Dependability Analysis of Systems With On-Demand and Active Failure Modes, Using Dynamic Fault Trees

Leila Meshkat, *Member, IEEE*, Joanne Bechta Dugan, *Fellow, IEEE*, and John D. Andrews

***Abstract*—Safety systems and protection systems can experience two phases of operation (standby and active); an accurate dependability analysis must combine an analysis of both phases. The standby mode can last for a long time, during which the safety system is periodically tested and maintained. Once a demand occurs, the safety system must operate successfully for the length of demand. The failure characteristics of the system are different in the two phases, and the system can fail in two ways:**

1) It can fail to start (fail on-demand), or

2) It can fail while in active mode.

Failure on demand requires an availability analysis of components (typically electromechanical components) which are required to start or support the safety system. These support components are usually maintained periodically while not in active use.

Active failure refers to the failure while running (once started) of the active components of the safety system. These active components can be fault tolerant and use spares or other forms of redundancy, but are not maintainable while in use.

The approach, in this paper, automatically combines the "availability analysis of the system in standby mode" with the "reliability analysis of the system in its active mode." The general approach uses an availability analysis of the standby phase to determine the initial state probabilities for a Markov model of the demand phase. A detailed method is presented in terms of a dynamic fault-tree model. A new "dynamic fault-tree construct" captures the dependency of the demand-components on the support systems, which are required to detect the demand or to start the demand system. The method is discussed using a single example sprinkler system and then applied to a more complete system taken from the offshore industry.

***Index Terms*—Dynamic fault-tree, Markov analysis, on-demand failure, standby system.**

ACRONYMS[1]

BDD	binary decision diagram
CBE	component Boolean expression
DDEP	demand dependency
DFT	dynamic fault-tree
DP	diesel pump
DPS	diesel power supply
EP	electric pump
EPS	electric power supply
FDEP	functional dependency
HSS	hypothetical sprinkler-system (an example)
hw	hardware
ISP	initial-state probability
ISPA	ISP algorithm
MC	Markov chain
MPS	multi-phase system
SBE	state Boolean expression
SDP	sum of disjoint products
sw	software
WDS	water deluge system.

NOTATION

$\{D\}$, $\{U\}$	set of Down, Up states
c_A	coverage probability for component A
Q_A	unreliability of component A (covered- and uncovered-failure)
f_A, r_A	failure, repair rates of component A
$A_{\mathrm{CO}}(x)$	availability of component x at the time of demand, irrespective of the support components
$S_{\{U\},\{D\}}$	set of states in the MC with demand-components partitioned into $\{U\}$, $\{D\}$
$\ell_{\{U\},\{D\}}$	$\lvert S_{\{U\},\{D\}}\rvert$: number of states in the MC with the demand inputs partitioned to the same $\{U\}$, $\{D\}$ states
$I_{\{U\}\{D\}}$	ISP for a state with demand-components partitioned into $\{U\}$, $\{D\}$
E_{A_j}	event: component j is available at the time of demand, irrespective of support components
E_{U_j}	$\overline{E_{A_j}}$: event that component j is unavailable at the time of demand, irrespective of support components
$\{SU\}_j$	set of support components for demand-component j
$\mathrm{SBE}_{\{U\},\{D\}}$	SBE for a state with demand-components partitioned into $\{U\}$, $\{D\}$
CBE_j	CBE for component j.

Manuscript received August 11, 2000; revised March 27, 2001. This work was supported in part by US NASA Langley Research Center under Contract NAS1-99098.

L. Meshkat is with the USC Information Sciences Institute, Marina Del Rey, CA 90292 USA (e-mail: Meshkat@isi.edu).

J. B. Dugan is with the Department of Electrical Engineering, University of Virginia, Charlottesville, VA 22904 USA (e-mail: JBD@virginia.edu).

J. D. Andrews is with the Department of Mathematical Sciences, Loughborough University, Loughborough - LE11 3TU U.K. (e-mail: J.D.Andrews@lboro.ac.uk).

Publisher Item Identifier S 0018-9529(02)06090-6.

[1]The singular and plural of an acronym are always spelled the same.

ASSUMPTIONS

1) The standby mode can last a long time, i.e., steady-state availability analysis is appropriate.
2) When the system is in standby mode, demand can occur at any time with equal likelihood.
3) When component lifetimes are considered in isolation, they are s-independent. When the components are used in a system, there might be functional dependencies that arise from the system-structure. Thus the model captures the s-dependencies explicitly, and assumes s-independence between the individual components.
4) Time-to-failure and time-to-repair are exponentially distributed, with constant parameters f_i and r_i, respectively, for component i.
5) The system is maintainable during standby mode and nonmaintainable during demand mode.

I. INTRODUCTION

RELIABILITY analysis of safety systems, for example sprinkler systems or other protection systems, requires considering two kinds of failures: failure on demand and failure during operation. That is, the system might fail to start when needed (on demand) or, once started, it might fail during use. Failure of the system to start when needed indicates its *unavailability on demand*; the failure, once started, indicates its *unreliability during demand*. The unavailability on demand of the system depends on the failure characteristics of its support components while in standby mode. These support components can be periodically tested and maintained while not in active mode.

The active components cannot be repaired/maintained during demand. The unreliability during demand depends on the failure characteristics of the active components during demand. The system reliability is the probability that the system is available upon demand, and successfully achieves the mission operation during demand. To conduct a reliability analysis on such systems, each phase is analyzed. Phase #1 is when the system is in standby mode, and phase #2 is when the system is operational. This would require an availability analysis of the support subsystem in standby mode and a reliability analysis of the active components during demand.

A general approach is presented to integrate the analyzes of these two phases within the context of a DFT model. DFT extend traditional fault trees by including special constructs to represent sequential relationships between events [4]. A new construct, DDEP, represents the dependencies between the components in the demand phase and its support components in the standby phase, where a component in the demand phase can require the availability of one or more support components in order to commence operation. A simple motivating example is used to define the problem, and the proposed method is described in detail. The method is illustrated in an example protection system adapted from the offshore oil industry. To keep the exposition simple, some standard complications, such as common-cause failures, are ignored. Standard approaches apply to the resulting model.

The $\times$ and $\prod$ are used with events to represent conjunction (logical AND); $+$ and $\sum$ are used with events to represent disjunction (logical OR).

II. MOTIVATING EXAMPLE

Consider a computer-controlled HSS as an example. HSS is computer-controlled, and is composed of three sensors, two pumps, and one digital controller. Each pump has a support stream composed of valves and filters; the pump requires that the pump stream be operational in order to start. The sensors send signals to the digital controller, and when temperature readings at two of the sensors are above threshold, the controller activates the pump. HSS is available on the demand, if at least two of the sensors are operational, and at least one of the pumps starts. HSS services the demand as long as at least one pump and the controller are functional.

If a pump activates on demand, then the filters and valves in the pump stream are in working condition. Once the pump starts, then the valves and filters will not fail. At least one pump is needed for the system to operate. There is a backup pump which runs if the primary pump fails. System failure occurs if both pumps fail.

Once a sprinkler system is activated, the sensors are no longer needed for reliable operation. However, at least one pump and digital controller must remain operational for a 10-h period. Once the pump system starts, the pump stream is unlikely to fail during operation.

If the distinction between standby and demand modes is ignored, and any maintenance that is conducted on the system while in standby mode is also ignored, then the dependencies can be modeled using the DFT in Fig. 1 [8]. In the DFT of Fig. 1, the functional dependence of each pump on its associated valves and filters is captured in the FDEP [4]. The functional dependency construct has a trigger input and one or more dependent inputs. When the trigger input occurs, the dependent inputs are forced to occur. The cold-spare relationship between the pumps is expressed with the CSP. A CSP gate is one of several dynamic gates introduced in [4] and is used to model several dependencies associated with the use of spares.

