

This article appeared in:

*Journal of Engineering Mechanics*, Trans. ASCE, 2009, **135**(10): 1089-1099.

[http://dx.doi.org/10.1061/\(ASCE\)EM.1943-7889.0000024](http://dx.doi.org/10.1061/(ASCE)EM.1943-7889.0000024)

## **Stochastic modeling of deterioration processes through dynamic Bayesian networks**

Daniel Straub<sup>1</sup>

### **Abstract**

A generic framework for stochastic modeling of deterioration processes is proposed, based on dynamic Bayesian networks (DBN). The framework facilitates computationally efficient and robust reliability analysis and, in particular, Bayesian updating of the model with measurements, monitoring and inspection results. These properties make it ideally suited for near-real time applications in asset integrity management and deterioration control. The framework is demonstrated and investigated through two applications to probabilistic modeling of fatigue crack growth.

### **Keywords**

Bayesian analysis; cracking; deterioration; inspection; Markov chain; monitoring; stochastic processes.

---

<sup>1</sup> Associate Professor, Engineering Risk Analysis Group, Technische Universität München, Arcisstr. 21, 80290 Munich, Germany, [straub@tum.de](mailto:straub@tum.de)

## **Introduction**

The modeling of deterioration is subject to significant uncertainty, which arises from a simplistic representation of the actual physical processes (typically through empirical or semi-empirical models) and from limited information on material, environmental and loading characteristics. This uncertainty has been addressed in stochastic models of deterioration processes in the past (Committee on Fatigue and Fracture Reliability 1982; Yang 1994; Melchers 1999). Additionally, observations of the deterioration processes or influencing factors (e.g., from inspection and monitoring) have been included in the models through Bayesian updating (Tang 1973; Madsen et al. 1986), in particular in the context of life-cycle optimization and inspection planning (Thoft-Christensen and Sørensen 1987; Pedersen et al. 1992; Straub and Faber 2006). In principle, the methods of structural reliability enable efficient Bayesian updating of any stochastic model with any kind of information (Madsen et al. 1986). In reality, however, algorithmic difficulties occur (Sindel and Rackwitz 1998), which can make the computations cumbersome and hinder the implementation in software that is run by engineers who are not experts in structural reliability methods.

This paper proposes a novel computational framework for evaluating stochastic deterioration models. The strength of the framework is its computational efficiency and robustness when performing Bayesian updating. It is based on Bayesian networks (BN), a modeling tool that originated in computer science (Pearl 1988; Jensen 2001; Russell and Norvig 2003), but has recently had a number of applications in engineering risk and reliability analysis (Faber et al. 2002; Friis-Hansen 2004; Grêt-Regamey and Straub 2006; Langseth and Portinale 2007). Few researchers have applied BN in the context of deterioration modeling. Friis-Hansen (2001) studies the application of BN for deterioration modeling and inspection planning by means of an example considering fatigue crack growth; Montes-Iturrizaga et al. (2009) use BN for optimizing inspection efforts for offshore structures subject to multiple deterioration mechanisms; Attoh-Okine and Bowers (2006) present an empirical model of bridge deterioration using Bayesian networks. In contrast to these previous publications, which make use of the BN

capabilities mainly for modeling the system aspects or for optimizing inspection and maintenance decision, the present paper focuses on the deterioration modeling. The resulting computational framework enables efficient and robust reliability updating for realistic deterioration models. By robust it is understood that the reliability updating can be performed in an automated manner, not requiring the input from an expert in reliability analysis. This is in contrast to existing efficient computational methods, such as first/second-order reliability methods (FORM/SORM), importance sampling or subset simulation, and facilitates the implementation in software that can be used by the lay engineer for the planning of inspection, repair and maintenance activities, as well as in automated alarm systems based on monitoring data.

The proposed framework is demonstrated and investigated through two applications considering fatigue crack growth, which are representative for a large number of deterioration mechanisms subject to uncertainty.

### **Relation to Markov process models**

Dynamic Bayesian networks (DBN) can be interpreted as a generalization of Markov process models, which have frequently been applied for the modeling of deterioration (Bogdanoff and Kozin 1985; Spencer and Tang 1988; Cesare et al. 1992; Ishikawa et al. 1993; Rocha and Schuëller 1996; Mishalani and Madanat 2002). Markov deterioration processes are characterized by the fact that for a given condition at time  $t_1$ , the condition at any future time  $t_2 > t_1$  is statistically independent of the condition at any past time  $t_0 < t_1$ . It is noted that the Markovian assumption does not hold in engineering practice, where epistemic uncertainties are prevalent (Yang 1994; Melchers 1999; Mishalani and Madanat 2002). Epistemic uncertainties are often time-invariant (e.g., uncertainties due to simplistic parametrical models, due to limited statistical data for empirical models, or due to incomplete knowledge of influencing parameters), thus invalidating the Markovian assumption. To overcome this shortcoming, here a deterioration model is formulated that corresponds to a Markov process model conditional on time-invariant random variables. The DBN technique enables the efficient computation of such models.

Among Markov process models, it can be distinguished between two fundamentally different approaches: models that are based on an underlying parametric model (Ishikawa

et al. 1993) and models that are purely empirical (Cesare et al. 1992; Mishalani and Madanat 2002). The latter are typically finite space Markov chain models. Although the DBN approach can be applied to both models, this paper focuses on parametric models, which are preferable in that they facilitate learning and transferability.

## Dynamic Bayesian networks

The textbooks by Pearl (1988), Jensen (2001) and Russell and Norvig (2003) provide an introduction to BN. In the following, a concise introduction to BN is given, limited to the case of discrete random variables, i.e., random variables that are defined in a finite space.

BN are probabilistic models based on directed acyclic graphs that represent  $p(\mathbf{x})$ , the joint probability mass function (PMF) of a set of random variables  $\mathbf{X}$ . The space of  $\mathbf{X}$ , i.e., the number of outcome states of  $\mathbf{X}$  for which  $p(\mathbf{x})$  must be computed, increases exponentially with the number of variables in  $\mathbf{X}$ , but BN enable an efficient modeling by factoring the joint probability distribution into conditional (local) distributions for each variable.

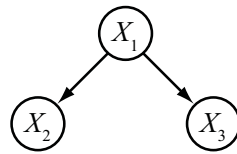


Figure 1. A simple Bayesian network.

A simple BN is illustrated in Figure 1. It consists of three discrete random variables  $X_1, X_2, X_3$ .  $X_1$  is a *parent* of  $X_2$  and  $X_3$ , which are *children* of the former. The PMF of each variable is defined conditional on its parents and the joint PMF of this network is given as a product of these conditional probabilities:

$$p(x_1, x_2, x_3) = p(x_1)p(x_2|x_1)p(x_3|x_1) \quad (1)$$

Wherein  $p(x_i | x_j)$  is the conditional PMF of  $X_i$  given  $X_j = x_j$ . More generally, the joint probability mass function for any BN having discrete variables is given as

$$p(\mathbf{x}) = p(x_1, \dots, x_N) = \prod_{i=1}^N p(x_i | \mathbf{pa}_i) \quad (2)$$

where  $\mathbf{pa}_i$  is the set of realizations of the parents of  $X_i$ . The basic supposition of BNs is that each variable  $X_i$  is independent of all other variables for given values of the variables in its Markov blanket, which includes the parents of  $X_i$ , the children of  $X_i$  and the parents of the children of  $X_i$ .

The BN allows entering evidence: probabilities in the network are updated when new information becomes available. For example, when the state of  $X_2$  in the network in Figure 1 is observed to be  $e$ , this information propagates through the network and the joint PMF of  $X_1$  and  $X_3$  change according to Bayes' rule to

$$p(x_1, x_3 | e) = \frac{p(x_1, e, x_3)}{p(e)} = \frac{p(x_1)p(e|x_1)p(x_3|x_1)}{\sum_{x_1} p(x_1)p(e|x_1)} \quad (3)$$

Consequently the marginal posterior probabilities of  $X_1$  and  $X_3$  are also updated. Note that the common influencing variable  $X_1$  introduces dependence between  $X_2$  and  $X_3$ , but evidence can change the dependence among variables in the network; in the above example, if  $X_1$  is known,  $X_2$  and  $X_3$  become independent. For given sets of evidence, it is possible to infer the independence assumptions encoded in the graphical structure using the rules of *d-separation* (Pearl 1988).

Dynamic Bayesian networks (DBN) are a special class of Bayesian networks, which represent stochastic processes. They consist of a sequence of slices, each of which consists of one or more BN nodes. The slices are connected by directed links from nodes in slice  $i$  to nodes in slice  $i+1$ . Figure 2 shows an example of a DBN. If the model structure and the conditional probability tables are identical for all slices except the first, then the DBN is homogenous. As for any BN, the joint PMF of the variables in the DBN is defined through Equation (2), but a number of inference algorithms are available that are developed especially for the DBN structure (Murphy 2002).

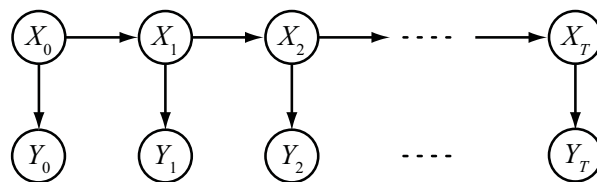


Figure 2. An example of a dynamic Bayesian network.

## Modeling deterioration processes using dynamic Bayesian networks

### Probabilistic modeling of deterioration processes

Consider a parametric deterioration model  $h$  that describes the state of deterioration as a function of time  $t$ , a set of time-invariant model parameters  $\boldsymbol{\theta}$ , a set of time-variant model parameters  $\boldsymbol{\omega}_t = \boldsymbol{\omega}(t)$  and the initial condition  $d_0$ . Deterioration is expressed by defect dimensions  $d_t$  or an empirical damage index  $d_t$  at time  $t$ , e.g., for fatigue,  $d_t$  is either (a) the depth and/or length of a crack when using a fracture mechanics (FM) based model or (b) the Palmgren-Miner damage when using the empirical model based on SN-curves. The model is written in generic form as

$$d_t = d(t) = h(t, d_0, \boldsymbol{\theta}, \boldsymbol{\omega}_1, \dots, \boldsymbol{\omega}_t), \quad t > 0 \quad (4)$$

The proposed DBN model does not replace the parametric deterioration model  $h$ . Instead, the DBN model provides a computational framework that allows accurate and efficient evaluation of  $d_t$  based on the prior stochastic model of the parameters and including all observations of the deterioration process and any of the parameters. The DBN also facilitates learning about the model parameters  $\boldsymbol{\theta}$  and  $\boldsymbol{\omega}_1, \dots, \boldsymbol{\omega}_t$  based on the observations.

