Towards Scalable k-out-of-n Models for Assessing the Reliability of Large-scale Function-as-a-Service Systems with Bayesian Networks

Otto Bibartiu Frank Dürr Kurt Rothermel
University of Stuttgart
Institute for Parallel and Distributed Systems (IPVS)
Stuttgart, Germany
{first name}.{last name}@ipvs.uni-stuttgart.de

Beate Ottenwälder Andreas Grau
Robert Bosch GmbH
Bosch IoT Cloud
Stuttgart, Germany
Beate.Ottenwaelder@de.bosch.com
Andreas.Grau2@de.bosch.com

***Abstract*—Typically, Function-as-a-Service (FaaS) involves state-less replication with very large numbers of instances. The reliability of such services can be evaluated using Bayesian Networks and k-out-of-n models. However, existing k-out-of-n models do not scale to the larger number of hosts of FaaS services. Therefore, we propose a scalable k-out-of-n model in this paper with the same semantics as the standard k-out-of-n voting gates in fault trees, enabling the reliability analysis of FaaS services.**

I. Introduction

The reliability of a Function-as-a-Service (FaaS) runtime environment, can be modeled by a k-out-of-n (k:n) reliability model with Bayesian networks (BNs). However, as we will see later, due to the very large number of hosts in server-less computing environments, state-of-the-art reliability models become intractable to model such large systems. For example, suppose we have a FaaS environment with hundreds of hosts, and we represent the failure probability of each host by a binary random variable. Then the conditional probability table of a k:n model, defined by the converging node K in Figure 1, would have 2^{100} entries.

To tackle this scalability problem, we provide an efficient BN implementation of k:n models in this paper. We call this model the *scalable* k:n model for BNs. The scalable k:n model enables the reliability modeling of large FaaS environments. We exploit the causal independence of the parent nodes from the converging BN structure by showing that the k:n formalism is an augmented instance of the temporal representation of the noisy adder model as proposed by Heckerman [1]. In detail, we make the following contributions: (1) we provide a scalabe k:n model for BNs to assess the availability of large-scale server-less computing environments, (2) we show that once the scalable k:n model is implemented, we can perform inference with any standard exact or approximate inference algorithm.

The remainder of the paper is structured as follows: First, we describe necessary background information on BNs in Section II. Afterwards, in Section III, we outline how to construct the scalable k:n gate by exploiting causal independence of the k:n semantic with help of the temporal transformation of the noisy adder. Finally, we conclude the paper in Section IV.

We gratefully acknowledge the support of this PhD project by Robert Bosch GmbH within the Doctoral Academy.

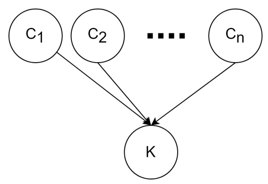

Fig. 1: Naive k:n model representation as Bayesian network

II. Background: Bayesian Networks

A (discrete) BN is a directed acyclic graph $G = (X, E)$, where the set $X = \{X_1, X_2, \ldots X_n\}$ represents discrete random variables and the set $E \subset X \times X$ models directed edges representing the direct cause-effect, i.e. influence between random variables. An edge is a tuple $(X_i, X_j) \in E$, where X_i is said to be a parent variable of X_i, and X_j is said to be a child variable of X_i.

The BN structure in Figure 1 is also called a common effect or *converging* network, due to the cause-effect relation modeled by edges between parent and child nodes. Random variables are visualized as nodes in the graph, thus, we will refer to random variables and their node representations simply as variables or nodes interchangeably. For instance, in Figure 1, node K represents a common effect of the nodes C_1 to C_n representing the causes. We encode the probability distribution of a variable by its conditional probability distribution $P(X_i|pa(X_i))$ given their parent nodes pa(X_i) as a conditional probability table (CPT). Finally, the BN structure defines the full joint probability distribution over X with $P(X) = \prod_{x \in X} P(x|\text{pa}(x))$, which is the product of its conditional probability distributions over all variables. Since the BN encodes all information about the joint distribution, it can be used to infer answers to any query on its random variables.

Once we have constructed the BN, we can perform inference on probabilistic queries to compute the posterior probability

 https://doi.org/10.1109/CLOUD.2019.00095

distribution $P(Q|K)$ for a subset of random variable $Q \subset X$ given a set of observed states of other random variables $K \subset X \backslash Q$.

However, converging BN structures, such as the BN in Figure 1, pose a challenge in terms of CPT size, which can make inference intractable due to space limitations. The CPT size grows exponentially in the number of parent nodes since we need to provide a conditional probability distribution for each parent state permutation.

III. Scalable k-out-of-n Model

The main problem is the space complexity of the node K in Figure 1. Node K has an exponential CPT size making inference intractable for large n. Therefore, we need to find a scalable BN formalism for the k:n model such that (1) it has the same semantics as naive k:n model and (2) it scales in the number of hosts w.r.t to space complexity.

A. Preliminaries: Temporal Noisy Adder Model

Our work is based on BN representations available in fault tree (FT) analysis [2]. Bobbio et al. [3] have proposed an algorithm to transform FTs into BNs. They show that the BN formalism has more modeling power than FTs. However, they only considered naive BN implementations of the AND, OR, and k:n voting gates, which lead to inference intractability for large numbers of events per gate.

