

# Solving Dynamic Fault Trees using a New Hybrid Bayesian Network Inference Algorithm

David Marquez<sup>†</sup>, Martin Neil <sup>\*\*</sup> and Norman Fenton<sup>†\*</sup>

<sup>†</sup> Department of Computer Science, Queen Mary, University of London

<sup>‡</sup> Agena Ltd, London

*Abstract — We present a Hybrid Bayesian Network (HBN) framework to analyse dynamic fault trees. By incorporating a new approximate inference algorithm for HBNs involving dynamically discretising the domain of all continuous variables, accurate approximations for the failure distribution of both static and dynamic fault tree constructs are obtained. Unlike in other approaches no numerical integration techniques or simulation methods are required. Moreover, no exact expression for the posterior marginal is needed and no conditional probability tables need to be completed. Sensitivity analysis, uncertainty, diagnosis, common cause failure analysis, can all be easily performed within this framework. Posterior estimates of parameterised marginal failure distributions can also be obtained using available raw failure data together with prior information from expert judgement.*

## I. INTRODUCTION

**D**YNAMIC fault trees (DFTs), [5], were developed primarily to capture the complex dynamic behaviour of the failure mechanisms of fault tolerant systems. DFTs increase the modelling capabilities of traditional (static) Fault Trees (FTs), [27], [30], by taking into account not only the combinations but also the sequential ordering of occurrence of component failures' that lead to system failure. Despite the analytical power of the DFT approach, in practice it has a severe drawback: DFTs are traditionally solved using Markov chain analysis techniques, based on the specification and modelling of the whole set of possible states of the system and the transitions between them. Consequently, the state-space generated grows exponentially with the number of components in the system, with a concomitant impact on computation times. Furthermore, the requirement for analytical tractability in Markov chain based approaches constrains the failure times of the system components to be modelled as exponential distributed variables. This makes DFTs much too inflexible for analysing general standby redundant systems; for

example, DFTs cannot handle the very common fault-tolerant system architecture of warm or cold spares with non-exponential time-to-failure distributions.

Recently, several works, [3], [19], [26], [31] have proposed that Bayesian Networks (BNs), [11], [25], and their extension for time-dependent modelling known as Dynamic Bayesian Networks (DBNs), [20], offer a unified framework for reliability modelling and analysis of complex dynamic systems. BNs have been shown to increase both the modelling capabilities and analysis power of DFT models, by including new features - like general component failure distributions, multi-state variables, noisy gates, common cause failures, and simple sequentially dependent failures - and general *a posteriori* diagnostic analysis, [29], [26], [3], [15]. More importantly, the BN framework allows a compact representation of the temporal (and functional) dependencies among the system components and event-dependent failure behaviours, characteristic of fault-tolerant systems, avoiding the state space explosion problem of the Markov Chain based approaches to DFT analysis.

In spite of all these modelling and analytical advantages, the real Achilles' heel of the applicability of BNs as a mainstream framework for reliability modelling of complex systems, is that previous attempts to apply BN models to reliability assessment have not adequately handled the necessary 'hybrid' models required in real real-world applications, i.e. models containing both continuous and discrete variables, with general static and time-dependent failure distributions. It is this problem that we address in this paper.

We will show how a simple event-based HBN reliability model, together with a new approximate inference algorithm for HBNs, can be used to perform DFT-like analysis of complex fault tolerant systems. The new approximate inference algorithm effectively integrates a dynamic discretisation schema on the domain of all continuous variables in the HBN, within existing robust local computation algorithms on BN architectures, such as Junction Trees.

D. Marquez is with Queen Mary, University of London, Mile End Road, London, England E1 4NS (phone: +44(0) 207 882 8027; e-mail: marquezd@dcs.qmul.ac.uk).

M. Neil is with Queen Mary, University of London, Mile End Road, London, England E1 4NS, (e-mail: martin@dcs.qmul.ac.uk).

N. Fenton is with is with Queen Mary, University of London, Mile End Road, London, England E1 4NS (e-mail: norman@dcs.qmul.ac.uk).

