Pseudo credal networks for inference with probability intervals

Hector Diego Estrada-Lugo
PhD student
Institute for Risk and Uncertainty, University of Liverpool, Chadwick building, L69 7ZF, Liverpool, UK
Email: h.d.estrada-lugo@liverpool.ac.uk

Silvia Tolo
Post Doctoral Research Associate
Institute for Risk and Uncertainty, University of Liverpool, Chadwick building, L69 7ZF, Liverpool, UK
Email: S.Tolo@liverpool.ac.uk

Marco de Angelis
Post Doctoral Research Associate
Institute for Risk and Uncertainty, University of Liverpool, Chadwick building, L69 7ZF, Liverpool, UK
Email: marco.de-angelis@liverpool.ac.uk

Edoardo Patelli
Senior Lecturer
Institute for Risk and Uncertainty, University of Liverpool, Chadwick building, L69 7ZF, Liverpool, UK
Email: Edoardo.Patelli@liverpool.ac.uk

ABSTRACT

The computation of the inference corresponds to an NP-hard problem even for a single connected credal network. The novel concept of *pseudo networks* is proposed as an alternative to reduce the computational cost of probabilistic inference in credal networks.

The method allows identifying the combination of intervals that optimizes the probability values of each state of the queried variable from the credal network. In the case of no evidence, the exact probability bounds of the query variable are calculated. When new evidence is inserted into the network, the outer and inner approximation of the query variable are computed by means of the marginalization of the joint probability distributions of the pseudo networks. The applicability of the proposed methodology is shown by solving numerical case studies.

1 Introduction

A Bayesian network is an effective probabilistic graphical model to study and analyze the natural dependencies of elements that constitute a complex system. Originally, this technique was developed as a tool for artificial intelligence as it was introduced in the paper presented at the AAAI National Conference on Artificial Intelligence in 1982 by Judea Pearl [1]. The probabilistic graphical model called Bayesian Network simplifies the variable conditional representation as well as the explanation of stochastic process outcomes and decision making under uncertainty [2]. There is a large variety of applications with satisfactory results in various fields from decision making to risk assessments, e.g. [3,4,5,6]. The graphical representation of the system facilitates the understanding of the problem even for non-expert personnel [7]). The graph of the network is made of nodes and edges representing the components of the system and their dependencies, respectively. The posterior probabilities obtained through the built network are based on the application of probability theory concepts, (joint and conditional probabilities, random variables, etc.), that provide quantitative support to the graphical representation of the model, [8].

Bayesian Networks characterize the uncertainty associated with the variables in the model through precise discrete distributions. Enhanced Bayesian Networks extend the characterization of uncertainty to continuous probability distributions [9, 10]. However, these approaches constraint the quantification of the epistemic uncertainty of the information available to model the system [11]. Credal networks follow the same graphical structure of their Bayesian counterpart but utilize a generalized model of uncertainty called credal set to represent probability mass functions. In this paper, a special class of credal set known as probability intervals is adopted to represent the epistemic uncertainty attached to a variable. The selection of such intervals to represent the nodes in the network makes the computation of posterior probabilities considerably harder.

Tractability of Credal networks is itself NP-hard even for simple networks. In practice, this means that the time to calculate a posterior probability, known as *inference computation*, increases exponentially with the number of nodes, as it does in Bayesian networks. Small models, with just about ten nodes, are suitable to use analytical methods to compute a posterior. However, for networks with several more tens of nodes or with a large number of connections, approximate algorithms must be used to save computational costs [2].

Pure analytical inference methods for Credal Networks are restricted to the use of relatively small networks due to the computational cost of the inference estimation, [12]. Cano et al. proposed a variable elimination method to evaluate the inner nodes in the search by using the branch-and-bound algorithm. This technique is reliable, however, computationally expensive for moderately large networks [13]. Antonucci and Cuzzolin, [14] suggested an outer approximation algorithm introducing non-optimal states in the extreme points of the Credal set.

A novel approach is here presented in order to reduce the inference computational time by reducing the search space to a small number of *pseudo networks*. The method allows finding the exact bounds of the queried variable in case of evidence present otherwise it produces an outer approximation from which an inner approximation can be identified in the case of evidence inserted. The approach works with any exact inference algorithm (e.g., joint tree, marginalization) that are commonly included on any Bayesian network toolbox.

The paper is organized as follows. Firstly, a brief explanation of Bayesian networks is given in section 2.1. Then, in section 2.2 credal networks and credal sets are described. The inference computation process is explained in section 2.3. The proposed method is detailed in section 3. Then, two numerical examples regarding a fire protection system and a railway system are explained. The results and discussions are given in sections 4 and 4.3 followed by the conclusions in section 5.

2 Theoretical background

In this section, the concepts regarding Bayesian and Credal networks and associated inference approaches are presented.

2.1 Bayesian networks

In Bayesian Networks random variables are used to model the nodes in the network, providing information about observable quantities or hypothesis of stochastic problems [15]. There are two main types of nodes in Bayesian networks, *parents* and *children*. A node is a child of a parent only if there is a direct connection, via an edge, from the latter to the former node. This connection represents, qualitatively, the child dependency of the parent. If a node has no parents it is known as a *root* node, which can be regarded as an original cause. On the other hand, nodes with no descendants are known as *leafs* (by referring to a "tree" analogy), representing a final effect [2].

The connections that a node maintains with the rest of the network constitute the key structure named, *Markov blanket*. Such arrangement consists of the necessary variables to forecast the behaviour of a variable of interest, obeying to the Markov property (given its parents, a variable is conditionally independent of its non-descendant non-parents). Dependencies among the variables in the network can be represented mathematically by the *Joint Probability Distribution*, $P(\mathbf{X})$, that contains the probability measures of each of the states of the variables in the Markov blanket [8]. Given a set of random variables, $X_i := X_1, X_2, ..., X_n$, that belongs to a given Bayesian network, the joint distribution of a particular value is given as $P(x_i) := P(X_1 = x_1, X_2 = x_2, ..., X_n = x_n)$, where x_i represents a specific probabilistic value of the random variable X_i. The expression $P(x_i)$, can be factorized by $P(\mathbf{X})$ using the chain rule (product of all the probability values in the network) for Bayesian networks as follows:

$$P(\mathbf{X}) = \prod_{i=1}^{n} P(x_i|\pi_i) \tag{1}$$

Here, $P(x_i|\pi_i)$ is known as the probability of x_i conditioned to its parents π_i. This term represents the *probability mass function* of the conditioned random variable X_i. Probability distribution functions shown on Eqn. 1, need to be determined before any further calculation is done. Thus, they receive the name of prior probabilities.

In the case of Bayesian networks such priors are stored in arrangements known as *Conditional Probability Tables* (CPT). In this way, the use of the information contained inside the CPTs as inputs in the joint distribution and, in combination with the Bayes' theorem, concede the capability of computing the posterior probability of any single or complex event modelled in the network. This process is known as *belief updating* or *probabilistic inference* and constitutes the warhorse of the predictive and diagnostic analyses as well as the *what-if* scenarios that can be produced with the Bayesian networks [16]. Further information about this matter will be treated later on section 2.3.

2.2 Credal networks

One of the main drawbacks of Bayesian networks is their restriction to work with precise probabilities, either discrete or continuous. Adoption of precise variables leaves aside the epistemic uncertainty attached to the quality of the information available. The use of interval probabilities has become a powerful tool to cope with the lack of data, contradictory judgments or imperfection in the beliefs of a decision-maker. As a response to these limitations, it has been proposed the use of interval probabilities in the Bayesian approach. However, there are two main problems with this approach [17]. The first has to be with the impossibility to directly apply the Bayes' rule to interval-valued probability measures hoping to obtain an interval-valued posterior [18]. However, the joint probability distribution can be separated into bounds and compute inference separately for each of the bounds. The second problem is the no-uniqueness of defining independence for interval probabilities [19].

