23

Reliability of Phased-mission Systems

Liudong Xing[1] and Suprasad V. Amari[2]

[1] Department of Electrical and Computer Engineering, University of Massachusetts – Dartmouth, USA
[2] Relex Software Corporation, Greensburg, USA

Abstract: In this chapter, a state-of-the-art review of various analytical modeling techniques for reliability analysis of phased-mission systems (PMS) is presented. The analysis approaches can be broadly classified into three categories: combinatorial, state-space oriented, and modular. The combinatorial approaches are computationally efficient for analyzing static PMS. A combinatorial binary decision diagram based method is discussed in detail. Methods to consider imperfect fault coverage and common-cause failures in the reliability analysis of PMS will also be presented.

23.1 Introduction

The operation of missions encountered in aerospace, nuclear power, and many other applications often involves several different tasks or phases that must be accomplished in sequence. Systems used in these missions are usually called phased-mission systems (PMS). A classic example is an aircraft flight that involves take-off, ascent, level-flight, descent, and landing phases. During each mission phase, the system has to accomplish a specified task and may be subject to different stresses as well as different dependability requirements. Thus, system configuration, success criteria, and component failure behavior may change from phase to phase [1]. This dynamic behavior usually requires a distinct model for each phase of the mission in the reliability analysis. Further complicating the analysis are statistical-dependencies across the phases for a given component. For example, the state of a component at the beginning of a new phase is identical to its state at the end of the previous phase in a non-repairable PMS [2]. The consideration of these dynamics and dependencies poses unique challenges to existing analysis methods.

Considerable research efforts have been expended in the reliability analysis of PMS over the past three decades. Generally, there are two classes of approaches to the evaluation of PMS: analytical modeling [1–5] and simulation [6, 7]. Simulation typically offers greater generality in system representation, but it is often more expensive in computational requirements [5]. On the other hand, analytical modeling techniques can incorporate a desirable combination of flexibility in representation as well as ease of solution. The analytical modeling approaches can be further categorized into three classes: state space oriented models [3, 5, 8–10], combinatorial methods [1, 2, 4, 11, 12], and a phase modular solution [13–16] that combines the former two methods as appropriate. The state space oriented approaches,

which are based on Markov chains and/or Petri nets, are flexible and powerful in modeling complex dependencies among system components. However, they suffer from state explosion when modeling large-scale systems. With an effort to deal with the state explosion problem of the state space oriented approaches, some researchers proposed combinatorial methods, which exploit Boolean algebra and various forms of decision diagrams to achieve low computational complexity and less storage space consumption.

This chapter will give a state-of-the-art review of the various analytical modeling methods. It then focuses on a combinatorial binary decision diagrams based method for the reliability analysis of a class of generalized PMS (GPMS). Traditionally in a PMS, the mission is assumed to fail if the system fails during any one phase [17]. GPMS extends this phase-OR failure requirement to the more general combinatorial phase requirement (CPR) [1]. The outcome of the GPMS may also exhibit multiple performance levels between binary outcome (success or failure). Methods to consider imperfect fault coverage and common-cause failures in the reliability analysis of GPMS will also be discussed in this chapter.

23.2 Types of Phased-mission Systems

PMS can be categorized in several ways:

- *Static versus dynamic PMS:* If the structure of the reliability model for any phase of PMS is combinatorial, *i.e.*, the failure of the mission in any phase depends only on the combinations of component failure events, the PMS is said to be static. If the order in which the component failure events occur affects the outcome, *i.e.*, the failure of the mission in any one phase depends on both the combinations of the component failure events and sequence of occurrence of input events, the PMS is said to be dynamic. Systems involving functional dependencies and/or spares management are also dynamic. In Section 23.3, various approaches to the analysis of static and dynamic PMS will be presented.
- *Repairable versus non-repairable PMS:* In a non-repairable PMS, once a component has failed in one phase, it remains failed in all later phases. In a repairable system, the state of the system depends on the failure characteristics of its components as well as the maintenance plans that are conducted on the system. Maintenance can be classified into three categories according to the reason why it is conducted [13]: 1) *failure-driven maintenance* occurs when maintaining a system upon the occurrence of a component failure; 2) *time-driven maintenance* is performed on a pre-determined schedule; and 3) *condition-driven maintenance* is performed based on the observed condition of a system, for example, a component is repairable whenever the component fails and the system does not fail; no repair is possible upon the system failure. Meshkat [13] investigated these maintenance plans and analysis of PMS with certain kinds of time-driven maintenance. Xing [18] studied the dependability modeling and analysis of PMS with the failure-driven maintenance and the scheduled maintenance. This chapter will focus on the reliability modeling and analysis of non-repairable PMS.
- *Coherent versus non-coherent PMS:* In a coherent PMS, each component contributes to the system state, and the system state worsens (at lease does not improve) with an additional component failure [19]. On the other hand, the structure function of a noncoherent system does not increase monotonically with the additional number of functioning components. Specifically, a noncoherent system can transit from a failed state to a good state by the failure of a component, or transit from a good state to a failed state by the repair of a component. In other words, both component failures and repairs can contribute to the system failure in a noncoherent system. The failure behavior of a noncoherent PMS can be described using noncoherent fault trees, which are characterized by inverse gates (for example, NOT and exclusive-OR gates) besides logic gates used in coherent fault trees. This chapter will focus on coherent PMS.

- *Series/phase-OR PMS versus combinatorial phase requirements (CPR):* In a series PMS, the entire mission fails if the system fails during any one phase [17]. For a PMS with CPR, its failure criterion can be expressed as a logical combination of phase failures in terms of phase-AND, phase-*K*-out-of-*N*, and phase-OR. Thus, a phase failure does not necessarily lead to a mission failure; it may just produce degraded performance of the mission [1].
- *Sequential versus dynamic choice of mission phases:* In a sequential PMS, the sequence of phases traversed by the system to accomplish its goals is always constituted by a single path from the first phase to the last one. Most of existing PMS analysis techniques focuses on the sequential PMS. There are indeed examples of PMS for which the sequence of phases is better represented by a more generic direct acyclic graph [9]. In this scenario, at the end of a phase, the next phase may be selected according to a probability distribution, or depending on the current internal state of the PMS. Methods for considering the probabilistic choice of mission phases were presented in [8, 9]. A brief discussion on these methods is also given in Section 23.3.2.2.

23.3 Analytical Modeling Techniques

Three classes of analytical approaches to the reliability analysis of coherent PMS are described in this section. Section 23.3.1 presents the combinatorial approaches to the analysis of static PMS. Section 23.3.2 presents the state space oriented methods. Section 23.3.3 presents the phase modular approach, which provides a combination of combinatorial solution for static phase modules and Markov chain solution for dynamic phase modules.

23.3.1 Combinatorial Approaches

Combinatorial methods for analyzing PMS assume that all components fail/behave *s*-independently within each phase. However, they deal with the *s*-dependence across phases for a given component.

23.3.1.1 The Mini-component Technique

Esary and Ziehms [4] proposed to deal with the *s*-dependence across phases by replacing the component in each phase with a system of components (called *mini-components*), performing *s*-independently and in series. For example, a component *A* in phase *j* of a non-repairable PMS is replaced by a set of *s*-independent mini-components $\{a_i\}_{i=1}^{j}$ in series. The relation between a component and its mini-components is: $A_j = a_1 \bullet a_2 \bullet \ldots \bullet a_j$, meaning that *A* is operational in phase *j* (represented by $A_j = 1$ or $\overline{A}_j = 0$) if and only if it has functioned in all the previous phases.

Figure 23.1 shows the reliability block diagram (RBD) and fault tree (FT) format of the mini-component solution. Esary and Ziehms [4] showed that reliability of the resulted new system after the above transformation is the same as the reliability of the original PMS. Most importantly, the evaluation of the new system can proceed without considering the *s*-dependence across phases for a given component.

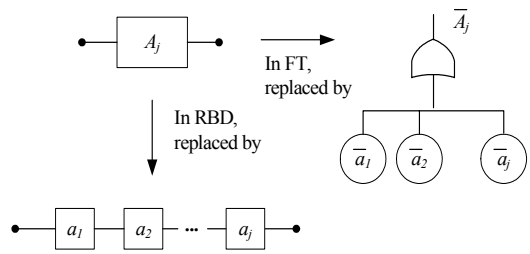


Figure 23.1. Mini-component method

Let *A(t)* be the state indicator variable of component *A*, and $q_{a_i}(t)$ be the failure function of mini-component a_i for component *A* in phase *i*, which conditionally depends on the survival of phase (*i*-1). The relationship between *A(t)* and $q_{a_i}(t)$ is:

$$q_{a_i}(t) = \begin{cases} \Pr(A(t) = 0) & i = 1, \\ \Pr(A(t + T_{i-1}) = 0 \mid A(T_{i-1}) = 1) & 1 < i \le j, t \le T_i. \end{cases} \quad (23.1)$$

In the system-level reliability analysis, $q_{a_i}(t)$ is given as the system input in the form of a

conditional failure distribution conditioned on the success of a_{i-1}.

Consider an example PMS with three components (A, B, and C) used in three non-overlapping consecutive phases (adapted from [3]). Figure 23.2 shows the failure criteria in each phase of the PMS in fault trees. In Phase 1, the system fails if all the three components fail. In Phase 2, the system fails if A fails or both B and C fail. In Phase 3, the system fails if any of the three components fails.

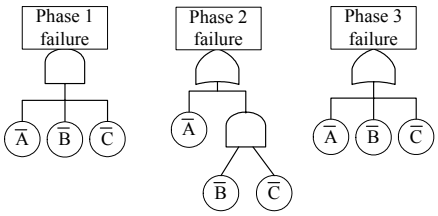


Figure 23.2. Fault tree model of a three-phase PMS

Figure 23.3 shows the equivalent system fault tree model in the mini-component method. Clearly the difficulty with this method is that the size of the problem becomes very large as the number of phases increases, for which a solution can be computationally very expensive.