Fig. 2 shows the MC corresponding to the fault tree in Fig. 1. The c_x are used to indicate covered-failures of component x [5]. Covered-failures of components might or might not lead to system failure, depending on the remaining redundancy of the system. Uncovered failures, however, always lead to immediate system failure.

Some difficulties are associated with this approach:

1) Although the pump streams and the sensors are used only to start up the system, the MC in Fig. 2 considers them throughout the analysis, and they create additional states in the corresponding MC. Real systems are more complex than HSS, and state space explosion is a problem often encountered when using MC.
2) In some instances, the components used to start up the system are also active during demand, but have different failure parameters.

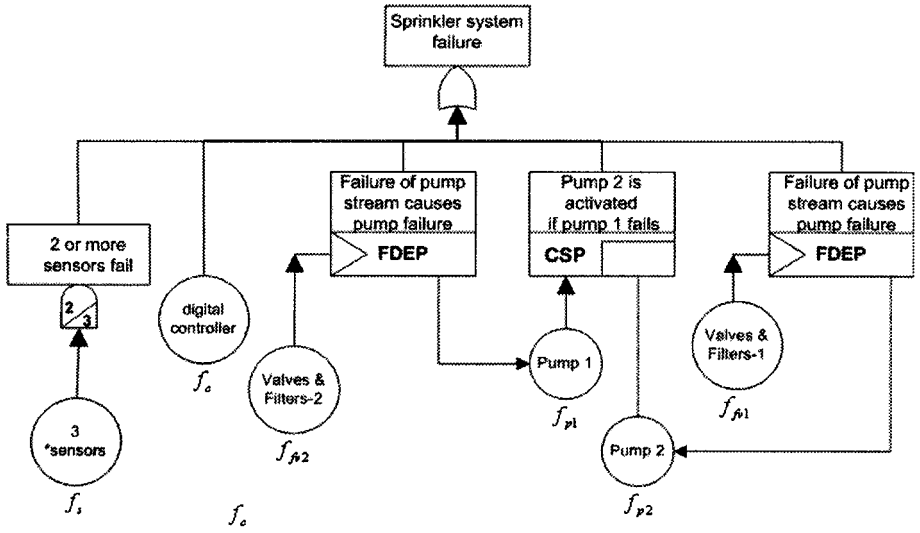

Fig. 1. DFT for HSS using functional dependency gate and ignoring the 2-phase nature of the system. The basic events are labeled with their failure rate.

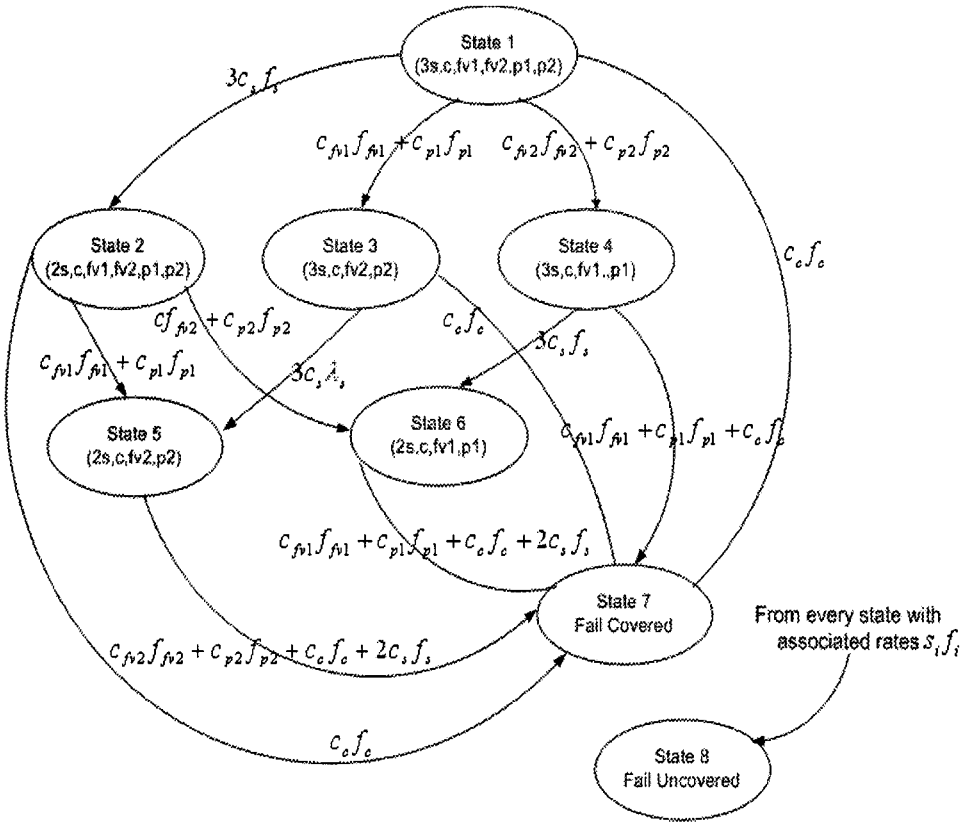

Fig. 2. MC corresponding to DFT shown in Fig. 1 for HSS. Repair arcs are not shown.

3) The support subsystem can be subject to repair and maintenance while the system is in standby mode, but the system is not allowed to be maintained during demand.

Such abrupt changes in the Markov model which occur at the time of demand are difficult to model, in general. A MPS approach could be applied directly if either the demand time were fixed (known) or if one could model the length of the standby mode (arrival of the demand) with a hazard function [11]. However, the MPS approach assumes a transient solution for each phase of the system, and has not been applied where steady-state analysis is appropriate for one of the phases. However, the MPS approach can be built on for the problem in this paper.

The basic MPS approach, as applied to the HSS, is shown conceptually in Fig. 3 where the standby mode is modeled separately from the demand mode. The Markov model for the standby phase can be further separated into three smaller Markov models, because the pump support streams and sensors are functionally independent. Thus there are three separate Markov models for the standby phase (shown on the left of the rectangle) and one Markov model for the demand phase. The

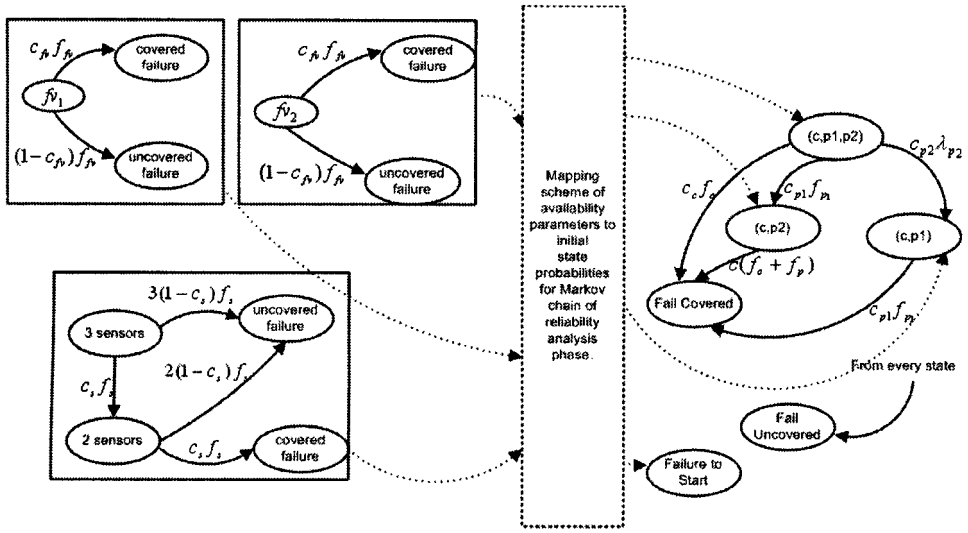

Fig. 3. Separate models for pump streams, sensors, and active subsystem.

