Available online at www.sciencedirect.com
ScienceDirect
Reliability Engineering and System Safety 93 (2008) 933–939
RELIABILITY ENGINEERING & SYSTEM SAFETY
www.elsevier.com/locate/ress

# Modelling dependable systems using hybrid Bayesian networks

Martin Neil[a,b,*], Manesh Tailor[b], David Marquez[a], Norman Fenton[a,b], Peter Hearty[a]

[a]*Department of Computer Science, Queen Mary, University of London, UK*
[b]*Agena Ltd., UK*

Available online 15 March 2007

**Abstract**

A hybrid Bayesian network (BN) is one that incorporates both discrete and continuous nodes. In our extensive applications of BNs for system dependability assessment, the models are invariably hybrid and the need for efficient and accurate computation is paramount. We apply a new iterative algorithm that efficiently combines dynamic discretisation with robust propagation algorithms on junction tree structures to perform inference in hybrid BNs. We illustrate its use in the field of dependability with two example of reliability estimation. Firstly we estimate the reliability of a simple single system and next we implement a hierarchical Bayesian model. In the hierarchical model we compute the reliability of two unknown subsystems from data collected on historically similar subsystems and then input the result into a reliability block model to compute system level reliability. We conclude that dynamic discretisation can be used as an alternative to analytical or Monte Carlo methods with high precision and can be applied to a wide range of dependability problems.


*Keywords:* Bayesian networks; Bayesian software; Systems dependability; Dynamic discretisation

## 1. Introduction

We have used Bayesian nets (BNs) in a range of real-world applications of system dependability assessment (see for example [1–3]). In such applications, it is inevitable that there will be a mixture of discrete and continuous nodes (the resulting BNs are called hybrid). The traditional approach to handling (non-Gaussian) continuous nodes is static: you have to discretise them using some pre-defined range and intervals. However, this approach is unacceptable for critical type systems where there is a demand for reasonable accuracy. To overcome this problem 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. The approach is influenced by the work of Kozlov and Koller [4] using entropy error as the basis for approximation. We differ from their approach by integrating an iterative approximation scheme within existing BN software architectures, such as in junction tree (JT) propagation [5]. Thus, rather than support separate data structures and a new propagation algorithm we use the data structures commonly used in JT algorithms.

The power and flexibility of the approach is demonstrated by applying it to estimate the reliability of repairable systems represented by a Bayesian hierarchical model. This problem represents a very simplified version of fragments of the wide range of models we have implemented as part of commercial and research projects. These have been in areas as diverse as data fusion, parameter learning, discrete systems simulation, RAM (reliability, availability and maintainability) evaluation and software defect prediction. The modelling has been made possible because our dynamic discretisation algorithm has recently been implemented in the commercial general-purpose BN software tool AgenaRisk [6].

## 2. Background

BNs have been widely used to represent full probability models in a compact and intuitive way. In the BN framework the independence structure in a joint distribution is characterised by a directed acyclic graph, with nodes representing random variables (which can be discrete or continuous, and may or may not be observable), and

*Corresponding author. Department of Computer Science, Queen Mary, University of London, UK.
*E-mail address:* martin@dcs.qmul.ac.uk (M. Neil).

0951-8320/$ - see front matter © 2007 Elsevier Ltd. All rights reserved.
doi:10.1016/j.ress.2007.03.009

directed arcs representing causal or influential relationship between variables [7]. The conditional independence assertions about the variables, represented by the lack of arcs, reduce significantly the complexity of inference and allow the underlying joint probability distribution to be decomposed as a product of *local conditional probability distributions* (*CPD*) associated with each node and its respective parents. 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. Since a BN encodes all relevant qualitative and quantitative information contained in a full probability model, it is an excellent tool for many types of probabilistic inference where we need to compute the posterior probability distribution of some variables of interest (unknown parameters and unobserved data) conditioned on some other variables that have been observed.

A range of robust and efficient propagation algorithms has been developed for exact inference on BNs with discrete variables [5,8,9]. The common feature of these algorithms is that the exact computation of posterior marginals is performed through a series of local computations over a secondary structure, a tree of clusters, enabling calculation of the marginal without computing the joint distribution. See also [10].

The present generation of BN software tools attempt to model continuous nodes by numerical approximations using static discretisation as implemented in a number of software tools [11,12]. Although discretisation allows approximate inference in a hybrid BN without limitations on relationships among continuous and discrete variables, current software implementations require users to define a uniform discretisation of the states of any numeric node (whether it is continuous or discrete) as a sequence of predefined intervals, which remain *static* throughout all subsequent stages of Bayesian inference regardless of any new conditioning evidence. 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 you do not necessarily know in advance where the posterior marginal distribution will lie on the continuum for all nodes and which ranges require the finer intervals. It follows that where a model contains numerical nodes having a potentially large range, results are necessarily only crude approximations.

Alternatives to discretisation have been suggested by Moral et al. [13] and Cobb and Shenoy [14], who describe potential approximations using mixtures of truncated exponential (MTE) distributions, Koller et al. [15] who combine MTE approximations with direct sampling (Monte Carlo) methods, and Murphy [16] who uses variational methods. There have also been some attempts for approximate inference on hybrid BNs using Markov Chain Monte Carlo (MCMC) approaches [17]. However, constructing dependent samples that mixed well (i.e., that move rapidly throughout the support of the target posterior distribution) remains a complex task.

## 3. Dynamic discretisation

Let $X$ be a continuous random node in the BN. The range of $X$ is denoted by $\Omega_X$, and the probability density function (PDF) of $X$, with support $\Omega_X$, is denoted by $f_X$. The idea of discretisation is to approximate $f_X$ as follows:

1. partition $\Omega_X$ into a set of interval $\Psi_X = \{w_j\}$, and
2. define a locally constant function $\tilde{f}_X$ on the partitioning intervals.

Discretisation operates in much the same way when $X$ takes integer values but in this paper we will focus on the case where $X$ is continuous.

As Kozlov and Koller [4], we use an upper bound of the Kullback–Leibler (KL) metric between two density functions $f$ and $g$:

$$D(f\|g) = \int_S f(x)\log\frac{f(x)}{g(x)}\,\mathrm{d}x$$

as an estimate of the relative entropy error induced by the discretised function. Under the KL metric, the optimal value for the discretised function $\tilde{f}$ is given by the mean of the function $f$ in each of the intervals of the discretised domain. The main task reduces then to finding an optimal discretisation set $\Psi_X = \{\omega_j\}$.

Our approach to dynamic discretisation searches $\Omega_X$ for the most accurate specification of the high-density regions given the model and the evidence, calculating a sequence of discretisation intervals in $\Omega_X$ iteratively. At each stage in the iterative process, a candidate discretisation, $\Psi_X$, is tested to determine whether the relative entropy error of the resulting discretised probability density $\tilde{f}_X$ is below a given threshold, defined according to the trade off between the acceptable degree of precision and computation time.

By dynamically discretising the model we achieve more accuracy in the regions that matter and incur less storage space over static discretisations. Moreover, we can adjust the discretisation any time in response to new evidence to achieve greater accuracy. In outline, dynamic discretisation follows these steps:

1. Convert the BN to a JT and choose an initial discretisation for all continuous variables.
2. Calculate the NPT of each node given the current discretisation.
3. Enter evidence and perform global propagation on the JT, using standard JT algorithms [5].
4. Query the BN to get posterior marginals for each node, compute the approximate relative entropy error, and check if it satisfies the convergence criteria.
5. If not, create a new discretisation for the node by splitting those intervals with highest entropy error.