Credal Networks are generalization of the Bayesian networks with relaxed parameters while preserving the same graphical structure and probabilistic characteristics [20]. However, probability measurements are not represented by probability mass functions ($P(X)$) but by *credal sets*. The credal set is an imprecise-probabilistic model strictly stated as the convex hull of a finite set of probability mass functions. The credal structure is extended to the conditional $K(x_i|\pi_i)$ and joint credal sets, each of them differently valued depending on their extreme points as presented in the first formalization of credal networks by Cozman [17]. A conditional credal set is formed by set of conditional densities $P(x_i|\pi_i)$, while a joint credal set $K(x_i)$ is defined as the convex hull of the joint probability distributions for all of the extreme points contained in the local credal sets [17]. This is known as the *strong extension*, the most popular type of credal network due to its similarity with normal Bayesian networks. It is represented as follows,

$$K(x_i) := CH\left\{P(\mathbf{X})\middle|P(\mathbf{x}) = \prod_{i=1}^{n} P(x_i|\pi_i)\right\} \tag{2}$$

Here, x_i is in the domain Ω_{X_i} and π_i is in Ω_{Π_i}, of the variables that form in the system. Each $P(x_i|\pi_i)$ is element of the extreme point $ext[K(X_i|\pi_i)]$. The strong extension is then, the largest joint credal set in which the extreme points contained in the set are associated with a quasi-Bayesian network described as in Eqn. 1. In fact, a credal network can be considered as a finite set of quasi-Bayesian networks, each of them, associated with different extreme points but preserving the same graphical structure.

An *extreme point* of a credal set, $ext[K(X)]$, is defined as that that cannot be given as a convex combination of another points in $K(X)$ [21, 22]. The sum of the probability values for all states of variable X must be 1, i.e. must be normalized. More strictly, any probability measures ($P(x_i)$ and $P(x_j)$ with $i \neq j$) forming the intervals ($[\underline{P}(x_i), \overline{P}(x_i)]$ and $[\underline{P}(x_j), \overline{P}(x_j)]$) of the extreme points must be consistent with the convex set in the $\mathbb{R}^n$ space and therefore meet the following conditions [23],

$$\overline{P}(x_i) + \sum_{\substack{j=1 \\ j\neq i}}^{n} \underline{P}(x_j) \leq 1 \quad \forall \quad i \in \mathbb{R}^n \tag{3}$$

$$\underline{P}(x_i) + \sum_{\substack{j=1 \\ j\neq i}}^{n} \overline{P}(x_j) \geq 1 \quad \forall \quad i \in \mathbb{R}^n \tag{4}$$

According to imprecise probability theory, the posterior probability of a credal network is part of the credal set (Eqn. 2) expressed as an interval probability. In the strong extension of credal networks, the lower and upper bounds of an event of interest (queried variable, $P(x_q)$) are obtained by minimizing and maximizing, respectively, the marginal probability of x_q from Eqn. 1. This is,

$$\underline{P}(x_q) = \min_{P(X_i|\pi_i)\in K(X_i|\pi_i)} \sum_{x_1,\ldots,x_n \setminus x_q} \prod_{i=0}^{n} P(x_i|\pi_i) \tag{5}$$

$$\overline{P}(x_q) = \max_{P(X_i|\pi_i)\in K(X_i|\pi_i)} \sum_{x_1,\ldots,x_n \setminus x_q} \prod_{i=0}^{n} P(x_i|\pi_i) \tag{6}$$

This approximation represents a non-linear optimization problem where the joint probability mass function $P(x_i|\pi_i)$ is a multi-linear objective function. This topic will be discussed in the coming sections. These concepts are used to compute inference with interval probabilities that, although it increases the complexity level, they preserve natural correspondence with the traditional Bayesian networks. The objective of this work, is to used the interval bounds representation of the strong extension to carry out the inference computation by using analytical algorithms resulting in an outer approximation of the queried variable. While an approximate but effective method is used to provide the inner approximation of such queries.

2.3 Probabilistic inference

Probabilistic inference or belief updating consists on the calculation of a posterior probability distribution of a node of interest, x_q, by using prior probabilities, $\{P(x_e|x)P(x)\}$, and evidence, x_e (although not necessarily), in any other nodes [2]. This procedure allows to adapt the network to new conditions and provide outputs according to such updates. Mathematically, the inference computation in Bayesian networks resides on the application of Bayes' theorem (Eqn. 7) over the joint probability $P(x_q, x_e)$, [8].

$$P(x_q|x_e) = \frac{P(x_e|x_q)P(x_q)}{P(x_e)} \tag{7}$$

In this equation $P(x_q|x_e)$ represents the posterior probability conditioned to some evidence. This approach is especially useful when the query variable is conditioned to another one (e.g., in diagnosis analyses).

Computation of $P(x_q)$ by the summation of the product of all possible combinations of a local joint probability is known as a *marginalization* process. This concept is closely related to the probabilistic inference over a queried variable x_q so that, $P(x_q)$ can be obtained by marginalizing out of the joint distribution (Eqn. 1) the variables different than x_q. This method is NP-hard since computational time increases exponentially with the number of nodes and connections in the network.

As an example, suppose the variable elimination method to infer the probability $P(x_q)$ of the queried variable x_q. The marginalization of the non-queried variables out of the joint probability $P(\mathbf{X})$ withing the variables universe U is given as,

$$P(x_q) = \sum_{U\setminus(x_q)}^{n} \prod_{i=1} P(x_i|\pi_i) \tag{8}$$

In the case of the queried probability, $P(x_q)$, conditioned to a given evidence, x_e, Eqn. 7 is modified as shown in Eqn. 9.

$$P(x_q|x_e) = \frac{\sum_{U\setminus(x_q)}^{n} P(x_i, x_e)}{P(x_e)} \tag{9}$$

Inference computation in credal networks is carried out over the lower and upper bounds, like shown in Eqn. 6, in an extension of the network. This task is now a combinatorial optimization due to the number of combinations that need to be tested in order to provide an output, [12]. Exact inference methods can be used to find the bounds of the queried variables. However, in cases of large networks, the use of approximations from the inside of the real bounds (inner approximation) and the outside (outer approximation) seems to be the feasible option [24]. This work introduces an efficient inference method providing outer and inner approximation of queried variables when no evidence is inserted, and exact bounds when evidence is present.

3 Methodology

3.1 Pseudo networks

This introduces the novel concept of *pseudo network*: a discrete network containing a particular combination of all the upper (or lower) bounds from the initial intervals of the credal network. Preliminary results were presented by some of the authors in [25]. The result is considered as *pseudo network* since the elements inside such array do not represent a real combination of the original interval bounds but it preserves the dependency among the variables in the joint distribution. In other words, there is no such element inside the pseudo network that corresponds to any convex combination of the initial credal sets $K(X)$.

Pseudo networks allow to reduce the complexity of inference calculation since the number of interval combinations is restricted to the number of crisp networks that actively contribute to the calculation of the query bounds, i.e. the crisp networks with probability values lying within the interval queried variable are hence discarded. This provides a significant reduction on the computational time required to compute the inference of a queried node.

The joint probability bounds for each pseudo network are defined as:

Definition 1. *The joint probability distributions $P(\boldsymbol{X})$ of a pseudo network, corresponds to the union over all the variables m in the network (since all variables are different in the network). The complete joint distribution contains the combination of the upper (or lower) bounds of the states n of each of the variables. Since each state is a mutually exclusive event of that variable, their combination is regarded as the union of the n states.*

The joint probability distribution can be written as follows,

$$P(\mathbf{X}) = \left\{ \bigcup_{n} \left[\bigcup_{m} P(x_n^m) \right] \right\} \tag{10}$$

Two joint probabilities are found, one containing all the upper bounds $\overline{P}(x_i)$ and the other containing all the lower bounds of each of the variable states $\underline{P}(x_i)$. Each element inside the array in Eqn. 10 corresponds to the *pseudo network* for each possible combination for the upper (or lower) bounds. All the components of the array can be computed with traditional inference algorithms (e.g. variable elimination, junction tree, etc).

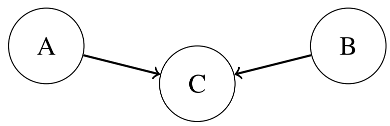

Fig. 1. A very simple credal network.