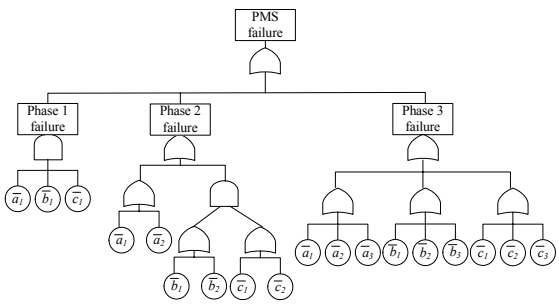


Figure 23.3. Equivalent mini-component system

23.3.1.2 The Boolean Algebraic Method

Another solution to the phased mission problem is to connect multiple phase models in series. Figure 23.4 shows the equivalent system at the end of mission in the Boolean algebraic method for the example PMS in Figure 23.2.

Based on the relation between a component and its mini-components, the failure function for component A in phase j can be calculated from $q_{a_i}(t)$ as (23.2):

$$F_{A_j}(t) = \begin{cases} q_{a_j}(t) & j = 1, \\ [1 - \prod_{i=1}^{j-1}(1 - q_{a_i}(T_i))] + [\prod_{i=1}^{j-1}(1 - q_{a_i}(T_i))] \bullet q_{a_j}(t) & j > 1 \end{cases} \quad (23.2)$$

where time t is measured from the beginning of phase j so that $0 \le t \le T_j$. T_j is the duration of phase j. The first term in (23.2) when $j > 1$ represents the probability that component A has already failed in the previous phases (1, 2, …, j-1). The second term denotes the probability distribution of lifetime of A in phase j.

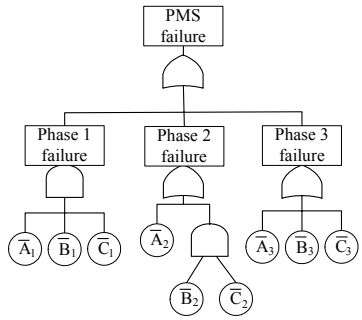


Figure 23.4. Example PMS in the Boolean algebra method

Because s-dependence exists among the same component in different phases, special treatment is needed for combination terms containing more than one A_i, $1 \le i \le m$, where m represents the total number of phases in the PMS. A set of Boolean algebraic rules called phase algebra rules was proposed to deal with the dependence (Table 23.1) [11, 20].

Table 23.1. Rules of phase algebra *($i < j$)*

$A_i \bullet A_j \to A_j$	$\overline{A}_i + \overline{A}_j \to \overline{A}_j$
$\overline{A}_i \bullet \overline{A}_j \to \overline{A}_i$	$A_i + A_j \to A_i$
$\overline{A}_i \bullet A_j \to 0$	$A_i + \overline{A}_j \to 1$

The phase algebra rules can be proved using the relation between the component and its mini-components ($A_j = a_1 \bullet a_2 \bullet \ldots \bullet a_j$) [2]:

- "$A_i \bullet A_j \to A_j$": the event "*A* is operational in phase *i* and the later phase *j*" is equivalent to the event "*A* is operational in the later phase *j*".

$$A_i \bullet A_j = (a_1 \bullet a_2 \bullet ... a_i)(a_1 \bullet a_2 \bullet ... a_j)$$
$$= a_1 \bullet a_2 \bullet ... a_j = A_j$$

- "$\overline{A}_i \bullet \overline{A}_j \to \overline{A}_i$": the event "*A* has failed in phase *i* and the later phase *j*" is equivalent to the event "*A* has failed in phase *i*".

$$\overline{A}_i \bullet \overline{A}_j = (a_1 \bullet a_2 \bullet ... a_i) \bullet \overline{(a_1 \bullet a_2 \bullet ... a_j)}$$
$$= \overline{a_1 \bullet a_2 \bullet ... a_i + a_1 \bullet a_2 \bullet ... a_j}$$
$$= \overline{a_1 \bullet a_2 \bullet ... a_i} = \overline{A}_i$$

- "$\overline{A}_i \bullet A_j \to 0$": the event "*A* has failed in phase *i*, but is operational in the later phase *j*" does not exist for a non-repairable PMS.

$$\overline{A}_i \bullet A_j = (\overline{a_1 \bullet a_2 \bullet ... a_i})(a_1 \bullet a_2 \bullet ... a_j)$$
$$= (\overline{a}_1 + \overline{a}_2 + ... + \overline{a}_i)(a_1 \bullet a_2 \bullet ... a_j) = 0$$

The three rules in the right column of Table 23.1 are just the complementary form of the rules in the left column, which have been proved in the above. Phase algebra rules do not account for $A_i \bullet \overline{A}_j$ and $\overline{A}_i + A_j$ combinations [2, 20]. $A_i \bullet \overline{A}_j$ means that *A* is operational until the end of phase *i* and then fails sometime between the end of phase *i* and the end of phase *j*; $\overline{A}_i + A_j$ has no physical meaning without considering repair. These phase algebra rules apply only to variables belonging to the same component.

23.3.1.3 Binary Decision Diagrams

Zang *et al.* [2] proposed a binary decision diagram (BDD) based method for the reliability analysis of static PMS with phase-OR requirement. As the first step of the method, phase algebra rules (Table 23.1) combined with heuristic variable ordering strategies are used to generate the PMS BDD model. Two types of ordering strategies were explored for variables that represent the same component in different phases: *forward* and *backward*. Thus, two types of phase-dependent operations (PDO) were proposed: *forward PDO* in which the variable order is the same as the phase order, and *backward PDO* in which the variable order is the reverse of the phase order. It is shown in [2] that in the PMS BDD generated by the backward PDO, the 0-edge always links two variables belonging to different components and the cancellation of common components can be done automatically during the generation of the BDD without any additional operation. So, the backward PDO is preferred in the PMS analysis.

After generating the PMS BDD, a recursive evaluation of the resulting PMS BDD yields the reliability/unreliability of the PMS. Special treatments are needed in the evaluation to deal with dependence among variables of the same component but different phases. The above BDD-based method [2] will be discussed in detail in Section 23.4.1. BDD-based methods for analyzing generalized PMS subject to imperfect fault coverage, modular imperfect coverage, and common-cause failures will also be discussed in Sections 23.4.2, 23.4.3, and 23.4.4, respectively.

23.3.2 State Space Based Approaches

Traditionally, if the failure criteria in any one phase of the PMS are dynamic, then a state space based approach must be used for the entire PMS. Section 23.3.2.1 presents Markov chains based methods for the reliability analysis of dynamic PMS. Section 23.3.2.2 presents Petri nets based methods for dynamic PMS analysis.

23.3.2.1 Markov Chains

Several different Markov chain based methods are available for the reliability analysis of PMS. The basic idea is to construct a single Markov chain to represent the failure behavior of the entire PMS or several Markov chains, each representing the failure behavior in each phase. These Markov models at once account for dependence among components within a phase as well as dependence across phases for a given component. Solving the Markov chain models yields the probability of the system being in each state. The system unreliability is obtained by summing all the failure state probabilities.

Specifically, Smotherman and Zemoudeh [5] (*SZ approach*) used a single non-homogeneous Markov chain model to perform the reliability analysis of a PMS. In their approach, the behavior of the system in each phase is represented using a different Markov chain, which may contain a different subset of states. The state transitions are described in terms of time dependent rates so as to include phase changes. Thus, state-dependent phase changes, random phase durations, time-varying failure and repair behavior can be easily modeled in the SZ approach.

Consider the example PMS in Figure 23.2. Assume the failure rates of the three components *A*, *B*, and *C* are *a*, *b*, and *c*, respectively. Figure 23.5 shows the Markov chain model of the entire PMS in the SZ approach. In the Markov chain representation, a 3-tuple represents a state indicating the status of the three components. A "1" represents the corresponding component is operational and a "0" represents the corresponding component has failed. For example, state (110) implies that *A* and *B* are operational and *C* has failed. A "F" represents a state in which the system has failed. A transition from one state to another state is associated with the failure rate of the failed component. The transitions $h_i(t)$ in Figure 23.5 represent the failure rates associated with the time at which phase changes occur.

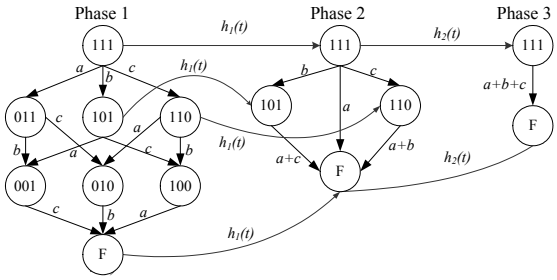


Figure 23.5. Markov chain model in the SZ approach

Since this model includes the configurations for all phases as well as the phase changes, it needs only be solved once. The major drawback of this approach, like the mini-component approach [4], is that a large overall model is needed. The size of the state space is as large as the sum of number of states in each of the individual phases. Since the state space associated with a Markov model of a system is exponential in the number of components in the worst case, the SZ method requires a large amount of storage and computational time to solve the model, thus limiting the type of system that can be analyzed.

Instead of generating and solving an overall Markov chain, Somani *et al*. [21] (*SRA approach*) suggested generating and solving separate Markov chains for individual phases of a PMS. The variation in failure criteria and system configuration from phase to phase is accommodated by providing an efficient mapping procedure at the transition time from one phase to another. While analyzing a phase, only states relevant to that phase are considered. Apparently, each individual Markov chain is much smaller than the overall Markov chain used in the SZ approach [5]. For the example three-phase PMS in Figure 23.2, Markov chains for the three phases are shown in Figure 23.5 (without considering the inter-phase mapping). In the SRA approach, three Markov chains with 8, 4, and 2 states, respectively, need to be solved. The reliability (or unreliability) of the system can be computed from the output of the last phase. While in the SZ approach, a single Markov chain with 12 states (after the three system failure states "F" are merged as one failure state) must be solved. Therefore, using the SRA approach, the computation time for large systems can be reduced significantly without compromising the accuracy of the results. Also, the SRA approach allows the phase duration to be modeled as fixed or random.

