Managing Risk in the Modern World

Applications of Bayesian Networks

A Knowledge Transfer Report from the London Mathematical Society and the Knowledge Transfer Network for Industrial Mathematics

By Norman Fenton and Martin Neil

MANAGING RISK IN THE MODERN WORLD
Applications of Bayesian Networks

By Norman Fenton and Martin Neil

Contents

November 2007

A Knowledge Transfer Report from the London Mathematical Society and
the Knowledge Transfer Network for Industrial Mathematics

London Mathematical Society
De Morgan House, 57/58 Russell Square
London WC1B 4HS

AUTHORS

Norman E Fenton is Professor of Computing at Queen Mary, University of London, and is also Chief Executive Officer of Agena, a company that specialises in risk management for critical systems. Norman is an Affiliated Professor to the University of Haifa, Israel. At Queen Mary he is the Director of Research in the Department of Computer Science and the Head of the Risk Assessment and Decision Analysis Research Group (RADAR). Norman has published 5 books and over 80 refereed articles on various topics in software engineering and risk analysis; his book *Software Metrics: A Practical and Rigorous Approach* has sold over 35,000 copies worldwide. Norman's current focus is on causal models (Bayesian Networks) for risk assessment in a wide range of application domains such as vehicle reliability, embedded software, transport systems, legal reasoning, TV personalisation and financial services. Norman has acted as an expert witness on a range of risk assessment issues in major legal cases. Norman is a Chartered Engineer, a Chartered Mathematician and is a Fellow of the British Computer Society.

Martin Neil is a Reader in Systems Risk at the Department of Computer Science, Queen Mary, University of London, where he teaches decision and risk analysis and software engineering. Martin is CTO of Agena, who develop and distribute AgenaRisk, a software product for modelling risk and uncertainty. Martin has over 18 years' experience in academic research, teaching, consulting, systems development and project management and has published or presented over 40 papers in refereed journals and at major conferences. His interests cover Bayesian modelling and risk quantification in diverse areas: operational risk in finance, systems and design reliability (including software), software project risk, decision support, simulation (using dynamic discretisation as an alternative to Monte Carlo), cost-benefit analysis, AI and personalisation, and statistical learning. Martin has consulted to Motorola, Philips, NATS, QinetiQ, Advantica, Dstl, ABSA, Ericsson and others, either providing advanced risk modelling expertise or systems deployment and integration using AgenaRisk. Before setting up Agena and joining academia Martin previously held senior positions with JP Morgan and Lloyds Register in the areas of software project governance and safety critical systems evaluation respectively. Martin earned a BSc in Mathematics, a PhD in Statistics and Software Metrics and is a Chartered Engineer.

Executive Summary

Businesses and governments must often assess and manage risk in areas where there is little or no direct historical data to draw upon, or where relevant data is difficult to identify. For example, the Barings Bank collapse in 1995 was not due to credit or market risk, where banks have sufficient data for prediction and mitigation of risk, but rather it was due to what is now called operational risk – the results of failures in everyday operational processes. The challenges are similarly acute when the source of the risk is novel: terrorist attacks, ecological disasters, major project failures, and more general failures of novel systems, market-places and business models.

Even though we may have little or no historical data, there is often an abundance of *expert* (but subjective) judgement, as well as diverse information and data on indirectly related risks.

These are the types of situation that can be successfully addressed using Bayesian Networks (BNs), even when classical, data-driven approaches to risk assessment are not possible. BNs describe "webs" of causes and effects, using a graphical framework that provides for the rigorous quantification of risks and the clear communication of results. They can combine historical data with expert judgement.

During the last decade, researchers have incorporated BN techniques into easy-to-use toolsets, which in turn have enabled the development of decision support systems in a diverse set of application domains, including medical diagnosis, safety assessment, forensics, procurement, equipment fault diagnosis and software quality. Further technology and tool advancements since 2000 mean that end-users, rather than just researchers, are now able to develop and deploy their own BN-based solutions. As a result, BN methods are beginning to penetrate mainstream business practice. Recent commercial case studies provide evidence of impressive returns on investment from these techniques.

Both the practice and research of BNs are mushrooming. This report provides a snapshot of this dynamic and exciting area, including an introduction to the underpinning ideas, recent case studies, emerging areas of application, current research challenges, and a summary of the key players.

Bayesian Networks: Background and Development

Basic definitions

A Bayesian Network (BN) is a way of describing the relationships between causes and effects, and is made up of nodes and arcs, as shown in Figure 1. The collection of nodes and arcs is referred to as the *graph* or *topology* of the BN. In addition, in a BN each node has an associated probability table, called the Node Probability Table (NPT).

The nodes represent variables. To ensure the example is as simple as possible we assume, in Figure 1, that all the nodes are discrete, having the two possible states "true" and "false" (in real-world examples a node such as "Norman late" could be continuous on a scale of zero to infinity representing the number of minutes late). The arcs in a BN represent causal or influential relationships between variables. Thus, a train strike can cause and/or influence both Martin and Norman to be late; Martin oversleeping can also cause or influence Martin to be late.

The key feature of BNs is that they enable us to model and reason about uncertainty. In our example, a train strike does not imply that Norman will definitely be late (he might leave early and drive), but there is an increased probability that he will be late. This information is captured in the NPT for the node "Norman late" as shown in Figure 2. The NPT for any node gives the *conditional probability* of each possible outcome given each combination of outcomes for its parent nodes. In this case the node "Norman late" has one parent node, "Train strike".

What the NPT tells us is that:

- The probability Norman is late given that there is a train strike is 0.8;
- The probability Norman is not late given that there is a train strike is 0.2;
- The probability Norman is late given that there is not a train strike is 0.1;
- The probability Norman is not late given that there is not a train strike is 0.9.

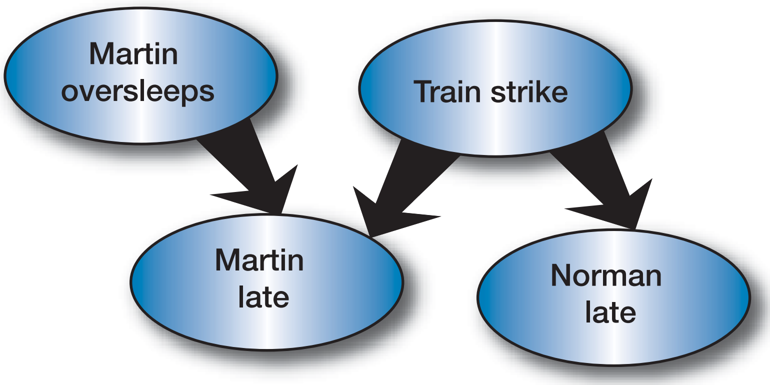


Figure 1: Simple Bayesian Network (BN)

	Train Strike	False	True
Norman Late	False	0.9	0.2
	True	0.1	0.8

Figure 2: NPT for "Norman Late"

The NPT for “Martin late”, as given by Figure 3, is more complicated since this node has two parents and so the number of combinations of parent states is four rather than two.

On the other hand, the NPTs for the nodes “Train strike” and “Martin oversleeps” (Figure 4) are both very simple; since these nodes have no parent nodes in this model (we call such nodes *root nodes*), we only have to assign a probability to each of the two possible values “true” and “false”.

Usually, there are several ways of determining the probabilities in any of the tables. For example, for the NPT for “Train strike” we might be able to base the probabilities on previously observed frequencies of days when there were train strikes. Alternatively, if no such statistical data is available we may have to rely on subjective probabilities entered by experts. A key feature of BNs is that we are able to accommodate both subjective probabilities and probabilities based on objective data.

Calculations in a BN

Having entered the probabilities we can now use Bayesian probability to do various types of analysis. Bayesian probability is all about revising probabilities in the light of actual observations of events. Suppose, for example, we find out that Norman is late. Then, intuitively, we feel that the probability of a train strike must have increased from its prior value of 0.1. But by how much? Bayes’ Theorem provides the answer as shown in Box 1. In this case the revised belief (called the *posterior*) is just under 0.5. Box 1 also shows how we use this information to revise our belief that Martin is late; in this case the prior probability of 0.446 increases to 0.542 once we know Norman is late. In practice (as we shall see later) there is no need for anybody to be concerned with the laborious calculations such as those in Box 1, since these are all done automatically in any BN tool.

The advantage of BNs over alternative techniques

When we enter evidence and use it to update the probabilities in this way we call it *propagation*. In theory we can enter any number of observations anywhere in the BN and use propagation to update the marginal probabilities of all the unobserved variables. This can yield some exceptionally powerful analyses that are simply not possible using other types of reasoning and classical statistical analysis methods. For example, without showing the computational steps involved, if we first enter the observation that Martin is late we get the revised probabilities shown in Figure 5.

	Martin oversleeps	False		True	
	Train Strike	False	True	False	True
Martin Late	False	0.7	0.4	0.4	0.2
	True	0.3	0.6	0.6	0.8

Figure 3: NPT for “Martin late”

False	0.9
True	0.1

NPT for “Train strike”

False	0.6
True	0.4

NPT for “Martin oversleeps”

Figure 4: NPTs for the root nodes

Box 1: Bayes' Theorem and Propagation

Suppose that T represents the statement "Train strike", and N represents "Norman late". We start with a prior probability of T, which we write as $P(T)$, but we are interested in knowing what is the (posterior) probability of T given the evidence N. We write this as $P(T|N)$. Bayes' Theorem is the following formula, due to the 18th century mathematician, the Reverend Thomas Bayes, for calculating $P(T|N)$:

$$P(T|N) = \frac{P(N|T)P(T)}{P(N)} \quad .$$

Now, we know from the NPTs that $P(N|T) = 0.8$ and that $P(T) = 0.1$. So the numerator in Bayes' Theorem is 0.08. The denominator, $P(N)$, is the so-called marginal (or unconditional) probability that Norman is late – it is the probability that Norman is late when we do not know any specific information about the events (in this case the train strike) that influence it. The NPTs do not provide this value directly, but they do provide it indirectly by virtue of the equation

$$\begin{aligned} P(N) &= P(N|T)P(T) + P(N|\text{not } T)P(\text{not } T) \\ &= 0.8(0.1) + 0.1(0.9) \\ &= 0.17. \end{aligned}$$

Hence, substituting this value of $P(N)$ in Bayes' Theorem we get $P(T|N) = 0.08/0.17 = 0.471$.
Thus, the observation that Norman is late significantly increases the probability that there is a train strike (up from 0.1 to 0.471). What we would like to do is to use this information to revise our belief that Martin is late. This is where doing things manually starts to get a little arduous even for such a simple example as this. "Martin late" is conditioned on two events rather than just one. Introducing O to represent "Martin oversleeps", the original marginal probability of Martin being late is

$$\begin{aligned} P(M) = {} & P(M|T, O)P(T)P(O) \\ & + P(M|T, \text{not } O)P(T)P(\text{not } O) \\ & + P(M|\text{not } T, O)P(\text{not } T)P(O) \\ & + P(M|\text{not } T, \text{not } O)P(\text{not } T)P(\text{not } O) \\ = {} & 0.032 + 0.036 + 0.216 + 0.162 \\ = {} & 0.446. \end{aligned}$$

But we know the revised marginal $P(T) = 0.471$ and hence $P(\text{not } T) = 0.529$. Using these values in the equation above yields the revised marginal $P(M) = 0.542$. Thus, the observation that Norman is late has also increased the probability that Martin is late.