For many expert systems, knowledge engineers were able to exploit *causal independence* to provide scalable BN structures and to ease knowledge acquisition [1], [4]. Important models are the noisy OR [4]–[6], noisy MAX [7], and the noisy adder [8], [9]. Our scalable k:n model is strongly related to the temporal definition of the noisy adder by Heckerman [1].

In general, converging BN structures have CPTs whose size grows exponentially with the number of parent nodes. However, sometimes the CPT of the converging node contains a special structure that can be exploited to enable inference without exponential growth of the CPT, know as causal independence. In our case, we use the *temporal model* of the noisy adder by Heckerman [1] to realize our scalable k:n BN structure.

The noisy adder is defined as follows. Suppose we have the converging BN structure as shown in Figure 1, with n binary random variables C_1 to C_n with state true and false, that influence the random variable K, where K represents a counter with the domain $[0, n]$. If C_i has the state true, it adds one to the counter with probability q_i, hence the name noisy adder. In this work, we only consider the case $q_i = 1$, thus the counter is always increased by 1 whenever a variable C_i is true. If a variable C_i is false, it does not change the value of the counter. With this model, the overall influence of the parent nodes is combined pairwise via a *contribution* variable E_i, which has the domain $[0, i]$, to model the causal impact of the variables C_1 to C_n on the variable $K = E_n$. Heckerman defines the CPT of the contribution variables as follows:

$$\begin{aligned} \forall i \in [1, n] : & P(E_i = k+1 | E_{i-1} = k, C_i = true) = 1 \\ & P(E_i = k | E_{i-1} = k, C_i = false) = 1 \end{aligned} \tag{1}$$

where we *increment* the contribution variable E_i by one, given the last state, i.e. last count of the previous contribution variable E_{i-1} given C_i is in state true. If C_i is false, the contribution variable E_i propagates the state of his predecessor E_{i-1}. Here, the CPT of E_0 is set to $P(E_0 = 0) = 1$.

According to Heckerman, the complexity for inference is $O(n^3 \times p)$ for the general noisy adder [1], which implies for our case with $q_i = p = 1$ that inference is performed in $O(n^3)$.

Next we show how to augment the noisy adder model to implement a scalable k:n model for BNs.

B. Scalable k:n Model Construction

As described by Bobbio et al. [3], the k:n voting gate can be represented by a converging BN structure where the CPT of the converging node, say K, is a binary random variable representing, for example, the availability of a system where at least k components are available. In contrast to the FT formalism of the k:n voting gate, we consider w.l.o.g. the parent nodes C_1 to C_n as binary random variables with the states $\{true, false\}$, where true refers to the availability of a component, i.e. (virtual) host, with the k:n semantic that at least k faults lead to a service failure.

First, we describe the naive k:n model and then transform the model step-by-step into the scalable k:n model. The CPT of node K in the naive k:n BN model is defined as follows:

$$P(K = true | c_1, \ldots, c_n) = \begin{cases} 1 & \sum_{i=1}^{n} \mathbf{1}_{true}(c_i) \geq k \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

where $\mathbf{1}_{true}(x)$ denotes an indicator function such that

$$\mathbf{1}_{true}(x) := \begin{cases} 1 & \text{if } x = true, \\ 0 & \text{otherwise.} \end{cases}$$

Based on Eq. 2, we implement the k:n semantic by computing the posteriori probability distribution $P(K = true)$ by marginalizing variable K. Using the BN formalism of the joint probability, we get the following marginal distribution for K:

$$\begin{aligned} & P(K = true) = \\ & \sum_{c_1, \ldots, c_n} P(K = true | c_1, \ldots, c_n) P(c_1) \ldots P(c_n) \end{aligned} \tag{3}$$

The conditional probability $P(K = true | c_1, \ldots, c_n)$ is 1 if at least k of the parent variables are true, and 0 otherwise. Therefore, the posterior probability is the summation of those products of parent variables $P(c_1) \times \cdots \times P(c_n)$ where at least k parents are true. If the parent variables are not independent, we can change the product $P(c_1) \times \cdots \times P(c_n)$ to the joint probability distribution $P(c_1 \times \cdots \times c_n)$ denoting that there are other dependencies between the causes.

With the notion of causal independence, we define a distinguished state, say false, as a neutral event that does not influence variable K. In Eq. 2, K is only influenced by those variables that are in the state true, i.e., K only depends on the sum of those parent variables that are true. Thus, we can rewrite the conditional probability distribution of $P(K = true | c_1, \ldots, c_2)$ as $P(K = true | N = n)$ where K is conditionally dependent on a random variable N with the domain $[0, n]$ representing the number of parent variables that are in state true. Variable N has the following CPT:

$$P(N = m|c_1, \dots, c_n) = \begin{cases} 1 & \sum_{i=1}^{n} \mathbf{1}_{true}(c_i) = m \\ 0 & \text{otherwise.} \end{cases} \quad (4)$$

Since N is a random variable that is influenced by the summation of the states of its parent variables, we can implement N as an adder that is semantically equivalent to the noisy adder model from Section III-A.