Let's consider a simple credal network as shown in Fig. 1. The network is formed by three nodes, two parents (A and B) and a child node (C). The nodes represent binary variables (i.e. only two states are allowed). The intervals of each of the variables, e.g., $\underline{P}(a_i) \leq P(a_i) \leq \overline{P}(a_i)$, are represented as $P(\overline{\underline{a_i}}) = [P(\underline{a_i}), P(\overline{a_i})]$. The term $P(\overline{A}) = \{p(\overline{a_1}), p(\overline{a_2})\}$ indicates the upper probability bounds of the states a_1 and a_2 of the event A, respectively. Following the definition of the *pseudo network* in Eqn.10, the array containing the combinations of the upper bounds is,

$$P(\overline{A}, \overline{B}, \overline{C}) = \left\{ \begin{matrix} [P(\overline{a_1}), P(\overline{b_1}), P(\overline{c_1})] & [P(\overline{a_1}), P(\overline{b_2}), P(\overline{c_1})] \\ [P(\overline{a_2}), P(\overline{b_1}), P(\overline{c_1})] & [P(\overline{a_2}), P(\overline{b_2}), P(\overline{c_1})] \\ [P(\overline{a_1}), P(\overline{b_1}), P(\overline{c_2})] & [P(\overline{a_1}), P(\overline{b_2}), P(\overline{c_2})] \\ [P(\overline{a_2}), P(\overline{b_1}), P(\overline{c_2})] & [P(\overline{a_2}), P(\overline{b_2}), P(\overline{c_2})] \end{matrix} \right\} \tag{11}$$

The computation of a queried variable namely, $P(a_1)$, by the marginalization method is obtained by summing over all the combination of states of the variables different than A, i.e., using Eqn. 8 (Eqn. 9 would be used in the case of a conditioned queried variable). Each of the factors for Eqn. 11 is obtained from the *pseudo network* array. Each element in the summation of Eqn. 8 corresponds to one element inside the *pseudo network*. This result is certainly higher than any probability measure that could have been obtained on any of the initial Bayesian networks included in the credal network. This is due to the complement rule (Eqns. 3,4) that states that no bound combination will include the upper bounds of complementary states. Thus, the marginalization over the joint probability of the lower and upper bounds, obtained through the *pseudo network* method, provides a coarse but efficient outer approximation.

3.2 Outer approximation computation

An outer approximation, i.e. an overestimation of the respective lower and upper bounds, is obtained by the direct marginalization of the joint probabilities (corresponding to the pseudo networks).

The outer approximation can be obtained for intervals as well for precise probability measures adopting existing exact inference algorithms available in a number of software toolboxes [26, 27, 28].

The procedure consists of the following steps,

1. *Pseudo networks definition.* Identification of bounds combination of the initial network that optimizes the probability values of each state (Eqn. 10).
2. *Inference computation.* Marginalization over the initial joint distribution containing the lower and upper combination of bounds obtained in the pseudo networks.
3. *Bound identification.* The lower and upper bounds identified using traditional inference methods over the optimal pseudo networks.

4. *Precise variable inclusion.* In the case that a discrete (non-interval defined) variable is queried, such variable must be included in the computation of the initial joint probability.

Variable combination has to be understood as the structure that will contain either an upper or lower bound of each of the states of the variables present in the joint distribution. Under the assumption of binary variables, combinations with the upper bound of a certain variable, e.g. $P(\overline{a_1})$ contains automatically its complement, $P(\underline{a_2})$.

For instance, if a bound $P(\overline{x})$ is given in a combination it will necessarily contain its counterpart $P(\underline{x})$ due to the *complement rule*, i.e., $\overline{P}(x) = 1 - \underline{P}(x^c)$ [21].

For the case of the lower bound of the marginal probability of the node A in state 1, $P(\underline{a_1})$. This quantity can be computed from the joint probability $P(\underline{A}, \underline{B}, \underline{C})$ by marginalizing out B and C as follows,

$$\begin{aligned} P(\underline{a_1}) &= \textstyle\sum_{\substack{i=1\\j=1}}^{n} [P(\underline{a_1}), P(\underline{b_i}), P(\underline{c_j})] \\ &= P(\underline{a_1}, \underline{b_1}, \underline{c_1}) + P(\underline{a_1}, \underline{b_2}, \underline{c_1}) + P(\underline{a_1}, \underline{b_1}, \underline{c_2}) + P(\underline{a_1}, \underline{b_2}, \underline{c_2}) \end{aligned} \tag{12}$$

After normalizing the result of Eqn. 12, the value computed necessarily corresponds to a lower probability bound obtained by variable elimination algorithm. However, it can be computed by any other exact inference method. This would correspond to a coarse outer approximation of the lower bound. If the same procedure is followed by taking into account now the upper bounds, the complement for the outer approximation of the interval query variable is found.

Furthermore, if the queried variable, D, is discrete, it has to be included in the joint probability of Eqn. 10. Thus, the distribution would be as, $P(\underline{A}, \underline{B}, \underline{C}, D)$ for the lower bound and $P(\overline{A}, \overline{B}, \overline{C}, D)$ for the upper one.

3.3 Inner approximation algorithm

In section 3.2 pseudo networks with the combinations of the states of all the variables were defined and then adopted to search the combination containing the combination relevant to compute the posterior distribution. This process produced an outer approximation since all the pseudo networks contain only the upper or lower bounds of each of the variables involved. Now, new combinations are defined over the marginal probability found (Eq. 12). This time, each combination is defined changing the bounds of each variable state. Then, a new search is done to find the combination that optimizes the posterior distribution. The result of this process is an inner approximation and the procedure is explained in this section.

Continuing with the example of section 3.2, if the marginal probability for a specific node, say A, is queried, one of the bounds for such query is now fixed (e.g., $\overline{a_1}$). Then, the bound combinations are built keeping the fixed bound, i.e., all the combinations where $\overline{a_1}$ appears, are the only ones taken into account. The upper bound of $P(a_1)$ is obtained from the maximization of the combination of the bounds for each of the variables different from $\overline{a_1}$ in the joint probability $P(\overline{a_1}, \underline{\overline{B}}, \underline{\overline{C}})$. This means finding the combination satisfying $P(\overline{a_1}) = max(P(\overline{a_1})_{comb_i}), i = 1, \ldots, n^{m-1}$ (where n corresponds to the states and m to the variables in the network). Suppose that variables in the the joint probability $P(\overline{a_1}, \underline{\overline{B}}, \underline{\overline{C}})$ have two states each. This means that the number of combinations of bounds will be 2^3. But, by fixing $\overline{a_1}$ the number of combinations is reduced to 2^2. For example, the combinations are defined as follows,

$$\begin{aligned} P(\overline{a_1})_{comb1} &= P(\overline{a_1}, \overline{b_1}, \overline{c_1}) + P(\overline{a_1}, \underline{b_2}, \overline{c_1}) + P(\overline{a_1}, \overline{b_1}, \underline{c_2}) + P(\overline{a_1}, \underline{b_2}, \underline{c_2}) \\ P(\overline{a_1})_{comb2} &= P(\overline{a_1}, \underline{b_1}, \overline{c_1}) + P(\overline{a_1}, \overline{b_2}, \overline{c_1}) + P(\overline{a_1}, \underline{b_1}, \underline{c_2}) + P(\overline{a_1}, \overline{b_2}, \underline{c_2}) \\ P(\overline{a_1})_{comb3} &= P(\overline{a_1}, \overline{b_1}, \underline{c_1}) + P(\overline{a_1}, \underline{b_2}, \underline{c_1}) + P(\overline{a_1}, \overline{b_1}, \overline{c_2}) + P(\overline{a_1}, \underline{b_2}, \overline{c_2}) \\ P(\overline{a_1})_{comb4} &= P(\overline{a_1}, \underline{b_1}, \underline{c_1}) + P(\overline{a_1}, \overline{b_2}, \underline{c_1}) + P(\overline{a_1}, \underline{b_1}, \overline{c_2}) + P(\overline{a_1}, \overline{b_2}, \overline{c_2}) \end{aligned}$$

Now, if combinations 1 and 2 are compared:

$$\begin{aligned} P(\overline{a_1})_{comb1} - P(\overline{a_1})_{comb2} = {} & P(\overline{a_1}, \overline{b_1}, \overline{c_1}) - P(\overline{a_1}, \underline{b_1}, \overline{c_1}) + P(\overline{a_1}, \underline{b_2}, \overline{c_1}) - P(\overline{a_1}, \overline{b_2}, \overline{c_1}) \\ & + P(\overline{a_1}, \overline{b_1}, \underline{c_2}) - P(\overline{a_1}, \underline{b_1}, \underline{c_2}) + P(\overline{a_1}, \underline{b_2}, \underline{c_2}) - P(\overline{a_1}, \overline{b_2}, \underline{c_2}) \end{aligned} \tag{13}$$

Since the only values known (from the previous pseudo nets computed) are those containing all lower/upper bounds, the law of total probability is considered:

$$P(A_1, B_1, C_1) = P(A_1, B_1 | C_1) P(C_1) \tag{14}$$

Considering that the bounds of the interval probability only affect the numerical value of its likelihood and not the nature of the event, Eqn. 14 can be applied to the unknown elements of Eqn. 13. For instance, $P(\overline{a_1}, \underline{b_1}, \overline{c_1}) = P(\overline{a_1}, \overline{b_1} | c_1) P(\underline{c_1})_{comb2}$.

