Dynamic and Dependent Tree Theory (D^2T^2): A Fault Tree Analysis Framework

John Andrews

Resilience Engineering Research Group, University of Nottingham, UK. E-mail: john.andrews@nottingham.ac.uk

Silvia Tolo

Resilience Engineering Research Group, University of Nottingham, UK. E-mail: silvia.tolo@nottingham.ac.uk

A modelling framework, known as Dynamic and Dependent Tree Theory, is provided which overcomes restrictions in the Kinetic Tree Theory developed in the 1970s for the analysis of fault trees. Kinetic Tree Theory assumes basic events to be independent and its implementation generally requires that the component failure models feature constant failure and repair rates and the options for representing maintenance processes are very limited. This paper describes the modelling framework which integrates Binary Decision Diagram methods, Petri nets and Markov approaches to remove the need for the restrictive assumptions which therefore enables the fault tree model to better represent actual system characteristics.

Keywords: Fault Tree Analysis, Binary Decision Diagrams, Petri Nets, Markov, Dependencies.

1. Background

Fault Tree Analysis has its origins back in the 1960's and its conception is attributed to Watson of Bell Laboratory when analysing the causes of an inadvertent launch of the Minuteman Intercontinental Ballistic Missile. The time dependent mathematical framework, known as Kinetic Tree Theory (KTT), (Vesely, 1970) was added at the end of the decade. In this methodology, the analysis of the fault tree is performed in two stages. The first delivers the qualitative analysis producing minimal cut sets, the second, quantitative stage then produces the system failure mode probability and frequency.

Performing the calculations requires assumptions about the operation and design of the system which will result in the independence of all basic events. In most commercial packages there are also very limited models for the component probabilities which assume constant failure and repair rates, and maintenance strategies limited to dealing with either non-repairable components, or repairable components whose failures are revealed and unrevealed.

Since the 1970s advances have been made in the technologies employed in systems design, along with their operation and maintenance that limit the ability of these traditional techniques to represent modern system performance. Advances have also been made in fault tree analysis methods and system failure modelling capabilities. Significant examples being the exploitation of Binary Decision Diagrams (BDDs), (Rauzy, 1993) and Petri Nets, (Andrews, 2017). These methods, along with Markov models, (Andrews, 2002), provide the tools to create a framework which removes the assumptions of:

- Component independence
- Component constant failure and repair rates
- Simplistic maintenance strategies

2. Dynamic and Dependent Tree Theory (D^2T^2) Algorithm

The foundation of the methodology is to exploit the BDDs ability to represent the causes of the fault tree top event in a logic equation which has a disjoint form. Since the paths through the BDD are mutually exclusive, the dependencies and complexities, modelled by the Petri net and Markov models, only need to be considered within the context of the paths through the BDD. In order to apply the algorithm, as shown in Fig.1, the problem needs to be broken down into a series of independent modules. Dependencies contained within each module are solved with Petri nets or Markov models whose results

Extended Abstract Collection of the 32nd European Safety and Reliability Conference
Edited by Maria Chiara Leva, Edoardo Patelli, Luca Podofillini, and Simon Wilson

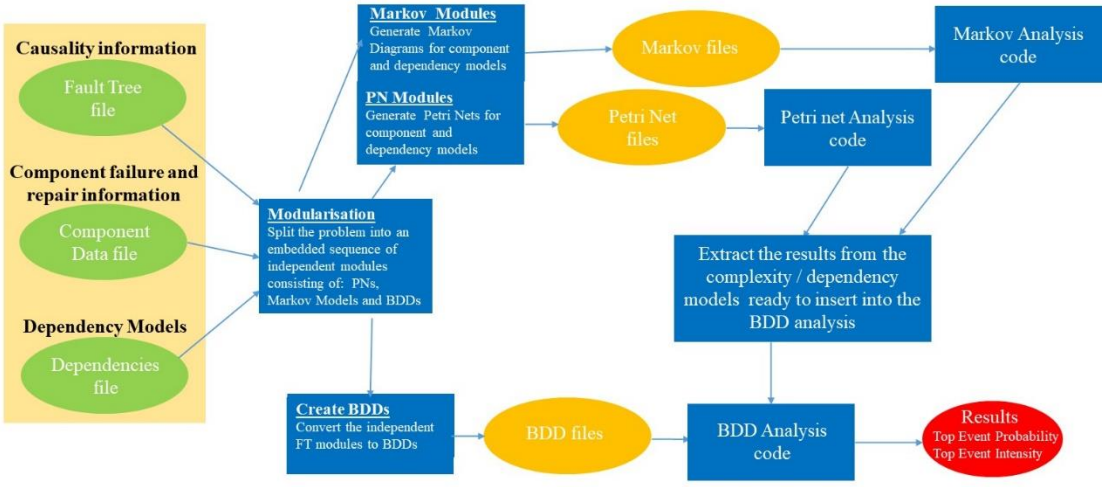

Fig. 1 D^2T^2 Modelling Process

Are substituted into the path calculations to derive the top event probability and failure intensity. To minimize the number and size of the paths through the BDD is essential in order to perform the analysis efficiently and so the modularisation process is key.

3. Modularisation

Basic Events which are mutually dependent are placed in a common dependency group. Modularization is carried then out in 2 stages. The first uses the Faunet approach with some extensions to account for events which appear in the same dependency group. This is applied to reduce the complexity of the fault tree. Then the algorithm of Rauzy & Dutuit (1998) is utilized to enable smaller independent modules of the fault tree structure to be identified. In order to be applicable for fault trees with dependent basic events, events in the same dependency group are given the same label when applying the algorithm.

4. Application

The application of the Dynamic and Dependent Tree Theory Framework is demonstrated using a pressure vessel cooling system example.

Acknowledgement

This work was supported by the Lloyd's Register Foundation, a charitable foundation in the U.K. helping to protect life and property by supporting engineering-related education, public engagement, and the application of research.

Andrews, J and Fecarotti, C. (2017). System design and maintenance modelling for safety in extended life operation, Reliability Engineering and System Safety, 163, 95-108.

Andrews, J.D. and Moss, T.R. (2002). Reliability and Risk Assessment, Professional Engineering Publishing Ltd.

Duituit E. and Rauzy, A. (1996) A linear-time algorithm to find modules in fault trees, IEEE Trans Relaibility, 45 No 3, 422-425.

Rauzy, A. (1993), New algorithms for fault tree analysis, Reliability Engineering and System Safety, 203-211.

Platz, O. and Olsen, J.V. (1976) FAUNET: A program package for evaluation of fault trees and networks, Riso Laboratories Report No 348. DK-4000 Roskilde.

Vesely, W.E. (1970). A time dependent methodology for fault tree evaluation, Nuclear Engineering and Design, 337-360.