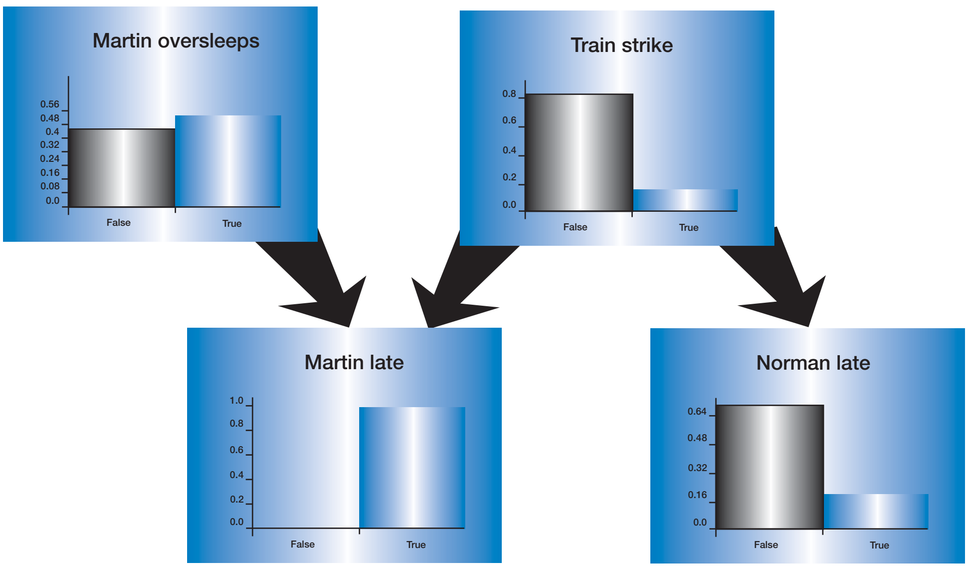


Figure 5: Revised probabilities for "Martin late"

The most likely explanation for Martin's lateness is Martin oversleeping – the revised probability of a Train strike is still low. However, if we now discover that Norman is also late (Figure 6) then "Train strike" (rather than Martin oversleeping) becomes the most likely explanation for Martin being late. This particular type of (backward) inference is called *explaining away*.

Classical statistics alone does not enable this type of reasoning and "what-if" analysis.

In fact, as even this simple example shows, BNs offer the following benefits:

- *Explicitly model causal factors:* It is important to understand that this key benefit is in stark contrast to classical statistics whereby prediction models are normally developed by purely data-driven approaches. For example, regression models are a standard method, in which historical data alone are used to produce equations relating dependent and independent variables. Such approaches not only fail to incorporate expert judgement in scenarios where there is insufficient data, but also fail to accommodate causal explanations. In a study [18] to predict software defects found in testing, the regression-based model was unable to predict the logical certainty that no testing would yield no defects. This is because there was insufficient data for such scenarios. But a causal model can predict this with perfect accuracy. Similarly, regression models cannot accommodate the impact of future process changes. In short, regression models are often good for describing the past, but poor for predicting the future.

- *Reason from effect to cause and vice versa:* A BN will update the probability distributions for every unknown variable whenever an observation is entered into any node. So entering an observation in an "effect" node will result in back propagation, i.e. revised probability distributions for the "cause" nodes and vice versa. Such backward reasoning of uncertainty is not possible in other approaches.

- *Overturn previous beliefs in the light of new evidence:* The notion of explaining away evidence is one example of this.

- *Make predictions with incomplete data:* There is no need to enter observations about all the "inputs", as is expected in most traditional modelling techniques. The model produces revised probability distributions for all the unknown variables when any new observations (as few or as many as you have) are entered. If no observation is entered then the model simply assumes the prior distribution.

- *Combine diverse types of evidence including both subjective beliefs and objective data.* A BN is "agnostic" about the type of data in any variable and about the way the NPTs are defined.

- *Arrive at decisions based on visible auditable reasoning:* Unlike black-box modelling techniques (including classical regression models and neural networks) there are no "hidden" variables and the inference mechanism is based on a long-established theorem (Bayes).

This range of benefits, together with the explicit quantification of uncertainty and ability to communicate arguments easily and effectively, makes BNs a powerful solution for all types of risk assessment. As we shall see in the next section, the availability of excellent tool support for BNs also makes them a practical solution compared with alternatives such as fuzzy logic [53] and Shafer-Dempster Theory [46].

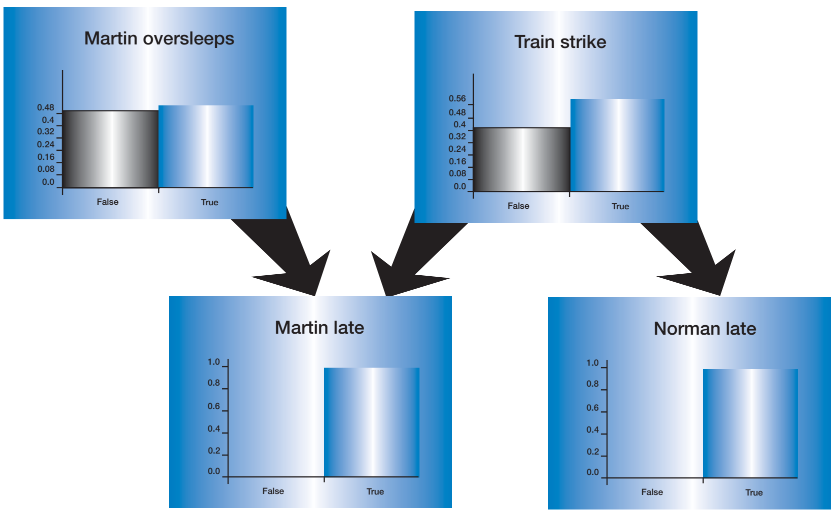


Figure 6: "Norman late" explains away Martin being late

Box 2: Dynamic BNs

Consider the simple BN flood model in Figure 7. The model is attempting to predict the probability of a flood given information about rainfall, water level and flood defences.

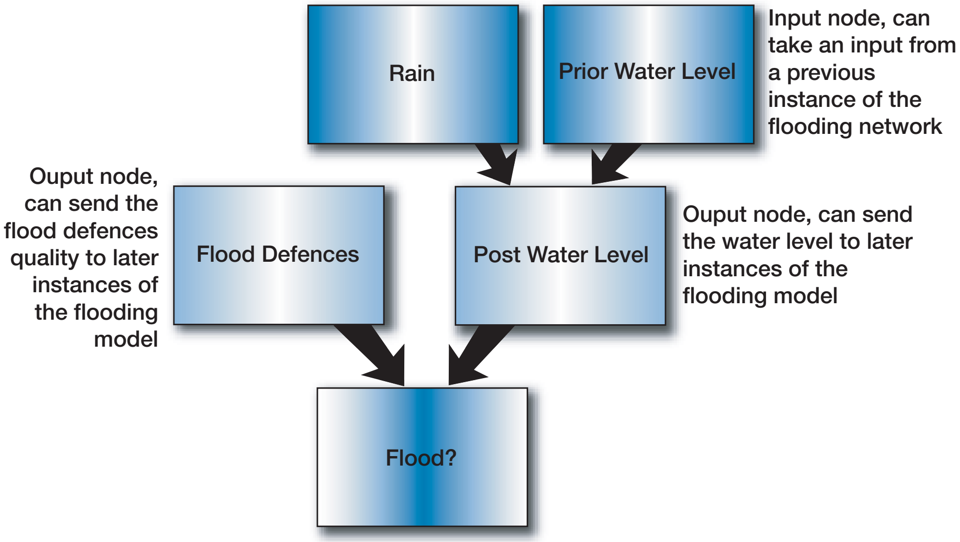


Figure 7: Simple flood model – a single iteration

The problem is that this model is a "one time" solution. Suppose, for example, we enter information about rainfall – this will give us a revised probability distribution for both "Flood" and "Post Water Level". This is very useful, but it is static. Suppose we are monitoring the situation at fixed time intervals. Then we want to be able to use the revised probability distribution for "Post Water Level" to replace the distribution for the "Prior Water Level" in the next time interval. In other words, we want to be able to use the same BN structure over and over again, but with the prior NPTs changed. This notion of "iteration" can only be implemented in (static) BNs by replicating the BN structure as many times as there are iterations and by linking appropriate input and output nodes. This is generally not practical for real-world problems because it leads to large, unmanageable and computationally inefficient models. Consequently, researchers have introduced the notion of Dynamic BNs (DBNs) [36] to extend BNs to take account of this type of temporal behaviour, and have developed algorithms that do the necessary propagation on the compact (rather than expanded) models. Work on DBNs is ongoing and some partial tool implementations are now available, such as in GeNie & SMILE and Netica. One approach uses the Object-Oriented BN solution (as in Figure 8 from AgenaRisk) whereby it is sufficient to link separate instances of the BN objects as opposed to copies of the full BN in each iteration. In this way there is only one object (with many instances of it) and the computational and storage overheads are therefore minimised.

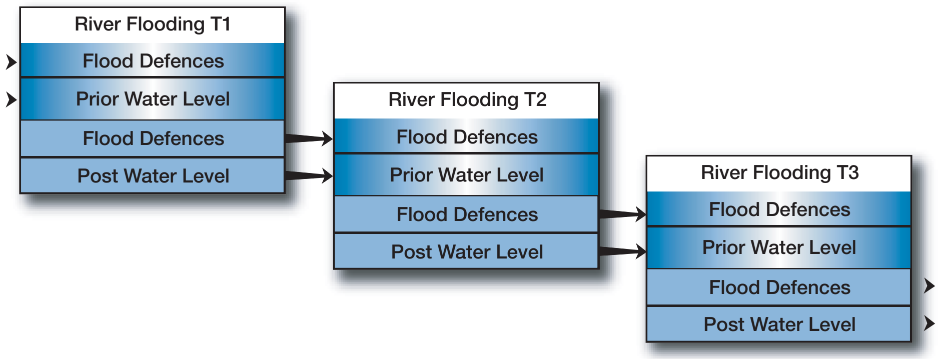


Figure 8: Flood model - multiple iterations

Key BN developments

Even in very simple BNs like the examples above the calculations involved in propagation are very time-consuming. When there are many variables and links between, as in most real-world models, and where the number of states for each variable is large this propagation becomes impossible to do manually. In fact, no computationally efficient solution for BN propagation is known that will work in all cases. This was the reason why, despite the known benefits of BNs over other techniques, there was for many years little appetite to use BNs to solve real-world decision and risk problems. However, a dramatic breakthrough in the late 1980s changed things. Researchers such as Lauritzen and Spiegelhalter [30] and Pearl [43] published algorithms that provided efficient propagation for a large class of BN models.