Here, $\underline{b_1}$ on the left-hand side of the equation is the marginalization of the joint distribution of combination one. While, $P(\underline{c_1})_{comb2}$ corresponds to the marginal probability from combination two. It must be noticed that some entries of Eqn. 13 must be computed by means of marginalization. For this reason, Bayes' theorem can be used to compute the undefined combinations. For instance,

$$P(\overline{a_1},\underline{b_1},\overline{c_1}) = \frac{P(\overline{a_1},\overline{b_1},\overline{c_1})P(\underline{b_1})_{comb2}}{P(\overline{b_1})_{comb1}} \tag{15}$$

$$P(\overline{a_1},\underline{b_2},\overline{c_1}) = \frac{P(\overline{a_1},\overline{b_2},\overline{c_1})P(\underline{b_2})_{comb1}}{P(\overline{b_2})_{comb2}} \tag{16}$$

Substituting Eqn. 15 and 16 in Eqn. 13 and grouping similar terms:

$$\begin{aligned} P(\overline{a_1})_{comb1} - P(\overline{a_1})_{comb2} = P(\overline{a_1},\overline{b_1},\overline{c_1})&\left[\frac{P(\overline{b_1})_{comb1} - P(\underline{b_1})_{comb2}}{P(\overline{b_1})_{comb1}}\right] \\ -P(\overline{a_1},\overline{b_2},\overline{c_1})&\left[\frac{P(\overline{b_2})_{comb2} - P(\underline{b_2})_{comb1}}{P(\overline{b_2})_{comb2}}\right] \\ +P(\overline{a_1},\overline{b_1},\underline{c_2}) - P(\overline{a_1},\underline{b_1},\underline{c_2}) - P(\overline{a_1},\overline{b_2},&\underline{c_2}) + P(\overline{a_1},\underline{b_2},\underline{c_2}) \end{aligned} \tag{17}$$

From the complement rule, the following conditions are given,

$$P(\overline{b_2})_{comb2} = 1 - P(\overline{b_1})_{comb2}, \qquad P(\overline{b_2})_{comb1} = 1 - P(\overline{b_1})_{comb1} \tag{18}$$

Subtracting combination 1 from combination 2,

$$P(\overline{b_2})_{comb2} - P(\overline{b_2})_{comb1} = P(\overline{b_1})_{comb1} - P(\overline{b_1})_{comb2} \tag{19}$$

These results are substituted in the fractional entries of Eqn. 17. Thus, after grouping similar elements and applying again the law of total probability, the following expression is obtained,

$$\begin{aligned} P(\overline{a_1})_{comb1} - P(\overline{a_1})_{comb2} = [P(\overline{a_1},\overline{c_1}|b_1) - P(\overline{a_1},\overline{c_1}|b_2)][P(\overline{b_1})_{comb1} - P(\underline{b_1})_{comb2}] \\ +P(\overline{a_1},\underline{c_2}|b_1)P(\overline{b_1})_{comb1} - P(\overline{a_1},\underline{c_2}|b_1)P(\overline{b_1})_{comb2} \\ -P(\overline{a_1},\overline{c_2}|b_1)P(\overline{b_1})_{comb2} + P(\overline{a_1},\underline{c_2}|b_1)P(\overline{b_1})_{comb1} \end{aligned} \tag{20}$$

grouping similar terms,

$$\begin{aligned} P(\overline{a_1})_{comb1} - P(\overline{a_1})_{comb2} = [P(\overline{a_1},\overline{c_1}|b_1) - P(\overline{a_1},\overline{c_1}|b_2) + P(\overline{a_1},\underline{c_2}|b_1) - P(\overline{a_1},\underline{c_2}|b_2)] \\ \times[P(\overline{b_1})_{comb1} - P(\underline{b_1})_{comb2}] \end{aligned} \tag{21}$$

simplifying,

$$P(\overline{a_1})_{comb1} - P(\overline{a_1})_{comb2} = [P(\overline{a_1}|b_1) - P(\overline{a_1}|b_2)][P(\overline{b_1})_{comb1} - P(\underline{b_1})_{comb2}] \tag{22}$$

Given that $P(\overline{b_1})_{comb1} > P(\underline{b_1})_{comb2}$, the last term of Eqn. 22 will be positive. Such conditions ($P(\overline{a_1}|b_1) > P(\overline{a_1}|b_2)$ and $P(\overline{a_1}|c_1) > P(\overline{a_1}|c_2)$), has to be satisfied by the middle term of Eqn. 22 so condition ($P(\overline{a_1})_{comb1} > P(\overline{a_1})_{comb2}$) will be verified. Since values of $P(\overline{a_1}|b_1)$ and $P(\overline{a_1}|b_2)$ are unknown, an estimation of such probability values is required to justify such situation.

By repeating the previously described procedure with all the possible combinations the following statements are obtained:

$$\begin{aligned}
P(\overline{a_1})_{comb1} > P(\overline{a_1})_{comb2} &\iff P(\overline{a_1}|b_1) > P(\overline{a_1}|b_2)\\
P(\overline{a_1})_{comb2} > P(\overline{a_1})_{comb3} &\iff P(\overline{a_1}|c_1) > P(\overline{a_1}|c_2)\\
P(\overline{a_1})_{comb3} > P(\overline{a_1})_{comb4} &\iff P(\overline{a_1}|b_1) > P(\overline{a_1}|b_2)\\
P(\overline{a_1})_{comb4} > P(\overline{a_1})_{comb1} &\iff P(\overline{a_1}|c_2) > P(\overline{a_1}|c_1)\\
P(\overline{a_1})_{comb1} > P(\overline{a_1})_{comb3} &\iff P(\overline{a_1}|c_1) > P(\overline{a_1}|c_2)\\
P(\overline{a_1})_{comb2} > P(\overline{a_1})_{comb4} &\iff P(\overline{a_1}|c_2) > P(\overline{a_1}|c_1)
\end{aligned} \tag{23}$$

From the bounds combinations shown in Eqn. 23 used as input help to identify the maximum value for the upper bound, i.e., $P(\overline{a_1}) = max(P(\overline{a_1})_{combi}), i = 1, ..., 4$. This maximum corresponds to the upper probability input that maximizes the marginal probability of the queried variable. Then, the quantities $max(P(\overline{a_1}|b_2))$ and $max(P(\overline{a_1}|c_2))$ can be calculated by using the probability bounds $P(\overline{A},\overline{B},\overline{C})$ and $P(\underline{A},\underline{B},\overline{C})$, obtained for the outer approximation. Thus,

$$max(P(\overline{a_1}|B)) = max\left[\frac{\sum_C P(\overline{a_1},\overline{B},\overline{C})}{\sum_{B,C} P(\overline{A},\overline{B},\overline{C})}\right] = max\left[\frac{P(\overline{a_1},\overline{B})_{pseudoUP}}{P(\overline{B})_{pseudoUP}}\right] \tag{24}$$

$$max(P(\overline{a_1}|C)) = max\left[\frac{\sum_B P(\overline{a_1},\overline{B},\overline{C})}{\sum_{A,B} P(\overline{A},\overline{B},\overline{C})}\right] = max\left[\frac{P(\overline{a_1},\overline{C})_{pseudoUP}}{P(\overline{C})_{pseudoUP}}\right] \tag{25}$$

Since Eqns. 24 and 25 result in $max(P(\overline{a_1}|B)) = max(P(\overline{a_1}|b_1))$ and $max(P(\overline{a_1}|C)) = max(P(\overline{a_1}|c_1))$, respectively, combination one delivers the outcome $P(\overline{a_1})_{comb1} = P(\overline{a_1}) = max(P(\overline{a_1})_{combi}), i = 1, ..., 4$. If the same procedure is followed to identify the combination that minimizes the outcome, the bounds combination that optimizes the queried variable is found. It has to be noticed that if there is no evidence inserted in the query, the joint probability outcomes correspond to the exact bounds. However, when inference is present, the proposed analysis produces an approximate result. In fact, the presence of evidence suggests that the joint probability identified by the pseudo network approach only delivers an approximation to the real bounds due to the failure of the method to estimate the normalization factor for each probability. Because of this restriction, the inference computation obtained by the present method corresponds to an inner approximation that is restricted to binary variables.

3.4 Software and hardware

A variety of software tools were adopted in order carry out the inference computation of the queries during the experiments presented on this paper.

The inner and outer approximation algorithms based on *pseudo networks* have been implemented on the open source tool for uncertainty quantification *OpenCossan* [29,27,25]. The software is coded in an object-oriented fashion of the Matlab programming language and integrated into the developed toolbox for credal network. The types of nodes supported by this system can be discrete, continuous, imprecise and hybrid (imprecise continuous nodes). The class called Enhanced Bayesian Network in *OpenCossan* is capable of performing inference computation and sensitivity analysis in traditional Bayesian, Enhanced Bayesian and credal networks [10].

