An open toolbox for the reduction, inference computation and sensitivity analysis of Credal Networks

Silvia Tolo [*,1], Edoardo Patelli[1], and Michael Beer[1,2,3]

[1]Institute for Risk and Uncertainty, University of Liverpool, UK
[2]Institute for Risk and Reliability, Leibniz University Hannover, DE
[3]Shanghai Institute of Disaster Prevention and Relief, Tongji University, CN

Abstract

Bayesian Networks are a flexible and intuitive tool associated with a robust mathematical background. They have attracted increasing interest in a large variety of applications in different fields. Furthermore, the fast growing availability of computational power on the one side and the implementation of efficient inference algorithms on the other, have additionally promoted the success of this method. In spite of this, inference in traditional Bayesian Networks is generally limited to only discrete variables or to probabilistic distributions (adopting approximate inference algorithms) that cannot fully capture the epistemic imprecision of the data available. In order to overcome these limitations, Credal Networks have been proposed to integrate Bayesian Networks with imprecise probabilities which, adopting non-probabilistic or hybrid models, allow to fully represent the information available and its uncertainty.

In this study a novel computational tool, implemented in the general purpose open software OpenCossan, is proposed. The tool provides the reduction of Credal Networks through the use of structural reliability methods, in order to limit the cost associated with the inference computation without impoverishing the quality of the information initially introduced. Novel algorithms for the inference computation

---

[*]s.tolo@liverpool.ac.uk

of networks involving probability bounds are provided. In addition, a novel sensitivity approach is proposed and implemented into the Toolbox in order to identify the maximum tolerable uncertainty associated with the inputs.

1   Introduction

The fast technological growth that has characterized the last century has progressively provided more efficient and advanced instruments for everyday life as well as for industrial and scientific applications. This progress goes along with an ever increasing grade of complexity which concerns the engineering field on any level, leading to face new and more challenging tasks from both mathematical and computational points of view. As a consequence of this, novel and efficient tools are strongly needed to adequately predict the behaviour of complex systems, optimize their performance, estimate their reliability and evaluate the risks they are subject to, especially with regards to safety-critical installations (e.g., reservoirs, dams, nuclear and chemical installations etc.).

The accuracy in estimating the actual risks to which a system is subject is clearly bound by our ability of capturing and representing reality. This means that any engineering analysis has to face the challenging task of formulating suitable numerical models in a quantitative manner without ignoring significant information on the one hand, and without introducing unwarranted assumptions on the other [4]. If this balance is violated, computational results may deviate significantly from reality and the associated decisions may lead to serious consequences due to risk underestimation or, conversely, to unnecessary costs in the case of over-conservative approach.

This challenging task implies two main bottlenecks: the first one is associated with the technological complexity of the systems under study. This includes the representation of elaborate networks of dependencies among subsystems and components interacting directly or indirectly with each other and determining the correct functioning of the overall system. To provide an oversimplified representation of these mechanisms, hence to introduce large model errors, can easily result in misleading estimates of the system

state or its reliability. The criticality of this task is also worsened by the difficulty of adequately quantifying the significance of the model error introduced and hence the weight of the simplifications adopted in the analysis. For this reason, it is of extreme importance to be able to adequately reproduce these interconnections in order to take into account the possible failure of each individual component and its consequences on a more or less wide range of others, potentially triggering a chain of events and leading to more serious accidents scenarios or failure modes.

Several methods (e.g., Fault Tree, Event Trees, Reliability Block Diagrams etc.) are available in the scientific literature and widely used in several fields of engineering for the representation and analysis of complex systems. Among these, Bayesian Networks (BNs) have attracted an increasing interest in the last decades spreading to several industrial and scientific application fields. The large success of BNs is linked to their capability of providing efficient factorization of joint probability distributions exploiting information about the conditional dependencies existing among the variables involved on the basis of a robust mathematical background such as Bayesian Statistics. In addition, their intuitive graphical framework has consolidated their attractiveness in quite different fields of science and engineering, from artificial intelligence to medical and economic areas [40]. They can be considered as the general case of more common methodologies, such as Fault Tree analysis [17], with respect to which they offer several advantages. Indeed, Bayesian Networks can model complex dependencies among components, uncertainty can be included in modelling and both forward and backward analysis is allowed, making them particularly attractive for both diagnosis and inference purposes [6].

The second of the bottlenecks previously mentioned refers to the necessity to deal with our ignorance, in terms of uncertainty, scarcity of data or even a more general lack of knowledge regarding the actual mechanisms of interaction involved in the analysis. This must be explicitly included in the analysis in order to be able to quantify its impact on the accuracy and robustness of the results obtained. Nevertheless, information available in real-world application involves sparse data, poor measurements and subjective information, hence results difficult to quantify and model. The adoption of traditional

mathematical models built on poor or scarce data, can lead the information modelled to be far from that actually available, introducing biases that lower the credibility and can even invalidate the results of the analysis [4]. For instance, this can occur when the samples related to certain event are not enough, or not robust enough, to be clearly attributed to a particular distribution family or to be associated confidently with precise parameter values for the distribution selected. A wide range of solutions are available to adequately represent different degrees of information without introducing biases or strong assumptions but reflecting in the predictions the uncertainty in input. The definition of imprecise probabilities embraces a wide set of approaches (e.g., interval probability, fuzzy probabilities, p-boxes) which provide a mathematical basis to deal with the representation of information when it is not sufficient for probabilistic modelling or rather suggests a set-theoretical approach.