These algorithms were based on the idea of message passing in a tree structure, called the *junction tree*, that is derived from the BN. Each node of the junction tree corresponds to a group of nodes in the original BN called a *cluster*. Provided these cluster sizes are not too large the resulting algorithm works very well. Real-world BN models of many hundreds of nodes are more often than not naturally structured in such a way that the cluster sizes are small; hence these propagation algorithms have proven to work on many real-world problems. Since the first published algorithms there have been many refined versions, including that of Shenoy-Shafer [47], which uses the idea of binary fusion to ensure that the junction tree is binary. In most cases this leads to faster computation times.

The first commercial tool to implement an efficient propagation algorithm was developed in 1992 by Hugin, a Danish company closely associated with Jensen's BN research team at Aalborg University. The Hugin tool (see Appendix 2) had a liberating effect; it enabled researchers who were not specialists in BNs and statistics to build and run BN models. Other BN tools quickly followed, such as Netica, Microsoft's MSBNX, and BayesiaLab.

While these tools enabled large-scale BNs to be *executed* efficiently, they provided little or no support for users actually to build large-scale BNs, nor to interact with them easily. Beyond a graphical interface for building the topology, BN-builders were left to struggle with the problems of handling large graphs that contained similar, but slightly different "patterns" of structure and of filling in the probabilities in many very large NPTs manually. In both cases, this could prove to be exceptionally time-consuming, error-prone and ultimately demoralising for the domain experts, in conjunction with whom the models were built and upon whom their accuracy depended. Consider, for example, the BN fragment shown in Figure 9. Such fragments are very typical of those that frequently occur in real-world models. They are characterised by the fact that node values are typically measurable only on a subjective ranked scale like {very low, low, medium, high, very high} and only extremely limited statistical data (if any) is available to inform the probabilistic relationship for Y given X1 and X2.

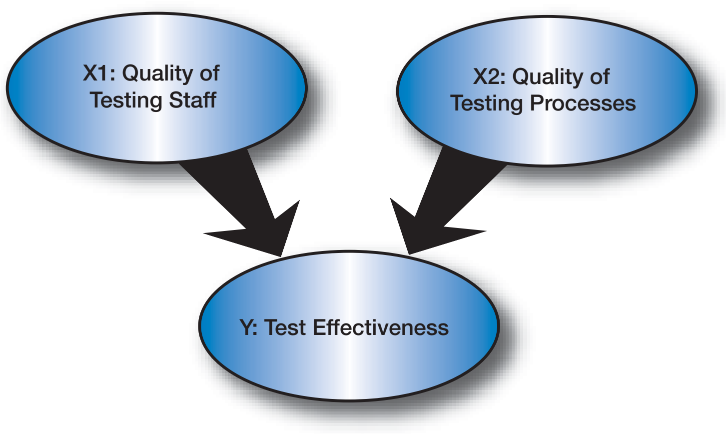


Figure 9: Typical qualitative BN fragment

Assuming each of the nodes has five states (in the many commercial studies we have been involved with, the experts are rarely satisfied with 3-point scales), the NPT for the node Y has 125 states. This is not an impossible number to elicit exhaustively, but from extensive experience we know that all kinds of inconsistencies arise when experts attempt to do so. If the number of states increases to seven (which experts commonly insist on) and/or the node Y has additional parents then exhaustive elicitation becomes infeasible, especially as real-world models invariably involve dozens of fragments like these.

On top of these constraints, none of the early BN tools was properly able to handle continuous, as opposed to discrete, node variables. Hence, until the late 1990s few commercial applications of BNs had been built outside research labs.

The key developments that have significantly improved the situation in recent years fall into the following broad categories:

- *Object-Oriented BNs:* A theoretical breakthrough in being able to scale-up BN-building came with Koller and Pfeffer's idea of object-oriented BNs [26]. Object-oriented BNs borrow ideas from object-oriented design and programming in that they allow complex problems to be described in terms of reusable abstract classes of objects with complex relationships. Large models can then be constructed, "building-block fashion", from smaller templates or fragments, with considerable productivity benefits. The SERENE project [13] further developed these ideas to the point whereby they could be deployed in practice (in the context of large-scale BNs for safety cases, where there were multiple occurrences of similar BN patterns). In particular, the techniques in [39] were subsequently incorporated into both the BN tools Hugin and AgenaRisk.

- *Efficient NPT elicitation:* The frustration of manually building large NPTs has been the biggest factor limiting more widespread use of BNs. Over the years a number of techniques have been introduced that enable large NPTs to be built with minimal effort for a range of special cases (see, for example, [9][28][50][51]). The most commonly used technique (especially in medical BNs) is the Noisy-OR method [22], but this has the disadvantage that it applies only to Boolean nodes and implicitly ignores the interaction effects between variables. What was still missing was a general, easily accessible approach that could be used directly with domain experts who are neither expert probability theorists nor mathematicians. A solution to the problem for a large class of NPTs involving *ranked nodes* such as in Figure 9 has recently been described in [17]. It enables the entire NPT to be defined in terms of simple weighted function expressions. This solution has been implemented in AgenaRisk (see Figure 10) and used with great effect in a number of real systems.

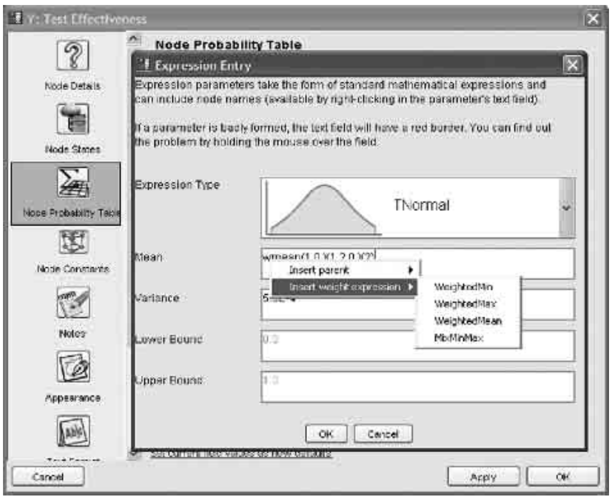


Figure 10: Defining an NPT for a ranked node using a single expression

- *Learning BNs from data* (see, for example [6][24][37]): Theoretically, with sufficient data about a set of potentially related variables it may be possible to *learn* both the qualitative (i.e. the graph structure) and the quantitative (i.e. the NPTs) components of a BN. Structure learning involves determining the conditional probability structure between variables from the strength of the interrelationships contained in the data set. Since there are many possible explanatory BN structures for any given data set, structural learning algorithms use scoring rules to penalise overly complex models using the principle of Occam's razor. At its simplest, structural learning looks at the correlation or mutual information shared between one or more variables and discards weak links between variables should the correlation fall below some acceptable threshold. When BNs are "learned" from data in this way they can be viewed as one of many machine learning AI techniques. However, this has also caused considerable confusion and underplays the major strength of BNs (there is little evidence that the structural relationships learned from data alone produce a model that makes sense from a causal perspective and structural learning cannot handle logical or deterministic relationships between variables). From this point of view, it is not surprising that many people mistakenly regard BNs as being in direct competition to Neural Networks. But whereas the latter always require significant data for learning accuracy, BNs can be built by experts with domain knowledge even when there is little or no data available – as is the case in many decision-making applications. In a comparison of models for predicting football results [25] the expert-built BN, which had been built without direct access to the database of results, significantly outperformed the BN built from data alone; but even more interesting is that it also outperformed other machine learning models.

  Notwithstanding these concerns, there is no doubt that, providing expert input is also used, learning BNs from data has great potential. One of the first commercial strength tools to implement a BN-learning algorithm was Bayesware Discoverer (Appendix 2), while Hugin and Netica have now added NPT learning components. A very powerful and completely free system is Powersoft, an implementation of Cheng's award-winning BN-learning algorithm [8]. Where there is extensive data these algorithms and tools can be very helpful in constructing NPTs and revising them automatically without the need for expert input as the database expands.

- *Dynamic BNs* (see Box 2): Whereas the basic BN is static, in many real-world problems we want to model the change in values of uncertain variables over successive time intervals. Dynamic BNs extend classical BN algorithms to support such modelling and inference. As such they have much in common with signal processing applications, such as Kalman filters. They have been used for image tracking, forecasting financial exchange rates, and online/offline condition monitoring and fault diagnosis in systems control applications. An especially intriguing recent application is to mind reading [4][11], where they model observable head and facial displays and corresponding hidden mental states over time. For a comprehensive overview of dynamic BNs and a comparison of competing algorithms, see [36].

- *Hybrid BNs:* Hybrid BNs are BNs that contain both discrete and continuous (numeric) nodes. Early BN tools did not allow numeric nodes at all. Then, when numeric nodes were first introduced into tools like Hugin in 1997 there were no facilities for defining NPTs as arithmetic or standard statistical functions. Fortunately, most current BN tools now include some kind of equation editor and predefined statistical functions that can be used to define NPTs for numeric nodes. However, most BN tools are unable to deal with numeric nodes accurately and until very recently this has been a constraint on the types of problems that can be solved using BNs.

What Can be Done with Bayesian Networks?

The first working applications of BNs (during the period 1988-1995) tended to focus on classical diagnostic problems, primarily in medicine [21] and fault diagnosis [4]. Indeed, it was an EU-funded project on diagnosing neuromuscular diseases that led to the Machine Intelligence Group at Aalborg University producing the MUNIN system [2]; this subsequently led to the first BN tool, Hugin. Like most of the medical applications, the MUNIN BN was wrapped up in a decision support system for the intended use of medical professionals. It is difficult to gauge the extent to which these systems have actually been used in practice. The suspicion is that, even when the natural scepticism by doctors to any kind of AI techniques is overcome in individual instances, the highly conservative nature of the profession overall has prevented any widespread take-up. Nevertheless the medical and biological/DNA decision-support domain has continued to be the most fruitful area for published BN applications; there have been hundreds of such publications — the online bibliography cited in [15] provides details of many of them.

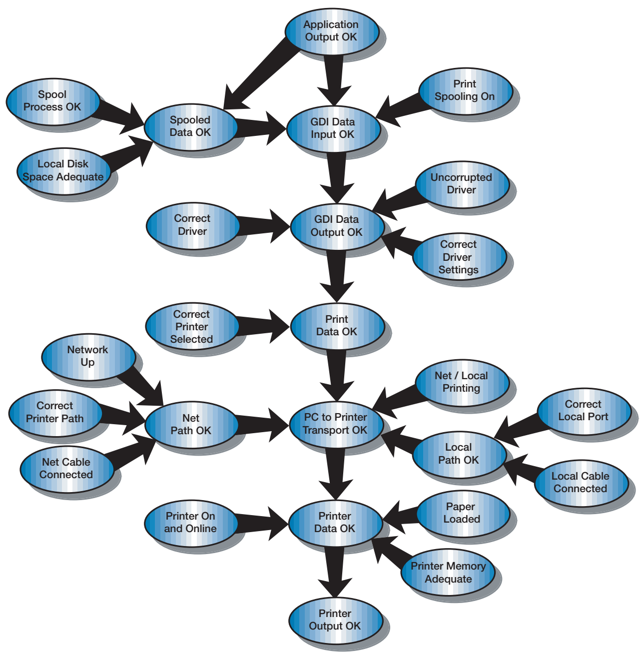
Whereas the take-up in practice of medical BNs has been limited, the take-up in other areas has been impressive. Companies such as Microsoft and Hewlett-Packard have used the early BNs for fault diagnosis, and in particular printer fault diagnosis, extensively. Box 3 describes this, and other uses of BNs, at Microsoft.