Hereafter, we limit ourselves to modeling deterioration as a discrete time process. Furthermore, in accordance with common deterioration models, we require that the dependence among the  $d_t$  is conditionally Markovian, i.e.,

$$f(d_t | d_0, \dots, d_{t-1}, \boldsymbol{\theta}, \boldsymbol{\omega}_1, \dots, \boldsymbol{\omega}_t) = f(d_t | d_{t-1}, \boldsymbol{\theta}, \boldsymbol{\omega}_t), \quad t = 1, 2, \dots, T \quad (5)$$

where  $f$  denotes the probability density function (PDF). Note that  $f(d_t | d_{t-1}, \boldsymbol{\theta}, \boldsymbol{\omega}_t)$  can vary with  $t$ , i.e., the conditional Markov process is not homogenous in the general case. Additionally, we require that the time-variant model parameters  $\boldsymbol{\omega}_t$  are a Markov process conditional on  $\boldsymbol{\theta}$  and  $d_{t-1}$ , i.e.,

$$f(\boldsymbol{\omega}_t | \boldsymbol{\omega}_1, \dots, \boldsymbol{\omega}_{t-1}, d_0, \dots, d_{t-1}, \boldsymbol{\theta}) = f(\boldsymbol{\omega}_t | \boldsymbol{\omega}_{t-1}, d_{t-1}, \boldsymbol{\theta}), \quad t = 1, 2, \dots, T \quad (6)$$

Since  $d_t$  is dependent on the time-invariant uncertain model parameters  $\theta$  for given  $d_{t-1}$ , the deterioration process is not Markovian in the unconditional case, i.e., in general it is  $f(d_t|d_0, \dots, d_{t-1}) \neq f(d_t|d_{t-1})$ .

### ***Limitations of the model framework***

In the generic model, the model parameters  $\omega_t$  and  $d_t$  are assumed to be Markov processes, conditional on  $\theta$ . This represents a limitation, which, however, is not critical in practice, because most stochastic models of deterioration fit at least approximately in this scheme. In particular, models with exclusively time-*invariant* random variables are a special instance of the framework. Examples of time-*variant* models that fit into the above framework are deterioration models based on a Gaussian process  $X(t)$  with exponential autocovariance function, Equation (7), since this process is Markovian (Ishikawa et al. 1993).

$$R_{XX}(t_1, t_2) = \text{cov}[X(t_1), X(t_2)] = \sigma_X^2 \exp(-|t_2 - t_1|/\alpha_X) \quad (7)$$

Fortunately, deterioration process models in the literature almost exclusively utilize an exponential autocovariance function. Even if the process is not Gaussian, the Markovian assumption might still be a reasonable approximation. As an example, consider the lognormal process  $Y(t)$  with median 1, which can be obtained from a zero-mean Gaussian process  $X(t)$  through the transformation  $Y(t) = \exp X(t)$ . Since  $\exp X(t) = 1 + X(t) + X^2(t)/2! + X^3(t)/3! + \dots$ , the first-order approximation of  $Y(t)$  is  $Y(t) \approx 1 + X(t)$  and the first-order approximation of the autocovariance function is  $R_{YY}(t_1, t_2) \approx R_{XX}(t_1, t_2)$ , see also (Ortiz and Kiremidjian 1988). If the process  $Y(t)$  has an exponential autocovariance function, then  $X(t)$  and  $Y(t)$  are approximately Markovian. This is utilized in an application later in this paper.

### **Representation of the deterioration model as a dynamic Bayesian network**

We represent the generic deterioration model as a DBN in Figure 3. Here, the vectors  $\theta_1, \dots, \theta_T$  are introduced to ensure that the time slices are identical. These vectors are related by the deterministic function  $\theta_t = \theta_{t-1}$ ,  $t = 2, \dots, T$  and  $\theta_1 = \theta$ . Introducing these additional parameter vectors has no effect on the computational efficiency of the model, yet it can facilitate the model building process and graphical representation of the model.

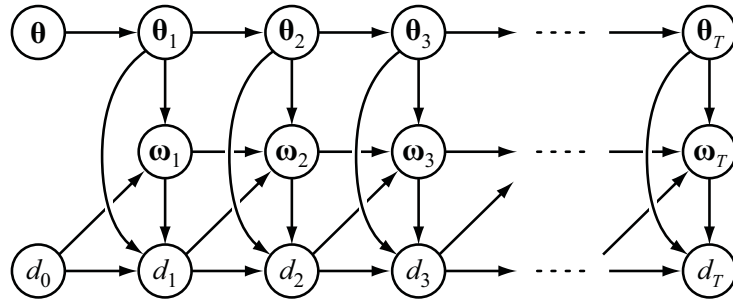


Figure 3. Generic DBN deterioration model.

The links in the DBN shown in Figure 3 correspond to the assumptions of statistical independence among the variables of the generic deterioration model described above. Removing links in a DBN is equivalent to making additional independence assumptions in the model. Therefore, any DBN that is created by removing links from the generic DBN in Figure 3 is a special case of the generic DBN model. As a consequence, the inference algorithm for the generic DBN as presented later can be applied to such DBNs.

The nodes in the generic model represent vectors of variables (e.g., most models have more than one time-invariant model parameter  $\theta$ ). Often, it will be computationally beneficial to consider the variables in these vectors as individual nodes in the network. To separate variables allows exploiting conditional independence of variables in the BN, which can increase the computational efficiency of the models.

### Including observations in the model

A main motivation for the DBN modeling approach is its computational efficiency and robustness for Bayesian updating when new evidence is available. Observations can be available of the initial conditions, the extent of the deterioration  $d_t$ , and the model parameters  $\theta$  and  $\omega_1, \dots, \omega_t$ . Examples of such observations are results from inspection and monitoring, observations of the environmental parameters (e.g., measurements of the chemical and physical properties of liquid and gas in a pipeline), observations of failure/survival events. Often these are indirect observations, i.e., observation that only provide an *indication* of the true parameter values.



Most observations will be independent of the past given the states of the DBN at the time of the observation. Under that condition, each time-slice of the generic DBN can be treated individually by adding the observations as variables in that time-slice. Figure 4 shows the inclusion of an inspection result  $Z_t$  in the model. Such an inspection result can be a discrete random variable (with two states “indication” and “no-indication” of a defect) or a continuous random variable (a measured defect size). In the former case, the inspection is characterized by a Probability of Detection (PoD) model, in the latter case through a measurement accuracy. In either case it is assumed that the inspection result at time  $t$  is independent of all other random variables for given deterioration condition  $d_t$ . The corresponding quantitative model is known as the likelihood function in the context of Bayesian updating,  $L(d_t|z_t) \propto \Pr(Z_t = z_t|d_t)$ . This function fully defines the variable  $Z_t$  in the DBN. Any inspection result is included in the model by instantiating the variable  $Z_t$  with the observed outcome (evidence). When evaluating the DBN with the evidence, the model is automatically updated.

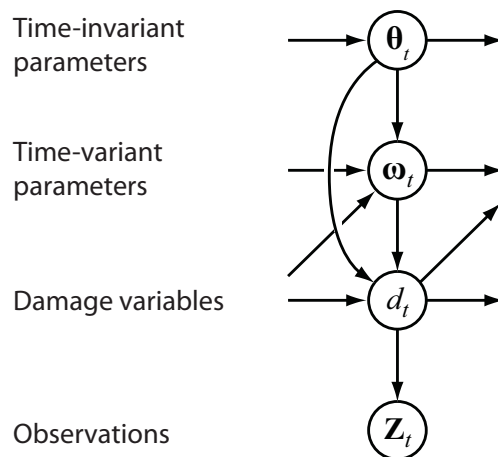


Figure 4. Including an inspection event (observation) in the DBN.

## Computational aspects

### Continuous versus finite state space models

Most variables in deterioration models are defined in a continuous space. Approximate inference algorithms such as Markov Chain Monte Carlo (MCMC) (Gilks et al 1996; Beck and Au 2002) allow working with BN that involve continuous random variables, yet this flexibility comes at a price. While it is possible to prove convergence of the algorithms in the limit, the rate of convergence is unknown and can be extremely slow, therefore, the algorithms might not converge in practical applications. This problem is intensified by the fact that the resulting DBN models can be very large, and that the Gibbs sampler, which is an efficient MCMC inference algorithm for large BN, cannot handle deterministic relations among variables (Hrycej 1990). Approximate inference algorithms might therefore not be suitable for many of the envisaged applications, which involve the updating of the probabilistic model in a near-real-time manner, and they will not be studied further here. However, as long as the amount of evidence is low, approximate inference algorithms such as stochastic simulation or evidence weighting perform well and are easily implemented. This will be used for validation purposes of the discrete model.

In the remainder of the paper, we focus on DBN models wherein all random variables are defined in a finite space. The nodes in the generic DBN model shown in Figure 3 and Figure 4 are defined through probability tables representing the conditional PMFs  $p(\theta_t | \theta_{t-1})$ ,  $p(\omega_t | \omega_{t-1}, d_{t-1}, \theta_t)$ ,  $p(d_t | d_{t-1}, \omega_t, \theta_t)$  and  $p(z_t | \omega_t, d_t, \theta_t)$ . If random variables are defined in a continuous space, they must be replaced by corresponding discrete random variables, as described later.

## Exact inference in the dynamic Bayesian network model

We can distinguish between a number of inference problems in temporal models (Russell and Norvig 2003). Those most relevant in the context of deterioration modeling are *filtering*, *prediction* and *smoothing*<sup>2</sup>. We can summarize these inference problems as:

- Filtering (or monitoring): The task of computing the posterior distribution over the state at time  $t$  given evidence up to time  $t$ , i.e.,  $p(\boldsymbol{\theta}_t, \boldsymbol{\omega}_t, d_t | \mathbf{z}_1, \dots, \mathbf{z}_t)$ .
- Prediction: The task of computing the posterior distribution over the state at a future time  $T$  given evidence up to time  $t$ , i.e.,  $p(\boldsymbol{\theta}_T, \boldsymbol{\omega}_T, d_T | \mathbf{z}_1, \dots, \mathbf{z}_t)$ , with  $t < T$ .
- Smoothing: The task of computing the posterior distribution over the state at a past time  $t$  given evidence up to time  $T$ , i.e.,  $p(\boldsymbol{\theta}_t, \boldsymbol{\omega}_t, d_t | \mathbf{z}_1, \dots, \mathbf{z}_T)$ , with  $t < T$ .