As another alternative to the reliability analysis of PMS using Markov models, Dugan [3] (*Dugan approach*) advocated generating a single Markov chain with state space equal to the union of the state spaces of the individual phases from the start. The transition rates are parameterized with phase numbers and the Markov chain is solved *n* times if the PMS has *n* phases. The final state probabilities of one phase become the initial state probabilities of the next phase. One potential source of the problem with the Dugan approach is that once a state is declared to be a system failure state in a phase, it cannot become an up state in a later phase. In practice, it is possible to have some states that are failure states in a phase but are up states in

a later phase. For example, if we swap the failure criteria of phase 1 and phase 3 in Figure 23.2, then the states of (011), (001), (010), and (100) are failure states in both phase 1 and phase 2, but are up states in phase 3. In the Dugan approach, all those states will be treated as forced failure states in phase 3. This problem would cause overestimated system unreliability.

23.3.2.2 Petri Nets

Mura and Bondavalli [9] (*MB approach*) proposed a hierarchical modeling and evaluation approach for analyzing PMS, where missions may evolve dynamically by selecting the next phase to perform according to the state of the system, and the duration of all phases are fixed and known in advance. Their approach combines the Markov analyses and Petri nets through a two-level modeling approach.

Specifically the upper level model in the MB approach is a single discrete-time Markov chain (DTMC), describing the overall behavior of the whole mission without any detail of the internal phase behavior. There are typically two absorbing states: loss of the entire mission and success of the mission. Each non-absorbing states in the DTMC represents a different phase in the mission. This allows simplifying the modeling of variety of mission scenarios by sequencing the phases in proper ways. Moreover, it allows probabilistic or dynamic choice of the mission phases according to the system states, which is not possible in other state space oriented approaches based only on Markov models. The lower level models are built using generalized stochastic Petri nets (GSPN). These lower level models are used to describe the system behavior inside each phase and they are built and solved separately. The separate modeling of each phase allows the reuse of the previously built models when the operation of a phase is repeated during the mission. The major advantages offered by the MB approach include the great flexibility by allowing the dynamic selection of mission phases and reusability of the defined phase models.

Later, Mura and Bondavalli [10] proposed a new methodology based on Markov regenerative stochastic Petri nets (MRSPN), which extended the MB approach by allowing random phase duration. This methodology is incorporated in the DEEM (dependability evaluation of multiple-phased systems) software package [8].

23.3.3 The Phase Modular Approach

Traditional approaches to PMS analysis are either combinatorial (Section 23.3.1) or state space based (Section 23.3.2). The combinatorial approaches are computationally efficient, but are applicable only when every phase of the PMS is static. Markov based approaches can capture the dynamic behavior such as functional dependencies, the sequence of failure events, and spares management. However, the major limitation with Markov methods is that if the failure criterion in only one phase is dynamic, then a Markov approach must be used for every phase. Due to the well-known state explosion problem of Markov approaches, it is often computationally intensive and even infeasible to solve the model.

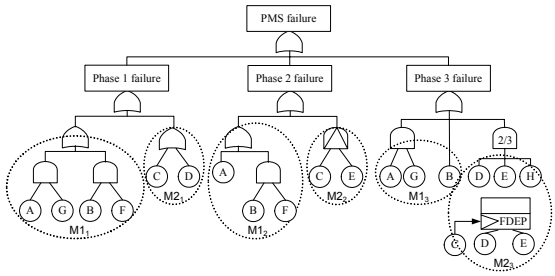


Figure 23.6. PMS fault free with defined modules

To take advantage of both types of solutions while addressing their limitations, a phase-modular fault tree approach employing both BDD and Markov-chain solution methods as appropriate was developed for the reliability analysis of PMS [13–16]. In applying this approach, first the modules of components that remain independent throughout the mission are identified, and then the reliability of each independent module in each phase is found using the appropriate solution technique. Finally, the modules are combined in a system-level BDD to find the system-level reliability. We illustrate the basic elements/steps of the phase-modular

approach using a simple example PMS, which has three phases and eight components (Figure 23.6) [22] as follows:

1) Represent each mission phase with a fault tree, and then link the phase fault trees with a system top event. For this example, the reliability of the PMS is the probability that the mission successfully achieves its objectives in all phases, the phase fault trees are linked using an OR gate to obtain the entire PMS fault tree.
2) Each phase fault tree is then divided into independent subtrees/modules. In Figure 23.6, Phase 1 fault tree has two main modules {A, G, B, F} and {C, D}. Phase 2 fault tree has two modules {A, B, F} and {C, E}. Phase 3 fault tree has three modules {A, G}, {B}, and {C, D, E, H}.
3) Characterize each phase module as static or dynamic. Static fault trees use only OR, AND, and *K*-out-of-*N* gates. Dynamic fault trees have at least one dynamic gate such as priority-AND gate, FDEP gate, or CSP/WSP/HSP gates. In Figure 23.6, both modules in Phase 1 fault tree are static; the module {A, B, F} in Phase 2 fault tree is static and the module {C, E} is dynamic; and Phase 3 fault tree has two static modules, {A, G} and {B}, and one dynamic module, {C, D, E, H}.
4) Identify each phase module as bottom-level (without child modules) or upper-level (with child modules). The module {C, D} in Phase 1 fault tree is a bottom-level module, and the module {A, G, B, F} is an upper-level module since it contains child modules {A, G} and {B, F} linked by an OR gate. The identification of child and parent modules is vital information used in solving for these modules' reliability.
5) Find the system-level independent modules. This identification is accomplished by finding the unions of components in all the phase modules that overlap in at least one component. The example PMS fault tree has two system-level independent modules, {A, B, F, G} and {C, D, E, H}.
6) Identify each system-level module as static or dynamic across the phases. Identification of a component as dynamic in at least one mission phase is sufficient for the identification of the corresponding system-level module as dynamic. In the example PMS, the system-level module {A, B, F, G} is static and {C, D, E, H} is dynamic.
7) Group the phase modules according to the corresponding system-level module. Components of {A, B, F, G} are labeled as $M1_i$ and components of {C, D, E, H} are labeled as $M2_i$, where i = mission phase (Figure 23.6). These are the modules that will be solved for the joint phase module probabilities.
8) Find the joint phase module probabilities for all system-level modules. The BDD method is used for modules that are static across all the phases, and the combined Markov chain method as presented in [13, 15] is used for modules identified as dynamic. Therefore, we can use the BDD method on the system-level module {A, B, F, G}; however, we must use the Markov chain method on the system-level module {C, D, E, H}.
9) Consider each module as a basic event of a static fault tree of the entire system and solve the corresponding fault tree using BDD to find the overall system reliability based on the reliability measures of the modules.

Each module's reliability is solved with a consideration of its own behavior in previous phases. For instance, for finding the reliability of $M1_2$, a combined BDD approach is used for $M1_1$ and $M1_2$; for finding the reliability of $M2_3$, the combined Markov chain approach is used for $M2_1$, $M2_2$, and $M2_3$. We then consider solving the static PMS fault tree with the basic events $M1_1$, $M2_1$, $M1_2$, $M2_2$, $M1_3$, and $M2_3$ using the combined BDD approach and the reliability measures for each individual phase module computed from previous steps. It is important to note that solving this simple PMS fault tree without using the modularization technique would involve solving a Markov chain with approximately 256 states, while the Markov chain involved in this example has a maximum of only 16 states. The phase-modular approach provides exact reliability measures for PMS with dynamic phases in an efficient manner. Readers may refer to [13, 15, 16] for more details about this approach.

23.4 BDD Based PMS Analysis

In this section, the binary decision diagrams (BDD) based approaches to the reliability analysis of PMS, PMS with imperfect fault coverage, and PMS with common-cause failures will be discussed. In the model for the BDD based PMS analysis, the following assumptions are made:

- Component failures are *s*-independent within each phase. Dependencies arise among different phases and different failure modes (when imperfect fault coverage is considered).
- Phase durations are deterministic.
- The system is not maintained during the mission; once a component transfers from the operation mode to a failure mode (either *covered* or *uncovered*), it will remain in that failure mode for the rest of the mission time.
- The system is coherent.

23.4.1 Traditional Phased-mission Systems

Reliability of a traditional phase-OR PMS is the probability that the mission successfully achieves the objective in all phases [17]. In the BDD-based method to the reliability analysis of PMS, three major steps are involved: 1) generating BDD for each phase fault tree, 2) combining single-phase BDD to obtain the entire PMS BDD, and 3) evaluating the PMS BDD to obtain the system reliability. Similar to the generation of BDD for non-PMS, the variable ordering can heavily affect the size of PMS BDD. Currently, there is no exact method of determining the best way of ordering basic events for a given fault tree structure. Fortunately, heuristics can usually be used to find a reasonable variable ordering.

In PMS, two kinds of variables need to be ordered: variables belonging to different components and variables that represent the same component in different phases. For the variables of different components, heuristics are typically used to find an adequate ordering. Several heuristics based on a depth-first search of the fault tree model can be found in [23]. For the variables of the same component in different phases, there are two ways to order them: *forward* and *backward*. In the forward method, the variable order is the same as the phase order, that is, $A_1 \prec A_2 \prec \ldots \prec A_m$, where A_i is the state variable of component A in phase i and m is the number of phases. In the backward method, the variable order is the reverse of the phase order, that is, $A_m \prec A_{m-1} \prec \ldots \prec A_1$.