Box 3: Use of BNs at Microsoft

In 1996 Bill Gates famously declared

"Microsoft's competitive advantage is its expertise in Bayesian networks." (Los Angeles Times, October 28, 1996)

There is no doubt that Microsoft has invested heavily in such expertise over the years, by bringing together some of the leading researchers in BNs like Eric Horvitz and David Heckerman in the Decision Theory and Adaptive Systems Group. Crucially, in addition to some significant contributions to core research in BNs this group has ensured that its work has often been implemented in real systems. One such key area where Microsoft has used BNs is in user-support and automated fault diagnostics. For example, its printer fault diagnostic system was based on a BN developed by Breese and Heckerman [5] (although interestingly it was Hewlett-Packard, rather than Microsoft, who subsequently patented a similar BN-based system [48]). A fragment of a fault diagnosis BN model is shown in Figure 11.



One of the most celebrated, but also most widely misunderstood, BN applications at Microsoft was the Office "paperclip" that has annoyed millions of MS Word users worldwide. In fact, as Horvitz reports in The Economist [10], although a BN lies at the heart of the system for learning the most likely user actions, the Office team employed a relatively simple rule-based system on top of the BN to bring the paperclip agent to the foreground with a variety of tips. Horvitz states:

"We had been concerned upon hearing this plan that this system would be distracting to users — and hoped that future versions of the Office Assistant would employ our Bayesian approach to guiding speculative assistance actions — coupled with designs we had demonstrated for employing non-modal windows that do not require dismissal when they are not used."

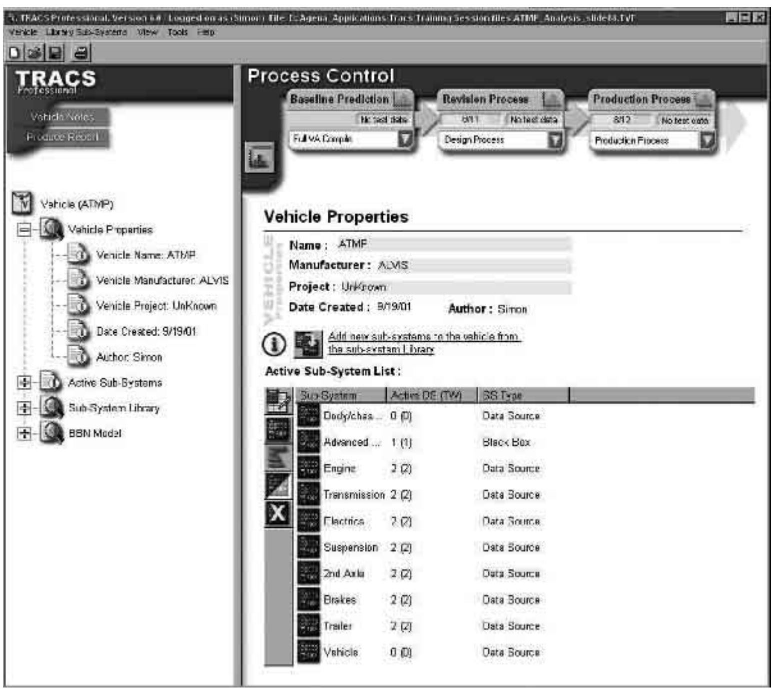
Figure 11: Simplification of Microsoft's printer fault diagnosis BN [5]

Unlike doctors, pharmaceutical companies are very keen to exploit BN models that potentially improve the efficiency and accuracy of their costly trials. In 2006 the FDA [12] announced its willingness to consider these kinds of Bayesian techniques in order both to speed up the approval process and increase overall safety. Researchers at Boston [45] have used data on both successful and unsuccessful drugs trials to develop a BN model that could reduce drug development costs by an estimated $283 million per approved drug and increase a drug's profitability by $160 million. The company Phorecaster (www.phorecaster.com) is exploiting these developments.

The ever-increasing need for improved decision support in critical systems — especially for assessing safety — has also resulted in a range of BN-based systems being used in practice. These include BN models for air traffic management [40], railway safety assessment [33], and terrorist threat assessment [29]. The VISTA system [20] helps users making high-stakes, time-critical decisions that involve complex visual information such as at NASA Mission Control Center in Houston. More generally, BNs have become a fairly standard means of modelling and tracking in vision applications [52]. There have also been numerous uses of BNs in military applications. For example, the TRACS system for predicting reliability of land vehicles (see Box 4) is one of many BN-based systems used routinely by QinetiQ.

Box 4: The TRACS System (Predicting Military Vehicle Reliability)

Determining which vehicle to choose, from competing tenders, to meet a new Ministry of Defence (MoD) requirement is difficult, time consuming and error prone. Traditional solutions used by QinetiQ involved a combination of extensive track-testing of prototypes supplied by competing manufacturers, and modelling based on analysing design specifications and using “sum-of-parts” reliability predictions. This process was hugely expensive and generally led to unsatisfactory predictions because there was no means of combining subjective judgements about likely design and manufacturing process quality. In collaboration with Agena [38], QinetiQ developed a BN-based model to predict vehicle reliability accurately based on information about the architecture and design process. The model was generated dynamically from a number of BN template models based on the particular subsystem architecture of a given vehicle specification. For each vehicle, any known information about the subsystem reliability was used. This was combined with information about the particular manufacturer and their design and manufacturing processes. Agena developed a decision-support system around the model, which enables QinetiQ’s engineers to generate solutions for each new proposed vehicle and enables them to interact with the BN model via a simple questionnaire GUI. For each proposed vehicle, various types of reliability analyses can be generated, along with all supporting reports, providing a full audit trail of assumptions.



The system, TRACS (Figure 12), saves QinetiQ time and money because accurate predictions are achieved without the need for track testing. Predictions made by TRACS are more accurate because they can combine subjective data about the process and manufacturer with hard data about component reliability. Moreover, TRACS helps identify process improvement opportunities and hence can lead to improved reliability and reduced whole life costs.

Many of the user interface ideas that were built into TRACS were subsequently incorporated into the AgenaRisk software; in particular the questionnaire interface enables non-programmers to generate instantly complete applications based around a BN, whilst exposing as many of the details as needed.

Figure 12: The TRACS BN-based system for predicting reliability of military vehicles

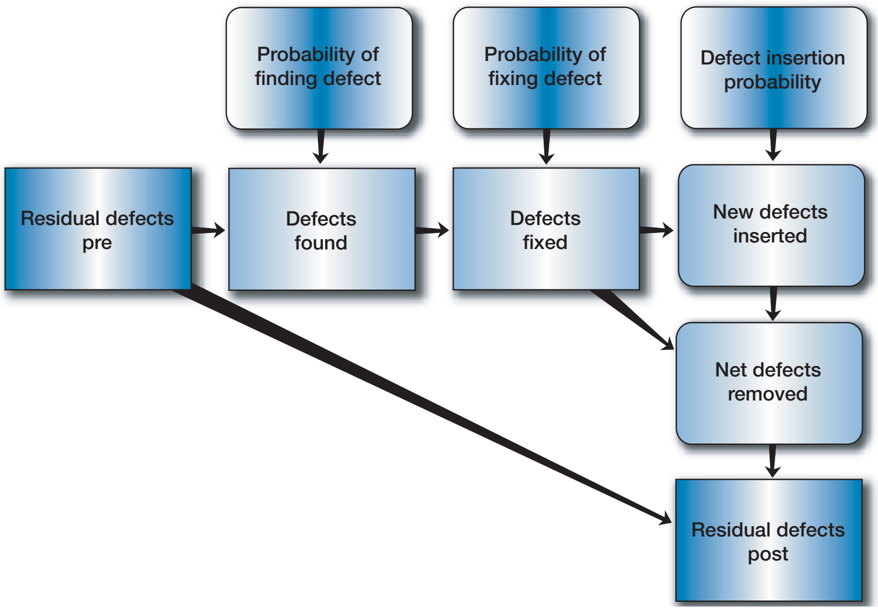
High-stakes applications are not limited to the safety-critical and military domains. The Basel 2 Banking Accord [3] places a regulatory requirement on all banks to provide an auditable quantification of operational risk. Previously they had only to provide quantification of credit and market risk, for which there are well-established data and models. In the absence of such data for operational risk the regulators proposed using models such as BNs. Moreover, the incentive to use such models was a lower capital charge. A number of such BN models have been developed (see, for example, [42][44]) and BN-based operational risk solutions are known to have been implemented at a number of major international banks. The special challenges involved in building these kinds of solutions are discussed further in the next section.

Another high-stakes application domain where BNs have been used extensively by commercial organisations is fault prediction. Box 5 describes some of the extensive work on software fault prediction. During 2005-2007 a leading technology company, known around the world for innovation and leadership in wireless and broadband communications, implemented an innovative BN-based solution to the problem of predicting hardware component failures in the field. With the implementation of the resulting quality control and reliability prediction system in its network and infrastructure division, this company expects to achieve savings of over $5 million in 2007 alone, including a 30 per cent improvement in warranty expenditure [1].

Box 5: BNs Provide Radical Improvements for Software Fault Prediction

The developers of any new complex software system will confirm that, no matter how much testing they perform, there will still be plenty of defects yet to be found. The hope is that, when the software is released, any defects found by end-users will have minimal impact. Hence, the decision about when to stop testing and release the software must always be balanced by the likely number (and criticality) of remaining defects. It follows that the ability to produce accurate predictions of "residual" defects in software systems is one of the most important and challenging tasks confronting software engineers. It is especially relevant for safety critical software (such as in transport and medical systems where software that is released with too many defects can have life-threatening impact); but the business of any commercial software producer can be devastated if they get their release decision wrong.

Since 1999 Philips Consumer Electronics has worked with researchers at Queen Mary, Surrey University, and Agena to evolve and validate BN models. Philips develop complex software that is embedded in electronic devices like TVs and DVDs. Being able to improve its decision-making about when to release the software is critical from a business perspective because faulty software can lead to the recall of entire batches of equipment. Figure 13 shows some BN models used to predict testing process quality and software product quality.