For DBN with discrete random variables, a number of exact inference algorithms exist for these tasks (Murphy 2002; Russell and Norvig 2003), some of which are implemented in commercial and free BN software (Murphy 2001). In the Appendix, an adopted version of the “forward-backward” algorithm is introduced. The main constituents of this algorithm are

- the forward operation (for time  $t$ ), which computes  $p(\boldsymbol{\theta}_t, \boldsymbol{\omega}_t, d_t | \mathbf{z}_1, \dots, \mathbf{z}_t)$  by means of a recursive algorithm as presented in the Appendix;
- the backward operation (for time  $t$ ), which computes  $p(\mathbf{z}_{t+1}, \dots, \mathbf{z}_T | \boldsymbol{\theta}_t, \boldsymbol{\omega}_t, d_t)$  by means of a recursive algorithm as presented in the Appendix.

Filtering is performed by simply applying the forward operation for time  $t$ . Predicting is performed by applying the forward operation for time  $T$ , whereby the likelihood

---

<sup>2</sup> Another inference task is *learning* of the conditional probability tables defining the nodes. However, since the DBN model is based on a parametric deterioration model, data is not used to learn the conditional probability tables, e.g.,  $p(d_t | d_{t-1}, \boldsymbol{\theta}, \boldsymbol{\omega}_t)$ . Instead, the probability distributions of the time-invariant model parameters  $\boldsymbol{\theta}$  are updated with the observations.

functions for  $\mathbf{z}_{t+1}, \dots, \mathbf{z}_T$  are set equal to one:  $p(\mathbf{z}_i | d_i) = 1$ ,  $i = t+1, \dots, T$  (i.e., no evidence is entered for these variables). Finally, smoothing is carried out by performing both the forward operation and the backward operation for time  $t$ . We then obtain

$$p(\boldsymbol{\theta}_t, \boldsymbol{\omega}_t, d_t | \mathbf{z}_0, \dots, \mathbf{z}_T) \propto p(\boldsymbol{\theta}_t, \boldsymbol{\omega}_t, d_t | \mathbf{z}_0, \dots, \mathbf{z}_t) p(\mathbf{z}_{t+1}, \dots, \mathbf{z}_T | \boldsymbol{\theta}_t, \boldsymbol{\omega}_t, d_t) \quad (8)$$

Equation (8) is an application of Bayes' rule, whereby the forward operation result  $p(\boldsymbol{\theta}_t, \boldsymbol{\omega}_t, d_t | \mathbf{z}_1, \dots, \mathbf{z}_t)$  is the prior probability. The likelihood function is  $p(\mathbf{z}_{t+1}, \dots, \mathbf{z}_T | \boldsymbol{\theta}_t, \boldsymbol{\omega}_t, d_t)$ , the result of the backward operation, because of independence of  $\mathbf{z}_{t+1}, \dots, \mathbf{z}_T$  from  $\mathbf{z}_1, \dots, \mathbf{z}_t$  for given  $\boldsymbol{\theta}_t, \boldsymbol{\omega}_t, d_t$ , as prescribed by the DBN structure.

Let the number of states of  $d_t, \boldsymbol{\omega}_t, \boldsymbol{\theta}_t$  be  $m_d, m_\omega, m_\theta$ , respectively. As demonstrated in the Appendix, the computation time for filtering is  $O[(m_d^2 m_\omega + m_d m_\omega^2) m_\theta t]$ , whereas it is  $O[(m_d^2 m_\omega + m_d m_\omega^2) m_\theta T]$  for predicting and smoothing. Therefore, the computational performance is determined by the number of states utilized for representing the variables, in particular  $d_t$  and  $\boldsymbol{\omega}_t$ . The discretization of the random variables defined in a continuous space is thus a critical part of the framework.

### Discretization of continuous random variables

All random variables that are defined in continuous space are replaced by equivalent variables defined in a finite space. It is suggested to perform this discretization sequentially, i.e., one variable at the time, and to proceed according to the hierarchy in the BN, i.e., parents are discretized before their children.

Consider discretization of a single random variable  $X$ . This variable has a set of parent variables  $\mathbf{X}_p = pa(X)$  and a set of children variables  $\mathbf{X}_c = ch(X)$ . Its corresponding discrete variable is  $\hat{X}$  and has  $m_x$  states, which are denoted by  $\hat{x}^{(k)}$ ,  $k = 1 \dots m_x$ . The  $m_x$  states of  $\hat{X}$  correspond to mutually exclusive, collectively exhaustive intervals in the space of the original variable  $X$ . Because of the hierarchal discretization order, all variables in  $\mathbf{X}_p$  are discrete.

The continuous random variable  $X$  is replaced in the BN with the discrete random variable  $\hat{X}$ . Therefore, the conditional PMF of  $\hat{X}$  given  $\mathbf{X}_p = \mathbf{x}_p^{(l)}$  must be defined, as well as the conditional distribution of  $\mathbf{X}_c$  given  $\hat{X} = \hat{x}^{(k)}$ . The conditional PMF of  $\hat{X}$  is

$$p\left(\hat{x}^{(k)} \mid \mathbf{x}_p^{(l)}\right) = F_X\left(x_k^+ \mid \mathbf{x}_p^{(l)}\right) - F_X\left(x_k^- \mid \mathbf{x}_p^{(l)}\right) \quad (9)$$

wherein  $F_X$  is the cumulative distribution function (CDF) of  $X$ , which is conditional on  $\mathbf{X}_p = \mathbf{x}_p^{(l)}$ , and  $x_k^-$  and  $x_k^+$  are the lower and upper boundaries of the interval corresponding to state  $k$ .

The distribution of  $\mathbf{X}_C$  conditional on  $\hat{X} = \hat{x}^{(k)}$  is approximated by

$$F_{\mathbf{X}_C}\left(\mathbf{x}_C \mid \hat{x}^{(k)}, \mathbf{y}_p\right) = \int_{x_k^-}^{x_k^+} F_{\mathbf{X}_C}\left(\mathbf{x}_C \mid x, \mathbf{y}_p\right) f_X\left(x \mid \hat{x}^{(k)}\right) dx \quad (10)$$

in which  $\mathbf{y}_p$  are realizations of all random variables that are parents to  $\mathbf{X}_C$  excluding  $X$ . The approximation in Equation (10) is twofold. First, it is assumed that the distribution of  $X$  is independent of  $\mathbf{Y}_p$  for given  $\hat{X}$ , which does not hold in the general case, although it does in many special instances. Second, an assumption about  $f_X(x \mid \hat{x}^{(k)})$  must be made, because  $f_X(x)$  is unknown. (Remember that the variable  $X$  in the BN is only defined conditional on  $\mathbf{X}_p$ .) A straightforward choice is the uniform distribution for  $f_X(x \mid \hat{x}^{(k)})$ :

$$f_X\left(x \mid \hat{x}^{(k)}\right) = \frac{1}{x_k^+ - x_k^-}, \quad x_k^- < x \leq x_k^+ \quad (11)$$

The uniform assumption is not suitable if the interval is only bounded on one side. In such a case, it is suggested to apply an exponential PDF, e.g., for the case of an interval with only a lower bound:

$$f_X\left(x \mid \hat{x}^{(k)}\right) = \lambda \exp\left[-\lambda\left(x - x_k^-\right)\right], \quad x_k^- < x \quad (12)$$

$\lambda$  can be freely selected to best represent the original distribution.

When  $X$  has no parents,  $f_X(x)$  is known and no approximation is required. Then  $f_X(x \mid \hat{x}^{(k)})$  is simply the original PDF of  $X$ , truncated in the range  $x_k^- < x \leq x_k^+$ .

### *Choice of the discrete intervals*

The approximation in Equation (10) introduces an unknown error, which becomes zero in the limit as the size of the discretization intervals approaches zero. The choice of these intervals, therefore, is crucial for the accuracy of the results as well as the computational

performance of the algorithms. This choice is application-specific, but some general considerations are made and a heuristic for the selection is presented in the following

Obviously, the goal is to select the discretization scheme that provides sufficient accuracy at maximum computational speed. Additionally, the scheme must ensure flexibility with respect to potential evidence that may become available in applying the model. Evidence will update the marginal distributions of the random variables in the model, thus increasing the difficulty in selecting an optimal discretization scheme a-priori. It is crucial that the discretization scheme leads to accurate results for all possible combinations of evidence. There is potential for an adaptive procedure, whereby the discretization scheme is adjusted as evidence becomes available (Neil et al 2007), but this is not further considered here since it is our aim to keep the procedure as simple as possible. For the same reason, only homogenous DBN models are considered, i.e., we require that the discretization scheme is the same for all time slices.

Consider discretization of a single random variable  $X$ . First, we identify the probable range of values of  $X$ . The probable range of values is defined so that the a-priori probability of a variable being outside that range is smaller than  $p$  for all time slices  $t$ . The value of  $p$  will determine the minimum probability value that can be computed accurately with the model. For most practical applications, in particular when the probability of the failure event is of interest,  $p = 10^{-6}$  will be sufficient.

Next, we determine the discretization intervals in the space of  $X$  within the probable range of values. These intervals are assigned without consideration of the a-priori distribution of  $X$ , to ensure flexibility with respect to potential evidence. The simplest choice would be intervals with equal lengths. However, it is recommended to account for the influence of  $X$  on the deterioration damage  $d_t$ . To this end, intervals of equal length are assigned in the space of  $T_x(x)$ , with  $T_x(x)$  being a suitable transformation such that the relation between  $T_x(x)$  and  $d_t$  is approximately linear. In most cases, it will be sufficient to select a simple functional form for  $T_x(x)$ , such as  $T_x(x) = \ln(x)$  or  $T_x(x) = \exp(x)$ . Finally, the interval borders  $\mathbf{x}_B$  are established as a function of the upper bound  $x_{ub}$  and the lower bound  $x_{lb}$  of the probable range of  $X$  as

$$\mathbf{x}_B(2:m_X) = T_x^{-1} \left[ T_x(x_{lb}) : \frac{T_x(x_{ub}) - T_x(x_{lb})}{m_X - 2} : T_x(x_{ub}) \right] \quad (13)$$

$\mathbf{x}_B(1)$  and  $\mathbf{x}_B(m_X + 1)$  are set equal to the borders of the physically admissible domain of  $X$ . The number of discrete states,  $m_X$ , is adjusted to achieve the optimal balance between accuracy and speed. A higher value is chosen for variables with a higher importance (e.g., as appraised by a sensitivity measure on the reliability index a-priori). Finally, the resulting discretization scheme must be validated by comparing results for a few selected evidence cases with results obtained by alternative computation methods (typically crude simulation, which has the advantage of being unbiased and robust).