MPS approach says to use the analysis of the standby phase to determine the initial state vector for the Markov model of the demand phase.

III. General Approach Using DFT

A. Overview and Applicability of Approach

The approach in this paper divides the reliability analysis of a safety system into two separate phases: standby and demand, and applies when the system exhibits the following characteristics.

a) The standby phase can be arbitrarily long, and system components are periodically tested and maintained while awaiting a demand. Thus a steady-state or average availability analysis is performed on the system in the standby phase.

b) When a demand occurs, the safety system must start and be able to service the demand for a fixed time period without maintenance or repair. The demand phase is analyzed for its reliability, which includes ability to start and, once started, to service the demand.

c) The safety system uses active redundancy during the demand phase, or otherwise exhibits some sequence-dependent behavior, so that a Markov model (rather than a simpler combinatorial model) is appropriate. The approach in this paper applies when the demand phase can be modeled combinatorially, but some simplifications might be appropriate.

d) The support components are passive, and are needed only to detect a demand or to start the demand system; they are not needed to service the demand once the safety system is activated.

e) Two special subsets of the system components can be identified: demand-components and support components. Demand components are necessary to service the demand, and thus appear in the reliability model for the demand phase. Support components are needed to detect the demand or to start the safety system, but are often not needed once the safety system has been activated. Each demand-component is associated with a (possibly empty) set of support components that must be available for the demand-component to start. Each support component is associated with a set of demand-components that depend on it.

Two subproblems must be considered:

1) How does an analyst specify the model? Such a specification must be precise enough to allow both: i) automatic generation of a set of models for the standby and demand phases and ii) mapping of the dependencies of the demand-components on their support components.
2) Once the models are generated, how does one solve the models and connect the required information between them?

Fig. 4 overviews the approach in this paper; it shows the two phases of the model and the connections between them, as well as the information and parameters required for each model.

The methodology is defined in terms of a DFT model; a new DFT construct to is defined to express the dependency of the demand-components on subsystems of support components. A steady-state availability analysis of the support components is used to determine the initial state vector for the Markov model of the demand phase.

The association between demand-components and support-components is general enough to allow multiple support-components for a single demand-component and to allow a support-component to support more than one demand-component. Thus the key to the methodology is the mapping between the two phases: determination of the ISP vector for the states in the MC of the demand phase.

Each state in the MC represents the operational or failed status of the components in the demand phase, and does not explicitly include the status of the support components. Thus, evaluating the ISP requires determining the corresponding status (operational at time of demand or not) of the support components, based on the status of the supported demand-components. For some states, it is possible to determine the status of

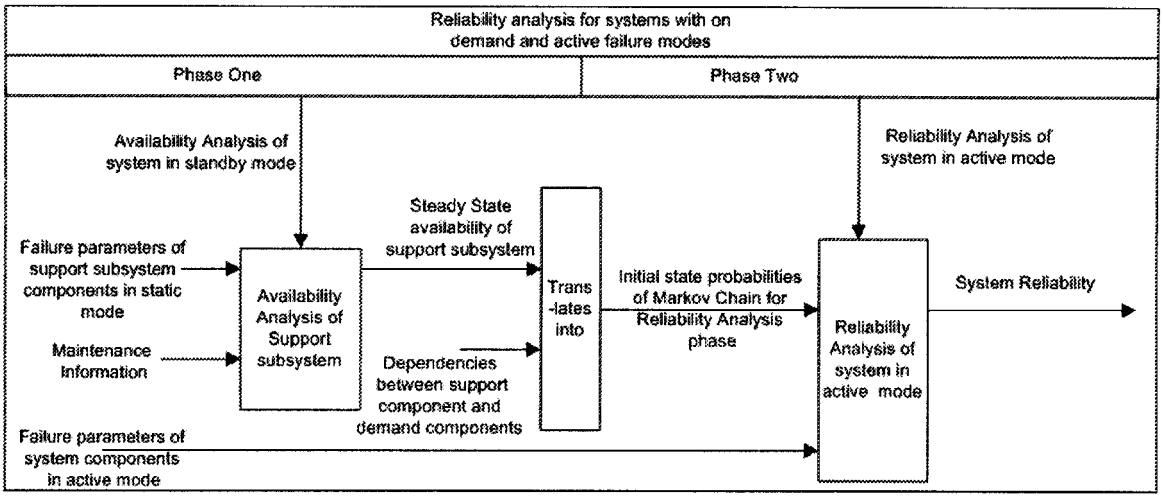

Fig. 4. Overview of general approach.

all support components, and thus the ISP is easily determined. In many cases, however, the status of the support components is ambiguous, and all possible cases must be considered.

The work in this paper builds on the preliminary approach to a similar problem in [9] and on the MPS approach in [1]–[3], [10]; [9] obtains the ISP of the MC associated with the reliability analysis phase using the availability measures of the system while in standby mode. This paper extends their approach to the more general case, and provides an explicit algorithm for considering nondisjoint sets of support components, and engineers the approach to work within the DFT analysis methodology.

B. DDEP Construct

Within the context of a DFT model, define a new fault-tree construct to model failure on demand, called DDEP and shown in Fig. 5. The first input (or support input) to DDEP is a fault tree describing the causes of failure-on-demand (or failure to start). This support input can be either a static or DFT whose components are characterized by failure parameters, repair rates, and/or maintenance intervals. The support subtree is solved for its unavailability: probability that the dependent subsystem is unable to start when demanded. The remaining inputs are dependent events which represent those system components whose functionality depends on the availability-on-demand of the support subsystem. If the support subsystem is unavailable at the time of demand, then the dependent events are forced to occur.

To facilitate automatic solution of systems using DDEP, the support input to a DDEP must be a module, *viz*, it must not share basic events with any other subtree in the system fault tree. A module can be replaced in the overall system fault-tree with a single event, whose availability is determined by solving the subtree rooted at the top node in the module. Thus the support subsystems can be treated as a basic event in the fault tree, and the analysis of the support system can be separated from the analysis of the demand system. (See [6], [12] for more information on modularizing fault trees.)

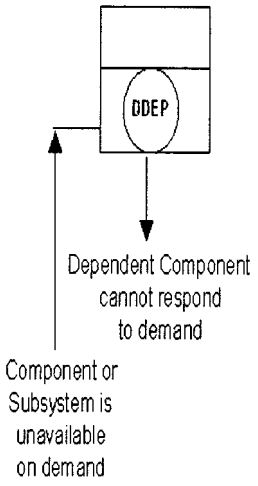

Fig. 5. DDEP construct.

C. Modeling HSS Using DDEP

Fig. 6 shows the fault-tree model of the HSS using DDEP. The dependence of the pump start-up on the correct functionality of at least two of the sensors is shown by the first DDEP construct. If these sensors are unavailable at the time of demand, the pumps become unavailable; hence the system cannot start up. On the other hand, the pumps also need their respective pump streams to be working in order to start up. This is shown by the other DDEP construct. Pump #2 is a backup pump, and is activated only if pump #1 is in a failed state. Therefore, if the pump stream for pump #1 is unavailable, but the pump stream for pump #2 two is available, and the sensors are available, then the system will start up using pump #2, instead of pump #1. The separate subsystems of HSS can be modeled using MC as in Fig. 3. Because the subsystems are static, they can be modeled using combinatorial methods such as BDD [6], which are mode efficient.

Fig. 7 shows this analysis of the HSS example. The initial failure probabilities for the pump streams and sensors are computed by solving the fault-tree models at the support input of each DDEP, producing an unreliability and a coverage value.