The aim of this work is to provide a novel computational tool for the efficient computation of Credal Networks (CNs). The approach proposed is based on the use of system reliability methods to integrate traditional Bayesian Networks with probabilistic, non-probabilistic and hybrid frameworks without renouncing to the robustness of traditional inference algorithms. The proposed algorithms have been implemented in the general purpose software OpenCossan [28] [16]. They allow to reduce the initial user defined models into a simpler but equivalent network. This, conversely from the initial model which can include a wide variety of variables type, embraces only crisp and interval probabilities and hence results into a BN containing probability bounds. The integration of these two approaches (namely BNs and probability intervals), strongly enhances the robustness of the analysis, but also introduces significant challenges. First, the inference computation on these models can easily become highly demanding. Second, the capability to track the propagation of uncertainty within the model is essential in order to ensure the significance of the output and to obtained the desired level of accuracy of the analysis at the lowest cost. This study aims to analyse and propose novel solutions to deal with these issues. Few available software packages for manipulation of graphical models with imprecise probabilities are available, so that this field is highly open to new contributions [10] and this study is characterized by strong novelty.

In the following, the theoretical background of the proposed method is analysed. The methodologies implemented in the toolbox and related to the reduction procedure, the inference computation and the sensitivity analysis of the reduced models are then shortly introduced. Finally, the structure of the toolbox and its main capabilities are accurately described, followed by a simple numerical example.

2 Theoretical Background

In this section, the theoretical background on which the present study relies is introduced. The theory of Bayesian Networks is outlined in Section 2.1. The main imprecise probability approaches available in the toolbox (Section 2.2) and their integration within cutting-edge system reliability methods (Section 2.3) are then briefly described as well as the state of the art for Credal Networks (Section 2.4).

2.1 Bayesian Networks

The graphical structure of a Bayesian Network consists mainly of *nodes*, which generally represent the individual variables of the problem modelled, and *edges*, which link the previous to each other and are characterized by specific directions (hence represented as arrows). According to their mutual relationships, nodes are referred through different definitions. If the edge linking two nodes, X1 and X2, is originated in the first and points toward the second, X1 is defined as the *parent* of X2 and, coherently, X2 as the *child* of X1. Conversely, a node without parents is referred as a *root* of the network.

The edges connecting the nodes represent the causal relationships existing among them. If no edge exists between two nodes, those are considered conditionally independent to each other. More precisely, each node is conditionally independent of its non-descendants given its parent variables, satisfying the local Markov property [33].

The numerical framework associated with the graphical structure consists of Conditional Probability Distributions (CPDs). As the definition suggests, these are distributions which quantify the strength of the conditional dependencies existing among the variables of the model. Each node is associated with a CPD, which contains the values of the probability of each possible outcome of the node given those of its parents. If the

5

values are crisp probabilities, the CPDs are also referred as CPTs (Conditional Probability Tables).

Overall, the simple scheme described, allows to easily compose complex models which, on the basis of the local information available (i.e., conditional probability values associated to every single node given its parents), provide the representation and quantification of much more elaborate events. Indeed, the probability of any query, e.g. marginal or complex event, can be calculated combining the numerical information of the CPDs on the basis of the Bayes' theorem. This process, generally referred as inference computation, is accomplished through the use of computational algorithms, a wide range of which is available in literature.

The main classification of inference algorithms refers to the nature of their outcome: exact inference algorithms provide the exact value of the probability query, hence obtained through an analytical approach, while approximation algorithms provide values which only approximate the true value of the probability. The adoption of an approximate approach is generally unavoidable in the case of continuous variables, but it is associated to unknown rate of convergence which can compromise the robustness of the analysis. Conversely, exact inference algorithms are restricted to the use of discrete variables.

Beyond the simple inference computation, the key feature of BNs is the so called *belief updating*. Indeed, BNs allow to revise the value of the posterior probability of a query if additional information (i.e., *evidence*) is available regarding the state of any node involved in the computation. This capability, together with the high flexibility and intuitive nature of the approach, have led to a large success of the method in many fields of science and engineering. Please refer to the work of Pearl and Russell for a complete overview on BNs [30].