### **Application I: Fatigue crack growth modeling involving time-invariant random variables only**

This example presents the application of the DBN model to a representative fracture mechanics (FM) based fatigue model that involves only time-invariant random variables. The aim of this example is to study the effect of discretization and to compare the performance of the DBN model with other numerical solutions, such as the second-order reliability method (SORM) and crude Monte Carlo simulation (MCS). For the sake of the example, a simple model is chosen from Ditlevsen and Madsen (1996). Crack growth is described by Paris' law, Equation (14).

$$\frac{da(n)}{dn} = C \left[ \Delta S \sqrt{\pi a(n)} \right]^m \quad (14)$$

$a$  is the crack length,  $n$  is the number of stress cycles,  $\Delta S$  is the stress range per cycle (constant stress amplitudes are assumed) and  $C$  and  $m$  are empirically determined model parameters. In this formulation of Paris' law, the geometry correction factor is one, which in theory corresponds to the case of a crack in a plate with infinite size. With the boundary condition  $a(n=0) = a_0$ , this differential equation can be solved for the crack size as a function of the number of cycles  $n$ , (Ditlevsen and Madsen 1996):

$$a(n) = \left[ \left( 1 - \frac{m}{2} \right) C \Delta S^m \pi^{m/2} n + a_0^{(1-m/2)} \right]^{(1-m/2)^{-1}} \quad (15)$$

The event of failure is described by the limit state function  $g$  as a function of  $a(n)$  and the critical crack length  $a_c$ :

$$g = a_c - a(n) \quad (16)$$

The performance of the structural element is represented through the binary variable  $E$ . Failure of the element  $\{E = 0\}$  occurs if  $g \leq 0$ , and its complement  $\{E = 1\}$ , the survival event, occurs if  $g > 0$ . The probabilistic model is summarized in Table 1.

*Table 1. Parameters of the fatigue crack growth application with time-invariant parameters.*

Variable	Distribution	Mean	Standard dev.	Correlation
$a_0$ [mm]	Exponential	1	1	-
$a_c$ [mm]	Deterministic	50	-	-
$\Delta S$ [Nmm <sup>-2</sup> ]	Normal	60	10	-
$\ln(C), m$ [* <sup>1</sup> ]	Bi-Normal	[-33; 3.5]	[0.47; 0.3]	$\rho_{\ln(C),m} = -0.9$

\*<sup>1</sup>) dimension corresponding to N and mm

Since this model contains only time-invariant parameters, we have, in terms of the generic model,  $\theta = [\Delta S, C, m]$  and  $d_t = a_t$ . In analogy to Equation (15), we obtain the deterministic function for  $a_t$ :

$$a_t = \left[ \left( 1 - \frac{m}{2} \right) C \Delta S^m \pi^{m/2} \Delta n + a_{t-1}^{(1-m/2)} \right]^{(1-m/2)^{-1}}, \quad t = 1, \dots, T \quad (17)$$

$\Delta n$  is the number of stress cycles during one time step. If the model were implemented with each variable as a separate node, the space of  $\theta$  would consist of  $m_\theta = m_{\Delta S} m_C m_m$  states. To reduce  $m_\theta$ , and consequently computation time, we make use of the fact that the two variables  $\Delta S, C$  can be replaced by a single variable  $q = (1 - m/2) C^{m/2} \Delta S^m \Delta n$  in Equation (17). The dimension of the space of  $\theta$  is reduced, and it is  $m_\theta = m_q m_m$ . The



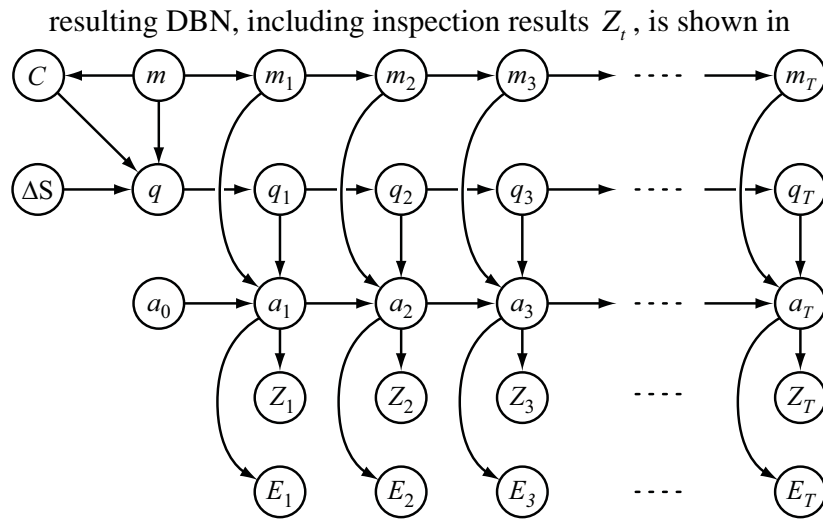


Figure 5.

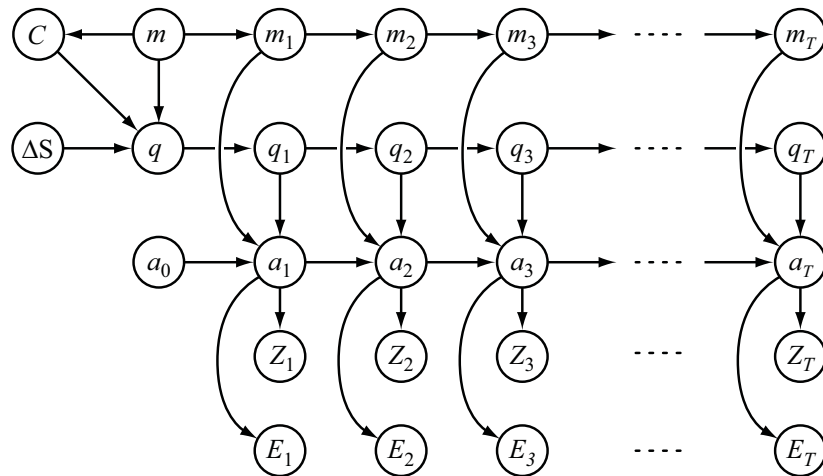


Figure 5. Fatigue crack growth as a DBN, case with time-invariant parameters.

In the numerical implementation, we choose  $\Delta n = 10^5$ . The discretization scheme for the random variables is summarized in Table 2. It is noted that this discretization scheme is likely suboptimal, because of the strong statistical dependence between  $q$  and  $m$ . To optimize the discretization scheme, the joint space of  $q$  and  $m$  should be considered, but the present discretization scheme is considered sufficiently accurate and efficient. The numbers of discrete states of  $S$  and  $\ln C$  have little effect on the computational speed,

because for given values of  $q$  and  $m$  the remaining variables of the model are statistically independent of  $S$  and  $\ln C$ . It is, therefore, sufficient to evaluate the prior distribution of  $q$  and  $m$ , then update all variables in the model according to the proposed algorithm, and then compute the posterior PMF of  $S$  and  $\ln C$  based on the posterior joint PMF of  $q$  and  $m$ . The computation time for this last step is orders of magnitude lower than for performing a forward or backward operation.

The PMF of  $\Delta S$  and  $m$  are obtained analytically using Equation (9);  $p(C|m)$  is computed from the cumulative bi-Normal distribution;  $p(q|\Delta s, m, C)$  and  $p(a_t|a_{t-1}, m_t, q_t)$  are evaluated using MC simulation with  $10^4$  samples. Finally,  $p(m_t|m_{t-1})$  and  $p(q_t|q_{t-1})$  are unit diagonal matrixes.

*Table 2. Discretization scheme.*

variable	Probable range	$T_x(x)$	Number of states $m_i$	Final interval boundaries
mm]	0.01 – 50	$\ln(a_t)$	80	$0, \exp\{\ln(0.01):[\ln(50)-\ln(0.01)]/78:\ln(50)\}, \infty$
	2.0 – 5.0	$\exp(m)$	30	$0, \ln\{\exp(2):[\exp(5)-\exp(2)]/28\}, \infty$
<sup>1]</sup>	-1.0 – (-10 <sup>-3</sup> )	$\ln(-q)$	35	$-\infty, -\exp[0:\ln(10^{-3})/33:\ln(10^{-3})], \infty$
[N/mm]	10 – 110	$\Delta S$	52	$0, 10:2:110, \infty$
*) [ <sup>1]</sup>	(-35.3) – (-30.7)	$\ln(C)$	48	$-\infty, -35.3:0.1:-30.7, \infty$

\*<sup>1</sup>) dimension corresponding to N and mm

### **Including evidence**

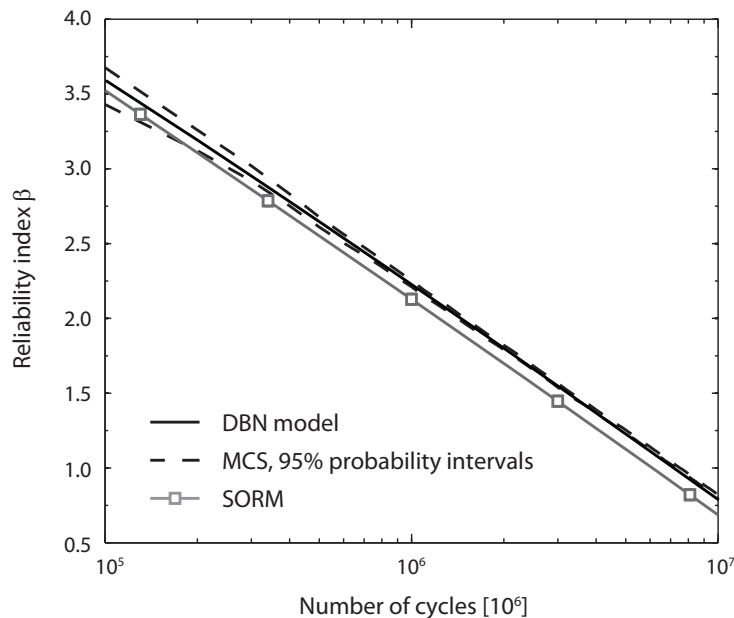
Inspections of the structural element are carried out in intervals of  $10^6$  cycles and it is assumed that all inspections result in no-indication (i.e., no defect is found). In the numerical example, we consider the following PoD model (with  $D$  being the event of detection):

$$\Pr(Z_t = D | a) = PoD(a) = 1 - \exp(-a/10\text{mm}) \quad (18)$$

Obviously, the probability of no-detection at time  $t$  conditional on  $a$  is  $\Pr(Z_t = \bar{D} | a) = 1 - PoD(a)$ . As described earlier, evidence is included by instantiating the inspection variables  $Z_t$  in the DBN with the observed events (here: no-indication) at the times of inspection.

