Reliability Engineering and System Safety 109 (2013) 5–17

Contents lists available at SciVerse ScienceDirect

Reliability Engineering and System Safety

journal homepage: www.elsevier.com/locate/ress

Risk-based design of process systems using discrete-time Bayesian networks

Nima Khakzad [a,*], Faisal Khan [a], Paul Amyotte [b]

[a] Process Engineering, Faculty of Engineering & Applied Science, Memorial University, St. John's, NL, Canada A1B 3X5
[b] Department of Process Engineering and Applied Science, Dalhousie University, Halifax, NS, Canada B3J 2X4

ARTICLE INFO

Article history:
Received 23 November 2011
Received in revised form
17 May 2012
Accepted 29 July 2012
Available online 29 August 2012

Keywords:
Discrete-time Bayesian network
Dynamic fault tree
Markov chain
Neutral dependency
Safety analysis
Probabilistic risk assessment

ABSTRACT

Temporal Bayesian networks have gained popularity as a robust technique to model dynamic systems in which the components' sequential dependency, as well as their functional dependency, cannot be ignored. In this regard, discrete-time Bayesian networks have been proposed as a viable alternative to solve dynamic fault trees without resort to Markov chains. This approach overcomes the drawbacks of Markov chains such as the state-space explosion and the error-prone conversion procedure from dynamic fault tree. It also benefits from the inherent advantages of Bayesian networks such as probability updating. However, effective mapping of the dynamic gates of dynamic fault trees into Bayesian networks while avoiding the consequent huge multi-dimensional probability tables has always been a matter of concern. In this paper, a new general formalism has been developed to model two important elements of dynamic fault tree, i.e., cold spare gate and sequential enforcing gate, with any arbitrary probability distribution functions. Also, an innovative *Neutral Dependency* algorithm has been introduced to model dynamic gates such as priority-AND gate, thus reducing the dimension of conditional probability tables by an order of magnitude. The second part of the paper is devoted to the application of discrete-time Bayesian networks in the risk assessment and safety analysis of complex process systems. It has been shown how dynamic techniques can effectively be applied for optimal allocation of safety systems to obtain maximum risk reduction.

1. Introduction

Among several techniques available to quantify the occurrence probability of accident scenarios or to estimate the failure probability of systems in the context of quantitative risk assessment, probabilistic safety analysis and reliability engineering, the fault tree (FT) method is the most recognized and widely used. FT is a deductive, user-friendly methodology constructed intuitively, dissecting the system for further detail until the primary causes of the system's failure or unavailability are known. FT could also be analyzed using well-established algorithms such as binary decision diagrams or analytical methods such as minimal cut sets. However, conventional or static fault trees (SFTs) are characterized by limitations constraining their application in complex systems where, for instance, redundant failures, multi-state variables and/or sequential and functional dependencies are common.

In recent years, Bayesian networks (BNs) have become popular for reliability and risk analysis of complex systems as a robust and viable alternative to most conventional methods such as reliability block diagrams [1], FT [2–4] and event tree (ET) analysis [5]. BN is a probabilistic method for reasoning under uncertainty which factorizes the joint probability distribution of a set of variables by considering local dependencies, significantly reducing both the system complexity and the computational time [2–4,6,7]. Most recently, Weber et al. [7] have given a statistical review of BN application and shown the appeal of Bayesian approaches in various areas of reliability, risk and maintenance engineering since 2000.

Many authors have shown the parallels between FT and BN and have examined the extent to which the limitations of the former can be relaxed by relying on the later. Bobbio et al. [2] were the first to map FT into BN to incorporate multi-state variables and common cause failures by means of the leaking noisy-or model. They also performed a sequentially dependent failure analysis which was an example of functional dependency, i.e., without considering the temporal sequence of failures (like the performance of the functional dependency gate in dynamic fault trees). Similar efforts have been made by Langseth and Portinale [3] to account for coverage factors in redundant systems by means of the noisy-and model, and also by Khakzad et al. [4] to explicitly model functional uncertainty and expert opinion in the safety analysis of process systems.

Dynamic fault tree (DFT) was introduced as an extension to SFT to model sequentially dependent failures in dynamic systems [8]. In a dynamic system, the failure sequence of events is as

* Corrsepondence to: Box 50, Faculty of Engineering, Memorial University, St. John's, NL, Canada A1B 3X5.
E-mail addresses: nkhakzadrostami@mun.ca (N. Khakzad), fikhan@mun.ca (F. Khan).

0951-8320/$ - see front matter © 2012 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.ress.2012.07.009

important as their combinations for the system to be unavailable or to fail. In other words, compared to SFT in which it only matters which components participate in a minimal cut set, in DFT the failure sequence of the participating components is also important [9]. DFT takes the sequential dependencies into account by using several dynamic gates such as a functional dependency gate (FDEP), cold spare gate (CSP), sequence enforcing gate (SEQ) [8] and priority-AND gate (PAND) [10].

Due to the sequential dependencies and dynamic behavior among the components of the system, DFT cannot be analyzed using conventional algorithms available for SFT. In this regard, DFT has traditionally been converted to the corresponding Markov chain model (MC) for which well-established and efficient solving techniques have been developed. Nevertheless, converting DFT into MC is an error-prone and cumbersome exercise [8]. Moreover, the state space of the MC (i.e., the set of its nodes) grows exponentially with the number of components of the corresponding DFT, making the MC very large and intractable. Indeed, for a MC equivalent to a DFT with m binary-state components (i.e., work/fail) for which k out of m components are sequentially dependent, the number of states is proportional to the product of 2^m (the number of state combinations) and $k!$ (the possible number of sequence combinations) [6]. This problem is frequently encountered in Markov processes and is referred to as the state space explosion. It should be noted that even a relatively simple DFT can result in a complicated and time-consuming MC, particularly in the presence of dynamic gates cascade [6,8,9,11]. Also, MC has been mentioned to have limitations in modeling dependencies among components with non-exponential failure time distributions [11].