After assigning each variable an index/order, for generating single-phase BDD in step 1), the traditional BDD operation rules based on Boolean algebra are applied. The reader may wish to review the traditional BDD operation rules in Chapter 38. In step 2), for combining single-phase BDD, dependence among variables of the same component but different phases is dealt with using the phase-dependent operation (PDO) [2]. According to the two ways to order variables of the same component, two types of PDO were developed: forward and backward. Assume component A is used in both phases i and j ($i < j$). A_i and A_j are state variables of A in phase i and phase j, respectively. A_i =0 or $\overline{A}_i$ =1 implies that A has failed in phase i. Using the *ite* format, the sub-BDD rooted at $\overline{A}_i$ and $\overline{A}_j$ respectively can be written as: $G = ite(\overline{A}_i, G_{\overline{A}_i=1}, G_{\overline{A}_i=0}) = ite(\overline{A}_i, G_1, G_2)$ and $H = ite(\overline{A}_j, H_{\overline{A}_j=1}, H_{\overline{A}_j=0}) = ite(\overline{A}_j, H_1, H_2)$. Let $\Diamond$ represent logic operation AND or OR, then we have:

$$G \Diamond H = ite(\overline{A}_i, G_1, G_2) \Diamond ite(\overline{A}_j, H_1, H_2) = \begin{cases} ite(\overline{A}_i, G_1 \Diamond H_1, G_2 \Diamond H) & \text{forward PDO} \\ ite(\overline{A}_j, G \Diamond H_1, G_2 \Diamond H_2) & \text{backward PDO} \end{cases} \quad (23.3)$$

The reader may refer to [2] for the proof of (23.3) using the phase algebra rules in Table 23.1. As discussed in Section 23.3.1.3, the backward PDO is preferred in the PMS analysis because in the PMS BDD generated by the backward PDO, the 0-edge always links two variables of different components and thus less dependence needs to be handled during the model evaluation.

Note that PDO of [2] is only applicable to non-repairable PMS. In addition, they can perform the task of combining BDD of individual phases into the overall PMS BDD correctly only given that the ordering strategies abide the following two rules:

- Orderings adopted in the generation of each single phase BDD are consistent or the same for all the phases.

- Orderings of variables that belong to the same component but to different phases stay together. In practice, this can be achieved by replacing each component indicator variable with a set of variables that represent this component in each phase after ordering components using heuristics.

These two rules are very stringent from the implementation point of view. Xing and Dugan relaxed the constraints by adding a removal procedure in the PMS BDD generation to allow arbitrary ordering strategies. For details, see [24].

After PMS BDD is generated, the final step to accomplish the reliability analysis is to evaluate the resulting PMS BDD. Note that 1-edges in the PMS BDD may link two variables of the same component but different phases. Dependence between these variables must be addressed during the evaluation. As a result, two different evaluation methods are needed for the PMS BDD generation. Specifically, consider the sub-BDD in Figure 23.7: The *ite* format is: $G = ite(x, G_1, G_2) = x \bullet G_1 + \bar{x} \bullet G_2$ $G_1 = ite(y, H_1, H_2) = y \bullet H_1 + \bar{y} \bullet H_2$. Let $p(x)$ be the failure probability of component represented by node x and P(G) be the unreliability with respect to the current sub-BDD rooted at node x. The recursive evaluation algorithm of PMS BDD is as follows:

- For 1-edge or 0-edge linking variables of different components, the evaluation method is the same as the ordinary BDD. For example, if x, y in Figure 23.7 belong to different components, the evaluation method is:

$$P(G)=P(G_1)+[1-p(x)]*[P(G_2)-P(G_1)] \quad (23.4)$$

- For 1-edge linking variables of the same component, for example, if x, y in Figure 23.7 belong to the same component, the evaluation method is:

$$P(G)= P(G_1)+[1-p(x)]* [P(G_2)-P(H_2)] \quad (23.5)$$

The phase algebra rules (Table 23.1) are applied to deal with the dependence between x and y in the derivation of (23.5). Refer to [2] for details of the derivation. Exit conditions of the recursive algorithm are: if G = 0, *i.e.*, the system is operational, then the unreliability P(G) = 0; if G = 1, *i.e.*, the system has failed, then P(G) = 1.

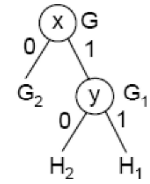


Figure 23.7. A PMS BDD branch

23.4.2 PMS with Imperfect Coverage

PMS, especially those devoted to safety-critical applications, such as aerospace and nuclear power, are typically designed with sufficient redundancies and automatic recovery mechanisms to be tolerant of faults or errors that may occur. However, the recovery mechanisms can fail, such that the system cannot adequately detect, locate, and recover from a fault occurring in the system. This uncovered fault can propagate through the system and may lead to an overall system failure, despite the presence of fault-tolerant mechanisms. As discussed in Chapter 22, the imperfect coverage (IPC) [25, 26] introduces multiple failure modes (*covered failure* and *uncovered failure*) that must be considered for accurate reliability analysis of fault-tolerant PMS. A covered component failure is local to the affected component; it may or may not lead to the system failure depending on the system configuration, failure criteria, and remaining redundancy. An uncovered component failure is globally malicious, and causes the system to crash.

This section presents a BDD-based approach called GPMS-CPR [1] for the reliability analysis of PMS with IPC, while considering the CPR and multiple performance levels for GPMS. The IPC behavior will be modeled using the fault/error handling model (FEHM) described in Figure 22.1. However, the near-coincident failure exit is not considered here. The probabilities of the three mutually exclusive exits R, C, and S in the FEHM are denoted as: r, c, and s, where $r + c + s = 1$.

The basic idea of the GPMS-CPR is to separate all the component uncovered failures from the combinatorics of the solution based on the simple and efficient algorithm (SEA) [1, 27] (Chapter 22) and the mini-component technique (Section 23.3.1.1). SEA represents a separable scheme for

incorporating IPC in the reliability analysis of single-phase systems. It cannot directly apply to PMS with *s*-dependence across phases. The mini-component concept can deal with the across-phase dependence. The basics of GPMS-CPR are to convert the PMS to an equivalent mini-component system so as to remove *s*-dependence, and then apply the SEA approach to address IPC. Figure 23.8 illustrates the GPMS-CPR approach.

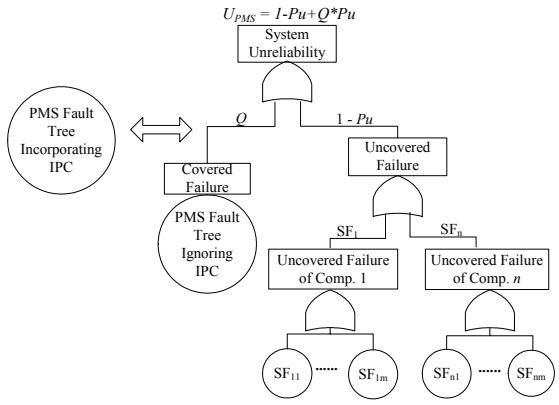


Figure 23.8. The separable GPMS-CPR approach

In Figure 23.8, SF_A denotes an event that component A fails uncovered. SF_A for different components are *s*-independent. SF_{a_i} represents an event that mini-component a_i fails uncovered. Different SF_{a_i} (i = 1,…,m) for the same component are not independent and the dependence must be addressed in the solution. The probability of no mini-component experiencing an uncovered failure (P_u) and the unreliability of the complementary perfect-coverage system (Q) are integrated using the total probability theorem:

$$U_{PMS} = 1 - P_u + Q * P_u \tag{23.6}$$

The derivation of (23.6) is similar to the derivation of the SEA in Chapter 22. Also, refer to [1] for details. The formulation of P_u in (23.6) is:

$$\begin{aligned} P_u &= \Pr(\overline{SF}_1 \cap \overline{SF}_2 \cap \ldots \cap \overline{SF}_n) \\ &= \prod_{A=1}^{n}(1 - \Pr(SF_A)) = \prod_{A=1}^{n}(1 - u[A]) \\ &= \prod_{A=1}^{n}(1 - u[A_m]) \end{aligned} \tag{23.7}$$

where n is the total number of components in the PMS, $u[A]$ is the probability that component A fails uncovered during the whole mission, that is, $u[A_m]$ is the probability that A has failed uncovered before the end of the last phase m. Let NF_{a_i}, CF_{a_i}, and SF_{a_i} denote events that A in phase i, namely, mini-component a_i does not fail, fails covered, and fails uncovered, respectively. The three events are mutually exclusive and complete. Define $n[a_i]$ = $\Pr(NF_{a_i})$, $c[a_i]$ = $\Pr(CF_{a_i})$, and $u[a_i]$ = $\Pr(SF_{a_i})$. According to the FEHM in Figure 22.1, these three probabilities can be calculated as:

$$\begin{aligned} n[a_i] &= 1 - q_{a_i}(t) + r_{a_i} \bullet q_{a_i}(t) \\ c[a_i] &= c_{a_i} \bullet q_{a_i}(t) \\ u[a_i] &= s_{a_i} \bullet q_{a_i}(t) \end{aligned} \tag{23.8}$$

Based on the relationship between a component and its mini-components depicted in Section 23.3.1.1 and on the fact that a component can fail uncovered in one phase only if it has survived all the previous phases, $u[A_j]$ can be calculated as:

$$\begin{aligned} u[A_j] &= \Pr(SF_{A_j}) \\ &= \Pr(A \text{ fails uncovered before the end of phase j}) \\ &= \Pr(\text{any mini - component } a_{i\in\{1\ldots j\}} \text{ fails uncovered}) \\ &= \Pr(SF_{a_1} \cup (NF_{a_1} \cap SF_{a_2}) \cup \ldots \\ &\qquad \cup (NF_{a_1} \cap \ldots \cap NF_{a_{j-1}} \cap SF_{a_j})) \\ &= u[a_1] + n[a_1] \bullet u[a_2] + \ldots \\ &\qquad + n[a_1] \bullet n[a_2] \bullet \ldots \bullet n[a_{j-1}] \bullet u[a_j] \\ &= u[a_1] + \sum_{i=2}^{j}(\prod_{k=1}^{i-1} n[a_k]) \bullet u[a_i] \end{aligned} \tag{23.9}$$

where j = 1, $u[A_1] = u[a_1]$. Similarly, the covered failure probability $c[A_j]$ and non-failure probability $n[A_j]$ can be calculated as in (23.10) and (23.11), respectively, when j = 1, $c[A_1 = c[a_1]$, $n[A_1] = n[a_1]$.

