 13-7

FORMULAS ASSOCIATED WITH A SECTION[4]

FORMULA	MEANING AND REQUIREMENTS	PARAMETERS
Formulas also permitted outside sections.		
FENO FGNO * FLNO * FTNO *	These formulas refer to pieces of equipment with no repair or replacement. Those with astericks after them cannot appear in a section with repair or replacement.	
Formulas only permitted within sections.		
FERE	Equipment with exponential failure and exponential repair. The repairman situation is described in the REPMEN E-vector.	RLAM - Failure rate. XMU - Repair rate.
FESI	Equipment with exponential failure and instantaneous replacement. The spares pools are described in the SPARES E-vector.	
FESE	Equipment with exponential failure and exponential replacement. The repairman situation is described in the REPMEN E-vector and the spares pools in the SPARES E-vector.	RLAM - Failure rate. SLAM - Replacement rate.
SECT	The first formula of a section. Its dependence on its subsystems is described in its UPSTATES E-vector.	None.
S	The formula of a group item within a section. Its dependence on its subsystem and pieces of equipment is described in its UPSTATES E-vector.	None.

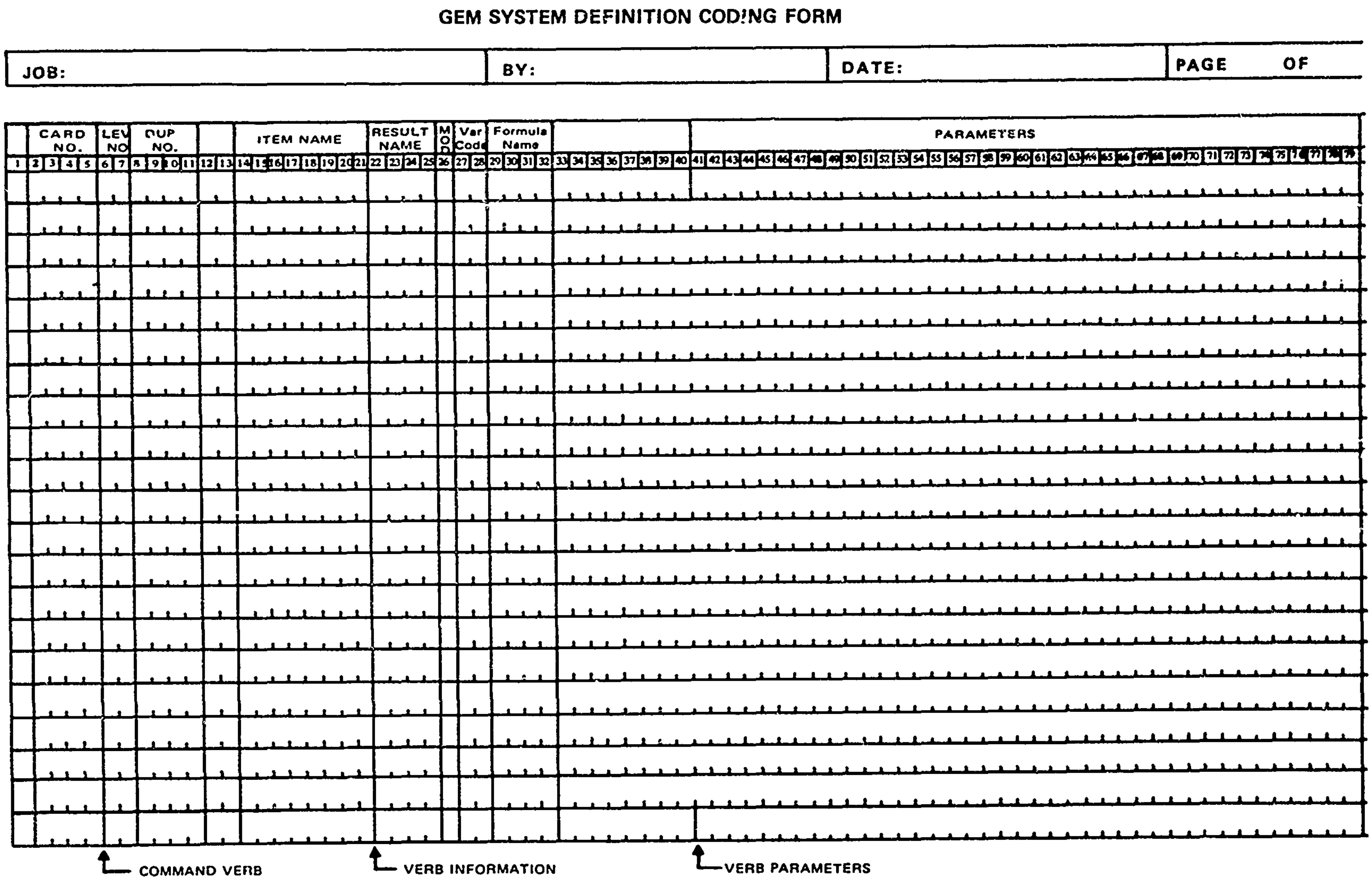


FIGURE 13-8. GEM System Definition Language Coding Form[4]

13-3.3.4 The Command Language

The System Definition Language gives the user the ability to describe a problem. The GEM Command Language is used to instruct the computer to do a computation and to modify the original problem.