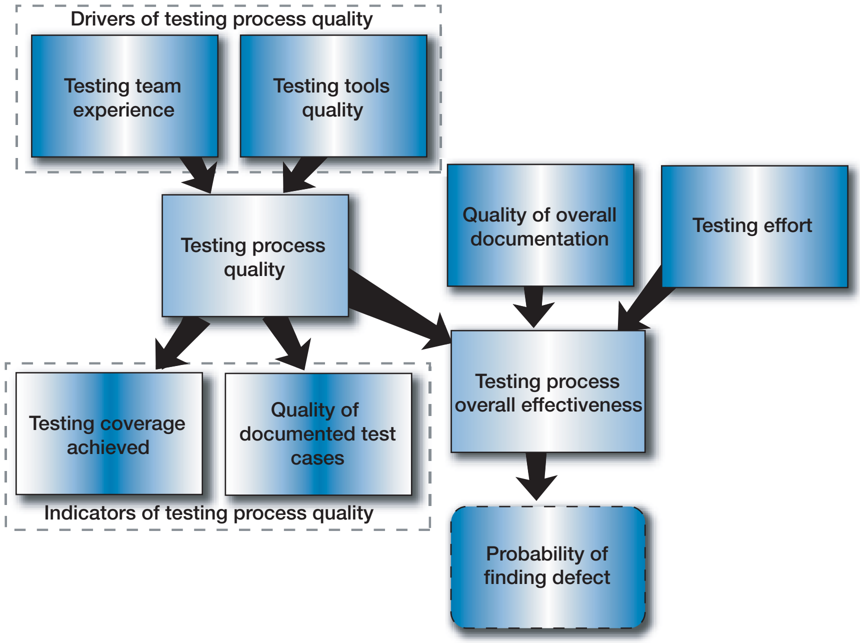


Figure 13: BNs used to predict testing process quality and product quality

The AID tool (described in [16]) was developed specifically for the Philips development environment and produced impressive results. In a subsequent EU-funded project MODIST [34], involving a number of companies in addition to Philips, a more general purpose BN solution was developed. This can be easily tailored to handle arbitrary software development processes, as shown in Figure 14 and hence the approach can be used by organisations whose development processes are very different from those of Philips.

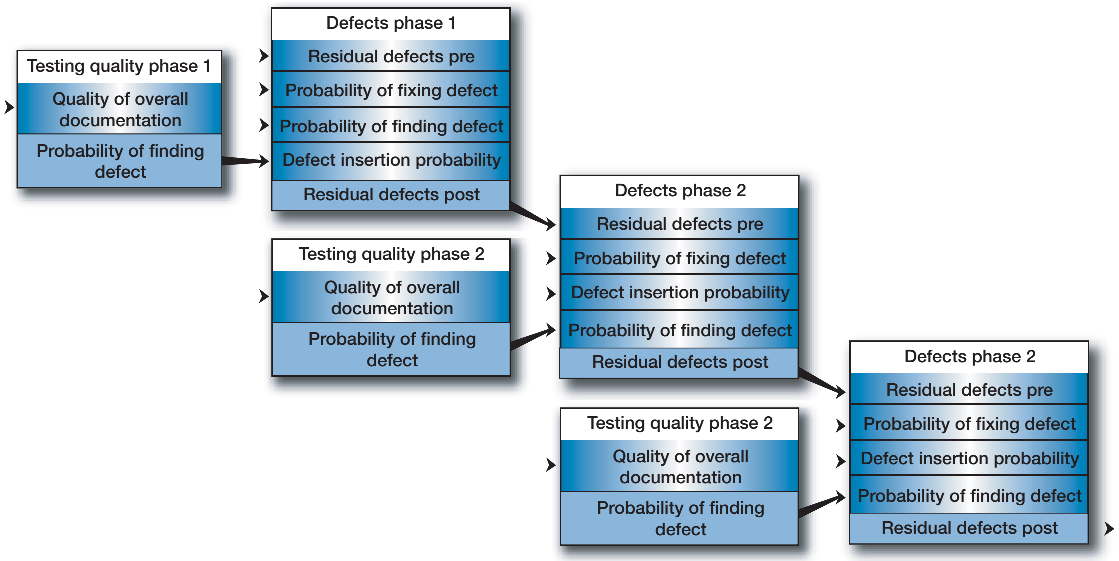


Figure 14: Arbitrary life-cycle stages modelled as a series of connected BNs

In 2004-2005 Philips were able to perform a comprehensive validation on 36 major projects based in Bangalore and Eindhoven [18]. The predictive accuracy of the causal models was outstanding — 93% — much better than could be achieved with traditional metric-based approaches. But even more important than the predictive accuracy was that the causal models and tools enabled project managers to do genuine risk assessment and "what-if" analysis that simply was not possible before.

Organisations including General Dynamics, Orange, Motorola, Siemens and Tellabs also now use these models and tools.

The number of applications of BNs has been increasing year-on-year and will continue to explode given the improvements in tool technology and the introduction of courses into many universities worldwide. The online bibliography [15] shows that, in addition to the range of applications discussed above, BNs have been used in SPAM filtering, personalisation systems, legal reasoning, ecology, security and many other fields.

Current Challenges

Most academic research remains focused on algorithm refinements because the general problem of exact BN propagation seems computationally intractable. Some of this work is on general-purpose algorithms (i.e. improving on the existing algorithms of Pearl, Lauritzen and Shenoy-Shafer) and there are promising results here with efficiency improvements enabling the generation and use of larger models containing thousands of variables. However, much of this work is restricted to models containing discrete variables and so other researchers are focused on special classes of BNs (and extended notions of BNs) for which the general algorithms either do not work at all, or will never run efficiently.

The most important special class of BNs is the hybrid BNs, that include continuous as well as discrete variables. The inability to handle continuous node variables accurately has been the “Achilles heel” of BN tools and hence represents the most important challenge for BN researchers. Exact inference in BNs can only be achieved when every node (with the exception of Gaussian variables) is discrete. Hence, the traditional approach to handling (non-Gaussian) continuous nodes is static: you have to discretise such nodes using some predefined range and intervals. Suppose, for example, that your BN model includes a node representing the *number of faults* found in a system. Instead of just specifying that the node ranges from zero to infinity, you would have to specify in advance how to break up this infinite range into a manageable number of intervals. The more intervals you define, the more accuracy you can achieve, but at a heavy cost of computational complexity. This approach also assumes you can identify and appropriately discretise the high-density regions for each node in the model, and do so in advance of any inference taking place. This is cumbersome, error prone and highly inaccurate. See Box 6 for an example from a real-world problem, and an indication of how it is being solved.

Box 6: Handling Continuous Node Variables

In the study [18] software size was measured in KLOC (thousands of lines of code). Typically, the size of a software module was between 10 and 20 KLOC, but this was by no means consistent. A part of the original BN with statically discretised nodes is shown in Figure 15. In the model the NPT for the node “Defects found” is defined as a Binomial distribution with parameters p equal to the probability of finding a defect and n equal to the value of “Defects inserted”. The NPT for the node “Residual defects” is simply defined by the deterministic function “Defects inserted” minus “Defects found”.

As with any attempt at discretisation, there was a need to balance the number of states (accuracy) against computational speed. There was much discussion, agonising and continual refinement of the discretisations. While predictions were generally reasonable within the “expected” range there were wild inaccuracies for modules whose properties were not typical. The inaccuracies were inevitably due to discretisation effects. For example, the model cannot distinguish between modules whose sizes are in the range 50 to 100 KLOC, so a module of size 51 KLOC is treated identically to one of 99 KLOC, and if we observe say 1501 defects found then the model cannot distinguish such an observation from 1999 defects found.

Such inaccuracies, as well as the wasted effort over selecting and defining discretisation intervals, are avoided using the dynamic discretisation described in [41] and implemented in AgenaRisk. Any node that is to be treated as continuous is simply flagged in the model and the modeller only has to specify a range, such as 0 to 1 for the node “Prob finding defect” and zero to infinity for the node “Size (KLOC)”. The resulting dynamically discretised model is shown in Figure 16.

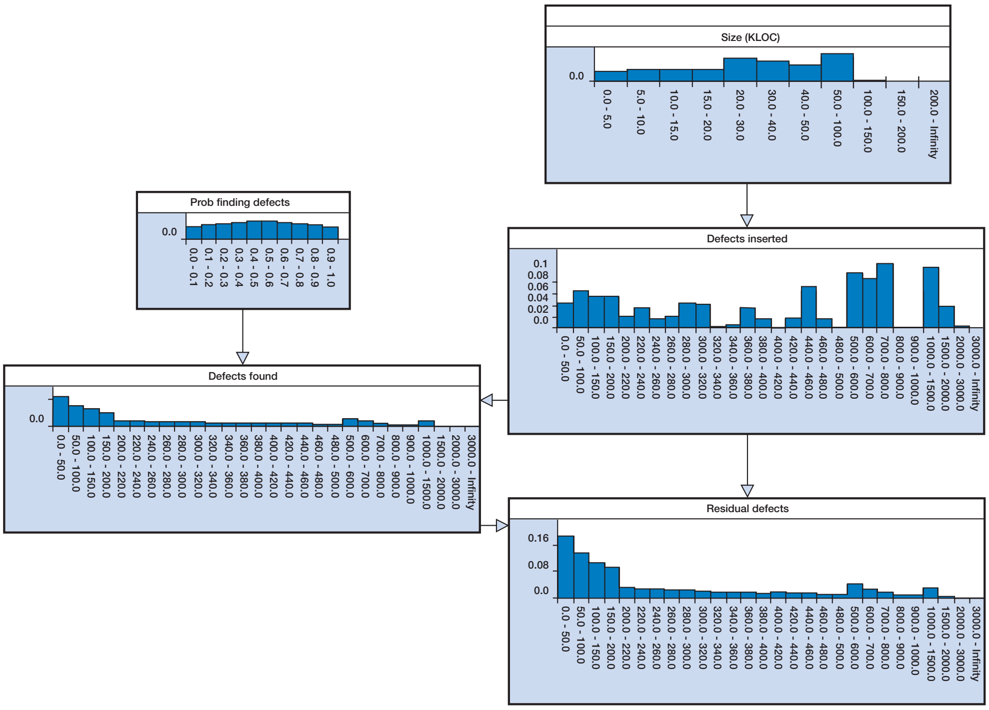


Figure 15: Statically discretised defects model with marginal distributions

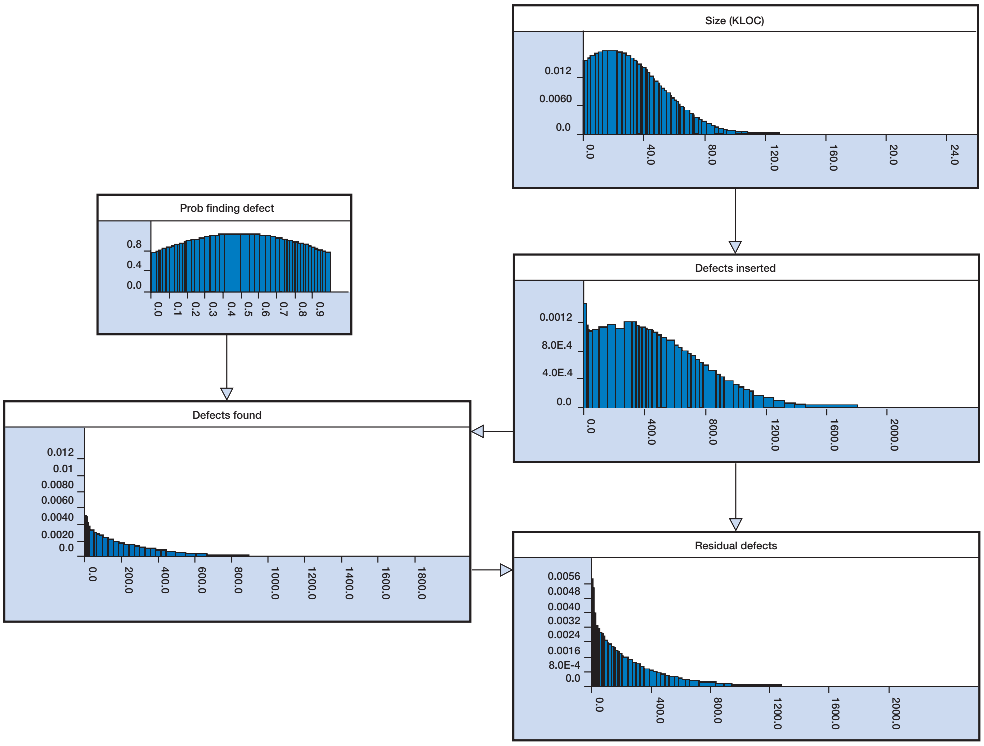


Figure 16: Dynamically discretised defects model with marginal distributions

This dynamic discretisation approach allows more accuracy in the regions that matter and incurs less storage requirement than static discretisations. In the AgenaRisk implementation of the algorithm the user can select the number of iterations and convergence criteria, and hence can go for arbitrarily high precision (at the expense of increased computation times).