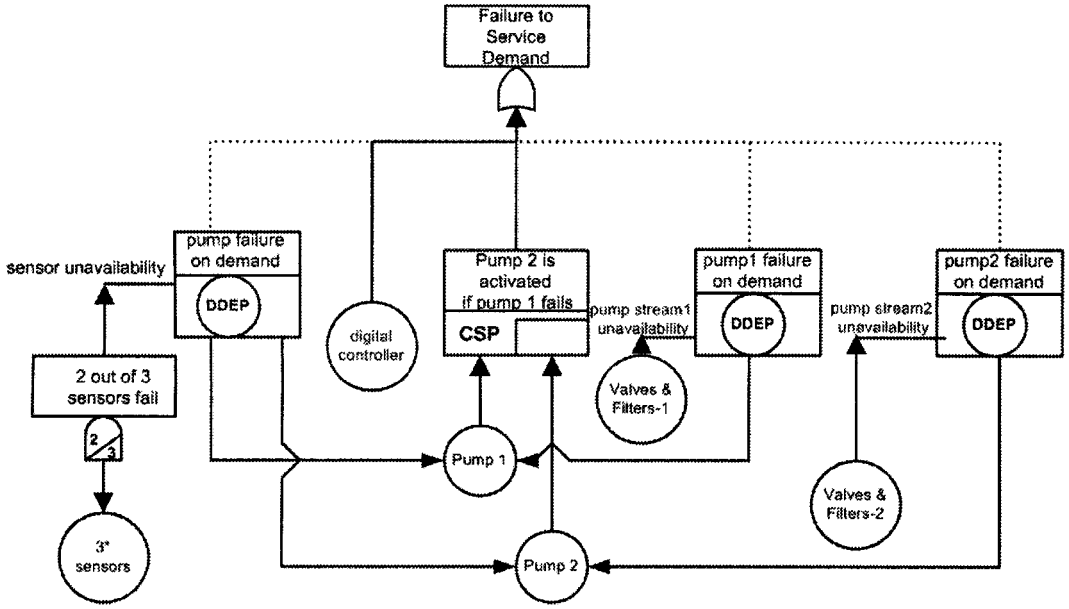

Fig. 6. Fault tree for HSS using DDEP.

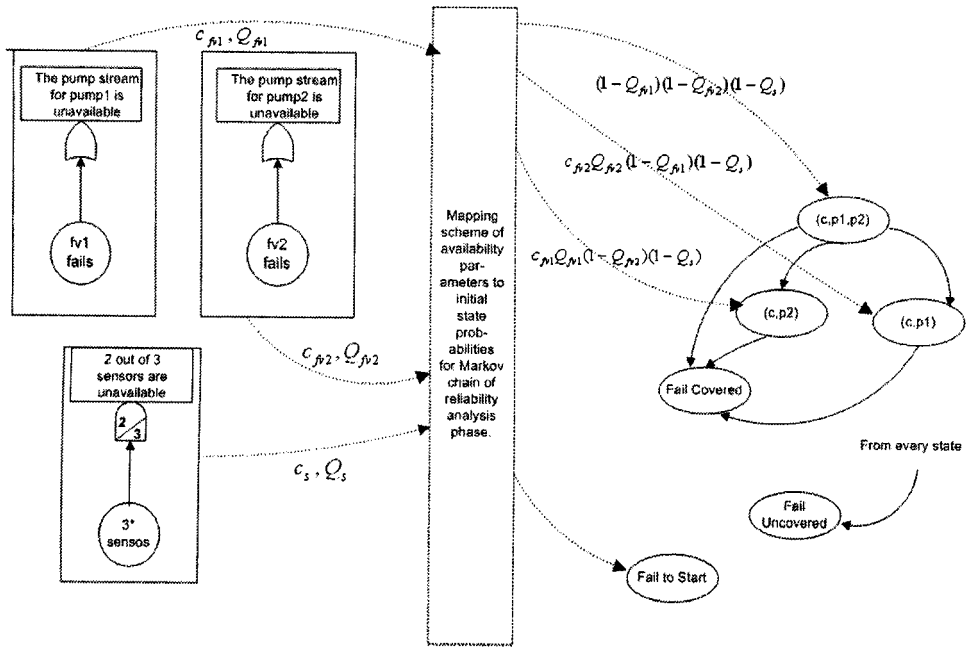

Fig. 7. Mapping scheme for ISP of the reliability-analysis phase.

The coverage parameter associated with each component is indicated by $c_{\text{component}}$, and the unreliability of that component by $Q_{\text{component}}$.

IV. Determining ISP for the Demand Phase Model

The key to the interface between the standby and demand phases is the determination of the ISP vector for the states in the Markov model of the demand phase. The solution of the standby phase for the availability of each of the support components and demand-components is used to determine the initial probability for each combination of demand-components in the MC. The Markov model of the demand phase is then solved for the reliability for the demand time, given the ISP vector as a state of the initial conditions for the demand phase.

The algorithm developed for determining the initial state iterates over each operational state in the Markov model. There can be more than one state in the MC associated with a particular combination of failed and operational demand-components, because the order in which the events occur can be important. For example, if components A and B share spare C, then there are two states in which both A and B have failed and C is operational: 1) C replacing A, and 2) C replacing B. Where there is more than one state associated with a particular combination,

this procedure assumes that each of these states is equally likely to be the initial state with this combination.

The approach for finding the ISP of the MC for the demand phase is based on identifying the respective sources of availability and unavailability for each operational and failed component in each state. The issue of shared support components complicates the calculation as follows:

1) If a demand-component is operational in a state, then infer that all the support components are operational.
2) If a demand-component is failed, then either the component failed on its own during the standby mode, or at least one of its support components can be unavailable.
3) If there is a support component that is shared between an up and down component in the same state, then the support component can not have been the cause of the failure for any of the down components.

A. ISP Algorithm

The ISPA takes as input:

a) A partition of the set of demand-components into $\{U\}$ (those which are up in the state) and $\{D\}$ (those which are down).
b) The number of states in the MC with this same partition, $\ell_{\{U\},\{D\}}$.
c) For each demand-component j, a set of support components, $\{SU\}_j$ which support component j.
d) The intrinsic availability of each demand-component during standby $A_{\mathrm{CO}}(j)$: at the time of demand. "Intrinsic" means that the availability of the component when all its support-components are available. If a demand-component j can not fail during standby then its intrinsic availability is simply 1.
e) The availability of each support-component x, $A_{\mathrm{CO}}(x)$, which is obtained by solving the fault trees which serve as support inputs to a DDEP gate.

The ISPA produces as output: $I_{\{U\},\{D\}}$, the ISP for an operational state in the Markov model of the demand phase.

ISPA iterates over each demand-component j, and builds a state Boolean expression $\mathrm{SBE}_{\{U\},\{D\}}$ representing the event associated with the state. The state Boolean expression is a conjunction of the component Boolean expressions CBE_j for each component j.