## Results

The results obtained with the DBN are compared to results obtained with MCS and SORM, in terms of the reliability index  $\beta = \Phi^{-1}[\Pr(E = 1)]$  with  $\Phi^{-1}$  being the inverse normal CDF. For the unconditional case (no observations), the results are shown in Figure 6.



*Figure 6. Results for the unconditional case.*

For the case including inspection results, we apply the filtering inference algorithm, i.e., the reliability index  $\beta$  after  $n$  cycles is computed by consideration of all inspection results up to  $n$  cycles, but neglecting later inspection results. The results are shown in Figure 7.

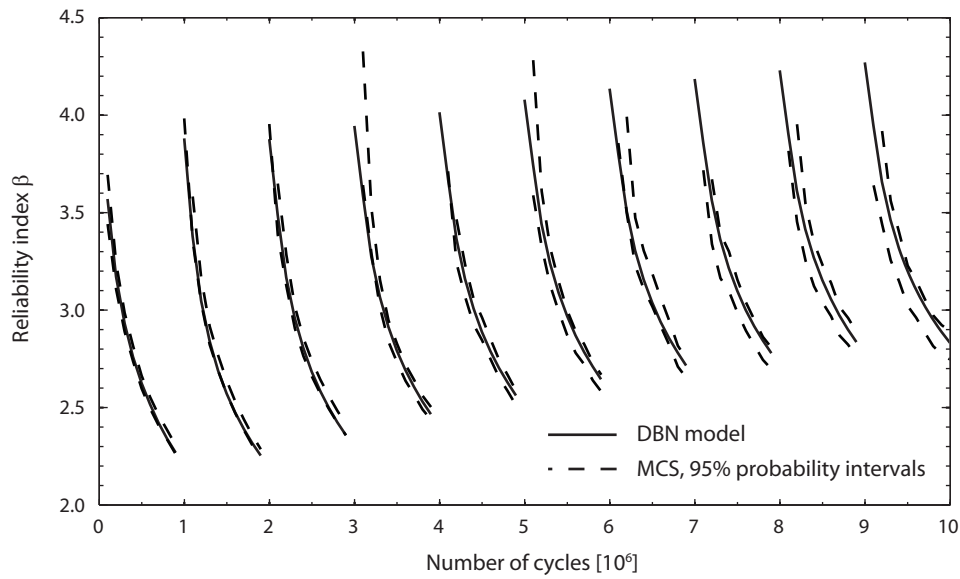


Figure 7. Results for the conditional case (no indication of a defect at all inspections).

The results in Figure 6 and Figure 7 demonstrate the accuracy of the DBN approach. Besides facilitating efficient computation of the reliability as a function of time, the DBN model enables updating of the distribution of all random variables in the model with given inspection results. As an example, Figure 8 presents the updated complementary CDF of the stress ranges  $\Delta S$  for different evidence cases.

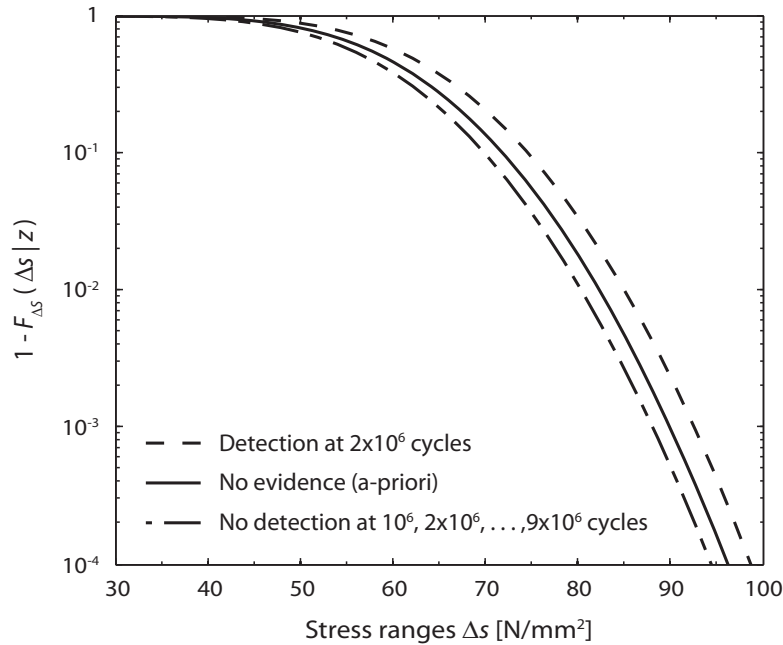


Figure 8. Updating of the distribution of stress ranges  $\Delta S$  with inspection results.

### Computational performance

With the DBN established, the computation of the results presented here takes in the order of 10 CPU seconds on a standard PC with a 2.0 GHz processor with a Matlab-based program. The computation time increases linearly with the number of time steps considered (the 10 CPU seconds correspond to 100 time steps), but is independent of the number of observations.

### **Application II: Fatigue crack growth as a stochastic process**

Fatigue crack growth has frequently been modeled by a random process model (Ortiz and Kiremidjian 1988; Lin and Yang 1985; Yang and Manning 1996; Zheng and Ellingwood 1998; Beck and Melchers 2004). To demonstrate the applicability of the DBN framework for such models, the model suggested by Yang and Manning (1996) is utilized in a numerical example. Equation (19) shows the model in its generic format.

$$\frac{da(n)}{dn} = Y(n) g_{cg} [a(n), \boldsymbol{\theta}] \quad (19)$$

$a(n)$  is the crack size after  $n$  cycles and  $g_{cg}[\cdot]$  can be any deterministic crack growth law as a function of  $a(n)$  and of time-invariant parameters  $\boldsymbol{\theta}$ . The simplistic crack growth law considered in this example is the same as in the previous example, i.e.,

$$g[a(n), \boldsymbol{\theta}] = C \left[ \Delta S \sqrt{\pi a(n)} \right]^m \quad (20)$$

$Y(n)$  in Equation (19) is a stationary lognormal process with a median value equal to 1 and standard deviation  $\sigma_Y$ . The autocovariance function of  $Y(n)$  is of an exponential form:

$$R_{YY}(n_1, n_2) = \text{cov}[Y(n_1), Y(n_2)] = \sigma_Y^2 \exp(-|n_2 - n_1|/\alpha_Y) \quad (21)$$

The parameter  $\alpha_Y$  is a measure of the correlation time for  $Y(n)$ .

For given values of  $\boldsymbol{\theta}$  and  $a_0$ , Yang and Manning (1996) present an analytical solution for the probability of the crack exceeding a given value  $a_c$ :

$$\Pr[a_c < a(n) | \boldsymbol{\theta}, a_0] = \Phi \left\{ \frac{\ln n - \ln [n_{0.5}(a_c | \boldsymbol{\theta}, a_0) / \lambda(n)]}{\psi(n)} \right\} \quad (22)$$

This solution is a second-order approximation, based on the assumption that  $W(\tau) = \int_0^\tau Y(n) dn$  is a lognormal process.  $n_{0.5}(a_c | \boldsymbol{\theta}, a_0)$  is the median number of cycles to reach  $a_c$ , computed as

$$n_{0.5}(a_c | \boldsymbol{\theta}, a_0) = \int_{a_0}^{a_c} \frac{da}{g[a, \boldsymbol{\theta}]} \quad (23)$$

For the crack growth law in Equation (20), it is

$$n_{0.5}(a_c | \boldsymbol{\theta}, a_0) = \frac{1}{C \Delta S^m \pi^{m/2}} \left[ \frac{a_c^{(1-m/2)} - a_0^{(1-m/2)}}{1 - m/2} \right] \quad (24)$$

The remaining parameters in Equation (22) are obtained as follows (for details refer to (Yang and Manning 1996)).

$$\lambda(n) = \exp(\sigma_{\ln Y}^2 / 2) \left[ 1 + \phi^2(n) \exp(\sigma_{\ln Y}^2) - \phi^2(n) \right]^{-1/2} \quad (25)$$

$$\psi(n) = \left\{ \ln \left[ 1 + \phi^2(n) \exp(\sigma_{\ln Y}^2) - \phi^2(n) \right] \right\}^{1/2} \quad (26)$$

$$\phi^2(n) = \frac{2\alpha_Y^2}{n^2} \left[ \exp\left(-\frac{n}{\alpha_Y}\right) + \frac{n}{\alpha_Y} - 1 \right] \quad (27)$$

where  $\sigma_{\ln Y}^2$  is the variance of  $\ln Y(n)$ .

The solution in Equation (22) is applied to verify the results obtained with the DBN model. The unconditional  $\Pr[a_c < a(n)]$  is computed from  $\Pr[a_c < a(n) | \boldsymbol{\theta}, a_0]$  by means of numerical integration.

In the numerical investigations, the parameter values provided in Table 3 are applied.  $\alpha_Y$ , the correlation length of  $Y(n)$ , is modeled as a random variable.

*Table 3. Parameters of the stochastic process crack growth model example.*

Variable	Distribution	Mean	Standard dev.	Correlation
$a_0$ [mm]	Exponential	1	1	-
$a_c$ [mm]	Deterministic	50	-	-
$\Delta S$ [Nmm <sup>-2</sup> ]	Deterministic	60	-	-
$\ln(C)$ [* <sup>1</sup> ]	Deterministic	-30.0	-	-
$m$ [-]	Deterministic	3.0	-	-
$Y(n)$ [-]	Lognormal	1.2	0.8	Equation (21)
$\alpha_Y$ [cycles]	Lognormal	10 <sup>6</sup>	10 <sup>6</sup>	-

\*<sup>1</sup>) dimension corresponding to N and mm

### **DBN modeling**

The DBN model for this example is shown in Figure 9. The binary variable  $E$  represents the performance of the structural element (failure  $\{E = 0\}$  or survival  $\{E = 1\}$ ). The potential evidence that is included in the DBN model are measurements  $M_t$  of the crack depth.  $M_t$  is assumed to be Normal distributed with mean equal to the true crack depth

$a_t$  and standard deviation  $\sigma_M = 1\text{mm}$ . This DBN model assumes that the Lognormal process  $Y(n)$ , represented by the variables  $Y_t$ , is Markovian, which represents a first-order approximation as discussed earlier.