Next, consider the evaluation of perfect-coverage unreliability Q in (23.6). According to the SEA method, Q should be evaluated given that no component experiences an uncovered failure.

$$
\begin{aligned}
&c[A_j] = \Pr(CF_{A_j}) \\
&= \Pr(A \text{ fails covered before the end of phase j}) \\
&= \Pr(\text{any mini - component } a_{i\in\{1...j\}} \text{ fails covered}) \\
&= \Pr(CF_{a_1} \cup (NF_{a_1} \cap CF_{a_2}) \cup ... \\
&\qquad \cup (NF_{a_1} \cap ... \cap NF_{a_{j-1}} \cap CF_{a_j})) \\
&= c[a_1] + n[a_1] \bullet c[a_2] + ... \\
&\qquad + n[a_1] \bullet n[a_2] \bullet ... \bullet n[a_{j-1}] \bullet c[a_j] \\
&= c[a_1] + \sum_{i=2}^{j} (\prod_{k=1}^{i-1} n[a_k]) \bullet c[a_i]
\end{aligned}
\tag{23.10}
$$

$$
\begin{aligned}
&n[A_j] = \Pr(NF_{A_j}) \\
&= \Pr(A \text{ has not failed before the end of phase j}) \\
&= \Pr(\text{all mini - components } a_{i\in\{1...j\}} \text{ are not failed}) \\
&= \Pr(NF_{a_1} \cap ... \cap NF_{a_{j-1}} \cap NF_{a_j})) \\
&= n[a_1] \bullet n[a_2] \bullet ... \bullet n[a_{j-1}] \bullet n[a_j] = \prod_{i=1}^{j} n[a_i]
\end{aligned}
\tag{23.11}
$$

Therefore, before evaluating Q, the failure function of each component A in each phase j needs to be modified as a conditional failure probability, denoted by $F_{A_j}(t)$, conditioned on there being no uncovered failure during the whole mission, that is,

$$
F_{A_j}(t) = \Pr(CF_{A_j} \mid \overline{SF}_A) = \frac{c[A_j]}{1 - u[A]} = \frac{c[A_j]}{1 - u[A_m]} \tag{23.12}
$$

Using these modified component failure functions, Q can be evaluated using the efficient PMS BDD method that does not consider IPC [2] (Section 23.4.1). In summary, GPMS-CPR can be described as the following five-step algorithm:

1) Compute the modified failure probability for each component at the end of each phase using (23.12).
2) Order components using backward PDO and heuristics. Generate BDD for each phase.
3) According to the specified CPR and mission performance criteria, combine the single-phase BDD using phase algebra and backward PDO to obtain the final PMS BDD.
4) Evaluate Q recursively from the final PMS BDD using the algorithm of Section 23.4.1 and using $F_{A_j}(t)$ generated in step (1) as the component failure probability.
5) Evaluate the imperfect coverage probability (1 - P_u). Then integrate it with Q using (23.6) to obtain final GPMS unreliability/performance.

Due to the nature of BDD and the beauty of the SEA method, the GPMS-CPR method has low computational complexity and is easy to implement, as compared to the other potential methods such as Markov chain based methods. The Markov methods can address IPC by expanding the state space and number of transitions, worsening the state explosion problem [28]. In addition, the GPMS-CPR is capable of evaluating a wider range of more practical systems with less restrictive mission requirements, while offering more human-friendly performance indices such as multi-level grading as compared to the previous PMS methods. Next, we consider the analysis of a data gathering PMS using GPMS-CPR.

23.4.2.1 The Data Gathering System and Analysis

A space data gathering system [1], which is loosely based on a practical system in NASA, consists of four types of components that are used in different configurations over three consecutive phases (Figure 23.9):

- A_a, A_b: needed for all phases; one of them must be functional during all the three phases.
- B_a: only needed for phases 1 and 2; it must be functional during these two phases.
- C_a, C_b: work during phases 1 and 3; both must be functional during phase 1, at least one of them must be functional during phase 3.
- D_a, D_b, D_c: work during phases 2 and 3; all of them must be functional during phase 2, at least two of them must be functional during phase 3.

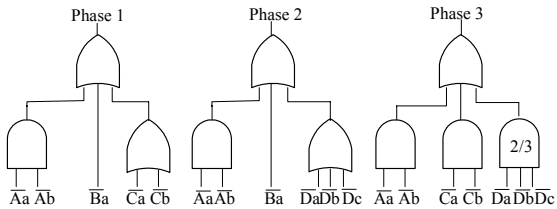


Figure 23.9. Data gathering system configuration

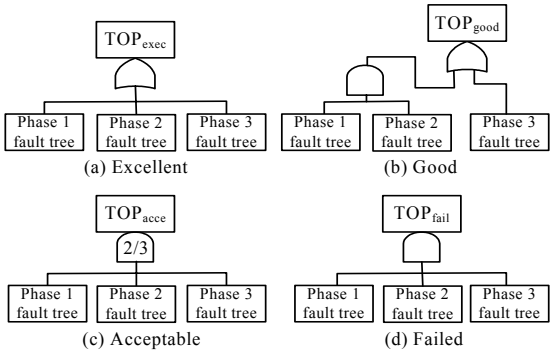


Figure 23.10. Four performance levels in the fault tree

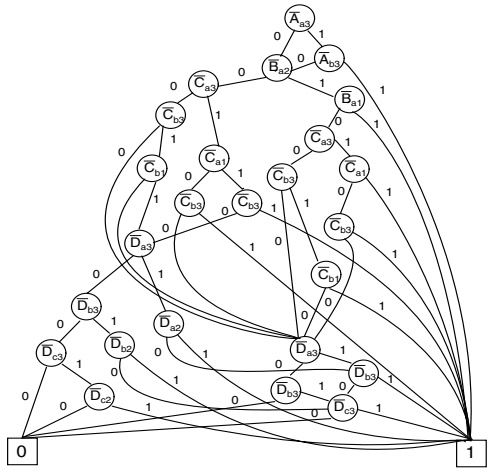


Figure 23.11. PMS BDD for the *good* level

According to the combination of data quality in the three phases, a four-performance-level result for the process can be defined as follows (Figure 23.10):

- *Excellent* level: data collection is successful in all the three phases.
- *Good* level: data collection is successful in phase 1 or 2 and in phase 3.
- *Acceptable* level: data collection is successful in only one of the three phases.
- *Failed* level: data collection fails in all the three phases.

Let P_{level} represent the multi-level reliability of the system, then we have:

$$P_{excellent} = 1 - \Pr(\text{TOP}_{\text{exce}}),$$
$$P_{good} = 1 - \Pr(\text{TOP}_{\text{good}}),$$
$$P_{acceptable} = \Pr(\text{TOP}_{\text{acce}}) - \Pr(\text{TOP}_{\text{fail}}),$$
$$P_{failed} = \Pr(\text{TOP}_{\text{fail}}). \quad (23.13)$$

For illustration purpose, the final PMS BDD for the *good* level is shown in Figure 23.11 [1]. The ordering of $A_a \prec A_b \prec B_a \prec C_a \prec C_b \prec D_a \prec D_b \prec D_c$ for variables of different components and backward ordering for variables of the same component and are used in the BDD generation.

By recursively traversing the PMS BDD of each performance level, the parameter Q in (23.6) is calculated. The U_{PMS}(level) is then found using (23.6). Lastly, the multi-level reliability P_{level} for each level is given as a simple and linear function of U_{PMS}(level) according to the corresponding grade-level performance criteria described in (23.13). Table 23.2 gives the input parameters (including phase duration, failure probabilities or rates, and coverage factors r, c, s) used in the analysis. Table 23.3 presents both the intermediate and final results for the analysis of the data gathering system.

Table 23.2. Input parameters (λ and λ_w are in 10^{-6}/hr; coverage factor r is 0 for all components in all phases)

Basic events	Phase 1 (33 hours)		Phase 2 (100 hours)		Phase 3 (67 hours)	
	p or λ	coverage c	p or λ	coverage c	p or λ	coverage c
A_a, A_b	0.0001	0.99	0.0001	0.99	0.0001	0.99
B_a	λ =1.5	0.97	λ =1.5	0.97	0.0001	0.97
C_a, C_b	0.0025	0.97	λ =1	0.99	λ_{Weibull} =1.6 α_{Weibull} =2	1
D_a, D_b, D_c	0.001	0.99	0.002	0.99	0.0001	0.97

Table 23.3. Analysis results of the data gathering system using GPMS-CPR

Performance level	Excellent	Good	Acceptable	Failed
P_u	0.999734	0.999734	0.999734	0.999734
Q	1.387e-2	1.261e-4	1.2602e-4	2.049e-7
$U_{PMS} = 1 - P_u + Q * P_u$.	0.0141326	3.9193e-4	3.9185e-4	2.6607e-4
Multi-level reliability: P_{level}	0.9858674	0.9996081	1.2578e-4	2.6607e-4

23.4.3 PMS with Modular Imperfect Coverage

In the traditional IPC, an uncovered component failure kills the entire mission. In the GPMS with CPR, however, the extent of the damage from an uncovered component fault can be just a phase loss, instead of the entire mission loss. Xing and Dugan proposed a generalized coverage model, called the modular imperfect coverage model (MIPCM) [29, 30], to exactly describe the behavior of a GPMS with CPR in the presence of a fault. As shown in Figure 23.12, MIPCM is a single entry, multiple exit black box. The model is activated when a fault occurs, and is exited when the fault is successfully handled or when the fault causes either a phase failure or the entire mission failure. The transient restoration exit R and permanent coverage exit C have the same meaning as in the traditional coverage model FEHM.