It is because of this historical limitation that even Bayesian statisticians have shunned BNs for problems that involve continuous variables and complex stochastic models. Instead they have used tools like the WinBUGS software package [49], which are based on intensive sampling algorithms collectively known as *Markov Chain Monte Carlo* (MCMC) methods. These methods require drawing tens of thousands of dependent samples from, usually, high-dimensional probability distributions. To obtain reliable results these tools rely on expert knowledge to calibrate the tool according to the model and data set and to monitor the outputs to ensure the model converges to a stable solution.

Fortunately, there have been some recent breakthroughs in algorithms for hybrid BNs. Building on the work of Koslov and Koller [27], Neil et al. [41] have developed and implemented a dynamic discretisation algorithm which works efficiently for a large class of continuous distributions. Users of a tool such as AgenaRisk, which implements this algorithm, can simply define continuous nodes by their range and distribution (see also Box 5). Without any of the complexities associated with the MCMC approach, they can achieve results of matching or greater accuracy for many classes of model, especially for models that include discrete variables.

There are, however, a number of research challenges to be overcome before the hybrid BN computation problem is fully resolved. These include handling inference from large data sets, handling periodic functions, dealing with multivariate situations, discovering maxima in high resolution problems, handling autoregressive time series models, modelling high-dimensional dependencies (such as copulas), and, in common with any type of BN, handling nodes with multiple parents (see Box 7).

On a wider scale there is considerable research into how to model extremely large problems involving hundreds of data points, with many variables, over long periods of time, or involving complex sequences of variables and data, such as in biosequences. A number of extensions to BNs beyond the classical inference algorithms are being used for this purpose, including:

Relational BNs — These extend BNs by representing objects, their attributes, and their relations with other objects [23]. The standard approach for inference with a relational model is based on the generation of a propositional instance of the model in the form of a classical BN, and then applying standard inference algorithms. Relational BNs can be used as a means of deriving and generating BNs given the entity-attribute structure declared in a relational database system.

Statistical parameter learning — Statistical parameter learning is not often done using BNs because of the discretisation problem discussed above (although recent advances mean that some classes of parameter learning problem can be meaningfully solved using BNs). But Bayesian statistical analysis, using closed form or approximate solutions, such as MCMC, are increasingly popular and make use of the graphical component of a BN to explain the interrelationships between variables in the model. For an overview of statistical parameter learning using Bayesian methods see [19].

Sensitivity analysis — The growth in the size and complexity of BN models gives rise to the problem of how best to assess the sensitivity of a change one variable might have on another. This is especially important when verifying model output against expert expectations or empirical results. For example, when making an investment decision it might be of interest to assess the sensitivity of financial returns to a small number of crucial assumptions. Samlam (see Appendix 2) is a research tool that provides a variety of sensitivity metrics including MAP (Maximum *a posteriori*) and MPE (Most Probable Explanation) [7].

From the observations above it is clear that the power and flexibility of BNs are growing considerably. In tandem with this growth is their expansion into different application areas where they are either complementing existing techniques (and in some cases supplanting them) or providing ways of modelling complex problems that were previously thought impossible. Some example applications where this is happening include:

Safety and reliability modelling — Traditionally, the reliability and safety of complex transport, nuclear and aviation systems has been assured through the use of a number of approximate techniques including Fault Tree Analysis (FTA), Event Tree Analysis (ETA), Monte Carlo simulation and Markov modelling. Each of these methods uses different approximate algorithms to model different aspects of the problem. It turns out that in all cases these can either be replaced or subsumed by BN methods. For example, fault trees are used to model the probability of loss events from knowledge of equipment failures and their interactions, but increases in the size of the state space mean that approximate algebraic methods have to be used for estimation; fortunately BNs provide exact solutions to these problems. See [31][33] for details on how BNs have been used in event tree modelling and [32][35] for details on how BNs can be used to model fault trees.

Box 7: Handling Nodes with Multiple Parents

Despite the extensive and continued improvements to BN algorithms, academic BN researchers have tended to ignore pragmatic solutions to common problems faced by practitioners. An especially common problem involves nodes with multiple parents. BN algorithms are generally unable to handle nodes with more than 5 parents efficiently if those nodes have multiple states. But for continuous variables it ought to be possible to provide sensible solutions. Consider, for example, a model involving simple addition of several variables – say we have a node "cost" that is defined as the sum of the costs X1,..,X8 of eight components. Despite its conceptual simplicity, no BN algorithm (static or dynamic) can handle such a model directly. Instead, users are forced to introduce an artificial factorisation of the BN structure, such as shown in Figure 17.

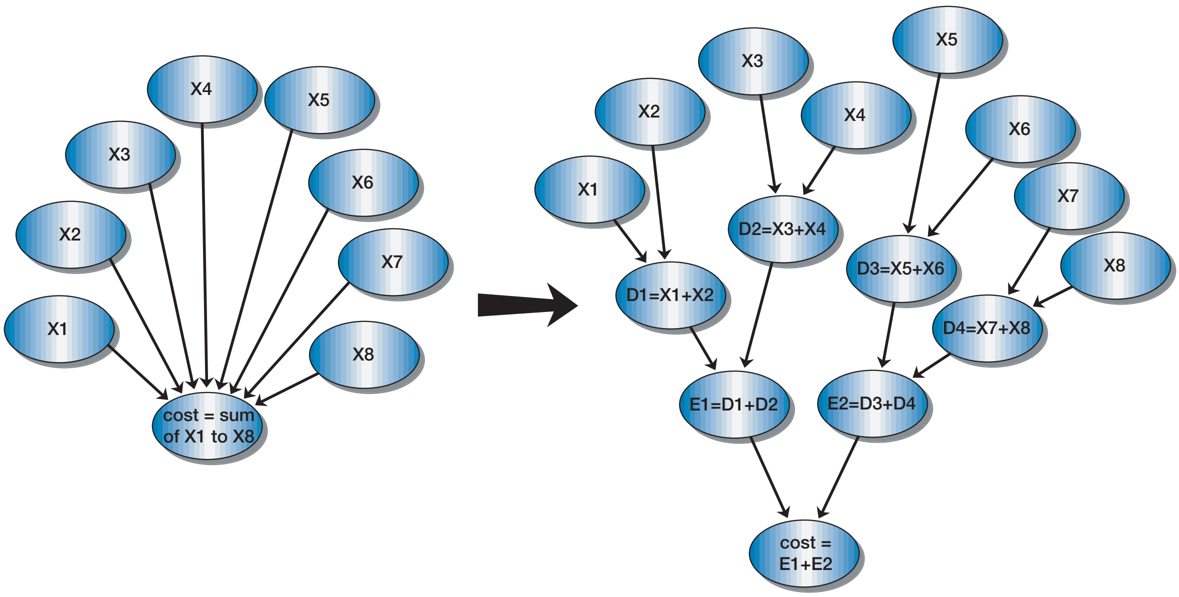


Figure 17: Binary factorisation

Clearly the need to do this for such conceptually simple problems is unacceptably laborious and error-prone. Key research challenges in this respect are therefore (1) to incorporate such optimal factorisations automatically; and (2) to handle deterministic functions within a BN more efficiently.

Operational risk in finance — As mentioned earlier, the Basel 2 regulations have forced banks to consider statistical loss distributions in financial operational risk scenarios with a focus on modelling unexpected loss events such as fraud or large scale IT failure. The major challenge banks face in such modelling is the lack of relevant data – incidents of major losses due to operational failures are relatively rare. Hence, the traditional data-driven approach to model-building is little use for well-managed banks; they will have suffered very few incidents and so have few data points from which to fit parametric loss models. BNs help overcome these hurdles because they combine qualitative data from experts, use information about the underlying processes followed in the organisation and use relevant quantitative data available in shared industry-wide databases. Crucially, given the role of expertise here, modern BN software tools can help organisations evaluate the sensitivity of the models to different expert assumptions and different sources of data in order to argue to the regulator that the resulting risk model is reasonable and sufficiently accurate [42].

Recommendation engines and information retrieval — Increasingly, information filtering and retrieval is used to collect information on user behaviour (such as when using the Web, Email, or TV) and to present relevant information back to the user, with the aim of providing a filtered list of relevant items of interest. For example a TV recommendation engine would use information about TV programmes viewed by a user, produce a model of the user's preferences, and then use this to recommend future TV programmes that the user might want to watch. Also, advertisers can target content to specific consumers directly. BNs and related Bayesian methods are increasingly used at the heart of these recommendation engines. Early recommendation engine technology (such as used in the TiVo system, www.tivo.com) relied heavily on a technique called collaborative filtering; this involves identifying groups of like-minded customers using a combination of Bayesian data mining and demographic analysis. New customers are then assigned to a group according to how well they fit the group's defining characteristics. Individual customer preferences can then be inferred and purchasing recommendations made. This approach to personalisation is indirect, but reasonably effective when it is possible to amass sufficient data to model an individual consumer accurately.

Crucially, modern approaches require little active involvement by the user. Instead BNs are used within the recommendation engine to learn user preferences statistically and provide accurate recommendations based on viewers' passive viewing or listening habits without the need to ask them to rate general classes of item. A typical BN-based architecture (which is the one encoded in [14]) is shown in Figure 18.

