Fault Tree analysis including component dependencies

Silvia Tolo[1], and John Andrews [1]

[1]Resilience Engineering Research Group, University of Nottingham, Nottingham, UK

Fault Tree analysis is not only the most common technique used in engineering practice for the estimation of system reliability, but is also a key tool shared between designers, analysts and regulators for safe operation and licensing purposes. In spite of its long lasting success, traditional Fault Tree analysis presents significant limitations in modelling a wide range of features frequently encountered in modern systems. The most critical of these is the assumption of failure events independence, which is often not justified by the realistic behaviour of engineering system, undermining modelling accuracy.

This paper introduces a novel methodology for the analysis of Fault Trees allowing for component dependencies and dynamic features. The proposed approach, based on the use of Binary Decision Diagrams, is demonstrated using a simple numerical application for verification. Its applicability and computational feasibility is discussed in details.

I. Introduction

The inability of Fault Trees (FTs) to model component dependencies is recognised to be a barrier towards more realistic modelling and the accurate representation of systems complexity. The underlying assumption of independence among systems' components on which FT analysis relies, rarely finds justification in engineering practice. On the contrary, modern systems strongly rely on the interaction (and hence mutual influence) of components operational conditions, resulting in elaborate, dynamic networks of dependencies which are further contributed to by shared environmental conditions and common cause failures. The problem is far from new, and several alternative tools have been either explicitly developed to address such limitations or borrowed from areas outside risk analysis. The first category embraces mainly dynamic FTs [1] [2]. This methodology relies on the conversion of the dynamic gates of the FT containing the dependencies into Markov Models. This implies the restriction of the analysis to dependent events occurring under the same gate and can result in very large state space models even for moderately sized problems. While this translates into a valid modelling strategy for the representation of sequence-dependent events, spares and dynamic redundancy management, dynamic FTs fail to offer a widely applicable solution for the treatment of the full range of dependency types. For such reason, the application of such technique in engineering practice remains still limited, in spite of numerous research efforts [3] [4] [5].

The second category entails modelling techniques such as Static and Dynamic Bayesian Networks and Petri Nets [6] [7], which while enhancing modelling flexibility, fail to meet the requirements dictated by industrial applications, such as modelling causality and computational feasibility.

This paper offers a mathematical solution tackling two main challenges associated with the analysis of dependencies in system reliability: modelling flexibility and computational feasibility. The method implemented allows the incorporation of dependencies within Fault Tree analysis regardless of their type or location within the system [8] [9]. Furthermore, it retains the familiarity and efficiency of the FT approach, so to match the needs and requirements of real-world industrial applications.

II. Background

A. Binary Decision Diagrams

Binary Decision Diagrams (BDDs) are acyclic graphs able to encode and manipulate Boolean functions [10] and represent an efficient tool for the analysis of FTs [11].

As shown in Fig.1, paths through the BDD originate from a *root node* and ends in *terminal vertices* which can assume the value 1, indicating the occurrence of the system failure, or 0, its non-occurrence. Each

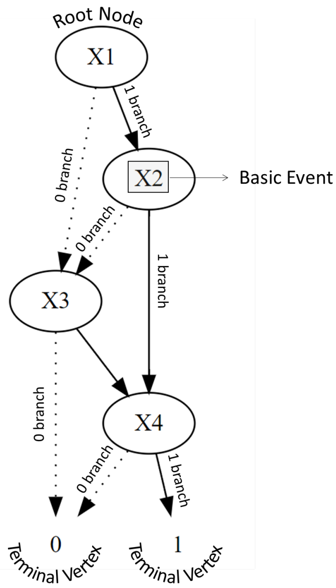

Fig. 1: Example BDD structure

non-terminal node is labelled according to the FT basic event to which it refers. Basic events are considered in a specified order and, as common in FT practice the failure of system components. In this paper, such events are labelled after the component of reference, while complimentary events (i.e. component working state) are instead indicated by a line over the variable name. The state space of a generic component X_i is then expressed as:

$$\mathbf{X_i} = \{X_i, \overline{X_i}\} \quad (1)$$

where the probability q associated with each state verifies the condition:

$$q(\overline{X_i}) = 1 - q(X_i) \quad (2)$$

Two edges originate from each node: one, namely 1-branch, refers to the occurrence of the associated basic event, the other (i.e. 0-branch) with its non-occurrence. The overall Boolean function encoded by the BDD structure is factorised node by node through the use of $if - then - else$ (*ite*) structure, such that:

$$N_k = ite(X_k, h1, h2) \quad (3)$$

where N_k refers to the Boolean structure of the $k - th$ node of the BDD and X_k to the failure event represented by the node. The expression in Eq. 3 translates as: *if* X_k fails *then* consider the Boolean function *h1* which lies on the 1-branch of N_k; else consider function *h2* which, lying on the 0-branch, requires the working state of component X_k [12]. Algorithms are available for the conversion of FTs to BDDs [13] [14]. This in fact a widely adopted strategy for the analysis of FTs, since it ensures the efficient and accurate computation of system reliability metrics, such as system failure probability, system failure intensity and component importance measures [15].

B. *Reliability metrics*

The proposed method focuses on the estimation of three reliability metrics: failure probability, component importance (Birnbaum's measure) and system failure intensity.

All BDD paths connecting the root node to a terminal 1 correspond to the associated FT cut sets and are referred to as paths to failure, P_i. The system failure probability Q_{system} can be then expressed as:

$$Q_{system} = \sum_{i=1}^{m} q(P_i) \quad (4)$$

where $q(P_i)$ indicates the probability associated with the $i - th$ of the m disjoint paths to failure represented by the BDD structure.