2.2   Imprecise Probability

In engineering practice and more generally in real-world applications, it is common to deal with the availability of only scarce data or little information or the disagreement of experts involved in the analysis. In such cases, the adoption of precise probability

distributions or crisp probability values could imply the necessity of adopting subjective hypothesis which is not provided (e.g., when the experimental samples are not sufficient for a robust calculation of the distribution parameters) or fully justified. This means not only to introduce biases in the analysis, which in worst cases can undermine the trustfulness of the results, but also to neglect the degree of uncertainty affecting the value. Failing in representing this latter results in the lost of information regarding the accuracy of the output, which would be intrinsically affected by an unknown degree of imprecision due to the "hidden" uncertainty of the input.

Several solutions to this problem have been proposed in the literature adopting different mathematical concepts, such as Bayesian approaches, interval probabilities, random sets, evidence theory, fuzzy stochastic concepts, info-gap theory etc. [4]. An overall analysis of all these concepts, generally embraced within the definition of imprecise probabilities, is beyond the purpose of this work.

Nevertheless, the methodologies adopted in this work rely on the use of interval probabilities and p-boxes, which are briefly presented in this section.

Interval Probabilities Interval probabilities specify bounds on probability for an uncertain quantity with underlying randomness that is not known in detail and, thus, they represent a special kind of imprecise probabilities. This would imply a more accurate depiction of the information and data available, avoiding the introduction of biases and assumptions which can lower significantly the accuracy of the analysis [41]. In practice, this approach is particularly preferable when no probabilistic information is available to specify univocally distribution functions so that modelling as probability sets is most appropriate. A complete overview of the historical developments of interval probabilities is provided by Kyburg [21].

According to interval probabilities, an event A can be characterized by a range of probability encloses between a lower limit $p(\underline{A})$ an upper limit $p(\overline{A})$. Since any value within the interval $\left[p(\underline{A}), p(\overline{A})\right] \subseteq [0, 1]$ is a plausible estimate of the probability of A, it derives that the interval must belong to the space of intervals on $[0, 1]$ [4]. Hence, if the outcome A with probability bounds $[p(\underline{A}), p(\overline{A})]$ is defined as the negation of an event B, the bounds for the probability of A can be calculated through the complementation

rule:

$$P(\overline{B}) = [1 - p(\overline{A}), 1 - p(\underline{A})] \tag{1}$$

Similarly, other logical operations can be easily computed with probability intervals adopting suitable formulations. For instance, the probability of the conjunction event $P(A \cap B)$, given two independent events A ($p(\overline{A}) = [p(\underline{A}), p(\overline{A})]$) and B ($p(\overline{B}) = [p(\underline{B}), p(\overline{B})]$), will belong to the interval:

$$P(\overline{A \cap B}) = [p(\underline{A}) \times p(\underline{B}), p(\overline{A}) \times p(\overline{B})] \tag{2}$$

More generally, if no assumption of independence is made, the interval can be calculated adopting Fréchet inequalities:

$$p(\overline{A \cap B}) = [max\left(0, p(\underline{A}) + p(\underline{B}) - 1\right), min\left(p(\overline{A}), p(\overline{B})\right)] \tag{3}$$

Likewise, the bounds for the probability of the disjunction event $p(\overline{A \cup B})$, can be calculated according to Eq.(4), or with the formulation in Eq.(5) if the events A and B are assumed independent.

$$p(\overline{A \cup B}) = [1 - (1 - p(\underline{A}))(1 - p(\underline{B})), 1 - (1 - p(\overline{A}))(1 - p(\overline{B}))] \tag{4}$$

$$p(\overline{A \cup B}) = [max(p(\underline{A}), p(\underline{B})), min(p(\overline{A}), p(\overline{B}))] \tag{5}$$

Probability Boxes   Epistemic uncertainty, i.e. the component of uncertainty related to the lack of knowledge on the data, can affect the information available on different levels. As mentioned in Section 2.2, only coarse specifications can be available for certain events (e.g., complex events observed phenomenologically); in other cases, probabilistic information, such as parameters of a probabilistic model, distribution type or, in a non-parametric description, the curve of the cumulative distribution function, may be available but only specified within some bounds. This latter case can be adequately described through the adoption of probability boxes. This definition refers to a numerical approach that allows the calculation of bounds on arithmetic combinations

of probability distributions when perhaps only bounds on the input distributions are known [4]. This kind of analysis can be considered as the integration of traditional probability theory with standard interval analysis, hence it is a generalization of the two: when the information regarding the distribution is large, the bounds will result very thigh, approximating the precise distribution; conversely, when only poor data is provided the bounds will be much wider, coherently with the weaker confidence about the specification of this distribution. A p-box represents a class of probability distributions consistent with constraints associated to the cumulative probability distribution function of a quantity: lower and upper bounds can be defined regarding the quantity's mean, variance and distributional shape, which hence are represented as intervals instead of crisp values, taking into account the epistemic uncertainty affecting them [15]. Let $\mathbb{D}$ represent the space of distribution functions on the real numbers $\mathbb{R}$, so that $\mathbb{D} = \{D|D : \mathbb{R} \rightarrow [0,1], D(x) \leq D(y) if x < y, \forall x, y \in \mathbb{R}\}$, and $\mathbb{I}$ the space of real intervals so that $\mathbb{I} = \{[i_1, i_2]|i_1 \leq i_2, \forall i_1, i_2 \in \mathbb{R}\}$. In light of this, a p-box is define as a set of distribution functions $F \in \mathbb{D}$ verifying the constraints:

$$F \in \mathcal{F} \tag{6}$$

$$\underline{F}(x) \leq F(x) \leq \overline{(F)}(x) \tag{7}$$

$$\int_{-\infty}^{\infty} x dF(x) \in m \tag{8}$$

$$\left(\int_{-\infty}^{\infty} x^2 dF(x)\right) - \left(\int_{-\infty}^{\infty} x dF(x)\right)^2 \in v \tag{9}$$

where $\overline{F}, \underline{F} \in \mathbb{D}$, while $m, v \in \mathbb{I}$ and $\mathcal{F} \subseteq \mathbb{D}$ [4]. The results of mathematical calculations on p-boxes rigorously enclose all possible distributions of the output variable so long as the input p-boxes were all sure to enclose their respective distributions.

This work have been focused on the adoption of distributional p-boxes, hence the description of variables through the adoption of this approach is restricted to the case in which the particular shape of the distribution (e.g., normal, uniform, beta, Weibull, etc.) is known, while its parameters are affected by imprecision, and hence represented by intervals. Such parameters will be referred as hyperparameters from now on. Further research will focus in the future to overcome this limitation, through the adoption of

suitable computational algorithms for the computation of non-parametric p-boxes.

2.3 Structural Reliability Methods

In the field of structural reliability, the domain bounding failure events is generally described by the so-called limited state functions $G(\mathbf{x})$, which represent the failure modes of the system under study. In light of this, considering an event F defined as a domain in the outcome space of m stochastic variables $\mathbf{x} = (x_1, x_2, ..., x_m)$, the m-dimensional space can be divided in a *safe region*, represented by the domain $\Omega_s = \{\mathbf{x} : G(\mathbf{x}) > 0\}$, whilst the failure domain can be expressed as $\Omega_f = \{\mathbf{x} : G(\mathbf{x}) \leq 0\}$. Hence, the probability of occurrence of the event F can be quantified solving the integral of the form:

$$P(f) = \int_{\mathbf{x} \in \Omega_f(\mathbf{x})} f(\mathbf{x}) d\mathbf{x} \tag{10}$$

where $f(\mathbf{x})$ is the joint probability density function associated to the m stochastic variables $\mathbf{x}$. Various methods for the solution of the integral in Eq.(10) are available such numerical integration techniques, Monte Carlo simulations [18] and asymptotic Laplace expansions [31]. Other common solutions (e.g., First-Order and Second-Order Reliability Methods) rely instead on the transformation of the reliability problem previously described into an optimization one [19]. In this case, the random variables in the vector $\mathbf{x}$ are mapped into independent standard normal variables and the minimum distance β between the limit state and the origin of the transformed space identified (i.e., the so called reliability index). The probability of failure is then computed on the basis of the assumption introduced. For example, in case of First Order Reliability Method (FORM) a linear approximation of the limit-state function is adopted leading to $P_f = \Phi(-\beta)$. A more generalized approach which aims to take into account epistemic uncertainty has to allow for uncertainty in both the structural parameters and those of the probabilistic models. As for the probabilistic structural reliability, different methods are available, although the related literature is by far more limited than the previous case. One of the most common approach consists of treating epistemic uncertainty using set of descriptor values, such as in the case of intervals [25], convex models [5], random sets [38] and fuzzy sets [42].

In this study the epistemic uncertainty affecting both the structural parameters of the model and the hyperparameters of the random variables have been taken into account. The current implementation has been restricted to the adoption of two main system reliability methods: the first, developed by Luo et al. [23], is based on a nested minimization problem. Thanks to the combination of uncertain parameters in the form of interval variables ($\mathbf{v}$) represented by convex models [20] and precise random variables ($\mathbf{u}$), the limit state can be expressed as a function of both the sets ($G(\mathbf{u}, \mathbf{v}) = 0$). This results in a cluster of limit-state surfaces in the standard normal space. The method allows to seek the worst-case combination of interval variables value, identifying the limit state surface on which the most probable failure point lies. The inverse of the normal cumulative distribution function of the distance between the identified point and the origin of the standard normal space is assumed to be the upper bound of the failure probability. This approach can be considered a more general case of the probabilistic method FORM. When the epistemic uncertainty drops the intervals representing the uncertain variables collapse into single values leading back to the traditional FORM procedure. The second method adopted has been developed by De Angelis et al. [13] and provides the estimation of set-valued failure probabilities. The approach consists of coupling advanced Monte Carlo methods (i.e., Adaptive Line Sampling) with optimization methods, in order to estimate the lower and upper bounds of the failure probability. Moreover, the method allows for both imprecise probability distribution functions (i.e., credal sets) and sets of bounded variables.