Our HBN based reliability framework allows us to accurately approximate the conditional probability distributions (CPDs) of the Time to Failure for all DFT constructs, as well as obtain posterior estimates of the parameterised marginal distribution of the Time to Failure of basic components of the system, using available raw failure data together with prior information from expert judgement. Furthermore, from the approximate failure distributions we can automatically obtain estimates of any reliability metric of interest, such as the reliability of the system at any mission time, warranty periods, and mean time to failure. Our framework enables us to solve any configuration of static and dynamic gates with any parametric or empirical distribution (so unlike previous approaches it is not restricted to the Exponential distribution), without using numerical integration techniques or simulation methods. All the example models shown in this paper are built and executed using a commercial general-purpose Bayesian Network software tool [1], in which our dynamic discretisation algorithm is now implemented.

## II. THE BAYESIAN NETWORK DYNAMIC FAULT TREE FRAMEWORK

### A. Bayesian Network Overview

A Bayesian Network (BN), [11], [25], consists of

1) a *qualitative part*: a directed acyclic graph (DAG), such as the one shown in Figure 1, with nodes representing random variables and directed arcs (from parent to child) representing causal or influential relationships between variables, and

2) a *quantitative part*: consisting of conditional probability distributions of each node given its respective parents, and marginal probability distributions of the nodes without parents (root nodes). Together, the qualitative and quantitative parts of the BN determine the joint probability distribution of all the random variables presented in the model.

The conditional independence assertions about the variables, represented by the lack of arcs in the DAG, allow decomposition of the underlying joint probability distribution as a product of CPDs. This significantly reduces the complexity of inference tasks on the BN [16]. If the variables are discrete, the CPDs can be represented as Node Probability Tables (NPTs), which list the probability that the child node takes on each of its different values for each combination of values of its parents. For continuous variables, the CPDs represent conditional probability density functions.

BNs are an effective and robust decision support framework for problems involving uncertainty and

probabilistic reasoning. They are mathematically sound and at the same time flexible and simple enough to allow interaction with domain experts and decision makers. To date BNs have proven useful in many areas of application such as medical expert systems, pattern matching, speech recognition, and diagnosis of failures of complex dynamic systems. We have applied BNs to a range of real-world dependability-type problems [21], [24]. In particular, in the area of software system reliability, we have shown the advantages of BNs over traditional regression methods, [6], [7], [8], [22].

### B. The Dynamic discretisation approach to inference

Once the BN structure and node probability distributions have been defined, inference analysis is carried out using standard BN propagation algorithm (Lauritzen et al. 1988, Jensen et al. 1990). Unfortunately, for hybrid BNs containing mixtures of discrete and continuous nodes with non-Gaussian distributions, exact inference becomes computationally intractable and only approximate solutions can be obtained. The traditional approach to handling (non-Gaussian) continuous nodes is static discretisation. This approach requires users to define a uniform discretisation of the domains of all continuous nodes, using some pre-defined sequence of intervals. These remain static throughout all subsequent stages of exact inference performed on the resulting discrete BN. The more intervals you define, the more accuracy you can achieve, but at a heavy cost of computational complexity. This is made worse by the fact that where a model contains numerical nodes having a potentially large range, results are necessarily only crude approximations. A static discretisation approach to an event-based BN reliability model is given in [4], where the time line is partitioned into a finite number of time intervals. To overcome these problems, we have developed a new and powerful approximate algorithm for performing inference in hybrid BNs. We use a process of dynamic discretisation of the domain of all continuous variables in the BN. Our approach to dynamic discretisation is influenced by work of Kozlov and Koller, [13], on using an approximation of the Kullback-Leibler (KL) metric between two density functions, [14], as an estimate of the relative entropy error induced by the discretisation.

The idea of discretisation is to approximate the target density function by first partitioning its domain into a set of intervals and then defining a locally constant function on the partitioning intervals. Under the KL metric, the optimal value for the discretised approximation of the target density function is given by the mean of the target density function in each of the intervals of the discretised domain. The discretisation task reduces then to finding an optimal partition of the domain of the target probability density function.