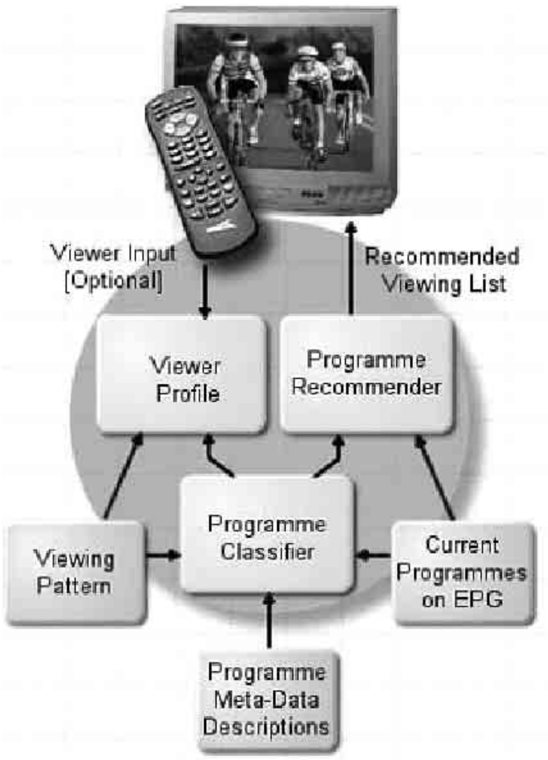


Figure 18: Architecture of typical BN-based recommendation system

Next Steps

Access tutorial material and case studies

For introductory and tutorial material there are a number of excellent web-based resources. We recommend the following be read in order:

- *An Intuitive Explanation of Bayesian Reasoning*
  Provides a comprehensive, but fun introduction to Bayesian probability, with interactive content.
  http://yudkowsky.net/bayes/bayes.html

- *Making sense of probability: Fallacies, Myths and Puzzles*
  Introduces some of the basics of Bayesian probability via a number of entertaining examples.
  www.dcs.qmw.ac.uk/~norman/papers/probability_puzzles/Making_sense_of_probability.html

- *RADAR Tutorial on Bayesian Networks*
  Provides a gentle introduction to BNs and probability theory, but also covers a number of more complex concepts beyond the scope of this report.
  www.dcs.qmul.ac.uk/~norman/BBNs/BBNs.htm

- *Kevin Murphy's Tutorial on Bayesian Networks*
  Provides some further material on more complex topics like Dynamic BNs, Hybrid BNs and learning BNs.
  www.cs.ubc.ca/~murphyk/Bayes/bnintro.html

Some BN companies have white papers, which both introduce the technology and explain its use in cases studies. We recommend www.agenarisk.com (start with the white paper entitled "Measuring Risk") and

www.cra.com (Charles River Analytics).

Build models

There is no better way to learn BNs than by actually building and running simple example models. The AgenaRisk tool (See Appendix 2) comes with a comprehensive set of example models and tutorials in categories ranging from introductory to advanced. A fully functioning (time-limited) version of the tool can be downloaded for free from www.agenarisk.com.

A number of models developed by the Machine Intelligence Group at Aalborg University are also available for download at http://oldwww.cs.aau.dk/research/MI/Misc/networks.html.
For an excellent free tool that learns BNs from data, while allowing users to specify their own links and bar others, we recommend Cheng's Powersoft tool (See Appendix 2).

Look at the bibliography of BN papers

To find out about BN applications of particular interest to you, look at the comprehensive bibliographic listing [15], available at www.agenarisk.com/resources/BN_refs.doc.

Gain an in-depth understanding of the algorithms

There is currently no simple introductory book on BNs (the BN books that are widely available require some reasonably in-depth understanding of probability and mathematics). For readers who seek a detailed mathematical treatment of BNs including propagation algorithms, the following books are recommended:

- Jensen, F., Bayesian Networks and Decision Graphs, Springer, 2001.
- Lauritzen, S. L., Graphical Models, Clarendon Press, Oxford, 1996.
- Neapolitan, R. E., Learning Bayesian Networks, Pearson Prentice Hall, 2004.
- Pearl, J., Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference, Morgan Kaufmann Publishers, Inc., San Francisco, California, 1988.

Attend a conference

There is no dedicated annual conference on BNs, because the range of application domains is so broad. However, there is one annual conference that has become the focus for most academic presentations on BNs, namely the Uncertainty in Artificial Intelligence (UAI) Conference. This conference and its associated workshops (which normally include one dedicated to BNs) is organised by the Association for Uncertainty in AI. Their website, www.auai.org, includes information on how to subscribe to their mailing list.

Appendix 1: Active Researchers and Practitioner Groups

In the UK

- **Agena:** They provide BN model-building consultancy and training.
  www.agenarisk.com
- **Durham University:** Michael Goldstein and David Wooff use BNs to support software testing.
- **ERA Technology, Safety Engineering Group (SEG):** They use BNs for safety and risk assessment and provide a consultancy service.
  www.era.co.uk/Services/networks.asp
- **Institute of Food Research:** Led by Gary Barker, they work on BN algorithms, but the main focus is using BNs for food safety assessment.
  www.ifr.ac.uk/science/Partnership/RCS/BBN.html
- **Leeds University:** Chris Needham and David Westhead (Institute of Molecular and Cellular Biology) work on BNs for protein function prediction and classification using uncertainty.
- **Manchester University, HCI Research Centre:** Led by Alistair Sutcliffe, this group uses BNs to predict errors in complex socio-technical systems.
- **Oxford University, Department of Statistics:** Steffen Lauritzen undertakes leading-edge research on BN algorithms and inference.
- **Queen Mary University of London, Risk and Decision Analysis Research Group (RADAR):** Norman Fenton, Martin Neil and William Marsh develop methods for building large-scale BNs, and improved algorithms for hybrid BNs and dynamic BNs. Main applications are in critical systems risk assessment.
  www.dcs.qmul.ac.uk/research/radar/
- **Queen's University Belfast, Centre for Statistical Science and Operational Research:** Led by Adele Marshall and Ronan Donaghy, they use BNs for a range of patient safety applications.
- **QinetiQ:** Numerous groups are working on BNs with applications such as capability support, reliability and availability, vehicle condition monitoring, data fusion and tracking.
  www.qinetiq.com
- **University College London, Department of Statistical Science:** Phil Dawid does fundamental work on BN theory and algorithms and is involved in BN applications to DNA profiling and forensic identification.
  www.ucl.ac.uk/~ucak06d/research.html
- **University of Edinburgh, School of Mathematics, Statistics Group:** Colin Aitken uses BNs for forensic science and legal reasoning.

Internationally

There are hundreds of research groups internationally doing work on BNs. Below is a selected list with special emphasis on application-oriented work.

- **Aalborg University, Machine Intelligence Group:** Led by Jensen, they do core research on BNs and applied work in autonomous agents (for example for computer games).
  http://oldwww.cs.aau.dk/research/MI/
- **Bayes, Stavanger:** Led by Andersen, they provide BN-based consultancy and models for the oil and gas industry and operational risk in banks, and work closely with the University of Stavanger.
  www.bayes.no
- **University of California, Berkeley, Centre for Intelligent Systems:** Led by Stuart Russell, they specialise in learning BNs and BNs for real-time decision-making.
  www.eecs.berkeley.edu/CIS/

- **Microsoft, Decision Theory and Adaptive Systems Group:** Key people are Horvitz and Heckerman, with a focus on learning models from data, BNs for diagnostics and troubleshooting, and intelligent user interfaces.
http://research.microsoft.com/dtas/
- **Charles River Analytics:** They build BN-based solutions for different application domains and offer model-building consultancy.
www.cra.com
- **NASA, Automated Learning Group:** Led by Cheeseman and Wolpert, they use BNs for a range of applications including data analysis.
http://ic.arc.nasa.gov/ic/projects/bayes-group/index.html
- **Seoul National University, Biointelligence Lab, Probabilistic Learning Research Group :** Main interests are in structural BNs and learning large-scale BNs from sparse data. Application areas: text mining and genomics/proteomics data analysis.
http://bi.snu.ac.kr/
- **Stanford University, Management Science and Engineering:** Ross Shachter works on applications of BNs and influence diagrams to medicine.
www.stanford.edu/dept/MSandE/people/faculty/shachter/research.html
- **UCLA, Automated Reasoning Group:** Led by Adnan Darwiche, they do work on efficient BN algorithms.
http://reasoning.cs.ucla.edu/
- **University of Alberta, Artificial Intelligence Research Group:** Greiner leads work on machine learning BNs. This is also the home of the Powersoft tool developed by Cheng.
www.cs.ualberta.ca/~greiner/
- **University of Helsinki, CoSCo (Complex Systems Computation Group):** Led by Myllymäki, they work on both theory and applications of BNs.
http://cosco.hiit.fi/
- **Université de Lausanne, Ecole des Sciences Criminelles:** Led by Franco Taroni, they specialise in BNs in forensic science.
www.unil.ch/esc
- **University of Massachusetts at Amherst, Multi-Agent Systems Lab:** Led by Victor Lesser, they use BNs to help with complex AI problem-solving.
http://dis.cs.umass.edu/
- **University of Pittsburgh, Decision Systems Laboratory:** Led by Druzdzel, this is the home of the GeNie & SMILE library of BN software.
http://dsl.sis.pitt.edu/
- **Universiteit Utrecht, Decision Support Systems:** Led by van der Gaag, they research building models with experts and efficient algorithms. Their key application area is medical decision problems in oncology with experts from the Netherlands Cancer Institute.
http://www.cs.uu.nl/groups/DSS/
- **Monash University, School of Computer Science and Software Engineering, Bayesian Artificial Intelligence:** Led by Korb and Nicholson, they specialise in learning BNs.
www.csse.monash.edu.au/bai/

Appendix 2: BN Tools

Commercial tools

- **Agenarisk:** From Agena (based in UK), www.agenarisk.com
- **BayesLab:** From Bayesia (based in France), www.bayesia.com
- **Bayesware Discoverer:** From Bayesware (based in UK), www.bayesware.com
- **BNet:** From Charles River Analytics (based in USA), www.cra.com/bnet
- **Hugin:** From Hugin A/S (based in Denmark), www.hugin.com
- **Netica:** From Norsys (based in Canada), www.norsys.com
- **SIAM & Causeway:** From SAIC (based in USA), www.inet.saic.com

Free and open source tools

- **GeNIe & SMILE:** http://genie.sis.pitt.edu
- **Microsoft MSBNx Bayesian Network Editor and Tool Kit:** http://research.microsoft.com/adapt/MSBNx/
- **OpenBayes:** www.openbayes.org
- **RISO:** http://sourceforge.net/projects/riso/
- **SamIam:** http://reasoning.cs.ucla.edu/samiam
- **Powersoft:** www.cs.ualberta.ca/~jcheng/bnsoft.htm

References

[1] Agena 2007, Press Release, http://www.agenarisk.com/agenarisk/case_13.shtml

[2] Andreassen S, Woldbye M, Falck B and Andersen SK, MUNIN: a causal probabilistic network for interpretation of electromyographic findings. Proceedings of the 10th International Joint Conference on Artificial Intelligence, Milan, Italy, 1987, pp. 366-372.

[3] Basel 2006, International Convergence of Capital Measurement and Capital Standards, Basel Committee on Banking Supervision, June 2006.

[4] BBC News, Computers set to read our minds, 26 June 2006, http://news.bbc.co.uk/1/hi/sci/tech/5116762.stm

[5] Breese JS and Heckerman D, Decision-theoretic troubleshooting: a framework for repair and experiment. Proceedings of the 12th Conference on Uncertainty in Artificial Intelligence, Portland, 1996, pp. 124-132, Morgan Kaufmann: San Francisco.