The Birnbaum's measure of importance, $G(X_j)$, of a generic component X_j, quantifies the likelihood of the system to be in a critical state for that component, so that the failure of the latter causes the system to pass from the working to the failed state. This can be calculated for each component as:

$$G(X_j) = \frac{\partial Q_{system}}{\partial q(X_j)} = Q_{system}(X_j) - Q_{system}(\overline{X_j}) \quad (5)$$

where $Q_{system}(X_j)$ and $Q_{system}(\overline{X_j})$ refer to the probability of failure of the system given the failure and working state of X_j respectively. Under the assumptions of components independence, Eq.5 can be rewritten in function of the BDD paths to failure as:

$$G(X_j) = \sum_{i|X_j \in P_i} \frac{q(P_i)}{q(X_j)} - \sum_{k|\overline{X_j} \in P_k} \frac{q(P_k)}{q(\overline{X_j})} \quad (6)$$

From this, the failure intensity of the system, i.e. F_{system}, can be calculated as:

$$F_{system} = \sum_{j=1}^{k} G(X_j) \cdot f(X_j) \quad (7)$$

where $f(X_j)$ refers to the failure intensity of the $j-th$ of the k components of the system.

C. System dependencies and their representation

The proposed method relies on the direct manipulation of joint probabilities, such as through marginalisation and conditioning. The first of these procedures allows the marginal contribution of one or more dependent variables to be determined. Consider two components X_i and X_j, and the set of joint probability values over all their possible states, i.e. $q(\mathbf{X_i}, \mathbf{X_j})$. The probability associated with the failure state X_i of $\mathbf{X_i}$ can be computed as:

$$q(X_i) = \sum_{\mathbf{X_j}} q(\mathbf{X_i} = X_i, \mathbf{X_j}) \quad (8)$$

where $q(\mathbf{X_i} = X_i, \mathbf{X_j})$ indicates the set of joint probability values covering the entire state space $\mathbf{X_j}$ but including only the state X_i for $\mathbf{X_i}$.

Conditional probability values can be obtain from joint probabilities through conditioning. This can be expressed as:

$$q(X_j \mid X_i) = \frac{q(X_i, X_j)}{q(X_i)} \quad (9)$$

where $q(X_j, X_i) = q(\mathbf{X_i} = X_i, \mathbf{X_j} = X_j)$, while $q(X_j \mid X_i)$ indicates the probability of component X_j to be in the failure state X_j given $\mathbf{X_i}$ to be in state X_i.

III. METHODOLOGY

This section describes, in detail, the algorithm developed for the calculation of the three reliability metrics of interest discussed:

- system failure probability (or top event probability);
- system failure intensity (or top event intensity);
- component importance measures.

A graphical overview of the proposed methodology is presented in Fig2.

TOP EVENT PROBABILITY

The probability of system failure can be calculated as the sum of all BDD paths probabilities (see Eq.4). Under the assumption of independence, these equal the product of the probabilities of individual events (working and failed) included in each path. Such procedure is not adequate in the presence of dependencies.

To take this into account, the proposed approach relies on the factorisation of each paths into $n+1$

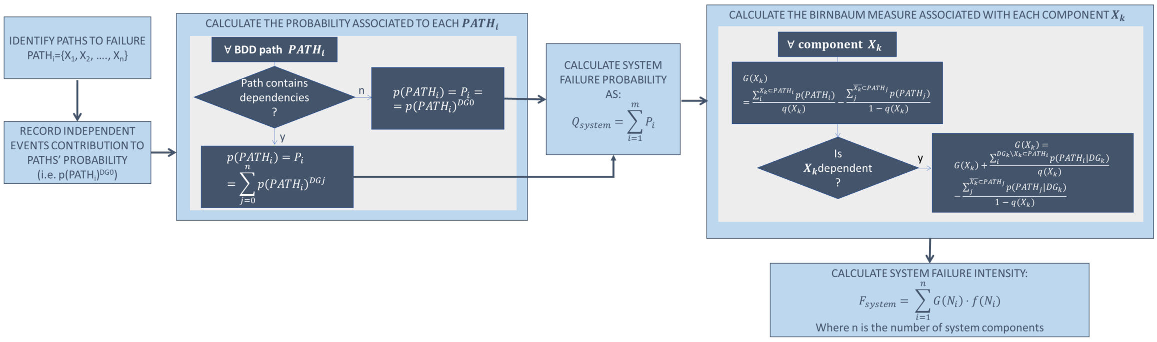

Fig. 2: Overview of the proposed methodology

groupings, corresponding to as many sets of components:

$$P_i = \{P_i^0, P_i^1, ..., P_i^n\} \tag{10}$$

where n is the number of dependency groups, **DG**, featured in the system, so that:

$$\mathbf{DG} = \{\mathbf{DG1}, \mathbf{DG2}, ..., \mathbf{DGn}\} \tag{11}$$

P_i refers to the $i - th$ path of the BDD set of m paths to failure ($PATHS$), so that:

$$PATHS = \{P_1, .., P_k, ..., P_m\} \tag{12}$$

Its first component P_i^0 refers to the set of all independent components included in P_i, so that:

$$P_i^0 \in \mathbf{DG0} \quad \forall i = 1, ..., m \tag{13}$$

$\mathbf{DG0}$ being the set including all system independent components.

The first step of the proposed strategy addresses the identification of the BDD paths and their components. The probability associated with the path independent components, i.e. $q(P_i^0)$, is estimated simultaneously, as described in step three. The third step targets the estimation of the contributions to each path probability associated with dependent components ($q(P_i^k)$ with $k = 1, ..., n$). Finally, system failure probability is computed in step four.