Our approach to dynamic discretisation generates a

sequence of discretisation intervals iteratively, searching the domain of the target marginal density functions for the most accurate specification of the high-density regions given the model and the evidence. By efficiently integrating our iterative approximation scheme within existing robust propagation algorithms on BN architectures, such as Junction Tree [17], we are able to iteratively propagate and query the BN to get approximations to the marginal probability density function for each node and then, at each iteration, split those intervals in their domain with highest entropy error. A number of interval-merging procedures are also applied in order to control the growth of the resulting discretisation sets after each discretisation step. At each iteration, a candidate partition is tested to determine whether the error of the corresponding discretised probability density is below a given threshold, defined according to the trade off between the acceptable degree of precision and computation time. This procedure continues to iterate until the model converges to an acceptable level of accuracy.

By performing finer discretisation on the regions that contribute more to the structure of the density functions, i.e., the regions with higher information content, our dynamic discretisation approach produces more accurate results and incurs less storage space over static discretisation. A detailed description of the dynamic discretisation algorithm is given in [23].

### C. The event-based BN approach to DFT analysis

We adopt a simple event-based HBN reliability model in which continuous nodes represent the time-to-failure of the components of the system. These can be either the time-to-failure of elementary components of the system or the time-to-failure of fault tree constructs. In the latter case, the nodes in the HBN are connected by means of incoming arcs to several components' time-to-failures. The DFT is therefore mapped into an equivalent HBN with continuous

nodes representing the time-to-failure of the elementary components and gates. Both the qualitative and quantitative parts of the HBN model the dependency and interactions between the basic components and gates of the DFT. The HBN structure is created in a modular way by combining a predefined set of HBN substructures designed to capture the failure mechanisms of the different (existing or new) DFT constructs. In the HBN framework, the nodes representing independent components of the system (inputs nodes in static FT analysis, or dynamic gates with independent inputs) are characterised by their marginal (prior) probability distributions. These are generally given by standard parametric probability density functions, where the values of the parameters are either obtained as prior information based on expert knowledge, or estimated in a previous reliability data analysis step. The basic idea of our HBN modelling approach is to define the time-to-failure of the fault tree constructs as a deterministic function of the input components, according to the type of fault tree construct. The CPDs of the time-to-failure of fault tree constructs are then defined as probability distributions of variables that are a deterministic function of its parents. For some simple configurations, such as static gates or dynamic gates with exponential time-to-failure components distributions, an exact closed-form analytical expression can be derived for the CPDs. However, for general components' failure distributions, a closed-form expression for the CPDs of dynamic gates may not be feasible, so numerical approximation methods need to be applied.

Figure 1 shows an HBN derived to solve a DFT, comprising two cascaded PAND gates. The shapes of the nodes in the HBN have been edited to conform to standard DFT symbols. Note that the fact that the HBN contains cascaded PAND gates makes this problem practically impossible to solve using Markov chain analysis techniques.