If demand-component j is up ($j \in \{U\}$) then it has not failed during the standby phase, and all of its support components must be operational.

$$\mathrm{CBE}_j = E_{A_j} \times \prod_{k \in \{SU\}_j} E_{A_k}, \qquad j \in \{U\}. \tag{1}$$

If a component is down, that could be caused by the component's failing on its own, or one or more of its support components failing. However, suppose there is a support component for a down demand-component which is also supporting an up component in the same state. Then, the support component must be up, and therefore cannot be contributing to the unreliability of the down demand-component. When constructing the CBE_j for a down component, include only the support components which are not supporting an up component in the same state.

$$\mathrm{CBE}_j = E_{A_j} + \sum_{\substack{k \in \{SU\}_j, \\ k \notin \{SU\}_{i \in \{U\}}}} E_{U_k}, \qquad j \in \{D\}. \tag{2}$$

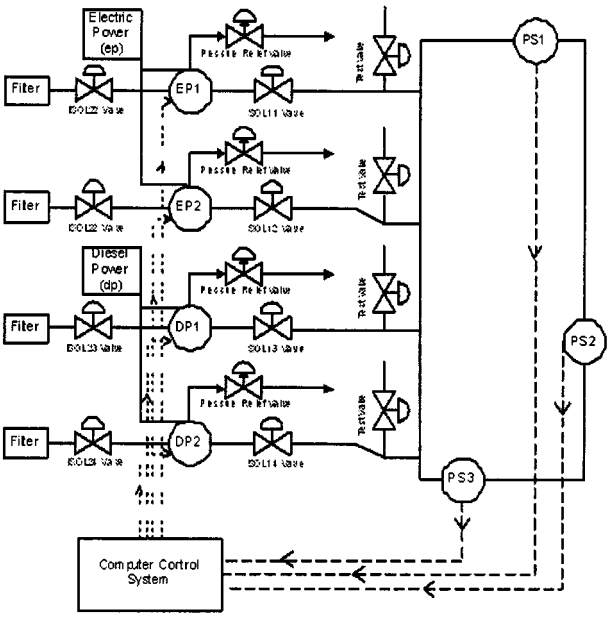

Fig. 8. Schematic representation of the deluge system pump stream.

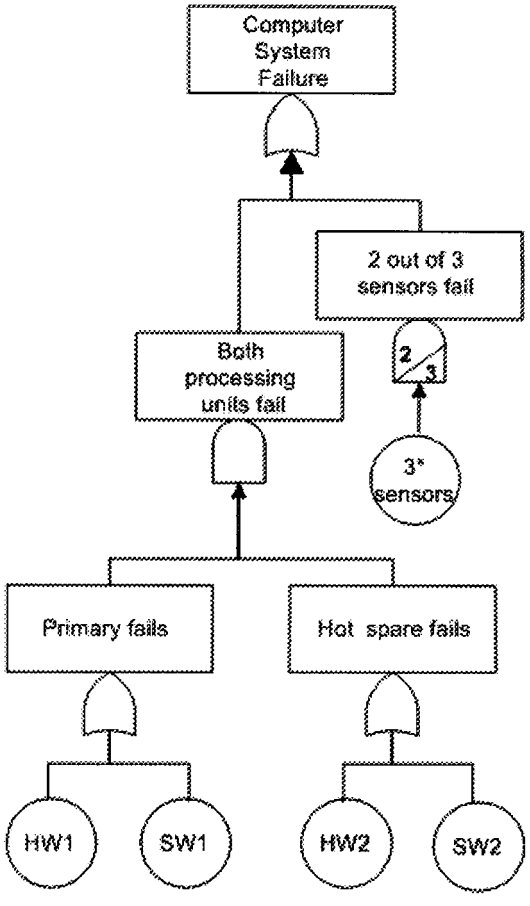

Fig. 9. Fault-tree model of computer-system in WDS.

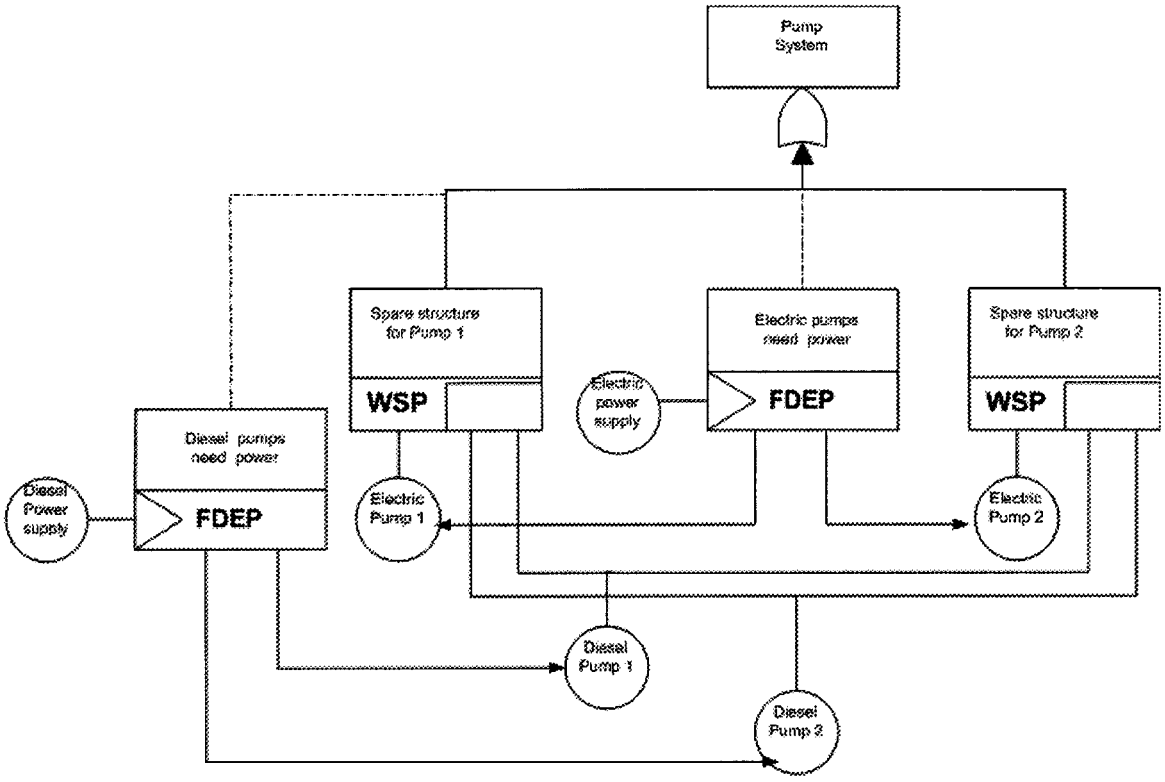

Fig. 10. Fault-tree showing pump-stream for WDS.

The state Boolean expression is then the conjunction of the Boolean expressions associated with each component.

$$\mathrm{SBE}_{\{U\},\{D\}} = \prod_j \mathrm{CBE}_j. \tag{3}$$

The SBE can then be expressed in disjoint products using an SDP algorithm [13] or converted to a BDD directly to evaluate its probability.

The ISP for a state where the set of demand-components is partitioned into $\{U\}$ (those which are up in the state) and $\{D\}$ (those which are down) is then

$$I_{\{U\},\{D\}} = \frac{1}{\ell_{\{U\},\{D\}}} \cdot \Pr\left\{\mathrm{SBE}_{\{U\},\{D\}}\right\}. \tag{4}$$

While iterating over all operational states, keep a running sum of the ISP. When the iteration is finished, the running sum is subtracted from one to produce the ISP for the Fail-to-Start state. The ISP for the Fail-Covered and Fail-Uncovered states are both 0, because these states record failures of the demand system during the demand.

V. Water Deluge System Example

Now, consider a more complete example: the water-deluge system [8] in Fig. 8. The features of this system are typical of water-spray systems used in many different off-shore industries. Four pumps are used to provide the water demand to the ringmain. The ringmain transports the water around the platform to the take-off points where it is used to protect against the hazards posed by hydrocarbon fires and explosions. Pressure in the ringmain is maintained by a jockey pump (not shown in the figure). When the take-off valves open, and water is delivered to the spray nozzles, then the ringmain pressure drops. Ringmain pressure is monitored and transmitted to the computer control system by the three pressure transmitters: PS1–PS3. When two of the three transmitters indicate a low ringmain pressure, then the main pumps are activated in the order indicated from top to bottom of the diagram: EP1, EP2, DP1, DP2. As long as two pumps are available, then water can be delivered at the required rate to satisfy demand. Four pumps provide redundancy in the system: pumps 1 and 2 are electric powered; pumps 3 and 4 are the diesel backups.