2.4   Credal Networks

The BN approach implies that every variable conditional on any configuration of the variables parents must be associated with an unique probability distribution. Nevertheless, in case of incomplete or vague beliefs, poor information or contradictory data, it may be not straightforward to identify point estimates of an event probability. To overcome this limitation, Credal Networks (CNs) have been introduced [12]. These are graphical models that realize the integration of BNs with credal sets, i.e. set of probability measures. In other words, a CN can be regarded as the representation of a

set of Bayesian networks which share the same graphical structure but are associated to different probability measures. Each of the BNs of the set represented by the CN is a plausible instantiation of the problem under study, hence there is no commitment as to whether one of these Bayesian networks is the 'correct' one [12]. In the case of CNs, each node of the network represents a random variable X_i which can be associated with conditional credal sets $K(X_i \mid pa(X_i))$, defined by the conditional distributions $p(X_i \mid pa(X_i))$.

The aim of the inference computation is to combine the local credal sets of the network into a set of joint distributions. Generally, there may be more then one set of joint distributions that are coherent with the marginal and conditional credal sets associated with the network: each of these joint distributions set will be referred as an extension of the marginal and conditional credal sets in input. In this work, the term inference is used with regards to CNs to indicate the computation of the probability bounds of the largest extension that satisfies the Markov condition (i.e., independence of each node of its non-descendant non-parents given its parents, as mention in Section 2.1) under the assumption of strong independence [11]. This is generally referred as strong extension, and results to be the convex hull of the set containing all joint distributions that factorize the overall joint probability of the network, as for BNs, but where the conditional distributions $p(Xi|pa(Xi) = \pi_k)$ are selected from the local sets $K(Xi|pa(Xi) = \pi_k)$. Both exact and approximate algorithms are available in literature [7] [1]. In spite of this, only polytree-shape networks with binary variables are suitable for efficient exact inferences while other types face extremely high computational challenges [14]. On the other hand, both outer or inner approximations can be computed through approximate algorithms: the first provides probability bounds which enclose the correct probability interval between lower and upper probabilities; the second provides in output intervals that are enclosed by the correct probability interval. Limited literature is available regarding CNs applications [2] [43].

3   Methodology

The toolbox presented in this paper integrates traditional and cutting-edge methods well established in the literature with novel proposed algorithms. This section is dedicated to the description of such novel approaches, introducing the methodologies behind them. They cover three main aspects of Credal Networks computation: the first is related to the reduction of the initial model, in order to remove part of the non-traditional variables simplifying the graphical structure of the network without impoverishing the information initially available (Section 3.2). The resulting reduced network contains only nodes associated with crisp values or interval probability. Several inference methods are available for the inference computation on this kind of networks: among these, a naive approach for the derivation of precise bounds on a query variable of interest consists of applying the standard Bayesian exact inference methods for each combination of probability bounds, and finally to minimize and maximize the results. However this approach can result highly expensive and suffers from combinatorial explosion. Section 3.3 briefly describes a novel algorithm for the computation of inference on the reduced network which aims to provide a valid solution to this issue. Finally, in Section 3.4, a novel approach to sensitivity analysis for the reduced network is proposed.

3.1   Aim and Motivation

The aim of the proposed methodology and related computational tool is to offer a novel approach for the implementation of credal networks able to include different mathematical frameworks for the representation of the available data. In spite of the restricted number of CNs applications available in the literature, the limitations of such approach in terms of data representation are blatant: most of the inference algorithms and software available for the computation of Credal Networks are indeed restricted to random variables which assume finitely many values (also called discrete or categorical variables). In many cases, this implies the necessity to adapt the representation of data to a discrete number of possible outcome states regardless the nature of the information available and hence potentially impoverishing its quality and the accuracy of the overall analysis.
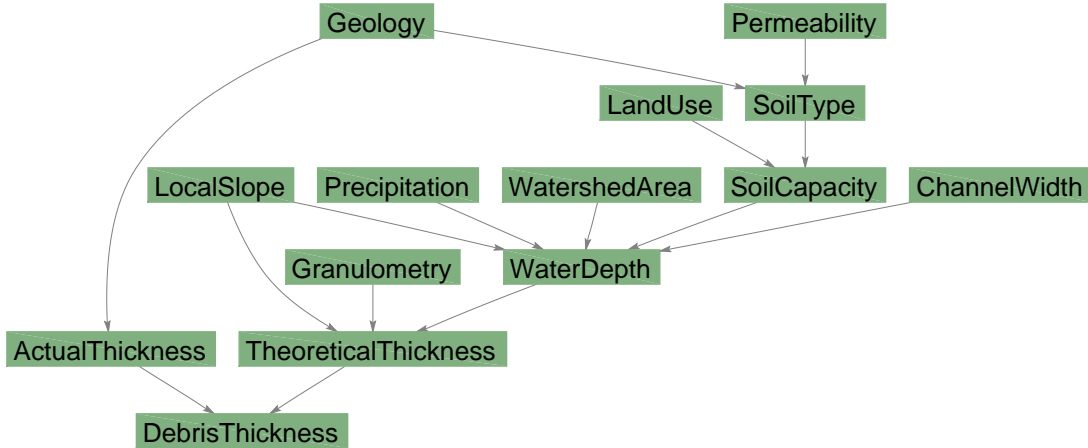
Figure 1: CN model for hazard assessment of debris flows proposed by Antonucci et al.