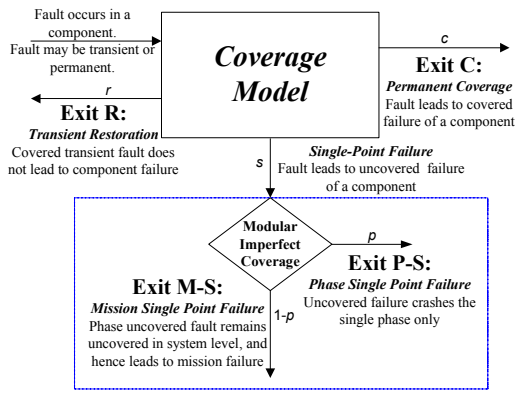


Figure 23.12. General structure of MIPCM

The following details the single-point failure exits. When a single fault (by itself) brings down a phase to which the fault belongs, single-point failure (or uncovered failure) is said to occur. Further, if such phase uncovered fault is covered at the higher system level, the phase single-point failure exit (labeled P-S) is reached, then a phase uncovered component failure occurs. If the phase uncovered fault remains uncovered at the system level, and hence leads to the failure of the entire mission, then the mission single-point failure exit (labeled M-S) is reached, and a mission uncovered failure is said to occur.

The four exits R, C, P-S, and M-S are mutually exclusive and complete. Define [*r, c, s*] to be the probability of taking the [transient restoration, permanent coverage, single-point failure] exit, given that a fault occurs, as in IPCM, and $r + c + s = 1$. Define p as a conditional probability that an uncovered fault fails a single phase, not the mission conditioned on an uncovered fault occurring in that phase. Then $s*p$ will be the probability of taking the P-S exit, and $s*(1-p)$ will be the probability of taking the M-S exit.

As compared with reliability analysis of PMS with traditional IPC, the analysis of GPMS with modular imperfect coverage (MIPC) is a more challenging task because the MIPC introduces more failure modes (covered failure, phase uncovered failure, and mission uncovered failure) and thus more dependencies into the system analysis. Building upon the above MIPCM, Xing and Dugan proposed two types of combinatorial methods for the reliability analysis of GPMS subject to MIPC: multi-state binary decision diagrams (MBDD) based method and ternary decision diagrams (TDD) based method. For each method, new phase algebra rules, new phase dependent operations for combining single-phase models into the overall system model, and new model evaluation algorithms were developed. The reader may refer to [29, 30] for the details of the MBDD-based method and the TDD-based method, respectively.

23.4.4 PMS with Common-cause Failures

Components in PMS can be subject to common-cause failures (CCF) during any phase of the mission. CCF are simultaneous component failures within a system that are a direct result of a common cause (CC) [31], such as extreme environmental conditions, design weaknesses, or human errors. It has been shown in many studies that the presence of CCF tends to increase a system's joint failure probabilities and thus contributes significantly to the overall unreliability of the system [32]. Therefore, it is crucial that CCF be modeled and analyzed appropriately. Considerable research efforts have been expended on the study of CCF for the system reliability analysis; refer to Chapter 38 for a discussion of various approaches, their contributions, and their limitations concerning the analysis of non-PMS. Actually, many of these limitations can also be found in the CCF models developed for PMS [33].

This section will present a separable solution that can address those limitations by allowing multiple CC to affect different subsets of system components and to occur *s*-dependently [34]. This separable approach is based on the efficient decomposition and aggregation (EDA) approach for the CCF analysis of single-phased systems (Chapter 38) and is easy to integrate into the existing PMS analysis methods.

Assume L_i elementary CC exists in each phase i of the PMS and they are denoted as: $CC_{11},.....,CC_{1L_1}$ for phase 1, $CC_{21},.....,CC_{2L_2}$ for phase 2, …, $CC_{m1},.....,CC_{mL_m}$ for the last phase m. Thus, total number of CC in PMS is: $L=\sum_{i=1}^{m}L_i$. According to the EDA approach, a common-cause event (CCE) space is built over a set of collectively exhaustive and mutually exclusive CCE that can occur in the PMS: $\Omega_{CCE}=\{CCE_1,CCE_2,...,CCE_{2^L}\}$. Each CCE in the set is a distinct and disjoint combination of elementary CC in the PMS:

$$CCE_1=\overline{CC_{11}}\cap...\cap\overline{CC_{1L_1}}\cap...\cap\overline{CC_{m1}}\cap...\cap\overline{CC_{mL_m}},$$
$$CCE_2=CC_{11}\cap...\cap\overline{CC_{1L_1}}\cap...\cap\overline{CC_{m1}}\cap...\cap\overline{CC_{mL_m}},$$
$$\cdots\cdots,$$
$$CCE_{2^L}=CC_{11}\cap...\cap CC_{1L_1}\cap...\cap CC_{m1}\cap...\cap CC_{mL_m}.$$

If $\Pr(CCE_j)$ denotes the occurrence probability of CCE_j, then $\sum_{j=1}^{2^L}\Pr(CCE_j)=1$ and $\Pr(CCE_i\cap CCE_j)=P(\phi)=0$ for any $i\neq j$.

As in the EDA approach, to find S_{CCE_j}, a set of components affected by event CCE_i is necessary. Define a common-cause group (CCG) as a set of components that are caused to fail due to the same CC. For non-PMS, S_{CCE_j} is simply the union of CCG whose corresponding CC occur. For example, assume $CCE_i=\overline{CC_1}\cap\overline{CC_2}\cap CC_3$ is a CCE in a non-PMS with three CC, S_{CCE_j} is simply equal to CCG_3 since CC_3 is the only active elementary CC. For a non-maintainable PMS, a component will remain failed in all later phases once it has failed in a phase. Therefore, S_{CCE_j} must be expanded to incorporate the affected components in all subsequent phases. The generation of S_{CCE_j} for PMS will be illustrated later on.

According to the total probability theorem, the unreliability of a PMS with CCF is calculated as:

$$U_{PMS}=\sum_{j=1}^{2^L}[\Pr(PMS\ fails\mid CCE_j)\Pr(CCE_j)] \quad (23.14)$$

$\Pr(PMS\ fails|CCE_j)$ in (23.14) is a conditional probability that the PMS fails conditioned on the occurrence of CCE_j. It is a reduced reliability problem, in which all components in S_{CCE_j} do not appear. Specifically, in the system fault tree model, each basic event appearing in S_{CCE_j} is replaced by a constant logic value "1" (true). After the replacement, a Boolean reduction can be applied to the PMS fault tree to generate a fault tree in which all components in S_{CCE_j} do not appear. Most importantly, the evaluation of the reduced problems can proceed without consideration of CCF. Thereby, the overall solution complexity is reduced.

Consider the *excellent* case of the data gathering PMS in Figure 23.9 with the following CCF scenario. The system is subject to CCF from hurricanes (denoted by CC_{11}) during phase 1, from lightning strikes (CC_{21}) during phase 2, and from floods (CC_{31}) during phase 3. A hurricane of sufficient intensity in phase 1 would cause A_a and C_a to fail, *i.e.*, $CCG_{11}=\{\overline{A_{a1}},\overline{C_{a1}}\}$, where A_{a1} is the

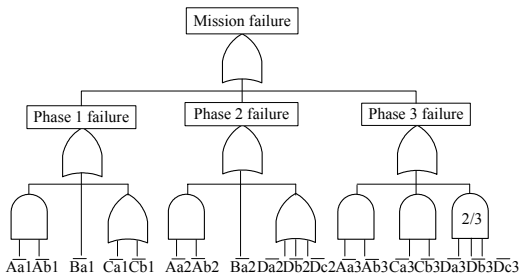
state indicator variable of component A_a in phase 1, and $\overline{A_{a1}}$ denotes the failure of A_a in phase 1. Serious lightning strikes in phase 2 would cause A_a, A_b, and B_a to fail, *i.e.*, $CCG_{21} = \{\overline{A_{a2}}, \overline{A_{b2}}, \overline{B_{a2}}\}$. Serious flooding in phase 3 would cause C_a and D_a to fail, *i.e.*, $CCG_{31} = \{\overline{C_{a3}}, \overline{D_{a3}}\}$. The probability of a hurricane occurring in phase 1 is $P_{CC_{11}} = 0.02$. The probability of a lightning strike occurring in phase 2 is $P_{CC_{21}} = 0.03$. Floods often occur in conjunction with hurricanes, and the *s*-dependence between the two CC can be defined by a set of conditional probabilities: the probability that floods occur in phase 3 conditioned on the occurrence of hurricanes in phase 1 is: $P_{CC_{31}|CC_{11}} = 0.6$. Similarly, $P_{CC_{31}|\overline{CC_{11}}} = 0.03$, $P_{\overline{CC_{31}}|CC_{11}} = 1 - P_{CC_{31}|CC_{11}}$, $P_{\overline{CC_{31}}|\overline{CC_{11}}} = 1 - P_{CC_{31}|\overline{CC_{11}}}$. These probabilities can typically be derived from available weather information.

Because there are three common causes in the example PMS, the CCE space is composed of $2^3 = 8$ CCE, as defined in the first column of Table 23.4. The second and third columns of the table show the set of components affected by each CCE (S_{CCE_j}) and occurrence probability calculation for each CCE based on statistical relation among those three CC, respectively.