The features on each pump stream are identical. Because the water supply is direct from the sea, a filter is fitted on each stream. Manual isolation valves are located on either side of the pump for maintenance purposes. A pressure-relief valve provides protection for the pump, and a test valve on each line enables individual pumps to be tested without fully activating the deluge system.

There are two failure-modes of concern for each stream:

1) It fails to start (unavailable);
2) It fails once running (unreliable).

If a pump stream activates on demand, it means that the filter, isolation valves, test valve, and pressure relief valve which are all (for this function) passive components are in working condition. Because they are passive, they are unlikely to fail in the relatively short running times if they work initially. These are static failure modes. The pump is, however, a dynamic component and can also fail when it is running. System failure occurs if fewer than two of the four streams can be activated: three of the four fail.

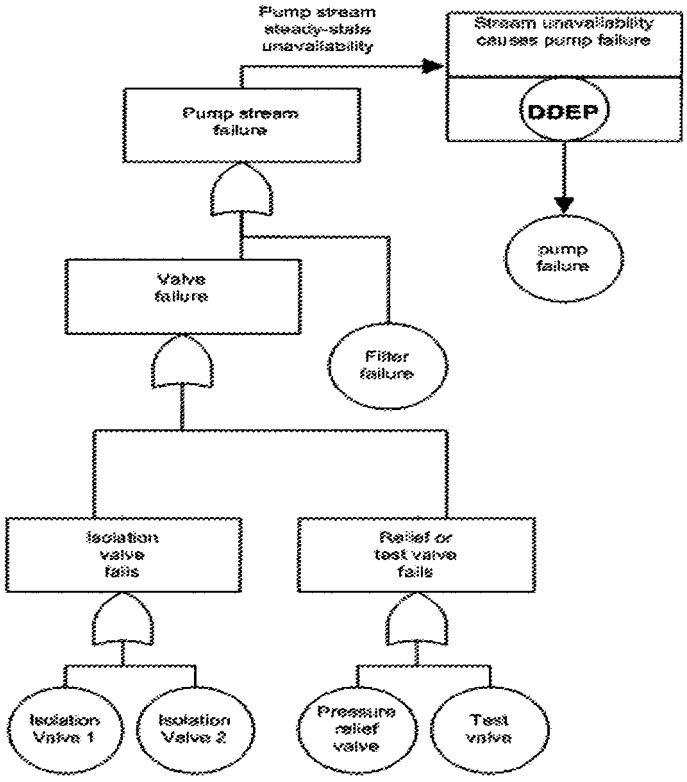

Fig. 11. Fault-tree showing pump-stream using the DDEP gate for WDS.

A. *Fault-Tree Model of WDS*

The computer-control system consists of three pressure-sensors (of which two are needed), plus the hardware and the software. The hardware consists of redundant processors in hot standby mode, each equipped with identical software. While the spare processor is in spare mode, it is monitoring the inputs and outputs of the primary, in order to provide detection and recovery in case of error. When an error is detected, control is switched to the backup processor. The computer-control system can thus tolerate a single (detected) hardware or software failure. However, an undetected error causes failure of the computer subsystem regardless of the state of the backup. This latter case (undetected error) is an example of an uncovered fault, which leads to immediate system-failure. Another example of an uncovered fault is a software fault that affects both processors simultaneously. One might anticipate, because the software on both processors are identical, that all software faults would affect both processors. However, there are field data to support the assumption that a large fraction of software faults affect only a single processor [7]. Fig. 9 shows a fault-tree model including the failure of the computer system, in which the basic events represent hardware (processors), software, and the sensor set.

Next, consider the pump system, consisting of the four pumps, their power sources (two are electric, and two are diesel) and their pump streams (associated valves and filters). For now, ignore the pump streams and power supplies, and concentrate on the four pumps.

The set of four pumps operate in standby redundancy: the two electric pumps are started first, and the diesel pumps provide replacements when the electric pumps are unavailable. On demand, pumps EP1 and EP2 are turned on. If one of these two fails, it is replaced by DP1. The second pump failure is replaced by DP2. This dynamic redundancy scheme introduces dependencies between the failures, and requires special modeling techniques. A pump which is in use experiences a different failure rate than one in standby. Therefore, one must keep track of which pumps are being used and which are in standby. A spare gate is used to model the failure dependencies that arise from the use of spares.

1) A component which is used as a spare has an associated dormancy-factor, $\alpha \in [0, 1]$, which is a multiplicative factor to the active failure-rate to produce the spare failure-rate. If the dormancy factor is 0, then the spare is a cold-spare; a cold-spare cannot fail before being switched into active operation (failure to activate is modeled as an uncovered failure). If the dormancy factor is 1, then the spare is a hot spare and can fail at the same rate as when active. The in-between situation is a warm spare; a warm spare can fail before switched into active operation, but does so at a lower rate than when active.
2) Pooled-spares are spares that can be used as a replacement for whichever of a set of components fails first. Modeling pooled-spares requires keeping track of not only the state of each component, but also the order in which they have failed, so that "which spare is being used where" can be determined.
3) Components might have preferences for replacements, in that there is a priority or order in which spares are used. This order might well be different for various components.

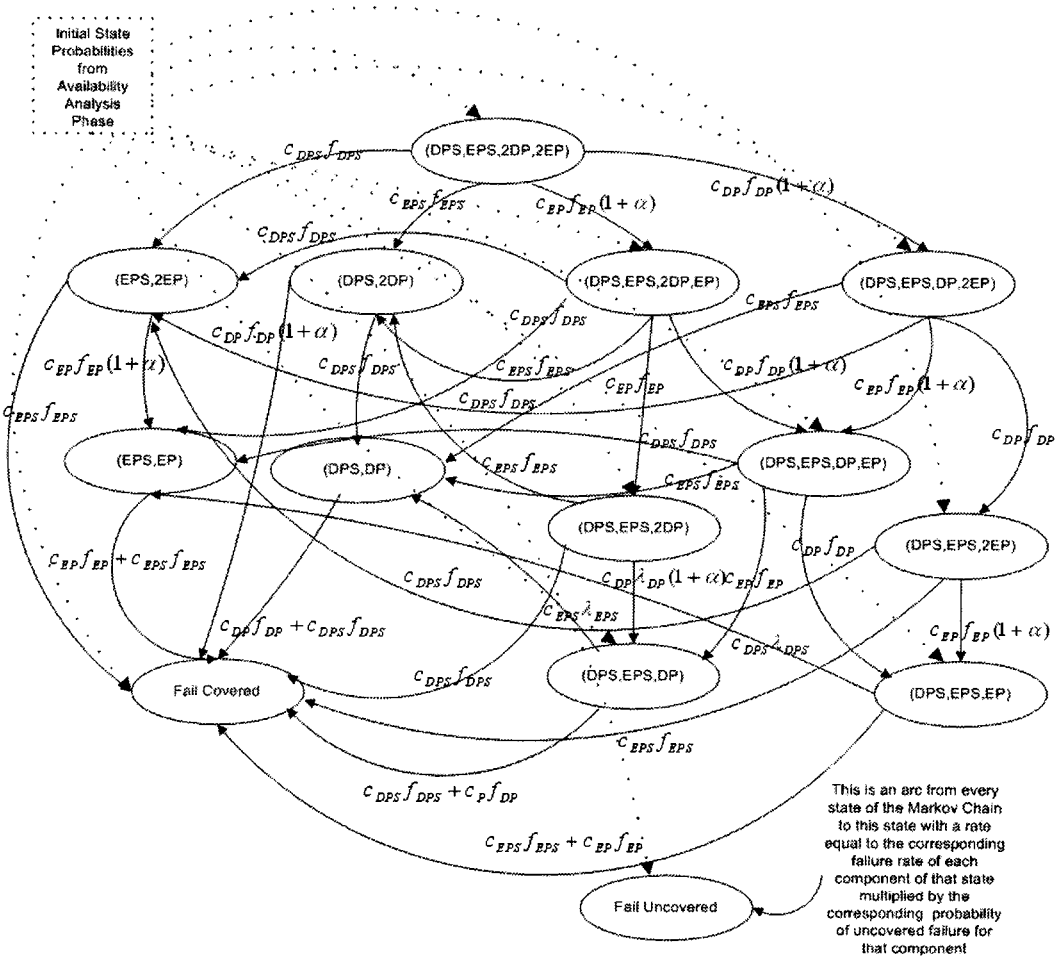

Fig. 12. MC for reliability-analysis phase of pump system.