As an example, Antonucci et al. [3] propose a CN model for the hazard assessment of debris flows, as shown in Fig.**??**. The model, developed for the Ticino canton in Switzerland, allows to fully exploit the advantages offered by CNs, such the potential of credibly dealing with qualitative uncertainty through the use of imprecise probabilities, enhancing the robustness of the inference computation. On the other hand, the authors highlight computational issues related to the updating process which could become widely more expensive and require different solutions in the case of application of the model to other geographical areas; moreover, the treatment of continuous variables in the network, in order to avoid the loss of information introduced by discretizations, is also strongly suggested. In spite of the large dataset available for the mentioned application and the random nature of some phenomenon modelled (e.g. precipitation intensity, granulometry), which would be more accurately captured through the use of continuous probabilistic variables, for computational reason the model is restricted to discrete nodes.

The computational approach implemented in the present study offers a solution strategy for both these issues: on the one hand it allows to depict the data available through the use of continuous variable, representing the randomness of the phenomenon involved as well as the imprecision affecting the available knowledge. On the other hand, it allows to simplify the initial network, reducing significantly the computational costs of updating, even for identical inference algorithms.

For instance, according to the nature of the underlying information in terms both of data availability and nature of the related physical phenomenon, soils characteristics such as *Permeability* and *Granulometry*, as well as the precipitation intensity (i.e. node *Precipitation*) would be better captured by continuous probabilistic distributions; similarly, features such as *LocalSlope*, *WatershedArea*, *SoilCapacity* and *ChannelWidth* can be reasonably known only with some imprecision and hence would be better represented as interval variables. Finally, the actual available debris thickness (node *ActualThickness*), obtained from observations and hence subject to a certain imprecision, could be represented as an imprecise random variable. The model modified according to such considerations is shown in Fig.2 and can be computed through the computational tool implemented resulting in the reduced CN shown in Fig.3. The network obtained involves only 4 of the 14 nodes of the initial model, and this ensures a sensible reduction of the inference computation costs, whatever inference algorithm would be adopted. This also suggests the possibility of using exact inference on reduced models even when the analytical approach on the initial network would results computationally prohibitive. In the event that the updating process involves a larger number of nodes, such as the width of the channel and the soil granulometry, these could be included in the reduced network implementing a low-invasive discretization procedure (as discussed in Section 4.1) resulting in the model of Fig.4 and subsequently in the CN shown in Fig.5. This approach ensures the possibility to introduce evidence in the nodes of interest still reducing the costs of computation as well as the impact of discretization. Hence, the most appealing aspects of the proposed approach lie with its ability to fully capture the nature of the data available, offering a wide and flexible range of mathematical frameworks suitable for the representation of the information available and hence reducing the need for discretization and underlying simplifying assumptions; at the same time, such methodology offers a robust strategy for the reduction of the computational costs associated with belief updating, which is commonly recognized as the main drawback of the CN approach. These advantages can be interpreted as the introduction, at a reduced computational cost, of a further degree of freedom in the representation of the uncertainty underlying the available information.
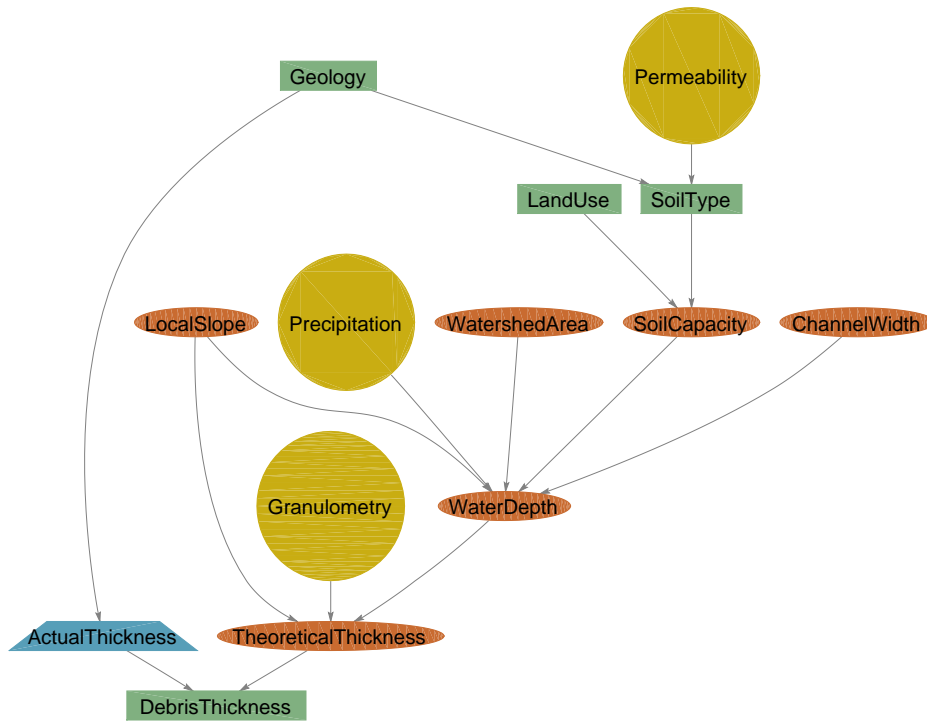
Figure 2: Modified CN model for hazard assessment of debris flows implemented in the proposed computational tool
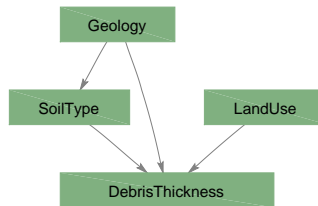