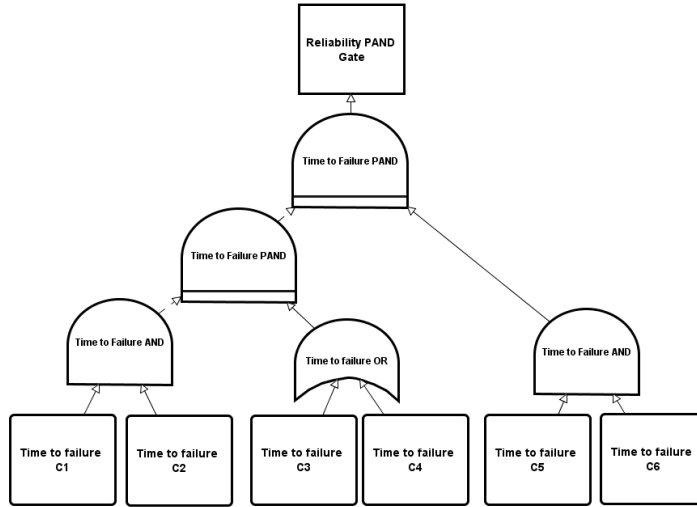


Figure 1. Cascaded PAND system represented as an HBN

In order to compute the CPDs of the time-to-failure of fault tree constructs we define the time-to-failure of the fault tree constructs as deterministic function of its inputs components. The output of the upper PAND gate will fail if all of its input components fail in a predefined order (left to right). Thus, if  $\tau_{PAND}$  represents the time-to-failure of the output event, then, the time-to-failure of the PAND gate can be written as

$$\tau_{PAND1} = \begin{cases} \tau_2 & \text{if } \tau_{PAND2} \leq \tau_{AND} \\ \infty & \text{otherwise} \end{cases} \quad (1)$$

Here,  $\tau_{PAND2}$  and  $\tau_{AND}$  represent the time-to-failure of the lower PAND and AND gates output events, respectively. The output of an AND gate fails if all the input components,  $C_i$ ,  $i=1,2$ , of the gate fail. Therefore, the time-to-failure of the AND gate is a random variable defined as a function of its parents by

$$\tau_{AND} = \max_{C_i} \{\tau_{C_i}\} \quad (2)$$

Finally, the output of the OR gate will fail if at least one of the input components of the gate fails. Then, we can define the time-to-failure of the OR gate by

$$\tau_{OR} = \min_{C_i} \{\tau_{C_i}\} \quad (3)$$

Once we have determined the conditional time-to failure distributions for the basic components, the CPDs for the DFT constructs are automatically estimated using the dynamic discretisation algorithm. The BN model for our cascade system example is depicted in Figure 2, with posterior marginal distributions superimposed on the BN graph. In this example we assumed that all the elementary

components,  $C_i$ ,  $i=1,\dots,6$ , are exponentially distributed<sup>1</sup>, with constant failure rates given in Table 1.

We also included in the HBN model a binary node (with an incoming arc from  $\tau_{PAND}$ ) representing the state of the system ('Fail', or 'On') at a particular time instance  $t$ . The NPT of the discrete node gives us an estimate of the reliability of the system at a given time. This is automatically computed from the system time-to-failure distribution by  $P_i(PAND = fail) = P(\tau_{PAND} \leq t)$ . In our example, an analytic closed form solution can be obtained for the reliability of the system at any mission time  $t$  (see Appendix). From the estimated failure distributions of the DFT constructs, we also obtain estimates for other reliability metrics of interest, like mean time to failure and warranty periods, for which analytical solutions might not be feasible.

TABLE 1. FAILURES RATES FOR THE CASCADED SYSTEM

Component	Failure rate
$C_1$	0.002
$C_2$	0.001
$C_3$	0.003
$C_4$	0.004
$C_5$	0.002
$C_6$	0.001

Running the model for 40 iterations results in an estimate

<sup>1</sup> As we have said, in our framework, any parametric density function can be used for the times to failure of the system's components. Once these have been defined, the CPDs for the DFT constructs are automatically estimated using the dynamic discretisation algorithm. No closed-form for the system failure distribution is required. Therefore, we can easily estimate the failure distribution of the above system for any (non-exponential) time-to-failure of the input components.

for the system reliability of 0.979 at  $t=1000$  h. This is very close to the exact result, 0.974, obtained from the analytical expression for the reliability of the (exponential) cascaded PAND system (see Appendix).

### CONCLUDING REMARKS

We have presented an effective and flexible event-based BN framework for DFT-like reliability modelling, based on a new dynamic discretisation algorithm for approximate inference general hybrid Bayesian Networks (HBNs). The new approximate inference algorithm permits reliability analysis of a complex system, comprised of a variety of dynamic fault tree constructs, including functional dependency and Warm Standby gates, with general

parametric or empirical time-to-failure distributions, avoiding the burden (and inaccuracies) associated with uniform discretisation of continuous nodes, and without recourse to numerical integration techniques or simulation methods.