Step 1: Paths identification

Each BDD path is uniquely identified by the combination of the component event identifiers. The aim of the first step of the algorithm is then to record the events included in each path P_i while classifying them according to the dependency group of reference. This is achieved node by node, in a bottom-up direction: starting from terminal 1, the set of parent node are identified. Each event associated with the branch linking the parent and child nodes is added to the component of path P_i according to the dependency group of belonging. For instance, let assume terminal 1 has only one parent node

Algorithm 1: Path identification

Result: $PATHS = \{P_1, P_2, ..., P_m\}$
[with m paths to failure]

$PATHS \leftarrow \{P_1\}$, where $P_1 = \{1\}$
[start from terminal 1]

For each $P_i = \{1, ..., N_m\} \in$ PATHS **do**
  for $h \leftarrow 1$ **to** m **do**
    $Pa(N_h) = \{N_{h-1}, N_{h-2}, ..., N_{h-n}\}$
    [set of n parent nodes of N_h] **for** $j \leftarrow 1$ **to** n **do**
      $P_{i-j+n} \leftarrow P_i$
      Add P_{i-j+n} to $PATHS$
      [duplicate path by parents]
      $N_{h-j} = ite(X, H, F), X \in \mathbf{DGk}$
      if $H = N_h$ **then**
        add X to P^k_{i-j+n}
      else if $F = N_h$ **then**
        add $\overline{X}$ to P^k_{i-j+n}
      end if
    end for
  end for
until *Root node* N_1 *is reached*;

N_h ($Pa(1) = N_h$), such that $ite(N_h) = (X, 1, 0)$ and X a dependent component of dependency group DGk. Hence, the event X is added to the $k-th$ component of path P_i such that, for the current stage of the procedure, $P^k_i = N_h$. The node N_h is then processed in the same way, starting with the identification of its parents. The procedure is in fact repeated for all queued parents until the root node is reached and the queue empty. The path is duplicated in the case of multiple parents, so to allow the differentiation of branches originating from the same child and their assignment to separate paths.

Step 2: Independent path element probability, P^0_i computation

The probability associated with the path independent component, i.e. $q(P^0_i)$, is estimated simultaneously with the progression of the path identification procedure. In fact, the probability of the independent component $q(P^0_i)$ of the $i{-}th$ path can be calculated as the product of all independent events lying on it: hence, any time an independent event X is added to the subset P^0_i of the $i-th$ path P_i (according to the former step), its probability $q(X)$ is multiplied by the product of the probabilities of independent events previous recorded within the same path. For the sake of clarity, this further step is represented in a separate pseudo code, shown in Algorithm 2, which expands on the underlined sections of Algorithm 1. This finally results in the computation of the contributions of each paths

Algorithm 2: Computation of independent probability components

Result: $q(PATHS^0) = \{q(P^0_1), q(P^0_2), ..., q(P^0_m)\}$
[with m BDD paths to failure]

When: Add X to P^k_i:
if *k=0* **then**
  $q(P^0_i) = q(P^0_i) \cdot q(X)$
end if

independent component grouping probability, i.e. $q(PATHS^0) = \{q(P_1^0), q(P_2^0), ..., q(P_m^0)\}$.

Step 3: Dependent path probability computation

The contribution of a dependent event X, where $X \in \mathbf{DGk}$ with $k \neq 0$, to the probability of a generic path P_i cannot be estimated as the product of marginal probabilities as done so far. Conversely, the value of X probability depends on the state of the other events from the same dependency group in the path. The contribution of dependent components to the overall path probability can hence be computed only after the path definition in Step 1 is completed. Once the events in P_i^k associated with dependency group $\mathbf{DGk}$ are known for each $i-th$ path to failure, their probability can be calculated on the basis of the joint probability in input, and finally multiplied by the total path probability since the group is independent from other dependency groups. This procedure is summarised in Algorithm 3.

Algorithm 3: Computation of dependent path probability components

Result: $q(PATHS) = \{q(P_1), q(P_2), ..., q(P_m)\}$ [with m BDD paths to failure]

```
do
    q(P_i) ← q(P_i^0)
    for k ← 1 to n do
        q(P_i) = q(P_i) · q(P_i^k)
    end for
for all P_i, i=1, ..., m;
```

Step 4: System failure probability

Once the probability of each BDD path has been computed, the failure probability of the system is calculated according to Eq. 4.

BIRNBAUM'S MEASURE OF IMPORTANCE

Steps 5 and 6 are dedicated to compute the criticality function (Birnbaum's measure of component importance) for independent and dependent system components respectively.

Step 5: Birnbaum's meausure of importance

For independent components, the value of the Birnbaum's measure of importance can be calculated according to Eq.6, as summarised in the pseudo code shown in Algorithm 4. The state of a dependent component, have a direct impact on the probability value associated with other members of the same dependency group. Hence, assumptions on the working or failure state of dependent compo-

Algorithm 4: Computation of Birnbaum's importance measure of independent component X

```
X_l ∈ DG0
for i ← 1 to m do
    if X_l ∈ P_i^0 then
        G(X_l) = G(X_l) + q(P_i)/q(X_l)
    else if ¬X_l ∈ P_i^0 then
        G(X_l) = G(X_l) − q(P_i)/(1−q(X_l))
    end if
end for
```

nents (as those implicitly adopted in the calculation of Birnbaum's measures), affect the probability of paths of belonging, as well as paths including other events from the same dependency group, even when excluding the component itself. In the first case, the contribution to the component Birnbaum's measures from paths including such component can be calculated following the same procedure discussed for independent component (Algorithm 4). The secondary contribution $G(Y_l)^i$ to the Birnbaum measure of component Y_l from the $i-th$ path including events of the same dependency group DGk but not Y_l, can be instead calculated as:

$$\begin{aligned} G(Y_l)^i &= q(Pi \mid Y_l) - q(Pi|\overline{Y_l}) \\ &= \frac{q(Pi)}{q(Pi^k)} \cdot (q(Pi^k \mid Y_l) - q(Pi^k \mid \overline{Y_l})) \end{aligned} \tag{14}$$

where $q(Pi \mid Y_l)$ and $q(Pi \mid \overline{Y_l})$ indicate the conditional probability associated to the $i-th$ path given the failure and working state of Y_l respectively, and $q(Pi^k)$ is the probability associated with the $k-th$ component of path Pi, which includes exclusively any number of members of the dependency group DGK except Y_l itself. Hence, the values $q(Pi^k \mid Y_l)$ and $q(Pi^k \mid \overline{Y_l})$ indicate the conditional joint probability of any component dependent on $\overline{Y_l}$ and lying on path Pi, and can then be calculated manipulating the joint probability in input through marginalisation and conditioning (see Eq.8 and 9). The overall procedure is discussed further in Section IV-C.