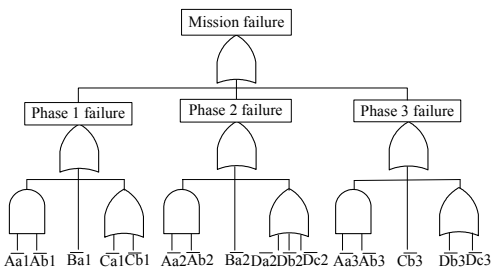
Table 23.4. CCE, affected components, and probabilities

CCE_i	S_{CCE_j}	$\Pr(CCE_i)$
$1: \overline{CC_{11}} \cap \overline{CC_{21}} \cap \overline{CC_{31}}$	ϕ	$P_{\overline{CC_{21}}} P_{\overline{CC_{11}}} P_{\overline{CC_{31}}\|\overline{CC_{11}}} = 0.9221$
$2: \overline{CC_{11}} \cap \overline{CC_{21}} \cap CC_{31}$	$\{\overline{C_{a3}}, \overline{D_{a3}}\}$	$P_{\overline{CC_{21}}} P_{\overline{CC_{11}}} P_{CC_{31}\|\overline{CC_{11}}} = 0.0285$
$3: \overline{CC_{11}} \cap CC_{21} \cap \overline{CC_{31}}$	$\{\overline{A_{a(2-3)}}, \overline{A_{b(2-3)}}, \overline{B_{a(2-3)}}\}$	$P_{CC_{21}} P_{\overline{CC_{11}}} P_{\overline{CC_{31}}\|\overline{CC_{11}}} = 0.0285$
$4: \overline{CC_{11}} \cap CC_{21} \cap CC_{31}$	$\{\overline{A_{a(2-3)}}, \overline{A_{b(2-3)}}, \overline{B_{a(2-3)}}, \overline{C_{a3}}, \overline{D_{a3}}\}$	$P_{CC_{21}} P_{\overline{CC_{11}}} P_{CC_{31}\|\overline{CC_{11}}} = 8.82e-4$
$5: CC_{11} \cap \overline{CC_{21}} \cap \overline{CC_{31}}$	$\{\overline{A_{a(1-3)}}, \overline{C_{a(1-3)}}\}$	$P_{\overline{CC_{21}}} P_{CC_{11}} P_{\overline{CC_{31}}\|CC_{11}} = 0.0078$
$6: CC_{11} \cap \overline{CC_{21}} \cap CC_{31}$	$\{\overline{A_{a(1-3)}}, \overline{C_{a(1-3)}}, \overline{D_{a3}}\}$	$P_{\overline{CC_{21}}} P_{CC_{11}} P_{CC_{31}\|CC_{11}} = 0.0116$
$7: CC_{11} \cap CC_{21} \cap \overline{CC_{31}}$	$\{\overline{A_{a(1-3)}}, \overline{C_{a(1-3)}}, \overline{A_{b(2-3)}}, \overline{B_{a(2-3)}}\}$	$P_{CC_{21}} P_{CC_{11}} P_{\overline{CC_{31}}\|CC_{11}} = 2.4e-4$
$8: CC_{11} \cap CC_{21} \cap CC_{31}$	$\{\overline{A_{a(1-3)}}, \overline{C_{a(1-3)}}, \overline{A_{b(2-3)}}, \overline{B_{a(2-3)}}, \overline{D_{a3}}\}$	$P_{CC_{21}} P_{CC_{11}} P_{CC_{31}\|CC_{11}} = 3.6e-4$

According to (23.14), the problem of evaluating the reliability of the data gathering system with CCF can be subdivided into eight reduced problems that need not consider CCF. Based on system configuration in Figure 23.9 and failure criteria for the *excellence* case described in Figure 23.10 (a), it is easy to derive that: Pr(PMS fails|CCE_j) = *1* for *j* = *3*...*8*. We apply the PMS BDD approach of [2] to evaluate the remaining two reduced problems, Pr(PMS fails|CCE_1) and Pr(PMS fails|CCE_2). Figure 23.13 (a) and (b) show the reduced fault tree models after applying the reduction procedure for removing components of S_{CCE_1} and S_{CCE_2}, respectively. Note that because no component is affected by CCE_1, the reduced fault tree in Figure 23.13(a) is actually the same as the original PMS fault tree (fault trees of the three phases in Figure 23.9 connected via an OR gate) but without considering CCF. Figures 23.14(a) and (b) show the PMS BDD generated from the fault tree models in Figures 23.13(a) and(b), respectively. Finally, results of the eight reduced problems are aggregated using (23.14) to obtain

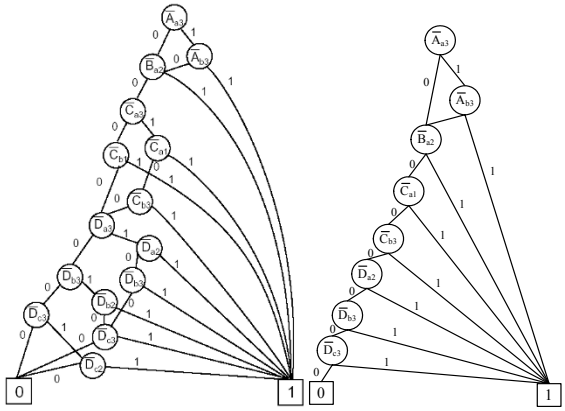


(a) PMS|CCE$_1$



(b) PMS|CCE$_2$

Figure 23.13. Reduced PMS fault trees



(a) PMS BDD|CCE$_1$ (b) PMS BDD|CCE$_2$

Figure 23.14. PMD BDD for reduced fault trees

the unreliability of the data gathering system with the consideration of CCF.

Figure 23.15 shows a conceptual overview of the separable approach for analyzing PMS with CCF. In summary, the methodology is to decompose an original PMS reliability problem with CCF into a number of reduced reliability problems based on the total probability theorem. The set of reduced problems does not have to consider dependence introduced by CCF, and thus can be solved using the efficient PMS BDD method [2]. Finally, the results of all reduced reliability problems are aggregated to obtain the entire PMS reliability considering CCF.

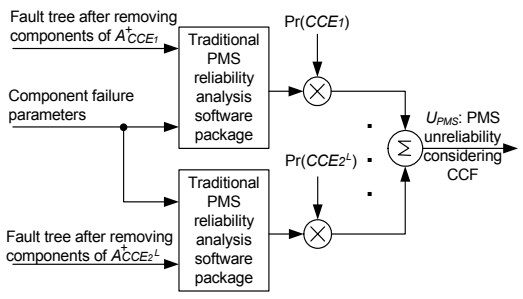


Figure 23.15. A conceptual overview

23.4.4.1 A Case Study: The Mars Orbiter System

To demonstrate this method, we considered a Mars orbiter mission system (originally described in [35]). As shown in the high-level dynamic fault tree (DFT) model of the system (Figure 23.16), this mission system involves launch, cruise, Mars orbit insertion (MOI), commissioning, and orbit phases. The triangles in the DFT are transfer gates to the DFT model for the Subsystem F.

Each mission phase is characterized by at least one major event in which the mission failure can occur. Examples of failure events for this system include the launch event during the launch phase, the deployment of the solar arrays (SA) and high-gain antennas (HGA), and the configuration of the heaters during the cruise phase, the propulsive capture into Mars' orbit during the MOI phase, and the release of an orbiting sample (OS) and the inclusion of a rendezvous and navigation (RAN) platform on the orbiter that might induce additional failure modes during orbit [35]. Table 23.5 gives occurrence probabilities of these failure events.

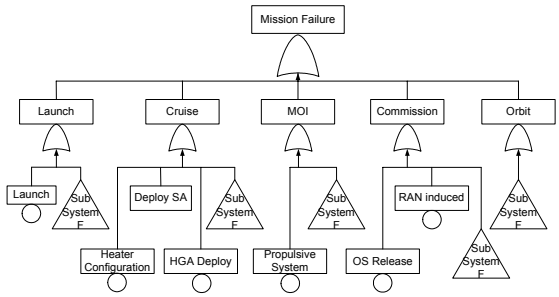


Figure 23.16. High-level DFT model

Table 23.5. Probabilities of failure events

Failure events	Probability
Launch	0.02
SA deployment	0.02
Heater configuration	0.02
HGA deployment	0.02
Propulsive capture	0.03
Orbiting sample release	0.02
RAN-induced failure	0.02

Subsystem F in Figure 23.16 consists of telecommunication, power, propulsion, the command and data handling system (CDS), the attitude control system (ACS), and thermal subsystems, which are connected through an OR gate (Figure 23.17).

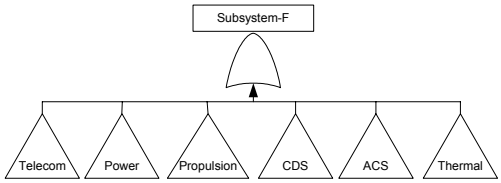


Figure 23.17. Fault tree of subsystem F

As described in [35], these subsystems can be subject to CCF due to two independent CC: CC_1 is a micrometeoroid attack that results in the failure of the entire system, and CC_2 is a solar flare that fails the subsystem's electronics, most notably the CDS in all pre-MOI phases. The orbiter will not be affected by solar flares after the MOI phase due to the increased distance of the orbiter from the sun. Assume that the occurrence probabilities of CC_1 and CC_2 are 0.01 and 0.02, respectively. Table 23.6 specifies the four CCE generated from the two CC, the set of components affected by CCE_i, and occurrence probability of each CCE_i, $\Pr(CCE_i)$.

A review of Table 23.6 implies that the CDS subsystem is the only subsystem affected by both CC_1 and CC_2 and therefore its failure will receive further analysis in this example. Figure 23.18 shows the fault tree model of the CDS subsystem.

Table 23.6. CCE, affected components, and probabilities

CCE_i	S_{CCE_j}	$\Pr(CCE_i)$
$CCE_1 = \overline{CC_1} \cap \overline{CC_2}$	ϕ	9.702e-1
$CCE_2 = CC_1 \cap \overline{CC_2}$	all spacecraft elements	9.980e-3
$CCE_3 = \overline{CC_1} \cap CC_2$	CDS	1.980e-2
$CCE_4 = CC_1 \cap CC_2$	all spacecraft elements	2.000e-4

Table 23.7 gives the failure rates for the CDS components and for the rest of the components (subsystems) of the subsystem F in each phase, as well as the phase duration. According to (23.14), the problem of evaluating the unreliability of the orbiter system with CCF is decomposed into four reduced problems that need not consider CCF. Based on the fault trees in Figures 23.16 through 23.18, we can derive that $\Pr(\text{orbiter fails}| CCE_i) = 1$ for $i = 2$, 3, and 4. Solving the phase-mission fault tree for a mission duration of 97368 hours using the PMS BDD method yields 0.14661 for $\Pr(\text{Orbiter fails}|CCE_1)$. Finally, according to (23.14), the unreliability of the proposed Mar's orbiter system with CCF is 0.172. This result is obtained by aggregating the results of $\Pr(\text{Orbiter fails} | CCE_i)$ and $\Pr(CCE_i)$ given in Table 23.6.