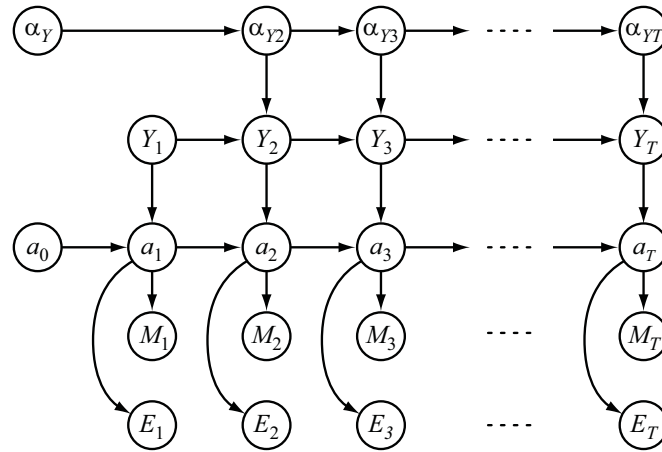


Figure 9. Fatigue crack growth as a DBN, random process model.

The discretization scheme is summarized in Table 4. The PMFs  $p(\alpha_Y)$ ,  $p(Y_1)$  and  $p(a_0)$  are calculated analytically from Equation (9);  $p(Y_t | Y_{t-1}, \alpha_{Y_t})$ ,  $p(a_t | a_{t-1}, Y_t)$  and  $p(M_t | a_t)$  are calculated using the approximations in Equations (11) and (12), assuming that  $Y_t$  is constant during one time slice. The correlation coefficient between  $Y_t$  and  $Y_{t-1}$  is given by Equation (21). The number of cycles between the slices is  $\Delta n = 10^5$ . For 100 time slices, this discretization scheme leads to CPU times in the order of 1-10 seconds on a standard PC with a 2.0 GHz processor for the presented results.



Table 4. Discretization scheme for the random process example.

variable	Probable range	$T_X(x)$	Number of states $m_i$	Final interval boundaries
$\bar{v}_t$ [mm]	0.01 – 50	$\ln(a_t)$	81	$0, \exp[\ln(0.01) : \frac{\ln(50) - \ln(0.05)}{80} : \ln(50)], \infty$
$Y_t$ [-]	0.0 – 7.0	$Y_t$	31	$0.0 : 0.175 : 7.0, \infty$
$\chi_Y$ [cycles]	$10^3 - 10^9$	$\log(\alpha_Y)$	21	$0, 10^{[3 : 0.3 : 9]}, \infty$
$M_t$ [mm]	Same discretization as $a_t$			

### Numerical results

Figure 10 presents a comparison of the DBN results with the solution based on second-order approximation, in terms of the reliability index for the case of no measurements. It is observed that the first-order Markovian approximation for  $Y_t$  has little influence on the reliability index. The deviation of the DBN result from the second-order results for small numbers of cycles is mainly due to the approximation in the discretization of  $Y_t$ , as found from additional numerical investigations, which are not reported here for brevity. When doubling the number of discrete states for  $Y_t$ , the difference between the two solutions becomes negligible.

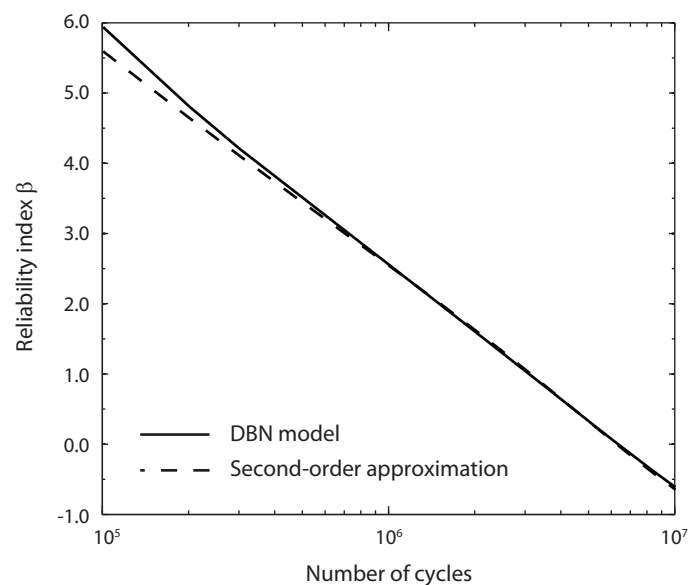


Figure 10. Result for the random process crack growth example – unconditional case.

To demonstrate the model updating with measurements of the deterioration size, the model is updated with the measurement results summarized in Table 5. Figure 11 presents the resulting posterior mean of the crack depth for filtering (including evidence up to  $t$ ) and for smoothing (including all evidence).

Table 5. Measurement results.

Measurement times [ $10^5$ cycles]	10	20	30	40	50	60	70	80	90
Measurements $M_t$ [mm]	0.3	0.3	0.3	0.5	1.0	3.5	5.0	10.0	12.0

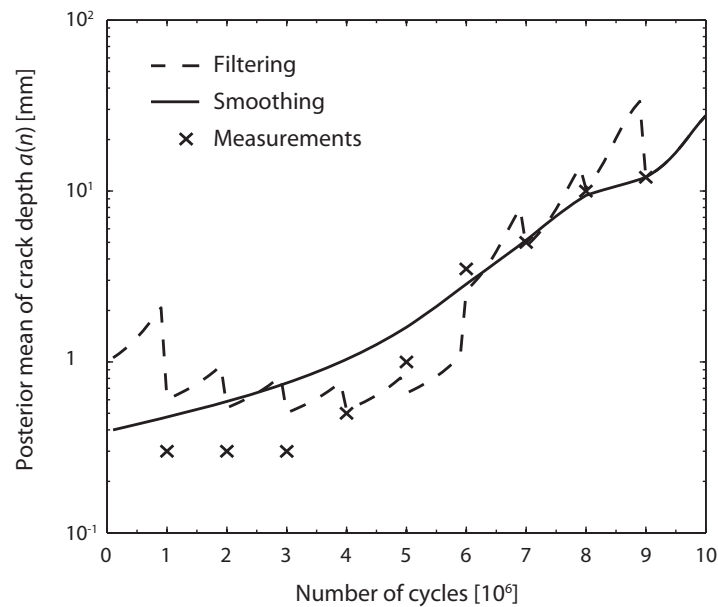


Figure 11. Posterior mean of the crack depth  $a(n)$ .

For the same measurements, Figure 12 presents the posterior mean of the stress range process  $\Delta S \cdot Y(n)^{1/m}$  for two values of the correlation length  $\alpha_y$ . For higher values of  $\alpha_y$ , larger differences are observed between the posterior and the prior mean of  $\Delta S \cdot Y(n)^{1/m}$ . (The latter is  $63.7\text{N/mm}^2$  in both cases.) This effect is expected, because higher correlation implies that a measurement at time  $t$  provides more information about the process at other times (Straub and Faber 2007).

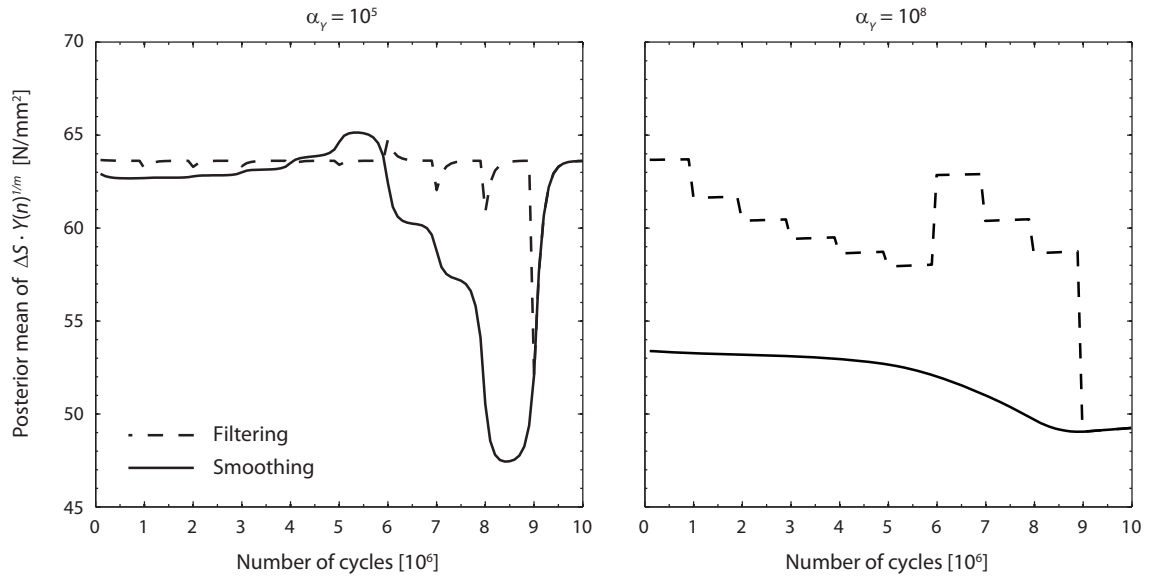


Figure 12. Posterior mean of the stress range process  $\Delta S \cdot Y(n)^{1/m}$  for two different values of the correlation length.

The updating of the probability distribution of model parameters based on inspection results is straightforward with the DBN. Exemplarily, the measurement results in Table 5 are utilized to update the distribution of the correlation length  $\alpha_\gamma$ . This results in a posterior distribution of  $\alpha_\gamma$  that has mean value  $1.1 \cdot 10^6$  and standard deviation  $0.95 \cdot 10^6$ . These values are close to the prior values, which indicates that the measurement results contain little information on  $\alpha_\gamma$ . This is in agreement with other results, which are not reported here, showing that the value of  $\alpha_\gamma$  has little effect on the reliability and the probability distribution of the crack depth, despite its significant impact on the posterior probability distribution of the stress range process. However, it is conceivable that learning about the correlation length is possible by collecting measurement data from a larger number of specimens, in particular in combination with other information, e.g., on the initial crack depth.