Our BN framework is mathematically sound and at the same time simple enough and sufficiently easy to use to allow the interaction with domain experts and decision makers. Sensitivity analysis, uncertainty, diagnosis, common cause failures, and warranty analysis can also be easily performed within this framework.

A free evaluation version of the AgenaRisk software is available for researchers wishing to replicate our results [1].

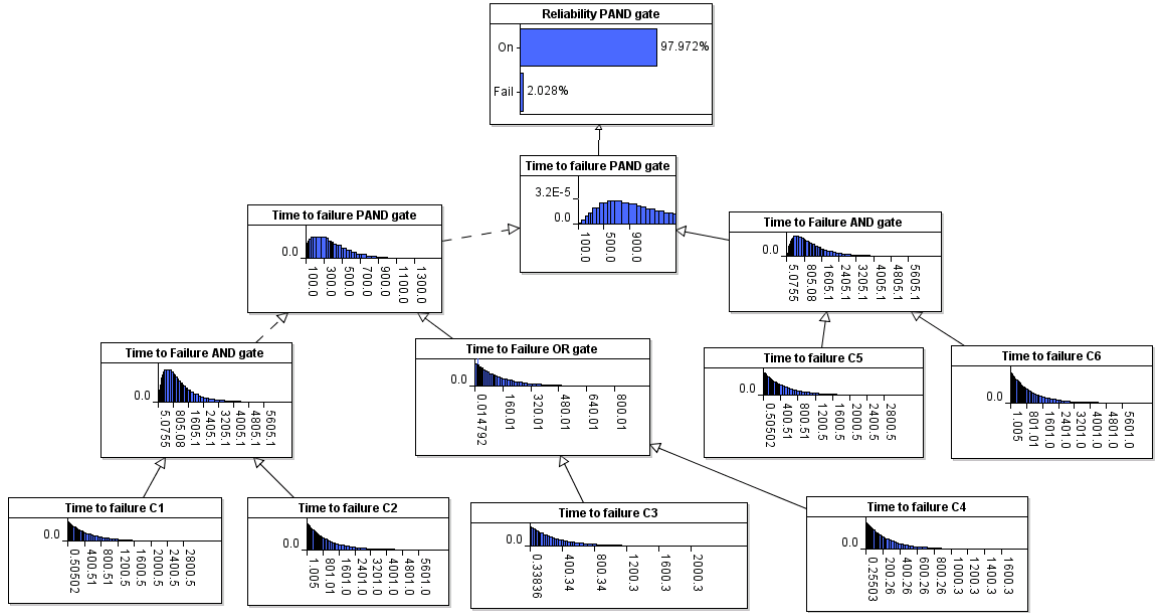


Figure 2. HBN model for PAND cascaded system with superimposed posterior marginal TTF and reliability distributions

### III. APPENDIX

The failure distribution of a PAND systems composed of one PAND subsystem and one AND subsystem, is given by

$$\begin{aligned} P(\tau_{PAND} \leq t) &= P(\tau_{PAND2} \leq \tau_{AND} \leq t) \\ &= \int_0^t f_{\tau_{AND}}(u) \left[ \int_0^u f_{\tau_{PAND2}}(y) dy \right] du \\ &= \int_0^t f_{\tau_{AND}}(u) F_{\tau_{PAND2}}(u) du \end{aligned}$$

The failure density  $f_{\tau_{AND}}(u)$  for the AND subsystem, composed of two input components,  $C_i$ ,  $i=5,6$ , is  $f_{\tau_{AND}}(u) = f_{C_5}(u) \times F_{C_6}(u) + F_{C_5}(u) \times f_{C_6}(u)$ . The second