The Branch-and-bound algorithm is part of the decision support tool called *Elvira system* [30]. This software is implemented on top of the Java platform JDK (j2sdk 1.5). *Elvira system* can perform different methods for inference, learning, abduction and decision making in Bayesian Networks and Influence Diagrams. In this case, only the inference method is applied for the purposes of this work.

The *A-LP* method adopts the optimization algorithm named linear programming to minimize/maximize the lower and upper bounds of the queried variable [31]. The code is written on Java and packed into a console application based on the Eclipse platform. To perform the linear programming tasks the free software makes use of the COIN-OR library.

A system with Intel(R) Xeon(R) CPU ES-2670 v2 at 2.50 GHz processors running Linux CentOS 6.7 was used to run the models on the different software tools during the experiment.

4 Experiments and results

The method proposed is tested on two different networks to provide a numerical validation. The performance is measured as the total computational time required by the procedure and the relative error of the approximation to the exact bounds. These parameters are further compared against similar inference algorithms available in the literature. More specifically, the inner approximation is compared with the A-LP algorithm developed by Antonucci et al., [31]. While the exact bounds are computed using the *branch-and-bound* approach described in Cano et al. [13].

4.1 Fire protection system

A small-sized network introduced by Tolo [10] is studied and shown in Fig. 2. The network refers to a benchmark problem about an expert system to assess the probability of having a fire in a building and consequences produced by the fire event. The network is made of six nodes representing binary random variables with interval probabilities. The node *fire* describes the probability of fire accident. This event can influence the appearance of *smoke* as well as activating the *alarm*. The possibility of the alarm being affected by *tampering* events is represented by the connection, through the edge, between the nodes *tampering* and *alarm*. In turn, smoke can alert the residents to evacuate the building. This event is modelled by node *leaving*. The node *report* represents the case of neighbours reporting the fire accident even when they are not being evacuated.

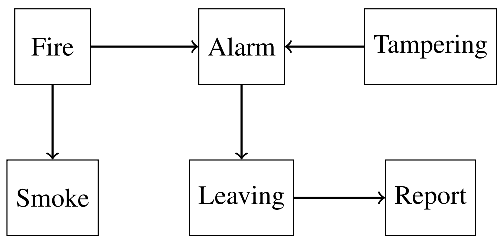

Fig. 2. Credal network of fire in a building as proposed in [25].

Two scenarios are considered. The first scenario contains the case when there is a small uncertainty associated to the probability of the events. This is represented by relative small sized intervals as shown in Appendix (Table 4). The events *smoke*, *report*, *alarm* and *leaving* are queried and the inner and outer approximation are compared against the exact bounds. The results of the inference computation with no evidence are shown in Fig. 3. While the posterior probabilities in the presence of evidence are shown in Fig. 4. The similarity of the values between the inner and the exact bounds demonstrate that the former bounds correspond to a very good approximation of the true bounds.

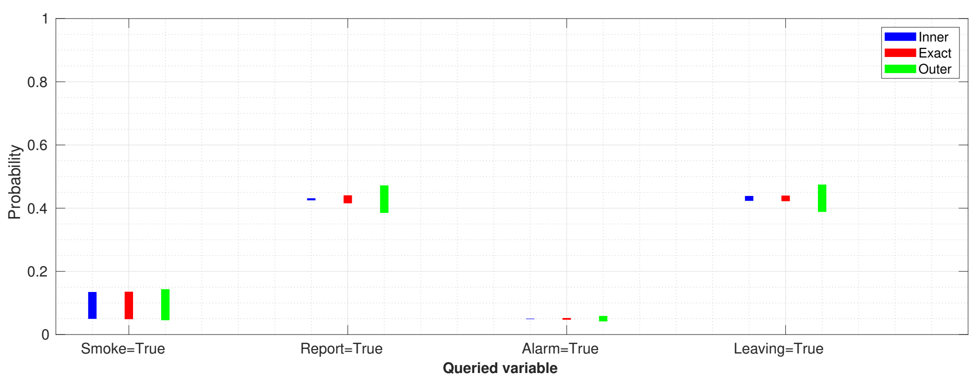

Fig. 3. Comparison of posterior probability intervals of queried variables without introducing evidence with small intervals (Table 4).

The second scenario represents the case when large uncertainty is associated to the definition of the probability tables (i.e. larger intervals). The values presented in [25] are adopted. The results are shown in Figs. 5 and 6 for the cases when no evidence and when evidence is introduced, respectively. In this case, the proposed approach still producing good approximations of the true bounds. Since the size of the input intervals is larger also the approximate inference produces larger bounds. The differences between the approximated and the exact bounds are quantified by means of the relative errors. Table 1 summarizes the relative errors of the proposed *Inner* and *Outer* approximated bounds with respect to the *Exact* solution obtained with exact inference tool.

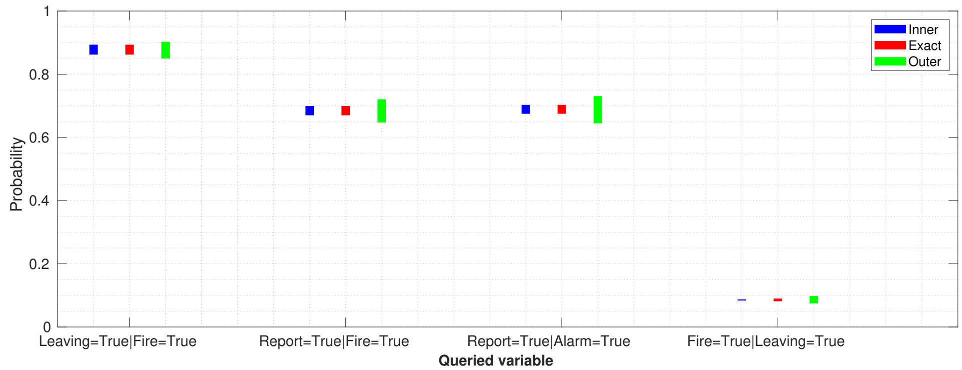

Fig. 4. Comparison of posterior probability intervals of queried variables after introducing evidence with small intervals (Table 4).

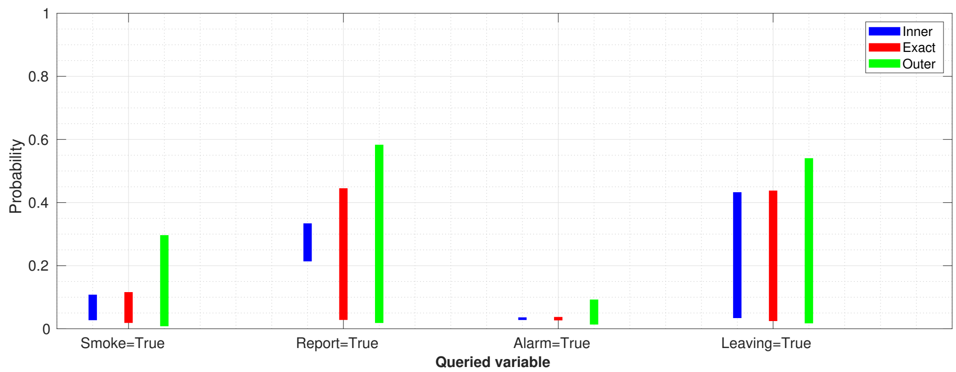

Fig. 5. Comparison of posterior probability intervals of queried variables without evidence adopting the intervals shown in [25]

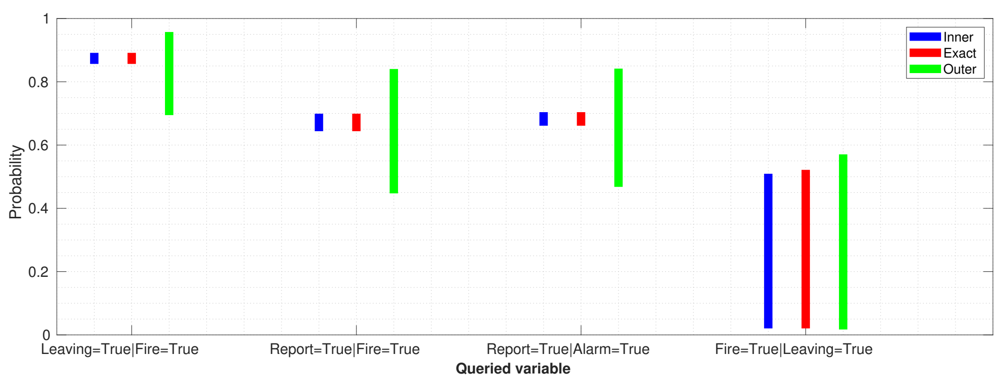

Fig. 6. Comparison of posterior probability intervals of queried variables with evidence adopting the intervals adopted in [25]

4.2 Railway model

A railway model is used to test the feasibility of the method to cope with larger networks containing multi-state variables since the inference computation complexity increases exponentially with the number of nodes. The model represents a generic railway system based on the design of [32] and [33]. The credal network consists of sixteen nodes with binary and multiple states. The interval probabilities associated with this model are shown in tables in Appendix (Tables 5-7).