The basic elements (vocabulary) of the Command Language are:

1. *Evaluation Verbs*:
   BEGIN
   END
   USE
   NAMING
   CALCULATE
2. *Modification Verbs*:
   DELETE
   ADD
   REPLACE
   ALTER
   VARY.

The two commands BEGIN and END are used to initiate and stop, respectively, the GEM program on the computer for the purpose of making a "run" on the machine. A run can consist of one or more problems. Each problem starts with the USE card and ends with the NAMING card. The NAMING statement is followed by any name the user wishes to give the computed answer to the problem.

The CALCULATE statement requests the calculation of a variable and is followed by the name of the variable. For variables which require the statement of a mission time, this information is stated after the name of the variable.

The verbs DELETE, ADD, REPLACE, ALTER, and VARY are used to modify a system description.

The command DELETE is used to drop a certain portion of the system description. If this command is applied to an 03 level item, for instance, then this item and all its lower level items will be dropped from the system description.

The command ADD will add to the system description either something that immediately follows the ADD command or a system (or portion thereof) which has been previously described or appears in the Systems Library.

The REPLACE command is a combination of the DELETE and ADD commands.

The ALTER command is used to change any one of the entries for an individual item, such as its parameters, name, resultant name, or level. Only the item specified is affected by the ALTER; its lower level items remain the same.

The VARY command is perhaps the most important one, because it gives the user the ability to make sensitivity analyses. It does this by allowing the user to vary the values of one or more parameters of items in the system description and see the effects of this on the value of the overall system answer. Thus, one can determine the sensitivity of the system Reliability with Repair to the failure rate and/or repair rate of an individual item or group of items appearing anywhere in the system description. The procedure followed in GEM is to compute the system answer for the requested variable for every value of the parameter specified in the VARY. Ref. 4 gives more specific examples of using GEM for a sample system; it includes block diagrams, GEM input, and GEM output.

13-4 OTHER PROGRAMS

Other computer programs for calculating various aspects of reliability are listed in *Part Two, Design for Reliability*, par. 4-5. In addition, most computer installations have statistical packages for performing routine estimations, and simulation languages for performing Monte Carlo simulation. Few people can know all about all available programs. Specialists can assist in selecting a few from the available many, then help an engineer become familiar with those few. It is better to be able to use handily a fairly good program than to have only a remote knowledge of several excellent programs.

REFERENCES

1. *MARSEP*, Mathematica Associates.
2. C. Sontz, S. Seltzer, and P. Giardano, *General Effectiveness Methodology*, Operational Research Society of America, Durham, North Carolina, October 18, 1966.

3. *The Generalized Effectiveness Methodology (GEM)*, Interim Report, U S Naval Applied Sciences Laboratory, Brooklyn, N.Y., 30 September 1966.
4. S. Orbach, *The Generalized Effectiveness Methodology (GEM) Analysis Program*, U S Naval Applied Science Laboratory, Brooklyn, N.Y., 8 May 1968.
5. *GEM Formula Library Reference Manual* and *GEM Maintenance Manual*, CAI Report NY-6453-II-002-U, Prepared for U S Naval Applied Sciences Laboratory, Brooklyn, N.Y., under Contract No. N00140-67-0350, April 1967.

INDEX

R

S

T

V

W

FOR THE COMMANDER:

ROBERT L. KIRWAN
Brigadier General, USA
Chief of Staff

OFFICIAL:

G. J. HAROLD
LTC, GS
Adjutant General

DISTRIBUTION:
Special

☆U.S. GOVERNMENT PRINTING OFFICE: 1976-0-711-372/39-A

ENGINEERING DESIGN HANDBOOKS

Available to DA activities from Letterkenny Army Depot, ATTN: AMXLE-ATD, Chambersburg, PA 17201. All other requestors--DOD, Navy, Air Force, Marine Corps, nonmilitary Government agencies, contractors, private industry, individuals, universities, and others--must purchase Handbooks from National Technical Information Service, Department of Commerce, Springfield, VA 22151. See Preface for further details and AMC policy regarding requisitioning of classified documents.