Algorithm 5: Computation of Birnbaum's importance measure of dependent component Y_l

$Y_l \in \mathbf{DGk}$ with $k \neq 0$
for $i \leftarrow 1$ **to** m **do**
  if $P_i^k \neq \emptyset$ **then**
    if $Y_l \in P_i^k$ **then**
      $G(Y_l) = G(Y_l) + \frac{q(P_i)}{q(Y_l)}$
    else if $\overline{Y_l} \in P_i^k$ **then**
      $G(Y_l) = G(Y_l) - \frac{q(P_i)}{q(\overline{Y_l})}$
    else
      $G(Y_l) = G(Y_l) + \frac{q(Pi)}{q(Pi^k)} \cdot \big(q(Pi^k \mid Y_l) - q(Pi^k \mid \overline{Y_l})\big)$
    end if
  end if
end for

TOP EVENT FAILURE INTENSITY

The availability of the components Birnbaum's measures of importance enables the calculation of the final reliability parameter of interest, system failure intensity, carried out in step six.

Step 6: Failure Intensity

The system failure intensity is computed according to Eq.7.

A. Computational Feasibility

The complexity of the methodology proposed depends on the BDD structure, being directly proportional to the number of paths to failure. This implies that, for extremely large BDDs, the technique may become computationally intractable.

A possible alternative could be shifting to an approximate solution: this implies the exclusion from the computation of any path whose probability falls under a pre-established threshold $q_{threshold}$. Such a truncation procedure could be carried out in the path identification phase, hence modifying the procedure proposed in Algorithm 1 as shown in Algorithm 6. While the system failure probability calculated

Algorithm 6: Path elimination procedure for approximate failure probability computation

```
When: Add X to P_i^k:
if k=0 then
    q(P_i^0) = q(P_i^0) · q(X)
    if q(P_i^0) < q_threshold then
        delete P_i
    end if
end if
```

with the truncated paths set would result in an underestimation of the real value, by recording the number of censored paths it is possible to provide an upper bound to the approximate output. In fact, since each eliminated path $P_j^{truncated}$ is associated to a probability value lower than $q_{threshold}$, the maximum contribution to the overall failure probability lost through the truncation procedure is less than:

$$Q_{system}^{truncated} = M \cdot q_{threshold} \tag{15}$$

where M is the number of eliminated paths. Hence, it results:

$$Q_{system}^{approx} < Q_{system} < \left(Q_{system}^{approx} + Q_{system}^{truncated}\right)$$

where Q_{system}^{approx} is the estimate approximating the real system failure probability Q_{system}.

IV. Numerical Application

In order to test the capabilities of the proposed methodology, a simple case study focusing on the fault tree structure shown in Fig.3 has been analysed, introducing multiple dependency groups.

A. Case study

The FT shown in Fig.3, represents a system of ten components: $X1$-$X10$. The top event *TOP* represents the simultaneous failure of two subsystems, in turn depicted by the FT subsections below gates *G6* and *G7*. Both subsystems embrace two components working in parallel, i.e. $X1$ and $X2$ for gate *G6*, $X7$ and $X8$ for gate *G7*. These components below each gate are considered to have mutual dependencies: it is assumed that the failure of one of the parallel components will put a larger load on the other, increasing its failure probability. This results in the definition of two dependency groups, $\mathbf{DG1} = \{X1, X2\}$ and $\mathbf{DG2} = \{X7, X8\}$. In order to test the generality of the proposed method, a further dependency relation embracing components belonging

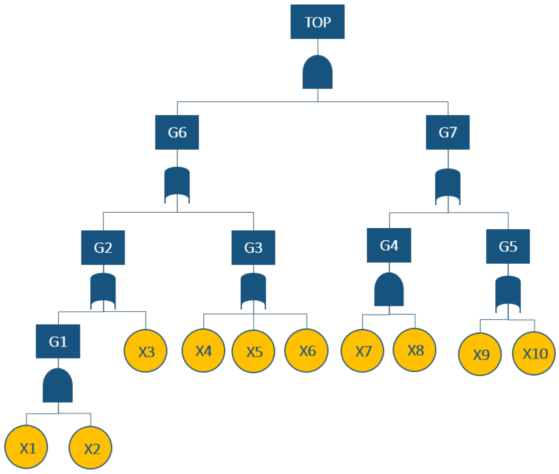

Fig. 3: Fault Tree structure for the case-study analysed

to different subsystem (i.e. $X5$ and $X9$) is also considered, resulting in a third dependency group $\mathbf{DG3} = \{X5, X9\}$. This could be representative of more complex types of dependencies, e.g. related to maintenance strategies. As discussed, the FT is first converted into a BDD structure: this is shown in Fig.4. For the sake of clarity, nodes associated with dependent components (i.e.$X1$, $X2$, $X5$, $X7$, $X8$ and $X9$), are represented as double line ellipses, while different dependency groups are highlighted in different color shades. The dependency groups so defined are accounted for numerically through the use of joint estimates, as discussed in Section II-C: these are listed in Table I-III for the dependency groups $\mathbf{DG1}$, $\mathbf{DG2}$ and $\mathbf{DG3}$ respectively.

The reliability information assumed for the remaining independent components are shown in Table IV.