In this railway model, the node called *Viaduct* refers to the probability that a part of the rail-track is intersected by a water or car flow path, i.e. the train transits over or under a bridge. This possibility is represented by the node *Overbridge*. As in many cases in the railway system, it is assumed that the majority of the tracks are placed neither over nor under the bridge but mostly on flat terrain representing a low probability for this event. The *Cut slope*, *Earthworks* and, *Embankment slope* describe the infrastructure management required to prevent the system from damages by natural threads namely, flooding, landslips and, debris flow [34], [35]. The quality of materials used to build the embankment slope is modelled by the *Fill*

Network	Query	Outer	Inner
Simple	$P(Smoke = True)$	[0.4600, 0.3422]	[0.0119, 0.0056]
	$P(Report = True)$	[0.4879, 0.3331]	[0.0079, 0.0052]
	$P(Alarm = True)$	[0.4121, 0.2865]	[0.0000, 0.0000]
	$P(Leaving = True)$	[0.4298, 0.2526]	[0.0000, 0.0000]

Table 1. Relative errors between the exact bounds and approximated bounds for the fire in a building example using the original intervals in [25].

material quality node. These variables condition the probability of having a well preserved *Terrain*. In turn, the probability of having *Obstructions* on the rail-tracks is conditioned to the circumstances of the terrain. In addition to that, the terrain status also determines the severity (low, medium, high) of the defects on the rail-track structures. This is modelled with the node *Track and structure defects* that influences the *Maintenance costs* node. A last section of the network regards the status of some primary elements of the train such as *Wheel defects* and *Fault in brakes*. These nodes affect directly the *Final train speed* assumed here to possess two states only (low, high) conditioned to the states of its parent nodes. Such node is considered to influence the occurrence of two different accidents: *Run over accident* and *Derailment*. These two nodes only hold the binary probability of having, or not, such accident.

The node *Derailment* was queried to know the probability of this event happening ($P(D = yes)$) and the probability of not having derailment ($P(D = no)$). Then, the conditional case is studied for both states of derailment by inserting evidence in node *Obstruction*. $P(D = yes|O = yes)$ and $P(D = no|O = yes)$ represent the probability of derailment or not-derailment given that the obstruction event is true. The results are shown in Fig. 8. In this figure, the differences between the *Exact* bounds, obtained via the Branch-and-bound algorithm, *B&B* and the bounds computed with the *A-LP* algorithm for inner approximation are reported and compared against the results obtained by means of the pseudo network method for the inner and outer approximation (displayed in the plot as *Inner* and *Outer*, respectively).

Table 2 summarizes the relative errors between the exact solution, *Exact*, the proposed *Inner* and *Outer* bounds and the Linear programming algorithm for inner approximation, (*A-LP*). Using the pseudo network method proposed, the outer approximation can be computed with or without the presence of evidence. However, the inner approximation can only be computed with evidence and the resulting interval corresponds to the exact bounds (see Table 2).

In order to provide another performance quantification, the proposed method was timed as well as the *B& B* and *A-LP* for each of the models presented in this work (i.e., the Fire protection and Railway system models). Such results are shown in Table 3 for comparison purposes. The time values reported in Table 3 represent the wall-clock time required by the methods for running the models shown on Figs. 2 and 7. It must be noticed that the inner approximation can be computed in the case

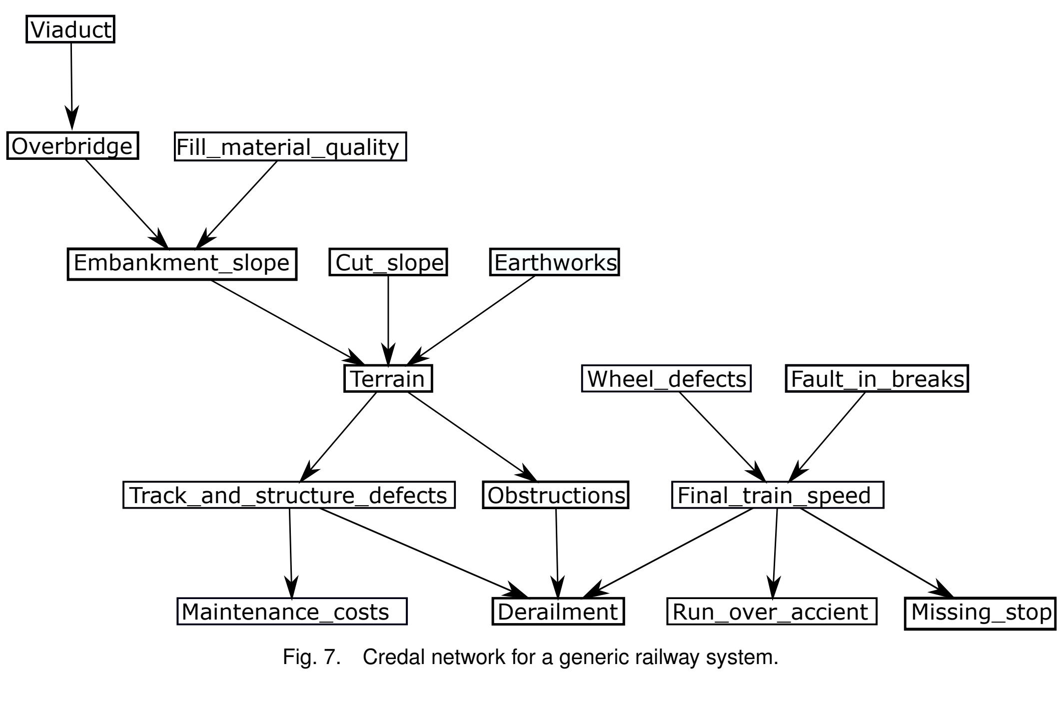

Fig. 7. Credal network for a generic railway system.

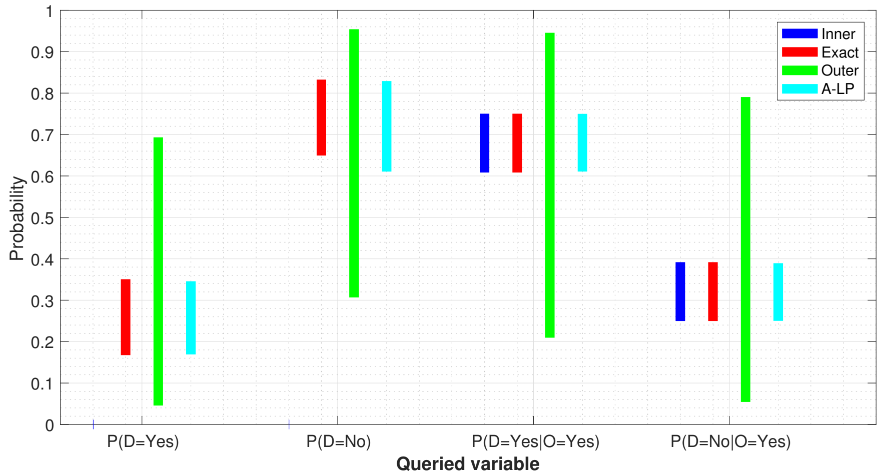

Fig. 8. Comparison of posterior probability intervals of variable *Derailment* (D) for both states with and without evidence from node *Obstruction* (O).

Network	Query	Outer	Inner	A-LP
Railway	$P(D=Yes)$	[0.7267, 0.9768]	[- , -]	[0.0107, 0.0140]
	$P(D=No)$	[0.5275, 0.1461]	[- , -]	[0.0597, 0.0042]
	$P(D=Yes\|O=Yes)$	[0.6556, 0.2604]	[0.0000, 0.0000]	[0.0018, 0.0115]
	$P(D=No\|O=Yes)$	[0.7823, 1.0176]	[0.0000, 0.0000]	[0.0015, 0.0062]

Table 2. Relative error of the approximate bounds with respect to the exact bounds for the railway model.

of a network with binary nodes, otherwise, the computational time cannot be measured as shown in Table 3.