Figure 3: Reduced model for hazard assessment of debris flows obtained with the suggested methodology
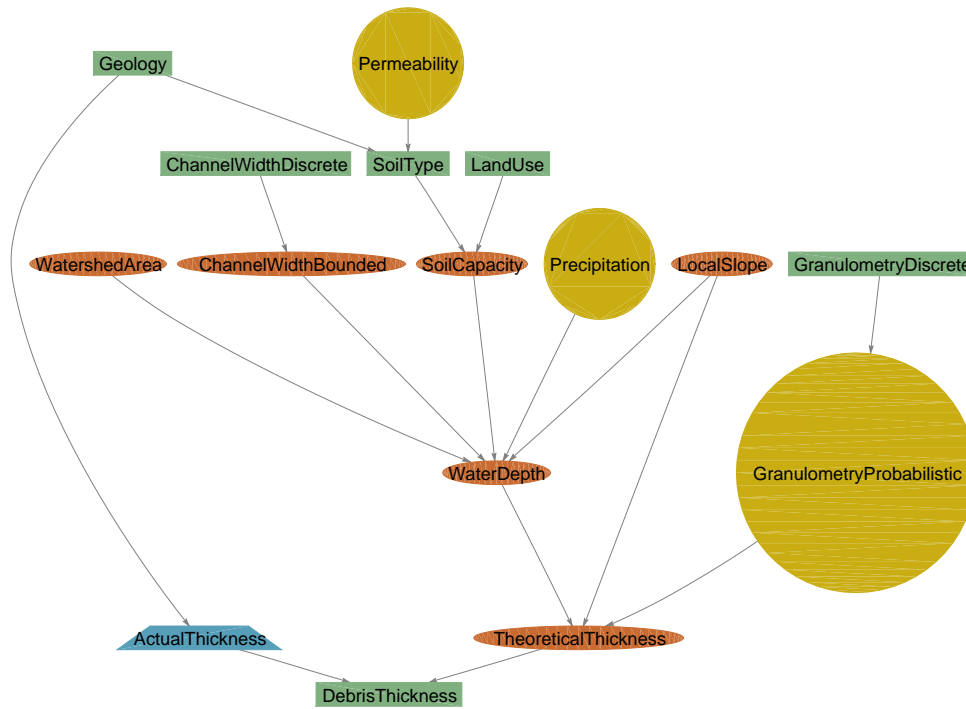
Figure 4: Modified CN model for hazard assessment of debris flows including additional discrete nodes for the variables *ChannelWidth* and *Granulometry*
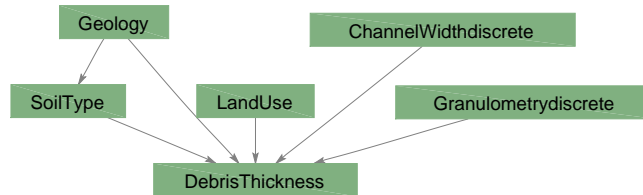

Figure 5: Reduced version of the model shown in Fig.3 obtained with the implemented computational tool

3.2 Credal Network Reduction

The aim of the methodology is to reduce the initial Credal Network in order to simplify the computation of inference. The initial model can contain both discrete or continuous variables: the first can be associated with crisp probabilities values, as in traditional BNs, or with probability bounds, and will be referred from now on as *discrete nodes*. The second can be interval variables (i.e., *interval nodes*) or continuous variables whose distribution is described by sets of probabilistic distributions (i.e., *probabilistic nodes*) or parametric p-boxes (i.e., *hybrid nodes*). Table 7 summarizes the types of nodes allowed in the analysis.