As an example, consider a parallel system consisting of three pumps A, B and C of different failure rates, in which B is planned to only operate as a standby to A. In other words, not only all three pumps have to fail for the system to fail, but also A must fail before B. Fig. 1 illustrates the SFT (left), the DFT (middle) and the equivalent MC (right) for the failure analysis of the system. As the SFT cannot capture sequential failures, it ignores the sequential dependence between A and B, approximating the system failure using an AND gate. On the other hand, the DFT employs a cascade of SEQ gate and AND gate to model the dynamic behavior. The DFT is then conventionally converted to the MC to be solved. Assuming a mission time of $t=100$ h and the failure rates 0.3E-03, 0.5E-03 and 0.7E-03 for A, B and C, respectively, the failure probability of the system is calculated as 9.76E-05 and 4.94E-05 using SFT and DFT (MC), respectively. This example demonstrates how the failure probability and also the consequently envisaged risk in dynamic systems can be overestimated (here by a factor of two) if dependency conditions are ignored or simplified through using static techniques.

Considering the abovementioned problems encountered in converting DFT into MC, temporal Bayesian networks (TBNs) have alternatively been proposed to explicitly incorporate time in the modeling of sequential dependencies without resort to MC. Accordingly, two different approaches have been adopted: instant-based (time-sliced) approach and interval-based (event-based) approach [12]. In the first approach, the time line is divided into a finite number of time instants (e.g., t_{i-1},t_i,t_{i+1}), and identical BN structures are generated for each time instant, connected to each other by means of temporal arcs (e.g., [13,14]). In the second approach, the time line is partitioned into a finite number of time intervals (e.g., $]t_{i-1},t_i]$,$]t_i,t_{i+1}]$), and only one BN is generated, each node of which has a finite number of states equal to the number of time intervals [6,11,12] (see Section 2.2). Fig. 2 illustrates how a CSP gate is converted into interval-based and instant-based (here, a 2-time-slice) BN structures.

Following the instant-based approach, Montani et al. [13] developed the RADYBAN software tool for reliability analysis of dynamic systems by converting DFT into a 2-time-slice BN. They also introduced the probability dependency gate (PDEP) as a probabilistic case of FDEP proposed by Dugan et al. [8]. Their work was further developed by Portinale et al. [14], enabling the modeling of repairable systems by introducing the repair box gate. The instant-based approach has been criticized for either being too general or resulting in unnecessarily large networks due

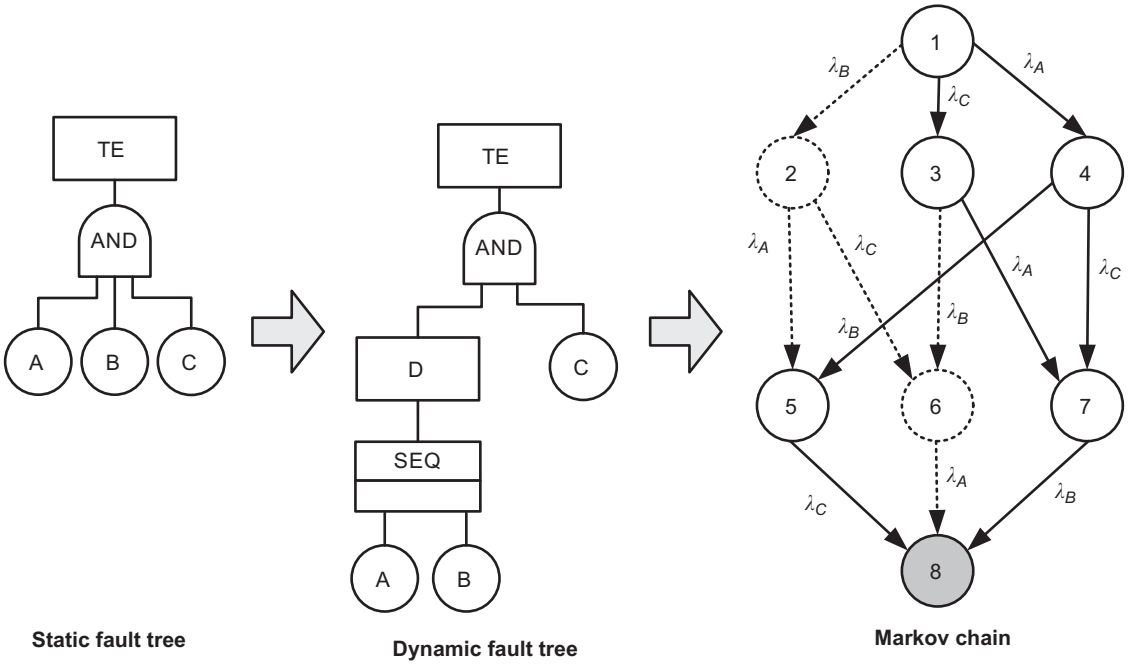

Fig. 1. SFT (left), DFT (middle) and MC (right) models for a three-component parallel system in which *A* must fail before *B*. The dashed parts in the MC are not accounted for in the system failure due to the representation of improper failure sequences. λ is the failure rate of components.