Network	Outer	Inner	A-LP	B&B
Fire (no evidence)	1.003	1.300	20.229	6.51
Fire (evidence)	0.887	1.356	21.063	7.24
Railway (no evidence)	1.628	-	23.663	19139.977
Railway (evidence)	1.440	6576.393	24.829	20354.925

Table 3. Total computational time (wall-clock time) for the inference computation expressed in seconds.

4.3 Discussions

The proposed inference method provides inner and outer approximation with binary nodes. An outer approximation can also be computed in the case of multi-state nodes. An inner approximation can be computed when evidence is inserted in any of the multi-state nodes yielding to the calculation of the exact bounds.

Significant overestimation of the bounds are observed in Table 2. This is a consequence of the normalization process that is carried out during the inference computation. When the lower bound of a query variable is larger than the upper bound of its complement state, the sum of the two events is smaller than one, the normalization of the marginal distribution produces an increase of the initial probability of the outer approximation. The quality of the outer approximation depends on the level of imprecision (interval width) in the prior probabilities. As shown in Fig. 3, the outer bounds are very close to the real bounds when small intervals are used. However, the accuracy decreases when the size of intervals is increased (see Fig. 5). An inner approximation is obtained when no evidence is inserted in the case of a network with binary variables (Table

1). Moreover, the inference computation when evidence is present yields inner bounds with a relative error of almost zero (in the order of 10^{-13}) even for the case of multi-state variables (Table 2).

The computational time increases exponentially with the number of nodes in the network (see Table 3) when computing the inner approximation. Such increase in time can be due to the number of combinations that are tested before finding the optimum solution. The accuracy of the bounds remains close to the probability intervals obtained with the A-LP method. As shown in Table 3, the outer approximation provided by the proposed approach are computationally more efficient than other methods making the the pseudo networks a fast support tool for decision making or any other diagnostic analysis. The computational cost is also not affected by the presence of evidence. Further research will focus on improving the efficiency of the inner approximation and supporting the inclusion of multi-state variables when no evidence is introduced in the analysis.

5 Conclusions

The novel concept of *pseudo networks* has been presented. This approach identifies the combination of interval endpoints from the initial credal network that extremizes the probability values for each state of the queried variable. This structure is used for the determination of outer and inner approximations of the probability bounds.

The proposed method significantly reduces the computational cost of the outer bounds for large credal networks. When evidence is inserted the approach produces the exact bounds. It also supports networks with multi-state variables overcoming the limitation of existing approaches. The main advantage of the proposed approach resides on its simplicity, making easy to implement and combine with different exact inference methods in order to find the posterior probability intervals of the query variable. The low computational cost required to compute the outer approximation allows an quasi-real-time analysis of the network, a feature required by many decision support tools.

Acknowledgements

The authors would like to thank Alessandro Antonucci and Andres Cano for their support with the comparison of the algorithms. The first author gratefully acknowledges the *Consejo Nacional de Ciencia y Tecnología (CONACyT)* for the scholarship awarded by the Mexican government for graduate studies. This research was partially funded by the Engineering & Physical Sciences Research Council (EPSRC) and the Economic & Social Research Council (ESRC) with grant no. EP/L015927/1.

References

[1] Pearl, J., 1982. "Reverend Bayes on inference engines: A distributed hierarchical approach". In Proceedings of the AAAI (ASSOCIATION FOR THE ADVANCEMENT OF ARTIFICIAL INTELLIGENCE) National Conference on AI, pp. 133–136.

[2] Korb, K. B., and Nicholson, A. E., 2004. *Bayesian Artificial Intelligence.* Chapman & Hall/CRC, Boca Raton, U.S.A.

[3] Wang, H., Yajima, A., Liang, R., and Castaneda, H., 2015. "Bayesian modeling of external corrosion in underground pipelines based on the integration of Markov chain Monte Carlo techniques and clustered inspection data,". *Computer-Aided Civil and Infrastructure Engineering,,* **30**, pp. 300 – 316.

[4] Sun, H., and Betti, R., 2015. "A hybrid optimization algorithm with Bayesian inference for probabilistic model updating". *Computer-Aided Civil and Infrastructure Engineering,* **30**, pp. 602 – 619.

[5] Castillo, E., Grande, Z., and Calvino, A., 2016. "Bayesian Networks-Based Probabilistic Safety Analysis for Railway Lines". *Computer-Aided Civil and Infrastructure Engineering,* **31**(9), pp. 681–700.

[6] Estrada-Lugo, H. D., Patelli, E., de Angelis, M., and D, D. R., 2018. "Bayesian networks with imprecise datasets : application to oscillating water column". In European Safety and Reliability Conference 2018, S. Haugen, A. Barros, C. Van Gulijk, T. Kongsvik, and J. E. Vinnem, eds., Taylor & Francis Group.

[7] Bobbio, A., Portinale, L., Minichino, M., and Ciancamerla, E., 1999. "Comparing Fault Trees and Bayesian Networks for Dependability Analysis". In Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, pp. 310–322.

[8] Jensen, F. V., and Nielsen, T. d., 2007. *Bayesian networks and Decision Graphs.* Springer New York, New York, USA.

[9] Straub, D., and Kiureghian, A. D., 2010. "Bayesian network enhanced with structural reliability methods: Methodology". *ASCE Journal of Engineering Mechanics,* **136**(10), pp. 1248–1258.

[10] Tolo, S., Patelli, E., and Beer, M., 2018. "An open toolbox for the reduction, inference computation and sensitivity analysis of Credal Networks". *Advances in Engineering Software,* **115**, January, pp. 126–148.

[11] Beer, M., and Patelli, E., 2015. "Editorial: Engineering analysis with vague and imprecise information". *Structural Safety Special Issue: Engineering Analyses with Vague and Imprecise Information,* **52, Part B**(0), p. 143. Engineering Analyses with Vague and Imprecise Information.

[12] Ferreira Da Rocha, J. C., Cozman, F. G., and Polpo De Campos, C., 2003. "Inference in Polytrees with Sets of Probabilities". In Proceedings of the 19th Conference on Uncertainty in Artificial Intelligence, pp. 217–224.

[13] Cano, A., Gómez, M., Moral, S., and Abellán, J., 2007. "Hill-climbing and branch-and-bound algorithms for exact and approximate inference in credal networks". *International journal of approximate reasoning*, pp. 261–280.

[14] Antonucci, A., and Cuzzolin, F., 2010. "Credal sets approximation by lower probabilities: Application to credal networks". *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics),* **6178 LNAI**, pp. 716–725. cited By 7.

[15] Pfeiffer, P. E., and Schum, D. A., 1973. *Introduction to applied probability*. Academic Press, Inc., New York, NY.

[16] Tolo, S., Patelli, E., and Beer, M., 2017. "Robust vulnerability analysis of nuclear facilities subject to external hazards". *Stochastic Environmental Research and Risk Assessment,* **31**(10), "December", pp. 2733—- 2756.

[17] Cozman, F. G., 2000. "Credal networks". *Artificial Intelligence,* **120**(2), "October", pp. 199–233.

[18] Chrisman, L., 1996. "Propagation of 2-Monotone Lower Probabilities on an Undirected Graph". In Proc. XII Conference on Uncertainty in Artificial Intelligence,, E. Horvitz and F. Jensen, eds., Morgan Kaufmann, pp. 178–186.

[19] Chrisman, L., 1996. "Independence with Lower and Upper Probabilities". In Proc. XII Conference on Uncertainty in Artificial Intelligence, E. Horvitz and F. Jensen, eds., Morgan Kaufmann, pp. 169–177.

[20] Ferson, S., Ginzburg, L., and Akçakaya, R., 2005. "Whereof one cannot speak: When input distributions are unknown". *Grey Room 21*, pp. 46–69.

[21] Augustin, T., Coolen, F. P. A., de Cooman, G., and Troffaes, M. C. M., 2014. *Introduction to imprecise probabilities*. John Wiley & Sons, Ltd, West Sussex, England.

[22] Walley, P., 1991. *Statistical Reasoning with Imprecise Probabilities*, 1st ed. Champan and Hall, London, U.K.

[23] Tessem, B., 1992. "Interval Probability Propagation". *International Journal of Approximate Reasoning,* **7**, January, pp. 95–120.

[24] De Campos, C. P., and Cozman, F. G., 2004. "Inference in Credal Networks using Multilinear Programming". In Proceedings of the Second Starting AI Researcher Symposium, IOS Press, pp. 50–61.