[6] Buntine W, A guide to the literature on learning probabilistic networks from data. IEEE Transactions on Knowledge and Data Engineering 8(2), 195-210 (1996).

[7] Chan H and Darwiche A, A distance measure for bounding probabilistic belief change. International Journal of Approximate Reasoning 38(2), 149-174 (2005).

[8] Cheng J, Greiner R, Kelly J, Bell D and Liu W, Learning Bayesian networks from data: an information-theory based approach. Artificial Intelligence 137(1-2), 43-90 (2002).

[9] Druzdzel MK and van der Gaag LC, Building probabilistic networks: where do the numbers come from? IEEE Transactions on Knowledge and Data Engineering 12(4), 481-486 (2000).

[10] The Economist, Son of paperclip. Print edition, 22 March 2001.

[11] el Kaliou RA, Mind-reading machines: automated inference of complex mental states. Technical Report No. 636, University of Cambridge Computer Laboratory, 2005, www.cl.cam.ac.uk/techreports/UCAM-CL-TR-636.pdf

[12] Federal Drugs Agency, Guidance for the use of Bayesian statistics in medical device clinical trials - draft guidance for industry and FDA staff. U.S. Department of Health and Human Services, Food and Drug Administration, April 2006, www.fda.gov/cdrh/osb/guidance/1601.html

[13] Fenton et al, The SERENE method manual (SafEty and Risk Evaluation using bayesian NEts), EC Project No. 22187 SERENE, SERENE/5.3/CSR/3053/R/1, 1999. Available at www.dcs.qmul.ac.uk/~norman/papers/serene.pdf

[14] Fenton NE and Neil M, Improved programme selection. International Patent Publication Number WO 03/090466 A2, World Intellectual Property Organisation International Bureau, 2003.

[15] Fenton NE, Bayesian Net references: a comprehensive listing, 2007, http://www.agenarisk.com/resources/BN_refs.doc

[16] Fenton NE, Krause P and Neil M, Software measurement: uncertainty and causal modelling. IEEE Software 10(4), 116-122 (2002).

[17] Fenton NE, Neil M, and Caballero JG, Using ranked nodes to model qualitative judgements in Bayesian Networks. IEEE Transactions on Knowledge and Data Engineering 19(10), 1420-1432 (2007).

[18] Fenton NE, Neil M, Hearty P, Marsh W, Marquez D, Krause P and Mishra R, Predicting software defects in varying development lifecycles using Bayesian Nets. Information & Software Technology 49, 32-43 (2007).

[19] Gelman A, Carlin JB, Stern HS, and Rubin DB. Bayesian Data Analysis (2nd Edition), Chapman and Hall, 2004, pp. 209–302.

[20] Horvitz E and Barry M, Display of information for time-critical decision making. Proceedings of the 11th Conference on Uncertainty in Artificial Intelligence, Montreal, 1995, pp. 296-305, Morgan Kaufmann: San Francisco.

[21] Horvitz E, Heckerman D, Ng K and Nathwani B, Heuristic abstraction in the decision-theoretic pathfinder system. Symposium on Computer Applications in Medical Care, Washington DC, IEEE Press: Silver Springs, MD, November 1989.

[22] Huang K and Henrion M, Efficient search-based inference for noisy-OR belief networks. Proceedings of the 12th Conference on Uncertainty in Artificial Intelligence, Portland, OR, 1996, pp. 325-331.

[23] Jaeger M, Complex probabilistic modeling with recursive relational Bayesian networks. Annals of Mathematics and Artificial Intelligence 32, 179–220 (2001).

[24] Jordan MI, Learning in Graphical Models (Adaptive Computation and Machine Learning), Morgan Kaufman, 1998.

[25] Joseph A, Fenton NE and Neil M, Predicting football results using Bayesian nets and other machine learning techniques. Knowledge Based Systems 19(7), 544-553 (2006).

[26] Koller D and Pfeffer A, Object-oriented Bayesian networks. Proceedings of the 13th Annual Conference on Uncertainty in Artificial Intelligence, Providence, Rhode Island, 1997, pp. 302-313.

[27] Kozlov AV, and Koller D, Nonuniform dynamic discretization in hybrid networks. In D Geiger and PP Shenoy (eds.), Uncertainty in Artificial Intelligence, 13: 314–325 (1997).

[28] Laskey KB and Mahoney SM, Network engineering for agile belief network models. IEEE Transactions on Knowledge and Data Engineering 12(4), 487-498 (2000).

[29] Laskey KB and Levitt TS, Multisource fusion for opportunistic detection and probabilistic assessment of homeland terrorist threats. Aerosense 2002.

[30] Lauritzen SL and Spiegelhalter DJ, Local computations with probabilities on graphical structures and their application to expert systems (with discussion). Journal of the Royal Statistical Society Series B 50(2), 157-224 (1988).

[31] Marquez D, Neil M and Fenton NE, A new Bayesian network approach to reliability modelling. Fifth International Mathematical Methods in Reliability Conference (MMR 07), Glasgow, July 2007.

[32] Marsh W and Bearfield G, Representing parameterised fault trees using Bayesian networks. In Proceedings of the 26th International Conference on Computer Safety, Reliability and Security, SAFECOMP 2007, Springer-Verlag, 2007.

[33] Marsh W and Bearfield G, Using Bayesian networks to model accident causation in the UK railway industry. International Conference on Probabilistic Safety Assessment and Management, PSAM7, Berlin, June 2004.

[34] MODIST, Models of Uncertainty and Risk for Distributed Software Development, Sept 2001 - Feb 2004, EC Framework 5 Project IST-2000-28749, www.modist.org.uk/

[35] Montani S, Portinale L and Bobbio A, Dynamic Bayesian networks for modeling advanced fault tree features in dependability analysis. Proceedings of the 16th European Conference on Safety and Reliability, Leiden, The Netherlands, A.A. Balkema, 2005, pp. 1415–1422.

[36] Murphy K, Dynamic Bayesian networks: representation, inference and learning. PhD thesis, Department of Computer Science, UC Berkeley, 2002.

[37] Neapolitan RE, Learning Bayesian Networks, Pearson Prentice Hall, 2004.

[38] Neil M, Fenton N, Forey S and Harris R, Using Bayesian belief networks to predict the reliability of military vehicles, IEE Computing and Control Engineering 12(1), 11-20 (2001).

[39] Neil M, Fenton NE, Nielsen L, Building large-scale Bayesian networks. The Knowledge Engineering Review 15(3), 257-284 (2000).

[40] Neil M, Malcolm B and Shaw R, Modeling an air traffic control environment using Bayesian belief networks. Twenty-first International System Safety Conference, August 4-8, Ottawa, Canada, 2003.

[41] Neil M, Tailor M and Marquez D, Inference in hybrid Bayesian networks using dynamic discretization. Statistics and Computing 17(3), September 2007.

[42] Neil M, Fenton N and Tailor M, Using Bayesian networks to model expected and unexpected operational losses. Risk Analysis Journal, August 2005.

[43] Pearl J, Fusion, propagation, and structuring in belief networks. Artificial Intelligence 29(3), 241-288 (1986).

[44] Ramamurthy S, Arora H and Ghosh A, Operational risk and probabilistic networks: an application to corporate actions processing. Infosys White Paper, 2005, www.infosys.com/industries/banking/white-papers/operational-risk-probabilistic-networks.pdf

[45] Schachter AD and Ramoni MF, 2007, http://marketplace.publicradio.org/shows/2007/02/01/PM200702013.html

[46] Shafer G, Perspectives on the theory and practice of belief functions. International Journal of Approximate Reasoning 4, 323-362 (1990).

[47] Shenoy P and Shafer G, Axioms for probability and belief-function propagation. Readings in Uncertain Reasoning, Morgan Kaufmann Publishers Inc, pp. 575–610, 1990.

[48] Skaanning C et al., Automated diagnosis of printer systems using Bayesian networks. US Patent 6535865, 2003, www.patentstorm.us/patents/6535865.html

[49] Spiegelhalter DJ, Thomas A, Best NG, and Gilks WR, BUGS: Bayesian inference using Gibbs sampling, Version 0.50. MRC Biostatistics Unit, Cambridge, 1995.

[50] van der Gaag LC, Renooija S, Witteman CLM, Aleman BMP and Taal BG, Probabilities for a probabilistic network: a case study in oesophageal cancer. Artificial Intelligence in Medicine 25(2), 123-148 (2002).

[51] Wellman MP, Fundamental concepts of qualitative probabilistic networks. Artificial Intelligence 44(3), 257-303 (1990).

[52] Xiang T and Gong S, On the structure of dynamic Bayesian networks for complex scene modelling. Proceedings of Joint IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance (VS-PETS), Nice, France, October 2003, pp. 17-22.

[53] Zadeh LA, Fuzzy algorithms. Information and Control 5, 94-102 (1968).

Managing Risk in the Modern World:

Applications of Bayesian Networks

A Knowledge Transfer Report from the London Mathematical Society and the Knowledge Transfer Network for Industrial Mathematics

by Norman Fenton and Martin Neil

The London Mathematical Society (LMS) is the UK's learned society for mathematics. Founded in 1865 for the promotion and extension of mathematical knowledge, the Society is concerned with all branches of mathematics and its applications. It is an independent and self-financing charity, with a membership of over 2600 drawn from all parts of the UK and overseas. Its principal activities are the organisation of meetings and conferences, the publication of periodicals and books, the provision of financial support for mathematical activities, and the contribution to public debates on issues related to mathematics research and education. It works collaboratively with other mathematical bodies worldwide. It is the UK adhering body to the International Mathematical Union and is a member of the Council for the Mathematical Sciences, which also comprises the Institute of Mathematics and its Applications and the Royal Statistical Society.

The Knowledge Transfer Network for Industrial Mathematics (KTN) brings together business, academia and government to boost innovation performance in the UK through the improved use of mathematics. It is managed by the Smith Institute for Industrial Mathematics and System Engineering, the UK's leading intermediate organisation in this area. Our vision is for companies to look increasingly towards mathematics as a key component in their innovation planning, and a powerful means for addressing challenges in design, operations, services and strategy. The KTN is a programme of the Government's Technology Strategy Board, and currently networks the activities of over 130 companies and 30 universities, providing them with a vehicle for developing new capability and exchanging knowledge and experience. It also provides Government with information to help shape science and technology strategy in the public sector.

The LMS-KTN Knowledge Transfer Reports are a new initiative, coordinated through the Smith Institute and the Computer Science Committee of the LMS. The reports are being produced as an occasional series, each one addressing an area where mathematics and computing have come together to provide significant new capability that is on the cusp of mainstream industrial uptake. They are written by senior researchers in each chosen area, for a mixed audience in business and government. The reports are designed to raise awareness among managers and decision-makers of new tools and techniques, in a format that allows them to assess rapidly the potential for exploitation in their own fields, alongside information about potential collaborators and suppliers.

The London Mathematical Society

Industrial Mathematics
Knowledge Transfer Network