### Concluding remarks

The aim of the DBN framework is to provide a computationally robust and efficient approach to probabilistically assess the condition and the reliability of structural elements subject to deterioration when observations are available. Figure 6, Figure 7 and Figure 10

demonstrate the accuracy of the framework for computing the reliability of deteriorating structural elements. Figure 8 illustrates the capability of the DBN approach for Bayesian updating of the uncertain model parameters. Finally, Figure 11 and Figure 12 demonstrate the capabilities of the DBN framework in monitoring the performance of structures subject to deterioration in a near-real time manner. Based on these results, it is concluded that the proposed framework has a huge potential for applications in monitoring, inspection, maintenance and repair planning.

The DBN framework is limited with respect to the number of random variables that can be included, due to the exponential increase in computation time with the number of random variables in the general case. However, as illustrated by the introduction of the auxiliary random variable  $q$  in the first application, it is often possible to reduce the number of time-invariant random variables by reformulating the problem. Furthermore, at present most practically relevant deterioration models contain only a small number of random variables.

Future work should aim at extending the presented framework to consider multiple structural elements. This is relevant for the asset integrity management of structural systems, but also for parameter estimation from experiments. Such an extension could be based on formulating the PMF of the time-invariant parameters  $\theta$  and the initial damage condition  $d_0$  conditional on so-called hyper-parameters  $\mathbf{A}$  that are common to all structural elements (i.e., the hyper-parameters are introduced as common parents to the variables  $\theta$  and  $d_0$  of all elements). The DBNs representing the individual elements are then connected through the  $\mathbf{A}$  node, which allows passing information between the elements, such that the information collected for one element is used to update the probabilistic model of all elements. Depending on the type of dependence among the system elements, different models can be envisaged and should be investigated. The flexibility of the BN methodology will facilitate such developments.

## **Acknowledgment**

This work was supported by the Swiss National Science Foundation (SNF) through grant PA002-111428.

## Appendix – Inference algorithm

This Appendix presents algorithms for the forward and the backward operation, which form the basis for all inference, as summarized earlier. The algorithms are based on algorithms originally developed for Hidden Markov Models (HMM) (Russell and Norvig 2003) and are here adopted to the proposed deterioration modeling framework. They approximately correspond to the frontier algorithm presented by Murphy (2002).

### Forward computation

We compute  $p(\boldsymbol{\theta}_t, \boldsymbol{\omega}_t, d_t | \mathbf{z}_1, \dots, \mathbf{z}_t)$  through a recursive algorithm. By considering the independence assumptions encoded in the BN structure, as shown in Figure 3 and Figure 4, we obtain the following relationships:

$$p(d_t, \boldsymbol{\omega}_t, \boldsymbol{\theta}_t | \mathbf{z}_0, \dots, \mathbf{z}_t) \propto p(d_t, \boldsymbol{\omega}_t, \boldsymbol{\theta}_t | \mathbf{z}_0, \dots, \mathbf{z}_{t-1}) p(\mathbf{z}_t | d_t) \quad (28)$$

$$\begin{aligned} & p(d_t, \boldsymbol{\omega}_t, \boldsymbol{\theta}_t | \mathbf{z}_0, \dots, \mathbf{z}_{t-1}) \\ &= \sum_{d_{t-1}} \sum_{\boldsymbol{\omega}_{t-1}} \sum_{\boldsymbol{\theta}_{t-1}} p(d_t, \boldsymbol{\omega}_t, \boldsymbol{\theta}_t | d_{t-1}, \boldsymbol{\omega}_{t-1}, \boldsymbol{\theta}_{t-1}) p(d_{t-1}, \boldsymbol{\omega}_{t-1}, \boldsymbol{\theta}_{t-1} | \mathbf{z}_0, \dots, \mathbf{z}_{t-1}) \\ &= \sum_{d_{t-1}} \sum_{\boldsymbol{\omega}_{t-1}} \sum_{\boldsymbol{\theta}_{t-1}} p(d_t | d_{t-1}, \boldsymbol{\omega}_t, \boldsymbol{\theta}_t) p(\boldsymbol{\omega}_t | d_{t-1}, \boldsymbol{\omega}_{t-1}, \boldsymbol{\theta}_t) p(\boldsymbol{\theta}_t | \boldsymbol{\theta}_{t-1}) p(d_{t-1}, \boldsymbol{\omega}_{t-1}, \boldsymbol{\theta}_{t-1} | \mathbf{z}_0, \dots, \mathbf{z}_{t-1}) \quad (29) \\ &= \sum_{d_{t-1}} p(d_t | d_{t-1}, \boldsymbol{\omega}_t, \boldsymbol{\theta}_t) \sum_{\boldsymbol{\omega}_{t-1}} p(\boldsymbol{\omega}_t | d_{t-1}, \boldsymbol{\omega}_{t-1}, \boldsymbol{\theta}_t) \sum_{\boldsymbol{\theta}_{t-1}} p(\boldsymbol{\theta}_t | \boldsymbol{\theta}_{t-1}) p(d_{t-1}, \boldsymbol{\omega}_{t-1}, \boldsymbol{\theta}_{t-1} | \mathbf{z}_0, \dots, \mathbf{z}_{t-1}) \end{aligned}$$

Equations (28) and (29) establish the relationship between  $p(d_{t-1}, \boldsymbol{\omega}_{t-1}, \boldsymbol{\theta}_{t-1} | \mathbf{z}_1, \dots, \mathbf{z}_{t-1})$  and  $p(\boldsymbol{\theta}_t, \boldsymbol{\omega}_t, d_t | \mathbf{z}_1, \dots, \mathbf{z}_t)$ . Starting from  $p(d_1, \boldsymbol{\omega}_1, \boldsymbol{\theta}_1 | \mathbf{z}_0, \mathbf{z}_1)$ , it is thus possible to compute all  $p(\boldsymbol{\theta}_t, \boldsymbol{\omega}_t, d_t | \mathbf{z}_1, \dots, \mathbf{z}_t)$  for any  $t$  recursively.

In principle it is not necessary to determine the proportionality constant in Equation (28), but to prevent underflow it is recommended to normalize  $p(\boldsymbol{\theta}_t, \boldsymbol{\omega}_t, d_t | \mathbf{z}_1, \dots, \mathbf{z}_t)$  at each time step of the recursive algorithm.

If there is no evidence at a time  $t$ , the corresponding likelihood function can be set equal to one, i.e.,  $p(\mathbf{z}_t | d_t) = 1$ . This is equivalent to omitting Equation (28) in the forward operation.

## Backward computation

The backward-part of the algorithm computes the likelihood  $p(\mathbf{z}_{t+1}, \dots, \mathbf{z}_T | \boldsymbol{\theta}_t, \boldsymbol{\omega}_t, d_t)$ . By considering the independence assumptions encoded in the BN structure, we obtain the following relationships:

$$\begin{aligned}
& p(\mathbf{z}_{t+1}, \dots, \mathbf{z}_T | \boldsymbol{\theta}_t, \boldsymbol{\omega}_t, d_t) \\
&= \sum_{d_{t+1}} \sum_{\boldsymbol{\omega}_{t+1}} \sum_{\boldsymbol{\theta}_{t+1}} p(\mathbf{z}_{t+2}, \dots, \mathbf{z}_T, \boldsymbol{\theta}_{t+1}, \boldsymbol{\omega}_{t+1}, d_{t+1}, \mathbf{z}_{t+1} | \boldsymbol{\theta}_t, \boldsymbol{\omega}_t, d_t) \\
&= \sum_{d_{t+1}} \sum_{\boldsymbol{\omega}_{t+1}} \sum_{\boldsymbol{\theta}_{t+1}} p(\mathbf{z}_{t+2}, \dots, \mathbf{z}_T | \mathbf{z}_{t+1}, \boldsymbol{\theta}_{t+1}, \boldsymbol{\omega}_{t+1}, d_{t+1}, \boldsymbol{\theta}_t, \boldsymbol{\omega}_t, d_t) p(\boldsymbol{\theta}_{t+1}, \boldsymbol{\omega}_{t+1}, d_{t+1}, \mathbf{z}_{t+1} | \boldsymbol{\theta}_t, \boldsymbol{\omega}_t, d_t) \\
&= \sum_{d_{t+1}} \sum_{\boldsymbol{\omega}_{t+1}} \sum_{\boldsymbol{\theta}_{t+1}} p(\mathbf{z}_{t+2}, \dots, \mathbf{z}_T | \boldsymbol{\theta}_{t+1}, \boldsymbol{\omega}_{t+1}, d_{t+1}) p(\boldsymbol{\theta}_{t+1}, \boldsymbol{\omega}_{t+1}, d_{t+1} | \boldsymbol{\theta}_t, \boldsymbol{\omega}_t, d_t) p(\mathbf{z}_{t+1} | d_{t+1}) \\
&= \sum_{d_{t+1}} \sum_{\boldsymbol{\omega}_{t+1}} \sum_{\boldsymbol{\theta}_{t+1}} p(\mathbf{z}_{t+2}, \dots, \mathbf{z}_T | \boldsymbol{\theta}_{t+1}, \boldsymbol{\omega}_{t+1}, d_{t+1}) p(d_{t+1} | d_t, \boldsymbol{\omega}_{t+1}, \boldsymbol{\theta}_{t+1}) p(\boldsymbol{\omega}_{t+1} | d_t, \boldsymbol{\omega}_t, \boldsymbol{\theta}_{t+1}) p(\boldsymbol{\theta}_{t+1} | \boldsymbol{\theta}_t) p(\mathbf{z}_{t+1} | d_{t+1}) \\
&= \sum_{d_{t+1}} p(\mathbf{z}_{t+1} | d_{t+1}) p(d_{t+1} | d_t, \boldsymbol{\omega}_{t+1}, \boldsymbol{\theta}_{t+1}) \sum_{\boldsymbol{\omega}_{t+1}} p(\boldsymbol{\omega}_{t+1} | d_t, \boldsymbol{\omega}_t, \boldsymbol{\theta}_{t+1}) \sum_{\boldsymbol{\theta}_{t+1}} p(\boldsymbol{\theta}_{t+1} | \boldsymbol{\theta}_t) p(\mathbf{z}_{t+2}, \dots, \mathbf{z}_T | \boldsymbol{\theta}_{t+1}, \boldsymbol{\omega}_{t+1}, d_{t+1})
\end{aligned} \tag{30}$$