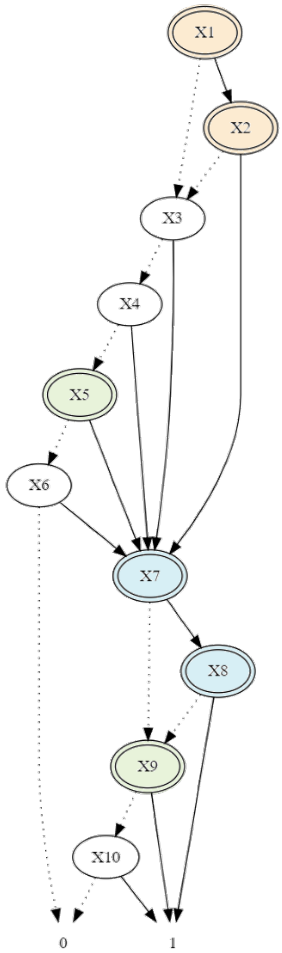

Fig. 4: BDD structure resulting from the conversion of the FT in Fig3. Double ellipses refer to dependent components, while color shades have been assigned to different dependency groups

B. System Failure Probability by Path

The application of the methodology described in Section III resulted in the identification of 45 disjoint failure paths, shown in Table V together with their individual probabilities. The overall system

Joint States	Probability	Intensity $[h^{-1}]$
$X1,\ X2$	$1.6205 \cdot 10^{-01}$	$8.4478 \cdot 10^{-01}$
$\overline{X1},\ X2$	$2.1658 \cdot 10^{-01}$	$2.7614 \cdot 10^{-01}$
$X1, \overline{X2}$	$2.1612 \cdot 10^{-01}$	$2.7478 \cdot 10^{-01}$
$\overline{X1},\ \overline{X2}$	$4.05250 \cdot 10^{-01}$	$6.75408 \cdot 10^{-01}$

TABLE I: Joint reliability information for **DG1**

Joint States	Probability	Intensity $[h^{-1}]$
$X7,\ X8$	$1.17647 \cdot 10^{-02}$	$7.05882 \cdot 10^{-02}$
$\overline{X7},\ X8$	$3.5294 \cdot 10^{-02}$	$4.58823 \cdot 10^{-01}$
$X7, \overline{X8}$	$3.5294 \cdot 10^{-02}$	$4.58823 \cdot 10^{-01}$
$\overline{X7},\ \overline{X8}$	$9.1765 \cdot 10^{-01}$	$9.1765 \cdot 10^{-01}$

TABLE II: Joint reliability information for **DG2**

failure probability is computed by the summation of such values as:

$$Q_{system} = \sum_{i=1}^{45} q(P_i) = 1.0906 \cdot 10^{-02} \qquad (16)$$

C. *Birnbaum's Measures of Importance*

Birnbaum's measures of importance were calculated, resulting in the values shown in Table VI.

Joint States	Probability	Intensity $[h^{-1}]$
$X5,\ X9$	$8.0842 \cdot 10^{-04}$	$7.6261 \cdot 10^{-03}$
$\overline{X5},\ X9$	$4.4276 \cdot 10^{-02}$	$7.9212 \cdot 10^{-02}$
$X5, \overline{X9}$	$4.6546 \cdot 10^{-02}$	$7.9727 \cdot 10^{-02}$
$\overline{X5},\ \overline{X9}$	$9.0837 \cdot 10^{-01}$	$1.5131 \cdot 10^{-01}$

TABLE III: Joint reliability information for **DG3**

Component	Probability	Intensity $[h^{-1}]$
$X3$	$1.3500 \cdot 10^{-06}$	$2.7000 \cdot 10^{-07}$
$X4$	$6.1136 \cdot 10^{-03}$	$2.2018 \cdot 10^{-03}$
$X6$	$2.3810 \cdot 10^{-05}$	$1.7000 \cdot 10^{-06}$
$X10$	$2.3810 \cdot 10^{-05}$	$1.7000 \cdot 10^{-06}$

TABLE IV: Independent components reliability info

As described in Section III, the computation of the criticality measure of independent component follows Algorithm 4, which factorises Eq.6 into individual path contributions. For the independent component $X3$ in the example, this results in:

$$\begin{aligned} G(X3) &= \frac{\sum_{j=6}^{15} q(P_j)}{q(X3)} - \frac{\sum_{i=16}^{45} q(P_i)}{q(\overline{X3})} \\ &= 4.7694 \cdot 10^{-02} \end{aligned} \qquad (17)$$

where the positive term refers to the probability of failure paths embracing the working state of $X3$ (i.e., paths 16 to 45 in Table V) conditional to the working state of the component. This is calculated dividing the joint probability of the paths by the probability of event $\overline{X3}$, according to the conditioning procedure of Eq.9. Similarly, the negative term is associated with paths implying the failure of $X3$ (i.e., 6 to 15 in Table V). Paths from 1 to 5 in Table V do not contain events $X3$ or $\overline{X3}$, hence contribute to system failure regardless of the state of the component: this implies the two resulting identical terms cancel themselves out when considered under both assumptions of $X3$ working and failure states, reducing to the expression in Eq.17.

While assumptions on the working or failure state of independent components affect only the failure probability of the individual component, similar hypothesis have a larger impact when entailing dependencies. For instance, assuming the dependent