No. AMCP 706-	Title
100	Design Guidance for Producibility
104	Value Engineering
106	Elements of Armament Engineering, Part One, Sources of Energy
107	Elements of Armament Engineering, Part Two, Ballistics
108	Elements of Armament Engineering, Part Three, Weapon Systems and Components
109	Tables of the Cumulative Binomial Probabilities
110	Experimental Statistics, Section 1, Basic Concepts and Analysis of Measurement Data
111	Experimental Statistics, Section 2, Analysis of Enumerative and Classificatory Data
112	Experimental Statistics, Section 3, Planning and Analysis of Comparative Experiments
[illegible]	Experimental Statistics, Section 4, Special Topics
[illegible]	Experimental Statistics, Section 5, Tables
[illegible]	Environmental Series, Part One, Basic Environmental Concepts
116	Environmental Series, Part Two, Natural Environmental Factors
117	Environmental Series, Part Three, Induced Environmental Factors
118	Environmental Series, Part Four, Life Cycle Environments
119	Environmental Series, Part Five, Glossary of Environmental Terms
120	Criteria for Environmental Control of Mobile Systems
121	Packaging and Pack Engineering
123	Hydraulic Fluids
124	Reliable Military Electronics
125	Electrical Wire and Cable
127	Infrared Military Systems, Part One
128(S)	Infrared Military Systems, Part Two (U)
130	Design for Air Transport and Airdrop of Materiel
132	Maintenance Engineering Techniques (MET)
133	Maintainability Engineering Theory and Practice (METAP)
134	Maintainability Guide for Design
135	*Inventions, Patents, and Related Matters
136	**Servomechanisms, Section 1, Theory
137	**Servomechanisms, Section 2, Measurement and Signal Converters
138	**Servomechanisms, Section 3, Amplification
139	**Servomechanisms, Section 4, Power Elements and System Design
140	Trajectories, Differential Effects, and Data for Projectiles
150	Interior Ballistics of Guns
158	Fundamentals of Ballistic Impact Dynamics, Part One
159(S)	Fundamentals of Ballistic Impact Dynamics, Part Two (U)
160(C)	Elements of Terminal Ballistics, Part One, Kill Mechanisms and Vulnerability (U)
161(C)	Elements of Terminal Ballistics, Part Two, Collection and Analysis of Data Concerning Targets (U)
162(SRD)	Elements of Terminal Ballistics, Part Three, Application to Missile and Space Targets (U)
163(S)	*Basic Target Vulnerability (U)
165	Liquid-Filled Projectile Design
170(S)	Armor and Its Applications (U)
175	Solid Propellants, Part One
176	*Solid Propellants, Part Two
177	Properties of Explosives of Military Interest
178	*Properties of Explosives of Military Interest, Section 2 (Replaced by -177)
179	Explosive Trains
180	Principles of Explosive Behavior
181	Explosions in Air, Part One
182(SRD)	Explosions in Air, Part Two (U)
185	Military Pyrotechnics, Part One, Theory and Application
186	Military Pyrotechnics, Part Two, Safety, Procedures and Glossary
187	Military Pyrotechnics, Part Three, Properties of Materials Used in Pyrotechnic Compositions
188	Military Pyrotechnics, Part Four, Design of Ammunition for Pyrotechnic Effects
189	Military Pyrotechnics, Part Five, Bibliography
190	*Army Weapon System Analysis
191	System Analysis and Cost-Effectiveness
192	Computer Aided Design of Mechanical Systems, Part One
193	Computer Aided Design of Mechanical Systems, Part Two
195	*Development Guide for Reliability, Part One, Introduction, Background, and Planning for Army Materiel Requirements
196	Development Guide for Reliability, Part Two, Design for Reliability
197	Development Guide for Reliability, Part Three Reliability Prediction
198	Development Guide for Reliability, Part Four, Reliability Measurement
199	*Development Guide for Reliability, Part Five, Contracting for Reliability
200	Development Guide for Reliability, Part Six, Mathematical Appendix and Glossary
201	Helicopter Engineering, Part One, Preliminary Design
202	*Helicopter Engineering, Part Two, Detail Design
203	Helicopter Engineering, Part Three, Qualification Assurance
204	Helicopter Performance Testing
205	Timing Systems and Components
210	Fuzes
211(C)	Fuzes, Proximity, Electrical, Part One (U)