$p(\mathbf{z}_{t+1}, \dots, \mathbf{z}_T | \boldsymbol{\theta}_t, \boldsymbol{\omega}_t, d_t)$  is computed recursively, starting from  $p(\mathbf{z}_{T+1} | \boldsymbol{\theta}_T, \boldsymbol{\omega}_T, d_T) = 1$ . Typically, values of  $p(\mathbf{z}_{t+1}, \dots, \mathbf{z}_T | \boldsymbol{\theta}_t, \boldsymbol{\omega}_t, d_t)$  can become very small. To prevent underflow, it is recommended to normalize  $p(\mathbf{z}_{t+1}, \dots, \mathbf{z}_T | \boldsymbol{\theta}_t, \boldsymbol{\omega}_t, d_t)$  at every step, i.e., we multiply  $p(\mathbf{z}_{t+1}, \dots, \mathbf{z}_T | \boldsymbol{\theta}_t, \boldsymbol{\omega}_t, d_t)$  with a normalization factor  $\alpha_t$  in Equation (30) above.  $\alpha_t$  can be chosen freely, the results computed in this paper were obtained with

$$\alpha_t = \left[ \sum_{\boldsymbol{\theta}_t} \sum_{\boldsymbol{\omega}_t} \sum_{d_t} p(\mathbf{z}_{t+1}, \dots, \mathbf{z}_T | \boldsymbol{\theta}_t, \boldsymbol{\omega}_t, d_t) \right]^{-1} \tag{31}$$

## Computational performance of the algorithms

Let the number of states of  $d_t, \boldsymbol{\omega}_t, \boldsymbol{\theta}_t$  be  $m_d, m_\omega, m_\theta$ , respectively. The computation time for calculating  $p(\boldsymbol{\theta}_t, \boldsymbol{\omega}_t, d_t | \mathbf{z}_1, \dots, \mathbf{z}_t)$  is then  $O[(m_d^2 m_\omega + m_d m_\omega^2) m_\theta t]$ . To verify this result, consider the last line of Equation (29) and note that the computation is performed from right to left.  $O[m_d m_\omega^2 m_\theta t]$  time is spent in multiplying the elements of  $p(\boldsymbol{\omega}_t | d_{t-1}, \boldsymbol{\omega}_{t-1}, \boldsymbol{\theta}_t)$  with the terms on its right, and  $O[m_d^2 m_\omega m_\theta t]$  time is spent in multiplying the elements of  $p(d_t | d_{t-1}, \boldsymbol{\omega}_t, \boldsymbol{\theta}_t)$  with the terms on its right. The summation over  $\boldsymbol{\theta}_{t-1}$  does not have to be performed; since  $p(\boldsymbol{\theta}_t | \boldsymbol{\theta}_{t-1})$  takes value one if  $\boldsymbol{\theta}_t = \boldsymbol{\theta}_{t-1}$  and zero otherwise, it is sufficient to exchange  $\boldsymbol{\theta}_{t-1}$  with  $\boldsymbol{\theta}_t$ . It is noted that a simple Markov chain model would require a computation time of  $O(m_d^2 m_\omega^2 m_\theta^2 t)$ , which is orders of magnitude higher. Based

on similar consideration, the computation time required for the backward operation is determined as  $O[(m_d^2 m_\omega + m_d m_\omega^2) m_0 (T - t)]$ . It follows that the computation time for filtering is  $O[(m_d^2 m_\omega + m_d m_\omega^2) m_0 t]$ , whereas it is  $O[(m_d^2 m_\omega + m_d m_\omega^2) m_0 T]$  for predicting and smoothing.

## References

- Attoh-Okine, N.O. and S. Bowers, A Bayesian belief network model of bridge deterioration. *Bridge Engineering*, 2006. 159(2): p. 69-76.
- Beck, A.T. and R.E. Melchers, Overload failure of structural components under random crack propagation and loading - a random process approach. *Structural Safety*, 2004. 26(4): p. 471-488.
- Beck, J.L. and S.-K. Au, Bayesian Updating of Structural Models and Reliability using Markov Chain Monte Carlo Simulation. *Journal of Engineering Mechanics*, Trans. ASCE, 2002. 128(4): p. 380-391.
- Bogdanoff, J.L. and F. Kozin, *Probabilistic Models of Cumulative Damage*. 1985, New York: John Wiley & Sons.
- Cesare, M.A., et al., Modeling Bridge Deterioration with Markov Chains. *Journal of Transportation Engineering*, Trans. ASCE, 1992. 118(6): p. 820-833.
- Committee on Fatigue and Fracture Reliability of the Committee on Structural Safety and Reliability of the Structural Division, *Fatigue and Fracture Reliability: A State-of-the-Art Review*. *Journal of the Structural Division*, ASCE, 1982. 108(1): p. 3-88.
- Ditlevsen, O. and H.O. Madsen, *Structural Reliability Methods*. 1996: John Wiley & Sons.
- Faber, M.H., et al., Risk Assessment of Decommissioning Options Using Bayesian Networks. *Journal of Offshore Mechanics and Arctic Engineering*, 2002. 124(4): p. 231-238.
- Friis-Hansen, A., *Bayesian Networks as a Decision Support Tool in Marine Applications*, in Department of Naval Architecture and Offshore Engineering. 2001, PhD thesis, Technical University of Denmark.
- Friis-Hansen, P. Structuring of complex systems using Bayesian network. in *JCSS Workshop*. 2004. DTU, Lyngby, Denmark.
- Gilks, W.R., S. Richardson, and D.J. Spiegelhalter, *Markov chain Monte Carlo in practice*. 1st ed. 1996, London ; New York: Chapman & Hall. xvii, 486.
- Grêt-Regamey, A. and D. Straub, Spatially explicit avalanche risk assessment linking Bayesian networks to a GIS. *Natural Hazards and Earth System Sciences*, 2006. 6(6): p. 911-926.
- Hrycej, T., Gibbs sampling in Bayesian networks. *Artificial Intelligence*, 1990. 46(3): p. 351-363.
- Ishikawa, H., et al., Reliability assessment of structures based upon probabilistic fracture mechanics. *Probabilistic Engineering Mechanics*, 1993. 8(1): p. 43-56.
- Jensen, F.V., *Bayesian Networks and Decision Graphs*. 2001, New York: Springer.
- Langseth, H. and L. Portinale, Bayesian networks in reliability. *Reliability Engineering & System Safety*, 2007. 92(1): p. 92-108.
- Lin, Y.K. and J.N. Yang, A Stochastic-Theory of Fatigue Crack-Propagation. *Aiaa Journal*, 1985. 23(1): p. 117-124.
- Madsen, H.O., S. Krenk, and N.C. Lind, *Methods of Structural Safety*, ed. W.J. Hall. 1986, New Jersey: Pentice-Hall, Inc., Englewood Cliffs, New Jersey 07632. 403.

- Melchers, R.E., Corrosion Uncertainty Modelling for Steel Structures. *Journal of Constructional Steel Research*, 1999. 52: p. 3–19.
- Mishalani, R.G. and S.M. Madanat, Computation of Infrastructure Transition Probabilities Using Stochastic Duration Models. *Journal of Infrastructure Systems*, Trans. ASCE, 2002. 8(4): p. 139-148.
- Montes-Iturrizaga, R., et al., Risk Based Structural Integrity Management Using Bayesian Probabilistic Nets. *Journal of Offshore Mechanics and Arctic Engineering Transactions of the ASME*, 2009. 131(1).
- Murphy, K., *The Bayes Net Toolbox for Matlab*. Computing Science and Statistics, 2001. 33.
- Murphy, K.P., *Dynamic Bayesian networks: representation, inference and learning*. 2002. p. 268.
- Neil, M., M. Taylor, and D. Marquez, Inference in hybrid Bayesian networks using dynamic discretization. *Statistics and Computing*, 2007. 17(3): p. 219-233.
- Ortiz, K. and A.S. Kiremidjian, Stochastic modeling of fatigue crack growth. *Engineering Fracture Mechanics*, 1988. 29(3): p. 317-334.
- Pearl, J., *Probabilistic reasoning in intelligent systems : networks of plausible inference*. The Morgan Kaufmann series in representation and reasoning. 1988, San Mateo, Calif.: Morgan Kaufmann Publishers.
- Pedersen, C., et al., Reliability Based Inspection Planning for the Tyra Field. *Journal of Offshore Mechanics and Arctic Engineering Transactions of the ASME*, 1992. 2: p. 255-263.
- Rocha, M.M. and G.I. Schuëller, A probabilistic criterion for evaluating the goodness of fatigue crack growth models. *Engineering Fracture Mechanics*, 1996. 53(5): p. 707-731.
- Russell, S.J. and P. Norvig, *Artificial intelligence: a modern approach*. 2nd ed. 2003, Englewood Cliffs, N.J.: Prentice-Hall.
- Sindel, R. and R. Rackwitz, Problems and Solutions Strategies in Reliability Updating. *Journal of Offshore Mechanics and Arctic Engineering*, Transactions of ASME, 1998. 120: p. 109-114.
- Spencer, B.F. and J. Tang, A Markov Process Model for Fatigue Crack Growth. *Journal of Engineering Mechanics*, Trans. ASCE, 1988. 114(12): p. 2134-2157.
- Straub, D. and M.H. Faber, Computational aspects of risk-based inspection planning. *Computer-Aided Civil and Infrastructure Engineering*, 2006. 21(3): p. 179-192.
- Straub, D. and M.H. Faber, Temporal Variability in Corrosion Modeling and Reliability Updating. *Journal of Offshore Mechanics and Arctic Engineering*, Trans. ASME, 2007. 129(4): p. 265-272.
- Tang, W.H., Probabilistic Updating of Flaw Information. *Journal of Testing and Evaluation*, 1973. 1(6): p. 459-467.
- Thoft-Christensen, P. and J.D. Sørensen, Optimal Strategy for Inspection and Repair of Structural Systems. *Civil Engineering Systems*, 1987. 4: p. 94-100.
- Yang, J.N. and S.D. Manning, A simple second order approximation for stochastic crack growth analysis. *Engineering Fracture Mechanics*, 1996. 53(5): p. 677-686.
- Yang, J.N. Application of reliability methods to fatigue, quality assurance and maintenance. in *Proceedings of the 6th International Conference of Structural Safety & Reliability 1994*. Balkema Rotterdam.
- Zheng, R.H. and B.R. Ellingwood, Stochastic fatigue crack growth in steel structures subject to random loading. *Structural Safety*, 1998. 20(4): p. 303-323.