We redefine the probability distribution $P(K = true)$ by introducing variable N:

$$P(K = true) = \sum_{c_1,\dots,c_n} \sum_{m\in[0,n]} P(K = true|N = m) \times P(N = m|c_1, \dots, c_n) \times P(C_1 = c_n) \dots P(C_2 = c_2) \quad (5)$$

where we substitute the initial conditional probability distribution of K, which was previously influenced by the variables C_1 to C_n, by a conditional probability distribution influenced by the counter N. Consequently, the CPT of K has linear table size. However, the problem of an exponential table size of the variable K has shifted to variable N now, due to its conditional dependency on variables C_1 to C_n. Next, we rearrange the summations to isolate the counting variable N, to apply the temporal transformation of the adder model. In the next step, we define

$$P(K = true) = \sum_{m\in[0,n]} P(K = true|N = m) \times \sum_{c_1,\dots,c_n} \underbrace{P(N = m|c_1, \dots, c_n)}_{B} \times P(c_1) \dots P(c_n) \quad (6)$$

where we substitute the conditional probability distribution of N by the temporal definition of the adder model in B, where B is defined as follows:

$$B = P(E_0 = 0) \sum_{\substack{e_1,\dots \\ e_n=m}} \prod_{i\in[1,n]} P(E_i = e_i|E_{i-1} = e_{i-1}, C_i = c_i)$$

The last term defines the noisy adder model w.r.t N, as proposed by Heckerman [1]. The resulting BN structure of Eq. 6 is shown in Figure 2. Variable K is now a child node of the last contribution variable E_n in the causal chain. Since E_n represents the total count of the causes C_1 to C_n with the states true, K enforces the k:n semantic by setting the conditional probability for K = true when the count is a least k with $P(K = true|N \geq k) = 1$, and $P(K = true|N < k) = 0$ otherwise.

The overall space complexity of our scalable k:n model is $O(n^3)$. A contribution variable E_i can have up to n states, and it is conditionally dependent on its previous contribution variable $E_i - 1$, which has up to $n - 1$ states. Thus, the size of the CPT grows as $O(n^2)$. Since we have n contribution variables, we have a total CPT size of the order $O(n^3)$. Therefore, we decreased the space complexity from exponential to polynomial by changing the structure of the BN network, thus inference can be still performed with any standard exact or approximate inference algorithm.

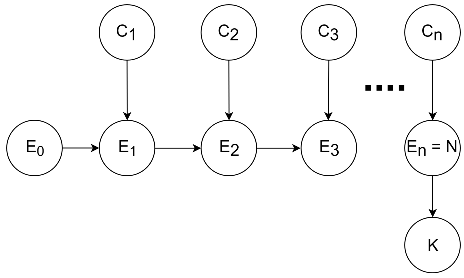

Fig. 2: Scalable BN structure of the k:n gate: The structure is composed of the temporal representation of the noisy adder and a deterministic random variable K that represents a binary relation operator, e.g. the greater-than-or-equal operator

IV. Conclusion and Future Work

In this paper, we proposed a scalable k:n reliability model to enable the modeling of large FaaS with BNs. In future work, we will show the practicality of modeling the reliability of FaaS and other cloud services with our scalable k:n BN structure.

References

[1] D. Heckerman, "Causal independence for knowledge acquisition and inference," in *Uncertainty in Artificial Intelligence, 1993*, pp. 122–127, Elsevier, 1993.

[2] E. Ruijters and M. Stoelinga, "Fault tree analysis: A survey of the state-of-the-art in modeling, analysis and tools," *Computer science review*, vol. 15, pp. 29–62, 2015.

[3] A. Bobbio, L. Portinale, M. Minichino, and E. Ciancamerla, "Improving the analysis of dependable systems by mapping fault trees into bayesian networks," *Reliability Engineering & System Safety*, vol. 71, no. 3, pp. 249–260, 2001.

[4] J. Kim and J. Pearl, "A computational model for causal and diagnostic reasoning in inference systems," in *International Joint Conference on Artificial Intelligence*, pp. 0–0, 1983.

[5] J. Pearl, "Probabilistic reasoning in intelligent systems: Networks of plausible reasoning," *Morgan Kaufmann Publishers, Los Altos*, 1988.

[6] K. Zhou, A. Martin, and Q. Pan, "The belief noisy-or model applied to network reliability analysis," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 24, no. 06, pp. 937–960, 2016.

[7] F. J. Diez, "Parameter adjustment in bayes networks. the generalized noisy or–gate," in *Uncertainty in Artificial Intelligence, 1993*, pp. 99–105, Elsevier, 1993.

[8] P. Dagum and M. Luby, "Approximating probabilistic inference in bayesian belief networks is np-hard," *Artificial intelligence*, vol. 60, no. 1, pp. 141–153, 1993.

[9] P. Dagum and A. Galper, "Additive belief-network models," in *Uncertainty in Artificial Intelligence, 1993*, pp. 91–98, Elsevier, 1993.