The spare gate has a set of at least two inputs: the first (leftmost) of which is the designated primary, and the second and subsequent (from left to right) are the spares. When the primary fails, it is replaced (in order) by the spares which are still available (not failed and not used elsewhere). The single output of the spare gate returns "true" when the primary and the spares have been exhausted. Basic events representing spares have failure rates, coverage factors, and dormancy factors.

Continuing to ignore the power supplies and pump streams, the fault-tree in Fig. 10 models the pumps and their spares. The pump-system fails when there are no longer two available pumps (thus the OR gate with two inputs). The basic events, EP1 and EP2 represent the two electric pumps, which are both initially active (on demand). The two diesel pumps, DP1 and DP2, are pooled-spares shared by both electric pumps. The first electric-pump failure is replaced by DP1, and the second by DP2. If EP2 is preferred to be replaced by DP2 then switch the order of DP1 and DP2 inputs on the second spare gate.

Consider the power supplies. There is an electrical power-supply for pumps EP1 and EP2 and a diesel-supply for DP1 and DP2. If a power-supply fails, then the associated pumps are unavailable (essentially failed). The functional-dependency gate can be used to model the functional dependence of the pumps on the power supplies: the power supply is the trigger event and the two pumps are the dependent events. The fault tree in Fig. 10 shows this. Using the DDEP-gate, separate the static-analysis of the pump stream from the dynamic-analysis of the pumps themselves as shown in Fig. 11.

B. WDS ISP

Fig. 12 shows the MC for the reliability analysis of the pump system of the water-deluge system; Fig. 13 shows the MC for the reliability analysis of the computer system. These MC can be derived automatically from the fault trees in Figs. 10 and 9, respectively. The dotted arrows in the MC emanating from the box show the states with nonzero ISP.

First, consider the Markov model of the pump system, in Fig. 12. The demand-components included in this model are the electric and diesel power supplies, the two electric pumps, and the two diesel pumps. The intrinsic availability of each of these components is 1; thus the only cause of unavailability on demand (failure to start) is the support systems. The four support systems are the valve and filter systems, one for each pump, of two types (DP and EP). Fig. 11 shows the detail for a typical pump-stream.

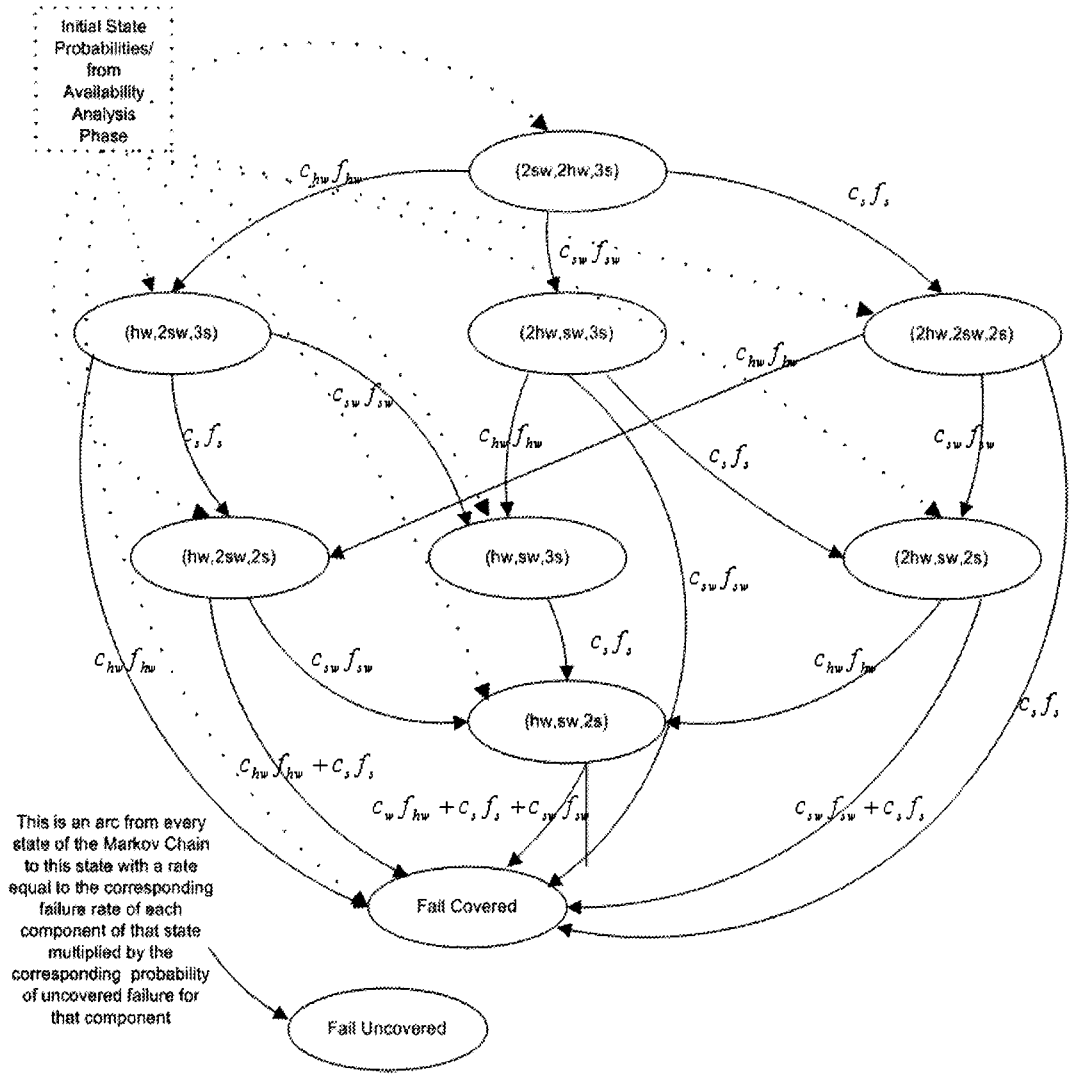

Fig. 13. MC for reliability-analysis phase of computer system.