212(S)	Fuzes, Proximity, Electrical, Part Two (U)
213(S)	Fuzes, Proximity, Electrical, Part Three (U)
214(S)	Fuzes, Proximity, Electrical, Part Four (U)
215(C)	Fuzes, Proximity, Electrical, Part Five (U)
235	Hardening Weapon Systems Against RF Energy
237	*Mortar Weapon Systems
238	*Recoilless Rifle Weapon Systems
239	*Small Arms Weapon Systems
240(C)	Grenades (U)
242	Design for Control of Projectile Flight Characteristics (Replaces -246)
244	Ammunition, Section 1, Artillery Ammunition--General, with Table of Contents, Glossary, and Index for Series
245(C)	Ammunition, Section 2, Design for Terminal Effects (U)
246	*Ammunition, Section 3, Design for Control of Flight Characteristics (Replaced by -242)
247	Ammunition, Section 4, Design for Projection
248	*Ammunition, Section 5, Inspection Aspects of Artillery Ammunition Design
249	Ammunition, Section 6, Manufacture of Metallic Components of Artillery Ammunition
250	Guns--General
251	Muzzle Devices
252	**Gun Tubes
253	*Breech Mechanism Design
255	Spectral Characteristics of Muzzle Flash
260	Automatic Weapons
270	Propellant Actuated Devices
280	Design of Aerodynamically Stabilized Free Rockets
281(SRD)	Weapon System Effectiveness (U)
282	*Propulsion and Propellants (Replaced by -285)
286	Structures
290(C)	Warheads--General (U)
291	*Surface-to-Air Missiles, Part One, System Integration
292	*Surface-to-Air Missiles, Part Two, Weapon Control
293	*Surface-to-Air Missiles, Part Three, Computers
294(C)	*Surface-to-Air Missiles, Part Four, Missile Armament (U)
295(C)	*Surface-to-Air Missiles, Part Five, Countermeasures (U)
296	*Surface-to-Air Missiles, Part Six, Structures and Power Sources
297(C)	*Surface-to-Air Missiles, Part Seven, Sample Problem (U)
300	Fabric Design
312	Rotational Molding of Plastic Powders
313	Short Fiber Plastic Base Composites
327	Fire Control Systems--General
329	Fire Control Computing Systems
331	Compensating Elements
335(SRD)	*Design Engineers' Nuclear Effects Manual (DENEM), Volume I, Munitions and Weapon Systems (U)
336(SRD)	*Design Engineers' Nuclear Effects Manual (DENEM), Volume II, Electronic Systems and Logistical Systems (U)
337(SRD)	*Design Engineers' Nuclear Effects Manual (DENEM), Volume III, Nuclear Environment (U)
338(SRD)	*Design Engineers' Nuclear Effects Manual (DENEM), Volume IV, Nuclear Effects (U)
340	Carriages and Mounts--General
341	Cradles
342	Recoil Systems
343	Top Carriages
344	Bottom Carriages
345	Equilibrators
346	Elevating Mechanisms
347	Traversing Mechanisms
350	Wheeled Amphibians
355	The Automotive Assembly
356	Automotive Suspensions
357	Automotive Bodies and Hulls
360	Military Vehicle Electrical Systems
361	Military Vehicle Power Plant Cooling
410	*Electromagnetic Compatibility (EMC)
411(S)	*Vulnerability of Communication-Electronic and Electro-Optical Systems (Except Guided Missiles) to Electronic Warfare, Part One, Introduction and General Approach to Electronic Warfare Vulnerability (U)
412(C)	*Vulnerability of Communication-Electronic and Electro-Optical Systems (Except Guided Missiles) to Electronic Warfare, Part Two, Electronic Warfare Vulnerability of Tactical Communications (U)
413(S)	*Vulnerability of Communication-Electronic and Electro-Optical Systems (Except Guided Missiles) to Electronic Warfare, Part Three, Electronic Warfare Vulnerability of Ground-Based and Airborne Surveillance and Target Acquisition Radars (U)
414(S)	*Vulnerability of Communication-Electronic and Electro-Optical Systems (Except Guided Missiles) to Electronic Warfare, Part Four, Electronic Warfare Vulnerability of Avionics (U)
415(S)	*Vulnerability of Communication-Electronic and Electro-Optical Systems (Except Guided Missiles) to Electronic Warfare, Part Five, Optical/Electronic Warfare Vulnerability of Electro-Optic Systems (U)
416(S)	*Vulnerability of Communication-Electronic and Electro-Optical Systems (Except Guided Missiles) to Electronic Warfare, Part Six, Electronic Warfare Vulnerability of Satellite Communications (U)
445	Sabot Technology Engineering
470	*Metric Conversion Guide for Military Applications

* UNDER REVISION -- not available

[illegible]