Path No	Events Contribution	Probability
1	$q(X1, X2) \cdot q(X7, X8)$	$1.9065 \cdot 10^{-03}$
2	$q(X1, X2) \cdot q(X7, \overline{X8}) \cdot q(X9)$	$2.7084 \cdot 10^{-04}$
3	$q(X1, X2) \cdot q(X7, \overline{X8}) \cdot q(\overline{X9}) \cdot q(X10)$	$2.2212 \cdot 10^{-07}$
4	$q(X1, X2) \cdot q(\overline{X7}) \cdot q(X9)$	$7.3126 \cdot 10^{-03}$
5	$q(X1, X2) \cdot q(X7) \cdot q(X9) \cdot q(X10)$	$5.9973 \cdot 10^{-06}$
6	$q(\overline{X1}) \cdot q(X3) \cdot q(X7, X8)$	$9.8761 \cdot 10^{-09}$
7	$q(\overline{X1}) \cdot q(X3) \cdot q(X7, \overline{X8}) \cdot q(X9)$	$1.4030 \cdot 10^{-09}$
8	$q(\overline{X1}) \cdot q(X3) \cdot q(X7, \overline{X8}) \cdot q(\overline{X9}) \cdot q(X10)$	$1.1507 \cdot 10^{-12}$
9	$q(\overline{X1}) \cdot q(X3) \cdot q(\overline{X7}) \cdot q(X9)$	$3.7882 \cdot 10^{-08}$
10	$q(\overline{X1}) \cdot q(X3) \cdot q(\overline{X7}) \cdot q(\overline{X9}) \cdot q(X10)$	$3.1068 \cdot 10^{-11}$
11	$q(X1, \overline{X2}) \cdot q(X3) \cdot q(X7, X8)$	$3.4325 \cdot 10^{-09}$
12	$q(X1, \overline{X2}) \cdot q(X3) \cdot q(X7, \overline{X8}) \cdot q(X9)$	$4.8763 \cdot 10^{-10}$
13	$q(X1, \overline{X2}) \cdot q(X3) \cdot q(X7, \overline{X8}) \cdot q(\overline{X9}) \cdot q(X10)$	$3.9992 \cdot 10^{-13}$
14	$q(X1, \overline{X2}) \cdot q(X3) \cdot q(\overline{X7}) \cdot q(X9)$	$1.3166 \cdot 10^{-08}$
15	$q(X1, \overline{X2}) \cdot q(X3) \cdot q(\overline{X7}) \cdot q(\overline{X9}) \cdot q(X10)$	$1.0798 \cdot 10^{-11}$
16	$q(\overline{X1}) \cdot q(\overline{X3}) \cdot q(X4) \cdot q(X7, X8)$	$4.4725 \cdot 10^{-05}$
17	$q(\overline{X1}) \cdot q(\overline{X3}) \cdot q(X4) \cdot q(X7, \overline{X8}) \cdot q(X9)$	$6.3537 \cdot 10^{-06}$
18	$q(\overline{X1}) \cdot q(\overline{X3}) \cdot q(X4) \cdot q(X7, \overline{X8}) \cdot q(\overline{X9}) \cdot q(X10)$	$5.2109 \cdot 10^{-09}$
19	$q(\overline{X1}) \cdot q(\overline{X3}) \cdot q(X4) \cdot q(\overline{X7}) \cdot q(X9)$	$1.7155 \cdot 10^{-04}$
20	$q(\overline{X1}) \cdot q(\overline{X3}) \cdot q(X4) \cdot q(\overline{X7}) \cdot q(\overline{X9}) \cdot q(X10)$	$1.4069 \cdot 10^{-07}$
21	$q(X1, \overline{X2}) \cdot q(\overline{X3}) \cdot q(X4) \cdot q(X7, X8)$	$1.5544 \cdot 10^{-05}$
22	$q(X1, \overline{X2}) \cdot q(\overline{X3}) \cdot q(X4) \cdot q(X7, \overline{X8}) \cdot q(X9)$	$2.2082 \cdot 10^{-06}$
23	$q(X1, \overline{X2}) \cdot q(\overline{X3}) \cdot q(X4) \cdot q(X7, \overline{X8}) \cdot q(\overline{X9}) \cdot q(X10)$	$1.8111 \cdot 10^{-09}$
24	$q(X1, \overline{X2}) \cdot q(\overline{X3}) \cdot q(X4) \cdot q(\overline{X7}) \cdot q(X9)$	$5.9622 \cdot 10^{-05}$
25	$q(X1, \overline{X2}) \cdot q(\overline{X3}) \cdot q(X4) \cdot q(\overline{X7}) \cdot q(\overline{X9}) \cdot q(X10)$	$4.8899 \cdot 10^{-08}$
26	$q(\overline{X1}) \cdot q(\overline{X3}) \cdot q(\overline{X4}) \cdot q(X7, X8) \cdot q(X5)$	$3.2780 \cdot 10^{-04}$
27	$q(\overline{X1}) \cdot q(\overline{X3}) \cdot q(\overline{X4}) \cdot q(X7, \overline{X8}) \cdot q(X9, X5)$	$1.7634 \cdot 10^{-05}$
28	$q(\overline{X1}) \cdot q(\overline{X3}) \cdot q(\overline{X4}) \cdot q(X7, \overline{X8}) \cdot q(\overline{X9}, X5) \cdot q(X10)$	$3.9372 \cdot 10^{-08}$
29	$q(\overline{X1}) \cdot q(\overline{X3}) \cdot q(\overline{X4}) \cdot q(\overline{X7}) \cdot q(X9, X5)$	$4.7611 \cdot 10^{-04}$
30	$q(\overline{X1}) \cdot q(\overline{X3}) \cdot q(\overline{X4}) \cdot q(\overline{X7}) \cdot q(\overline{X9}, X5) \cdot q(X10)$	$1.0631 \cdot 10^{-06}$
31	$q(X1, \overline{X2}) \cdot q(\overline{X3}) \cdot q(\overline{X4}) \cdot q(X7, X8) \cdot q(X5)$	$1.1393 \cdot 10^{-04}$
32	$q(X1, \overline{X2}) \cdot q(\overline{X3}) \cdot q(\overline{X4}) \cdot q(X7, \overline{X8}) \cdot q(X9, X5)$	$6.1287 \cdot 10^{-06}$
33	$q(X1, \overline{X2}) \cdot q(\overline{X3}) \cdot q(\overline{X4}) \cdot q(X7, \overline{X8}) \cdot q(\overline{X9}, X5) \cdot q(X10)$	$1.3684 \cdot 10^{-08}$
34	$q(X1, \overline{X2}) \cdot q(\overline{X3}) \cdot q(\overline{X4}) \cdot q(\overline{X7}) \cdot q(X9, X5)$	$1.6548 \cdot 10^{-04}$
35	$q(X1, \overline{X2}) \cdot q(\overline{X3}) \cdot q(\overline{X4}) \cdot q(\overline{X7}) \cdot q(\overline{X9}, X5) \cdot q(X10)$	$3.6947 \cdot 10^{-07}$
36	$q(\overline{X1}) \cdot q(\overline{X3}) \cdot q(\overline{X4}) \cdot q(X7, X8) \cdot q(\overline{X5}) \cdot q(X6)$	$1.6531 \cdot 10^{-07}$
37	$q(\overline{X1}) \cdot q(\overline{X3}) \cdot q(\overline{X4}) \cdot q(X7, \overline{X8}) \cdot q(X9, \overline{X5}) \cdot q(X6)$	$2.4174 \cdot 10^{-08}$
38	$q(\overline{X1}) \cdot q(\overline{X3}) \cdot q(\overline{X4}) \cdot q(X7, \overline{X8}) \cdot q(\overline{X9}, \overline{X5}) \cdot q(X10) \cdot q(X6)$	$1.9233 \cdot 10^{-11}$
39	$q(\overline{X1}) \cdot q(\overline{X3}) \cdot q(\overline{X4}) \cdot q(\overline{X7}) \cdot q(X9, \overline{X5}) \cdot q(X6)$	$6.5268 \cdot 10^{-07}$
40	$q(\overline{X1}) \cdot q(\overline{X3}) \cdot q(\overline{X4}) \cdot q(\overline{X7}) \cdot q(\overline{X9}, \overline{X5}) \cdot q(X10) \cdot q(X6)$	$5.1928 \cdot 10^{-10}$
41	$q(X1, \overline{X2}) \cdot q(\overline{X3}) \cdot q(\overline{X4}) \cdot q(X7, X8) \cdot q(\overline{X5}) \cdot q(X6)$	$5.7455 \cdot 10^{-08}$
42	$q(X1, \overline{X2}) \cdot q(\overline{X3}) \cdot q(\overline{X4}) \cdot q(X7, \overline{X8}) \cdot q(X9, \overline{X5}) \cdot q(X6)$	$8.4016 \cdot 10^{-09}$
43	$q(X1, \overline{X2}) \cdot q(\overline{X3}) \cdot q(\overline{X4}) \cdot q(X7, \overline{X8}) \cdot q(\overline{X9}, \overline{X5}) \cdot q(X10) \cdot q(X6)$	$6.6844 \cdot 10^{-12}$
44	$q(X1, \overline{X2}) \cdot q(\overline{X3}) \cdot q(\overline{X4}) \cdot q(X7, \overline{X8}) \cdot q(X9, \overline{X5}) \cdot q(X6)$	$2.2684 \cdot 10^{-07}$
45	$q(X1, \overline{X2}) \cdot q(\overline{X3}) \cdot q(\overline{X4}) \cdot q(X7, \overline{X8}) \cdot q(\overline{X9}, \overline{X5}) \cdot q(X10) \cdot q(X6)$	$1.8048 \cdot 10^{-10}$