The nonzero ISP for operational states in the Markov model of the pump subsystem of the WDS are

$$
\begin{aligned}
I_{(\mathrm{DPS,\,EPS,\,2DP,\,2EP})} &= A_{\mathrm{DP}}^2 \cdot A_{\mathrm{EP}}^2 \\
I_{(\mathrm{DPS,\,EPS,\,DP,\,2EP})} &= 2A_{\mathrm{DP}} \cdot (1 - A_{\mathrm{DP}}) \cdot A_{\mathrm{EP}}^2 \\
I_{(\mathrm{DPS,\,EPS,\,2EP})} &= (1 - A_{\mathrm{DP}})^2 \cdot A_{\mathrm{EP}}^2 \\
I_{(\mathrm{DPS,\,EPS,\,2DP,\,EP})} &= 2A_{\mathrm{DP}}^2 \cdot (1 - A_{\mathrm{EP}}) \cdot A_{\mathrm{EP}} \\
I_{(\mathrm{DPS,\,EPS,\,2DP})} &= A_{\mathrm{DP}}^2 \cdot (1 - A_{\mathrm{EP}})^2 \\
I_{(\mathrm{DPS,\,EPS,\,DP,\,EP})} &= 4A_{\mathrm{DP}} \cdot (1 - A_{\mathrm{DP}}) \cdot A_{\mathrm{EP}} \cdot (1 - A_{\mathrm{EP}}) \\
I_{(\mathrm{DPS,\,EPS,\,DP})} &= 2A_{\mathrm{DP}} \cdot (1 - A_{\mathrm{DP}}) \cdot (1 - A_{\mathrm{EP}})^2 \\
I_{(\mathrm{DPS,\,EPS,\,EP})} &= 2(1 - A_{\mathrm{DP}})^2 \cdot A_{\mathrm{EP}} \cdot (1 - A_{\mathrm{EP}}). \qquad (5)
\end{aligned}
$$

Next, consider the Markov model of the computer system, shown in Fig. 13. The computer system has no support subsystems, but each individual component is subject to failure and maintenance during the standby phase. For each operational state, therefore, the ISP is simply the probability of the combination of demand-component availabilities and unavailabilities as defined by the state

$$
\begin{aligned}
I_{(2\mathrm{hw,\,2sw,\,3s})} &= A_{\mathrm{hw}}^2 \cdot A_{\mathrm{sw}}^2 \cdot A_s^3 \\
I_{(\mathrm{hw,\,2sw,\,3s})} &= 2A_{\mathrm{hw}} \cdot (1 - A_{\mathrm{hw}}) \cdot A_{\mathrm{sw}}^2 \cdot A_s^3 \\
I_{(2\mathrm{hw,\,sw,\,3s})} &= A_{\mathrm{hw}} \cdot 2A_{\mathrm{sw}} \cdot (1 - A_{\mathrm{sw}}) \cdot A_s^3 \\
I_{(2\mathrm{hw,\,2sw,\,2s})} &= A_{\mathrm{hw}}^2 \cdot A_{\mathrm{sw}}^2 \cdot 3A_s^2 \cdot (1 - A_s) \\
I_{(\mathrm{hw,\,2sw,\,2s})} &= 2A_{\mathrm{hw}} \cdot (1 - A_{\mathrm{hw}}) \cdot A_{\mathrm{sw}}^2 \cdot 3A_s^2 \cdot 1 - A_s \\
I_{(\mathrm{hw,\,sw,\,3s})} &= 2A_{\mathrm{hw}} \cdot (1 - A_{\mathrm{hw}}) \cdot 2A_{\mathrm{sw}} \cdot (1 - A_{\mathrm{sw}}) \cdot A_s^3 \\
I_{(2\mathrm{hw,\,sw,\,2s})} &= A_{\mathrm{hw}}^2 \cdot 2A_{\mathrm{sw}} \cdot (1 - A_{\mathrm{sw}}) \cdot 3A_s^2 \cdot (1 - A_s) \\
I_{(\mathrm{hw,\,sw,\,2s})} &= 2A_{\mathrm{hw}} \cdot (1 - A_{\mathrm{hw}}) \cdot 2A_{\mathrm{sw}} \cdot (1 - A_{\mathrm{sw}}) \\
&\quad \cdot 3A_s^2 \cdot (1 - A_s). \qquad (6)
\end{aligned}
$$

REFERENCES

[1] J. A. Ritcey, A. K. Somani, and S. H. L. Au, "Computationally-efficient phased-mission reliability analysis for systems with variable configurations," *IEEE Trans. Reliability*, vol. 41, pp. 504–511, Dec. 1992.

[2] A. Pedar and V. V. Sarma, "Phased-mission analysis for evaluating the effectiveness of aerospace computing systems," *IEEE Trans. Reliability*, vol. R-30, pp. 429–436, Dec. 1981.
[3] B. E. Aupperle, J. F. Meyer, and L. Wei, "Evaluation of fault-tolerant systems with nonhomogeneous workloads," in *Proc. Fault Tolerant Computing Symp.*, 1989, FTCS-19, pp. 159–166.
[4] J. B. Dugan, S. Bavuso, and M. Boyd, "Dynamic fault tree models for fault tolerant computer systems," *IEEE Trans. Reliability*, vol. 41, Sept. 1992.
[5] J. B. Dugan and K. S. Trivedi, "Coverage modeling for dependability analysis of fault-tolerant systems," *IEEE Trans. Computers*, vol. 38, pp. 775–787, 1989.
[6] R. Gulati and J. B. Dugan, "A modular approach for analyzing static and dynamic fault trees," in *Proc. Reliability & Maintainability Symp.*, 1997, pp. 57–63.
[7] I. Lee and R. K. Dyer, "Faults, symptoms, and software fault tolerance in the Tandem Guardian90 operating system," in *Proc. 23rd Int. Symp. Fault Tolerant Computing*, 1993.
[8] J. D. Andrews and J. B. Dugan, "Dependency modeling using fault tree analysis," in *Proc. 17th Int. System Safety Conf.*, 1999.
[9] J. D. Andrews and L. M. Ridley, "Analysis of systems with standby dependencies," in *Proc. 16th Int. System Safety Conf.*, 1998, pp. 80–88.
[10] M. Alam and U. M. Al-Saggaf, "Quantitative reliability evaluation of repairable phased-mission systems using Markov approach," *IEEE Trans. Reliability*, vol. R-35, pp. 498–503, Dec. 1986.
[11] L. Meshkat, "Dependency modeling and phase analysis for computer based systems," Ph.D. dissertation, Univ. Virginia, Aug. 2000.
[12] A. Rauzy, "New algorithms for fault tree analysis," *Reliability Engineering and System Safety*, vol. 40, pp. 203–211, 1993.
[13] M. Veeraraghavan and K. S. Trivedi, "An improved algorithm for symbolic reliability analysis," *IEEE Trans. Reliability*, vol. 40, pp. 347–358, Aug. 1991.

Leila Meshkat (M'00) received the B.S. degree (1994) in applied mathematics from Sharif University of Technology, the M.S. degree (1997) in operations research from George Washington University, and the Ph.D. degree (2000) in systems engineering from the University of Virginia.

She is a Research Associate at the USC Information Sciences Institute. Her research interests include reliability and risk analysis.

Dr. Meshkat is a member of Omega Rho and INFORMS.

Joanne Bechta Dugan (F'00) received the B.A. degree (1980) in mathematics and computer science from La Salle University, Philadelphia, PA, and the M.S. and Ph.D. degrees in 1982 and 1984, respectively, in electrical engineering from Duke University, Durham, NC.

She is Professor of Electrical and Computer Engineering with the University of Virginia. She has performed and directed research on the development and application of techniques for the analysis of computer systems which are designed to tolerate hardware and software faults. Her research interests include hardware and software reliability engineering, fault tolerant computing, and mathematical modeling using dynamic fault trees, Markov models, Petri nets, and simulation.

Dr. Dugan was an Associate Editor of the IEEE TRANSACTIONS ON RELIABILITY for 10 years, and is Associate Editor of the IEEE TRANSACTIONS ON SOFTWARE ENGINEERING. She served on the USA National Research Council Committee on Application of Digital Instrumentation and Control Systems to Nuclear Power Plant Operations and Safety.

John D. Andrews is a Professor with the Department of Mathematical Sciences at Loughborough University. He joined this department in 1989, having previously gained nine years of industrial research experience with British Gas and two years of lecturing experience in the Mechanical Engineering Department at the University of Central England. His current research interests concern assessment of the safety and risk of potentially hazardous industrial activities. This research has been heavily supported by industrial funding. Over recent years, grants have been secured from the MOD, Rolls Royce Aero Engines, Mobil North Sea, and Bechtel. He has numerous journal/conference publications along with a jointly authored book *Risk and Reliability Assessment*, now in its second edition.