PAND2 gate is in turn composed by a AND and a OR subsystem. The failure distribution is given by

$$F_{\tau_{PAND2}}(t) = \int_0^t f_{\tau_{OR}}(u) F_{\tau_{AND}}(u) du$$

where

$$\begin{aligned} F_{\tau_{AND}}(u) &= F_{C_1}(u) \times F_{C_2}(u) \\ f_{\tau_{OR}}(u) &= f_{C_3}(u) R_{C_4}(u) + R_{C_3}(u) f_{C_4}(u) \end{aligned}$$

Assuming that all components have an exponentially distributed failure distribution, we obtain, after some rudimentary algebra:

$$P(\tau_{PAND} \leq t) =$$

$$\int_0^t (\lambda_5 e^{-\lambda_5 u} + \lambda_6 e^{-\lambda_6 u} - (\lambda_5 + \lambda_6) e^{-(\lambda_5 + \lambda_6)u}) F_{PAND2}(u) du$$

where

$$F_{PAND2}(u) =$$

$$(\lambda_3 + \lambda_4) \left[ \frac{1 - e^{-(\lambda_3 + \lambda_4)u}}{\lambda_3 + \lambda_4} - \frac{1 - e^{-(\lambda_1 + \lambda_3 + \lambda_4)u}}{\lambda_1 + \lambda_3 + \lambda_4} - \frac{1 - e^{-(\lambda_2 + \lambda_3 + \lambda_4)u}}{\lambda_2 + \lambda_3 + \lambda_4} + \frac{1 - e^{-(\lambda_1 + \lambda_2 + \lambda_3 + \lambda_4)u}}{\lambda_1 + \lambda_2 + \lambda_3 + \lambda_4} \right]$$