TABLE V: Failure paths for the BDD in Fig.4

Component	Birnbaum Measure
X1	$2.4390 \cdot 10^{-02}$
X2	$2.4360 \cdot 10^{-02}$
X3	$4.7694 \cdot 10^{-02}$
X4	$4.7987 \cdot 10^{-02}$
X5	$1.9498 \cdot 10^{-02}$
X6	$4.7695 \cdot 10^{-02}$
X7	$4.9035 \cdot 10^{-02}$
X8	$4.9035 \cdot 10^{-02}$
X9	$1.7898 \cdot 10^{-01}$
X10	$1.9384 \cdot 10^{-01}$

TABLE VI: Components Birnbaum's Measure of Importance

component $X2$ to be working correctly implies:

$$\begin{aligned} & q(X2) = 0, q(\overline{X2}) = 1 \\ & q(X1, X2) = 0, q(X1, \overline{X2}) = q(X1|\overline{X2}) \\ & q(X1) = q(X1|\overline{X2}), q(\overline{X1}) = q(\overline{X1}|\overline{X2}) \end{aligned} \quad (18)$$

Hence, the calculation of the system failure probability conditional on the working or failure state of the dependent component $X2$ has to take into account not only the paths involving $X2$ itself, but also those including the other components of the dependency groups **DG1** to which $X2$ belongs (i.e. $X1$).

When $X2$ is working correctly, BDD paths from 1 to 5 shown in Table V assume probability equal to 0 (due to $q(X1, X2) = 0$, as for Eq.18). Paths 21 to 25, 31 to 35 and 41 to 45 contain the simultaneous occurrence of dependent events $X1$ and $\overline{X2}$, hence their probability contains the joint value $q(X1, \overline{X2})$. According to the conditions in Eq. 18, this is equal to the conditional probability $q(X1|\overline{X2})$ under the assumption of $X2$ working. Hence, in this case the contribution of these paths to the system failure probability can be estimating dividing their individual probability by the $q(\overline{X2})$, which is equivalent to substituting the joint value $q(X1, \overline{X2})$ with the conditional probability $q(X1|\overline{X2})$ in the expression of the individual path probability.

The probability associated with the paths including event $\overline{X1}$ (i.e., paths 6 to 20, 26 to 30 and 36 to 40) needs to be 'updated' according to the working assumption substituting the probability $q(\overline{X1})$ with the conditional value $q(\overline{X1}|\overline{X2})$. This can be achieved multiplying the unconditional probability associated with such paths by $\frac{q(\overline{X1}|\overline{X2})}{q(\overline{X1})}$. Overall, the system failure probability conditional to the working state of $X2$ can then be expressed as:

$$\begin{aligned} & Q_{system}(\overline{X2}) \\ & = \frac{\sum_{i=21}^{25} q(P_i) + \sum_{j=31}^{35} q(P_j) + \sum_{k=41}^{45} q(P_k)}{q(\overline{X2})} \\ & + \left(\sum_{l=6}^{20} q(P_l) + \sum_{w=26}^{30} q(P_w) + \sum_{h=36}^{40} q(P_h) \right) \\ & \cdot \frac{q(\overline{X1}|\overline{X2})}{q(\overline{X1})} = 1.6826 \cdot 10^{-03} \end{aligned} \quad (19)$$

Similarly, the failure probability of the system conditional to the failure of $X2$, can be computed as:

$$
\begin{aligned}
&Q_{system}(X2) \\
&= \frac{\sum_{i=1}^{5} q(P_i)}{q(X2)} \\
&+ \left(\sum_{l=6}^{20} q(P_l) + \sum_{w=26}^{30} q(P_w) + \sum_{h=36}^{40} q(P_h) \right) \\
&\cdot \frac{q(\overline{X1}|X2)}{q(\overline{X1})} = 2.6042 \cdot 10^{-02}
\end{aligned} \tag{20}
$$