[25] Tolo, S., Patelli, E., and Beer, M., 2018. "An inference method for bayesian networks with probability intervals". In Proceedings of the joint ICVRAM ISUMA UNCERTAINTIES conference.

[26] Murphy, K. P., 2001. The Bayes Net Toolbox for Matlab. Tech. rep., University of California, Berkeley, Berkeley, CA.

[27] Patelli, E., Tolo, S., George-Williams, H., Sadeghi, J., Rocchetta, R., de Angelis, M., and Broggi, M., 2018. "OpenCossan 2.0: an efficient computational toolbox for risk, reliability and resilience analysis". In Proceedings of the joint ICVRAM ISUMA UNCERTAINTIES conference.

[28] Andersen, S. K., Olesen, K. G., Jensen, F. V., and Jensen, F., 1990. HUGIN*-a Shell for Building Bayesian Belief Universes for Expert Systems. Tech. rep., Aalborg University, Aalborg, Denmark.

[29] Patelli, E., 2016. *Handbook of Uncertainty Quantification*. Springer International Publishing, Cham, ch. COSSAN: A Multidisciplinary Software Suite for Uncertainty Quantification and Risk Management, pp. 1–69.

[30] Consortium, E., 2002. "Elvira: An environment for creating and using probabilistic graphical models". *Proceedings of the First European Workshop on Probabilistic Graphical Models*, 01, pp. 222–230.

[31] Antonucci, A., De Campos, C. P., Huber, D., and Zaffalon, M., 2015. "Approximate credal network updating by linear programming with applications to decision making". *International Journal of Approximate Reasoning,* **58**, October, pp. 25–38.

[32] Wang, T., Qu, Z., Nichol, T., Yang, Z., Dimitriu, D., Clarke, G., and Bowden, D., 2018. "Impacts of climate change on rail systems: A new climate risk analysis model". In Safety and Reliability - Safe Societies in a Changing World, Taylor & Francis Group, pp. 2771 – 2779.

[33] Aarsland, D. W., and Vatn, J., 2018. "Contributors to successful safety level in the Norwegian railway sector". In Safety and Reliability − Safe Societies in a Changing World, Taylor & Francis Group, pp. 233 – 240.

[34] on Climate Change, C., and Transportation, U., 2008. *TRB Special Report 290: The Potential Impacts of Climate Change on U.S. Transportation*. National Academy of Sciences, Washington, D.C.

[35] Melillo, J. M., Richmond, T. T., and Yohe, G. W., 2014. *Climate Change Impacts in the United States: The Third National Climate Assessment*. United States Global Change Research Program.

Appendix

		Tampering = False	**Tampering** = True
		[0.98999, 0.99111]	[0.00889, 0.01001]
		Fire = False	**Fire** = True
		[0.958978, 0.959989]	[0.040011, 0.041022]
Tampering	**Fire**	**Alarm** = False	**Alarm** = True
False	False	[0.999800, 0.999997]	[0.000003, 0.000200]
False	True	[0.010000, 0.012658]	[0.987342, 0.990000]
True	False	[0.100000, 0.119999]	[0.880001, 0.900000]
True	True	[0.400000, 0.435894]	[0.564106, 0.600000]
	Fire	**Smoke** = False	**Smoke** = True
	False	[0.897531, 0.915557]	[0.010000, 0.102469]
	True	[0.090000, 0.110000]	[0.890000, 0.910000]
	Alarm	**Leaving** = False	**Leaving** = True
	False	[0.585577, 0.599999]	[0.400001, 0.414423]
	True	[0.100000, 0.129999]	[0.870001, 0.900000]
	Leaving	**Report** = False	**Report** = True
	False	[0.809988, 0.828899]	[0.171101, 0.190012]
	True	[0.240011, 0.250000]	[0.750000, 0.759989]

Table 4. Probabilities of events for the fire protection system in section 4.1.

		Viaduct = Yes	**Viaduct** = No
		[0, 0.01]	[0.99, 1]
		Fill material quality = High	**Fill material quality** = Low
		[0.7, 0.8]	[0.2, 0.3]
		Cut slope = Yes	**Cut slope** = No
		[0.3, 0.5]	[0.5, 0.7]
		Earthworks = Yes	**Earthworks** = No
		[0.0001, 0.0009]	[0.9991, 0.9999]
		Wheel defects = Yes	**Wheel defects** = No
		[0.0001, 0.0005]	[0.9995, 0.9999]
		Fault in brakes = Yes	**Fault in brakes** = No
		[0.0001, 0.0005]	[0.9995, 0.9999]
	Viaduct	**Overbridge** = Yes	**Overbridge** = No
	Yes	[0.99, 1]	[0, 0.01]
	No	[0.8, 0.9]	[0.1, 0.2]
	Terrain	**Obstructions** = Yes	**Obstructions** = No
	Yes	[1, 1]	[0, 0]
	No	[0.1, 0.15]	[0.85, 0.9]
	Final train speed	**Run-over accident** = Yes	**Run-over accident** = No
	High	[0.002, 0.006]	[0.994, 0.998]
	Low	[0.0001, 0.0002]	[0.9998, 0.9999]
	Final train speed	**Missing stop** = Yes	**Missing stop** = No
	High	[0.005, 0.01]	[0.995, 0.999]
	Low	[0, 0.0001]	[0.9999, 1]
Fill material quality	**Overbridge**	**Embankment slope** = Steep	**Embankment slope** = Gradual
High	Yes	[0.9, 1]	[0, 0.1]
High	No	[0.6, 0.7]	[0.3, 0.4]
Low	Yes	[0.1, 0.2]	[0.8, 0.9]
Low	No	[0.1, 0.2]	[0.8, 0.9]
Fault in brakes	**Wheel defects**	**Final train speed** = High	**Final train speed** = Low
Yes	Yes	[0.6, 0.7]	[0.3, 0.4]
Yes	No	[0.8, 0.9]	[0.1, 0.2]
No	Yes	[0.1, 0.2]	[0.8, 0.9]
No	No	[0.0001, 0.003]	[0.9997, 0.9999]

Table 5. Probability tables of Railway model in section 4.2.

	Track/structure defects		
Terrain	Severe	Medium	Minor
Yes	[0.7, 0.8]	[0.18, 0.29]	[0.01, 0.02]
No	[0.05, 0.2]	[0.1, 0.15]	[0.7, 0.8]
	Maintenance costs		
Track and structure defects	High	Medium	Low
Severe	[0.8, 0.9]	[0.1, 0.15]	[0, 0.05]
Medium	[0.18, 0.29]	[0.5, 0.55]	[0.16, 0.32]
Minor	[0.1, 0.15]	[0.3, 0.4]	[0.45, 0.6]

Table 6. Probability tables of Railway model in section 4.2: Track/structure defects and Maintenance costs

Embankment slope	**Cut slope**	**Earthworks**	**Terrain** = Yes	**Terrain** = No
Steep	Yes	Yes	[0.5, 0.7]	[0.3, 0.5]
Steep	Yes	No	[0.35, 0.4]	[0.6, 0.65]
Steep	No	Yes	[0.3, 0.4]	[0.6, 0.7]
Steep	No	No	[0.02, 0.025]	[0.975, 0.98]
Gradual	Yes	Yes	[0.4, 0.5]	[0.5, 0.6]
Gradual	Yes	No	[0.3, 0.35]	[0.65, 0.7]
Gradual	No	Yes	[0.2, 0.3]	[0.7, 0.8]
Gradual	No	No	[0.001, 0.005]	[0.995, 0.999]
Track and structure defects	**Obstructions**	**Final train speed**	**Derailment** = Yes	**Derailment** = No
Severe	Yes	High	[0.9999, 1]	[0, 0.0001]
Severe	Yes	Low	[0.75, 0.8]	[0.2, 0.35]
Severe	No	High	[0.8, 0.99]	[0.01, 0.2]
Severe	No	Low	[0.5, 0.6]	[0.4, 0.5]
Medium	Yes	High	[0.7, 0.8]	[0.2, 0.3]
Medium	Yes	Low	[0.5, 0.6]	[0.4, 0.5]
Medium	No	High	[0.3, 0.4]	[0.6, 0.7]
Medium	No	Low	[0.2, 0.3]	[0.7, 0.8]
Minor	Yes	High	[0.99, 0.9999]	[0.0001, 0.01]
Minor	Yes	Low	[0.6, 0.7]	[0.3, 0.4]
Minor	No	High	[0.75, 0.8]	[0.2, 0.35]
Minor	No	Low	[0.001, 0.005]	[0.995, 0.999]

Table 7. Probability tables of Railway model in section 4.2: Nodes Terrain and Derailment