## REFERENCES

- [1] Agena Ltd. 2007. AgenaRisk software package, www.AgenaRisk.com.
- [2] Amari, S., Dill, G., Howald, E. (2003). A new approach to solve dynamic fault trees. Reliab Maintainability Symp., 374-379.
- [3] Bobbio A, Portinale L, Minichino M, and Ciancamerla E. (2001). Improving the analysis of dependable systems by mapping fault trees into Bayesian networks. Reliability Engineering and System Safety 71 (3), 249-260.
- [4] Boudali H, Dugan J. B. (2005). A discrete-time Bayesian network reliability modeling and analysis framework. Reliability Engineering and System Safety, vol. 87, no. 3, 337-349.
- [5] Dugan J.B., Bavuso S. J. and Boyd M.A. (1992). Dynamic Fault Tree models for Fault Tolerant Computer Systems, IEEE Trans. Reliability, vol 41, 363-377.
- [6] Fenton NE, Krause P, Neil M. (2002). Probabilistic Modelling for Software Quality Control, Journal of Applied Non-Classical Logics 12(2), 173-188.
- [7] Fenton N, Littlewood B, Neil M, Strigini L, Sutcliffe A, Wright D. (1998). Assessing Dependability of Safety Critical Systems using Diverse Evidence, IEE Proceedings Software Engineering, 145(1), 35-39.
- [8] Fenton N and Neil M. (1999). 'A Critique of Software Defect Prediction Models, IEEE Transactions on Software Engineering, 25(5), 675-689.
- [9] Ghahramani, Z. (1997). Learning dynamic Bayesian networks. In C. Giles and M. Gori (Eds.), Adaptive Processing of Sequences and Data Structures, Lecture Notes in Artificial Intelligence, pp. 168-197. Berlin: Springer Verlag.
- [10] Gulati R. and Dugan J. B. (1997). A modular approach for analyzing static and dynamic fault trees, in Proc. Ann. Reliability and Maintainability Symp., 57-63.
- [11] Jensen F. (2001). Bayesian Networks and Decision Graph, Springer.
- [12] Jensen F, Lauritzen S.L, Olesen K. (1990). Bayesian updating in recursive graphical models by local computations. Computational Statistics Quarterly, 4, 260-282.
- [13] Kozlov A.V, Koller D. (1997). Nonuniform dynamic discretization in hybrid networks, in D. Geiger and P.P. Shenoy (eds.), Uncertainty in Artificial Intelligence, 13, pp. 314-325.
- [14] Kullback, S., and Leibler, R. A., 1951, On information and sufficiency, *Annals of Mathematical Statistics* 22: 79-86.
- [15] Langseth, H., and Portinale, L. (2006). Bayesian networks in reliability, Reliability Engineering and System Safety. (Also available as Technical Report No. TR-INF-2005-04-01-UNIPMN, Dept. of Computer Science, University of Eastern Piedmont "Amedeo Avogadro", Alessandria, Italy).
- [16] Lauritzen S.L. (1996). Graphical Models, Oxford.
- [17] Lauritzen S.L., and Spiegelhalter D.J. (1998). Local Computations with Probabilities on Graphical Structures and their Application to Expert Systems (with discussion), Journal of the Royal Statistical Society Series B, Vol. 50, No 2, 157-224.
- [18] Marquez D, Neil M, and Fenton N. (2007). Improved reliability modelling using Bayesian networks and dynamic discretisation, (submitted to IEEE Trans. Reliability).
- [19] Montani S, Portinale L, Bobbio A. (2005). Dynamic Bayesian networks for modeling advanced fault tree features in dependability analysis, In: Proceedings of the sixteenth European conference on safety and reliability. Leiden, pp. 1415-22. The Netherlands: A.A. Balkema.
- [20] Murphy K. (2002). Dynamic Bayesian Networks: Representation, Inference and Learning, PhD thesis, Dept. Computer Science, UC Berkeley.
- [21] Neil M, Fenton N, Forey S, Harris R. (2002). Using Bayesian Belief Networks to Predict the Reliability of Military Vehicles, IEE Computing and Control Engineering, 12(1), 11-20.
- [22] Neil M., Krause P., and Fenton N. (2003). Software Quality Prediction Using Bayesian Networks in Software Engineering with Computational Intelligence, edited by Khoshgoftaar T. M. The Kluwer International Series in Engineering and Computer Science, Volume 73.
- [23] Neil M, Tailor M, Marquez D. Inference in Bayesian Networks using Dynamic Discretisation, Statistics and Computing Journal, Springer Netherlands, Vol. 17, Number 3, September 2007.
- [24] Neil M, Tailor M, Marquez D, Fenton N. and Hearty P. (2007) Modelling Dependable Systems using Hybrid Bayesian Networks, Reliability Engineering and System Safety, (to appear).
- [25] Pearl, J. (1993). Graphical models, causality, and intervention, Statistical Science, Vol. 8, no. 3, 266-273.
- [26] Portinale L, Bobbio A. (1999). Bayesian networks for dependability analysis: an application to digital control reliability, In: Proceedings of the fifteenth conference on uncertainty in artificial intelligence. San Francisco, CA: Morgan Kaufmann Publishers.
- [27] Schneeweiss W. G. (1999). The Fault Tree Method. LiLoLe Verlag.
- [28] Solano-Soto J, Sucar, LE. (2001). A methodology for reliable system design, In: Proceedings of the 4th international conference on industrial and engineering applications of artificial intelligence and expert systems, Lecture notes in artificial intelligence, vol. 2070, pp. 734-45. Berlin, Germany: Springer.
- [29] Torres-Toledano JG, Sucar LE. (1998). Bayesian networks for reliability analysis of complex systems, In: Proceedings of the 6th Ibero-American conference on AI (IBERAMIA 98), Lecture notes in artificial intelligence, vol. 1484, pp. 195-206. Berlin, Germany: Springer.
- [30] Watson H. A. and Bell Telephone Laboratories. (1961). Launch Control Safety Study, Bell Telephone Laboratories, Murray Hill, NJ USA.
- [31] Weber P, Jouffe L. (2003). Reliability modeling with Dynamic Bayesian Networks, 5th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes, Washington, D.C., USA. J.B. Dugan, S. J. Bavuso and M.A. Boyd, "Dynamic Fault Tree models for Fault Tolerant Computer Systems", IEEE Trans. Reliability, vol 41, 1992, pp. 363-377.