As a result, the Birnbaum importance measure for the dependent component $X2$ can be calculated as:

$$
\begin{aligned}
&G(X3) = Q_{system}(X2) - Q_{system}(\overline{X2}) \\
&= \frac{\sum_{i=1}^{5} q(P_i)}{q(X2)} \\
&- \frac{\sum_{i=21}^{25} q(P_i) + \sum_{j=31}^{35} q(P_j) + \sum_{k=41}^{45} q(P_k)}{q(\overline{X2})} \\
&+ \left(\sum_{l=6}^{20} q(P_l) + \sum_{w=26}^{30} q(P_w) + \sum_{h=36}^{40} q(P_h) \right) \\
&\cdot \left(\frac{q(\overline{X1}|X2)}{q(\overline{X1})} - \frac{q(\overline{X1}|\overline{X2})}{q(\overline{X1})} \right) \\
&= 2.4360 \cdot 10^{-02}
\end{aligned} \tag{21}
$$

which corresponds to the sum of all path contributions obtained with Eq.14 according to the procedure described in Algorithm 5.

D. System Failure Intensity

Once the Birnbaum's measure of component importance are computed, the failure intensity of the overall system, i.e. F_{system}, can be calculated. As for the failure probability values, the individual failure intensity of dependent components can be calculated through the marginalisation procedure discussed in the previous sections. This finally results in:

$$
\begin{aligned}
F_{system} &= \sum_{i=1}^{10} G(X_i) \cdot f(X_i) \\
&= G(X_1) \cdot f(X_1) + G(X_2) \cdot f(X_2) \\
&+ G(X_3) \cdot f(X_3) + G(X_4) \cdot f(X_4) \\
&+ G(X_5) \cdot f(X_5) + G(X_6) \cdot f(X_6) \\
&+ G(X_7) \cdot f(X_7) + G(X_8) \cdot f(X_8) \\
&+ G(X_9) \cdot f(X_9) + G(X_{10}) \cdot f(X_{10}) \\
&= 1.2440 \cdot 10^{-01} h^{-1}
\end{aligned} \tag{22}
$$

V. Conclusions

A novel FT analysis methodology based on the use of BDDs and allowing for component dependencies is proposed. The approach relies on the conversion of FT to BDD and the identification of the associated paths to failure. Each path probability is estimated taking into account the contribution of independent and dependent component, adopting joint probabilities to capture the relationships of dependent events. The resulting approach guarantees preserve the familiarity and efficiency of Fault Tree analysis while enhancing modelling flexibility. This is achieved allowing for the inclusion in the analysis of any type of dependency regardless of their location within the Fault Tree structure.

Acknowledgment

This work is funded by the Lloyd's Register Foundation, an independent global charity that helps to protect life and property at sea, on land, and in the air, by supporting high quality research, accelerating technology to application and through education and public outreach.

References

[1] J. B. Dugan, S. J. Bavuso, and M. A. Boyd, "Dynamic fault-tree models for fault-tolerant computer systems," *IEEE Transactions on reliability*, vol. 41, no. 3, pp. 363–377, 1992.

[2] M. Čepin and B. Mavko, "A dynamic fault tree," *Reliability Engineering & System Safety*, vol. 75, no. 1, pp. 83–91, 2002.

[3] K. D. Rao, V. Gopika, V. S. Rao, H. Kushwaha, A. K. Verma, and A. Srividya, "Dynamic fault tree analysis using monte carlo simulation in probabilistic safety assessment," *Reliability Engineering & System Safety*, vol. 94, no. 4, pp. 872–883, 2009.

[4] M. Volk, S. Junges, and J.-P. Katoen, "Fast dynamic fault tree analysis by model checking techniques," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 1, pp. 370–379, 2017.

[5] H. Xu and J. B. Dugan, "Combining dynamic fault trees and event trees for probabilistic risk assessment," in *Annual Symposium Reliability and Maintainability, 2004-RAMS*, pp. 214–219, IEEE, 2004.

[6] S. Kabir, M. Walker, and Y. Papadopoulos, "Dynamic system safety analysis in hip-hops with petri nets and bayesian networks," *Safety science*, vol. 105, pp. 55–70, 2018.

[7] W. Reisig, *Petri nets: an introduction*, vol. 4. Springer Science & Business Media, 2012.

[8] J. Andrews and S. Tolo, "Dynamic and dependent tree theory (d2t2): A framework for the analysis of fault trees with dependent basic events," *Reliability Engineering & System Safety*, vol. 230, p. 108959, 2023.

[9] S. Tolo and J. Andrews, "An integrated modelling framework for complex systems safety analysis," *Quality and Reliability Engineering International*, vol. 38, no. 8, pp. 4330–4350, 2022.

[10] A. Rauzy, "Binary decision diagrams for reliability studies," in *Handbook of performability engineering*, pp. 381–396, Springer, 2008.

[11] R. Drechsler and B. Becker, *Binary decision diagrams: theory and implementation*. Springer Science & Business Media, 2013.

[12] R. M. Sinnamon and J. Andrews, "New approaches to evaluating fault trees," *Reliability Engineering & System Safety*, vol. 58, no. 2, pp. 89–96, 1997.

[13] L. Bartlett and J. Andrews, "Choosing a heuristic for the "fault tree to binary decision diagram" conversion, using neural networks," *IEEE Transactions on Reliability*, vol. 51, no. 3, pp. 344–349, 2002.

[14] L. M. Bartlett and J. D. Andrews, "Comparison of two new approaches to variable ordering for binary decision diagrams," *Quality and Reliability Engineering International*, vol. 17, no. 3, pp. 151–158, 2001.

[15] R. M. Sinnamon and J. Andrews, "Improved accuracy in quantitative fault tree analysis," *Quality and reliability engineering international*, vol. 13, no. 5, pp. 285–292, 1997.