Table 23.7. Failure rates (10^{-7}/hr) of components in CDS and subsystem F

CDS components/ subsystem in F	Launch (504 hrs)	Cruise (5040 hrs)	MOI (144 hrs)	Comm. (4080 hrs)	Orbit (87600 hrs)
EPS-interface	0.05	0.04	0.05	0.05	0.04
Mass memory	0.02	0.01	0.02	0.02	0.01
AC-DC converter	0.02	0.01	0.02	0.02	0.01
CMIC (A and B)	0.03	0.02	0.03	0.03	0.02
FlightProc (A and B)	0.04	0.03	0.04	0.04	0.03
Bus (A and B)	0.02	0.01	0.02	0.02	0.01
IO-card (A and B)	0.02	0.01	0.02	0.02	0.01
PACI-card	0.01	0.005	0.01	0.01	0.005
ULDL-card	0.01	0.005	0.01	0.01	0.005
Telecommunication	0.03	0.2	0.3	0.3	0.2
Power	0.02	0.1	0.2	0.2	0.1
Propulsion	0.3	0.2	0.3	0.3	0.2
ACS	0.04	0.03	0.04	0.04	0.03
Thermal	0.02	0.01	0.02	0.02	0.01

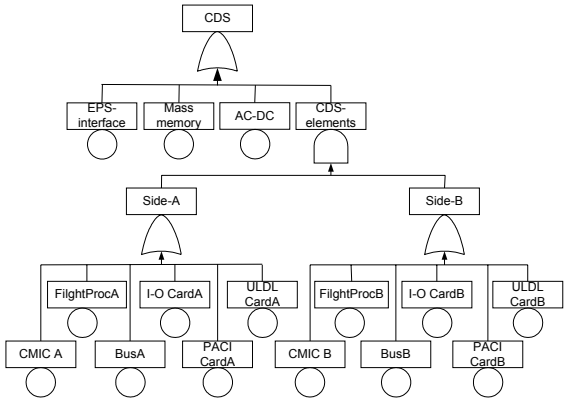


Figure 23.18. DFT model of the CDS

23.5 Conclusions

This chapter presented three classes of analytical approaches to the reliability analysis of PMS, which subject to multiple, consecutive, and non-overlapping phases of operations. The combinatorial approaches are computationally efficient but are limited to the analysis of static PMS only. The state space oriented approaches are powerful in modeling the various dynamic behaviors and dependencies, but are limited to the analysis of small-scale systems due to the state explosion problem. A better solution is the phase modular approach that combines the advantages of both combinatorial analyses and state space oriented analyses. This chapter also discussed the efficient BDD based methods to the analysis of PMS with imperfect coverage or common-cause failures in detail. Since they are combinatorial, the BDD-based methods are applicable to static PMS only. Recently, a separable solution based on the phase modular approach was proposed for the reliability analysis of dynamic PMS subject to CCF. The reader may refer to [22] for details.

References

[1] Xing L, Dugan JB. Analysis of generalized phased-mission systems reliability, performance and sensitivity. IEEE Transactions on Reliability 2002; 51(2): 199–211.

[2] Zang X, Sun H, Trivedi KS. A BDD-based algorithm for reliability analysis of phased-mission systems. IEEE Transactions on Reliability 1999; 48(1): 50−60.

[3] Dugan JB. Automated analysis of phased-mission reliability. IEEE Transactions on Reliability 1991; 40(1): 45−52, 55.

[4] Esary JD, Ziehms H. Reliability analysis of phased missions. In: Barlow RE, Fussell JB, Singpurwalla ND, editors. Reliability and fault tree analysis: Theoretical and applied aspects of system reliability and safety assessment. Philadelphia, PA, SIAM, 1975; 213−236,

[5] Smotherman MK, Zemoudeh K. A non-homogeneous Markov model for phased-mission reliability analysis. IEEE Transactions on Reliability 1989; 38(5): 585−590.

[6] Altschul RE, Nagel PM. The efficient simulation of phased fault trees. Proceedings of IEEE Annual Reliability and Maintainability Symposium, Philadelphia, PA, Jan. 1987; 292−296.

[7] Tillman FA, Lie CH, Hwang CL. Simulation model of mission effectiveness for military systems. IEEE Transactions on Reliability 1978; R-27: 191−194.

[8] Bondavalli A, Chiaradonna S, Di Giandomenico F, Mura I. Dependability modeling and evaluation of multiple-phased systems using DEEM. IEEE Transactions on Reliability 2004; 53(4): 509−522.

[9] Mura I, Bondavalli A. Hierarchical modeling and evaluation of phased-mission systems. IEEE Transactions on Reliability 1999; 48(4): 360−368.

[10] Mura I, Bondavalli A. Markov regenerative stochastic Petri nets to model and evaluate phased mission systems dependability. IEEE Transactions on Computers 2001; 50(12): 1337−1351.

[11] Somani AK, Trivedi KS. Boolean algebraic methods for phased-mission system analysis. Technical Report NAS1-19480, NASA Langley Research Center, Hampton, VA, 1997.

[12] Tang Z, Dugan JB. BDD-based reliability analysis of phased-mission systems with multimode failures. IEEE Transactions on Reliability 2006; 55(2): 350−360.

[13] Meshkat L. Dependency modeling and phase analysis for embedded computer based systems. Ph.D Dissertation, Systems Engineering, University of Virginia, 2000.

[14] Meshkat L, Xing L, Donohue S, Ou Y. An overview of the phase-modular fault tree approach to phased-mission system analysis. Proceedings of the 1st International Conference on Space Mission Challenges for Information Technology, Pasadena, CA, July 2003; 393−398.

[15] Ou Y. Dependability and sensitivity analysis of multi-phase systems using Markov chains. PhD Dissertation, Electrical and Computer Engineering, University of Virginia, 2002; May.
[16] Ou Y, Dugan JB. Modular solution of dynamic multi-phase systems. IEEE Transactions on Reliability 2004; 53(4): 499−508.
[17] Alam M, Al-Saggaf UM. Quantitative reliability evaluation of repairable phased-mission systems using Markov approach. IEEE Transactions on Reliability 1986; R-35(5): 498−503.
[18] Xing L. Dependability modeling and analysis of hierarchical computer-based systems. Ph.D. Dissertation, Electrical and Computer Engineering, University of Virginia, 2002; May.
[19] Andrews JD, Beeson S. Birnbaum's measure of component importance for noncoherent systems. IEEE Transactions on Reliability 2003; 52(2): 213−219.
[20] Somani AK, Trivedi KS. Phased-mission system analysis using Boolean algebraic methods. Proceedings of the ACM Sigmetrics Conference on Measurement and Modeling of Computer Systems 1994; 98−107.
[21] Somani AK, Ritcey JA, Au S. Computationally efficient phased-mission reliability analysis for systems with variable configuration. IEEE Transactions on Reliability 1992; 42: 504−511.
[22] Xing L, Meshkat L, Donohue S. An efficient approach for the reliability analysis of phased-mission systems with dependent failures. Proceedings of the 8th International Conference on Probabilistic Safety Assessment and Management, New Orleans, LA, May 14–18, 2006.
[23] Bouissou M, Bruyere F, Rauzy A. BDD based fault-tree processing: a comparison of variable ordering heuristics. Proceedings of ESREL Conference 1997.
[24] Xing L, Dugan JB. Comments on PMS BDD generation in "a BDD-based algorithm for reliability analysis of phased-mission systems". IEEE Transactions on Reliability 2004; 53(2): 169−173.
[25] Doyle SA, Dugan JB, Patterson-Hine A. A combinatorial approach to modeling imperfect coverage. IEEE Transactions on Reliability 1995; 44(1): 87−94.
[26] Dugan JB, Doyle SA. New results in fault-tree analysis. Tutorial notes of the Annual Reliability and Maintainability Symposium, Philadelphia, PA, Jan. 1997.
[27] Amari SV, Dugan JB, Misra RB. A separable method for incorporating imperfect coverage in combinatorial model. IEEE Transactions on Reliability 1999; 48(3): 267−274.
[28] Gulati R, Dugan JB. A modular approach for analyzing static and dynamic fault trees. Proceedings of the Annual Reliability and Maintainability Symposium 1997.
[29] Xing L, Dugan JB. Generalized imperfect coverage phased-mission analysis. Proceedings of the Annual Reliability and Maintainability Symposium 2002; 112−119.
[30] Xing L, Dugan JB. A separable TDD-based analysis of generalized phased-mission reliability. IEEE Transactions on Reliability 2004; 53(2): 174−184.
[31] Rausand M, Hoyland A. System reliability theory: models, statistical methods, and applications (2nd edition). Wiley Inter-Science, New Jersey, 2004.
[32] Vaurio JK. An implicit method for incorporating common-cause failures in system analysis. IEEE Transactions on Reliability 1998; 47(2): 173−180.
[33] Tang Z, Xu H, Dugan JB. Reliability analysis of phased mission systems with common cause failures. Proceedings of the Annual Reliability and Maintainability Symposium, Washington D.C., Jan. 2005; 313−318.
[34] Xing L. Phased-mission reliability and safety in the presence of common-cause failures. Proceedings of the 21st International System Safety Conference, Ottawa, Ontario, Canada 2003.
[35] Xing L, Meshkat L, Donohue S. Reliability analysis of hierarchical computer-based systems subject to common-cause failures. Reliability Engineering and System Safety 2007; 92(3): 351−359.