The key assumption for the application of the method is that each node, if child of at least one continuous node, has to be defined as a domain or a set of domains in the outcome space of its parents. Under this assumption, system reliability methods can be adopted to compute the probability of such nodes, cutting the number of non-discrete variables and thus reducing the initial model to an equivalent network containing only discrete nodes characterized by crisp and interval probabilities. This simplification is obtained without impoverishing the information available but storing the data provided by the continuous variables in input into their discrete children. Indeed, since each discrete child of at least one probabilistic, bounded or hybrid node is characterized by domains in the outcome space of its parents, the probability value associated to it can be computed as a reliability problem where its parents appear to be the variables in input. If the node under study has only probabilistic nodes as parents, the methodology leads back to the so called Enhanced Bayesian Network approach, suggested by Straub and Kiureghian [34]. In this case, the probability measure computed is a crisp value and the resulting node a discrete one. If also bounded and hybrid nodes are involved, the output of the reduction is a discrete node characterized by probability bounds. After the reliability computation is carried out, the information initially provided by the continuous parents is stored in the probability measure of the child node, hence the dependency links between the continuous parents and the computed child become meaningless and can be erased. Repeating this procedure for each node child of probabilistic, bounded or hybrid nodes, it is possible to progressively remove the edges linking the continuous

Variable Type	CPD	Node Definition	Graph
Discrete	Crisp probabilities	Discrete	Rectangle
Discrete	Probability bounds	Discrete	Rectangle
Random	Probability Distributions	Probabilistic	Circle
Interval	Crisp probabilities	Bounded	Ellipse
Hybrid	Parametric p-boxes	Hybrid	Trapezoid

Table 1: Types of nodes allowed

variables to the network: when these nodes are fully isolated from the rest of the model, they can be removed without consequences since they do not contribute any more to the joint probability distributions associated with the reduced CN.

The overall procedure described consists of the following steps:

1. Identification and computation of continuous nodes children of bounded, probabilistic or hybrid nodes. As for discrete nodes, also continuous nodes children of non-discrete nodes are defined as domains in the outcome space of their continuous parents and must be derived from their parent's distributions. Once the computation is completed the links among the continuous parents and the continuous child are removed. In this case no system reliability analysis is required and the result are continuous distributions.

2. Identification of Markov blankets of bounded, probabilistic and hybrid nodes. The Markov blanket of a node A is a set of nodes including the parents, children and spouses of the node itself. In other words, it contains all the variables that shield the node from the rest of the network. The variables within the Markov blanket of a node are the only needed to predict the behaviour of that node. Hence, in the case of the reduction procedure, the Markov blankets of non-discrete nodes of the network identify the groups of variables to be involved in system reliability computations in order to reduce the initial CN. The Markov blankets identified embrace both the input variables and the definition of the reliability problem, expressed as the domains associated with the children nodes of the blanket.

3. System reliability analysis computations and removal of unnecessary edges. Once the former steps are completed, the reliability analysis problems identified can be

computed. Let consider the example in Fig.6: the model on the left hand side of the figure shows the initial CN while on the right hand side the reduced network is represented. Assuming the variables independent, the joint probability associated
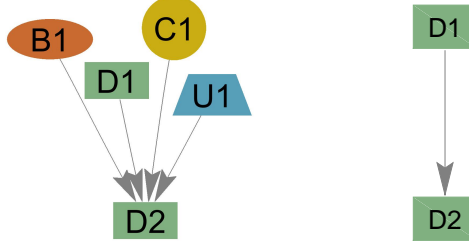

Figure 6: Example of an elementary CN and its reduced network, where C1 refers to a probabilistic node, B1 to an interval node, U1 to an hybrid node, D1 and D2 to discrete nodes, respectively

to the initial CN can be computed solving the integral in Eq.(11):

$$P(D1, D2) = \int_{B_1, C_1, U_1} p(D_1)p(D_2|B_1, D_1, C_1, U_1)f(B_1)f(C_1)f(U_1)dB_1 dC_1 dU_1$$

(11)

where $p(D_1)$ and $p(D_2|B_1, D_1, C_1, U_1)$ are the probability values associated with the discrete nodes D_1, D_2 whilst $f(B_1)$, $f(C_1)$, $f(U_1)$ are the probability density function associated to the continuous interval node B_1, the probabilistic node C_1 and the hybrid node U_1, respectively. Considering the Markov condition, hence the independence of the node D_1 from the continuous node C_1, the solution of the integral in Eq.(11) is reduced to:

$$P(D1|D2) = \int_{B_1, C_1, U_1} p(D_2|B_1, D_1, C_1, U_1)f(B_1)f(C_1)f(U_1)dB_1 dC_1 dU_1 \quad (12)$$

Since the state of the node D_2 can be expressed as domain in the outcome space of the nodes B_1 D_1 C_1 and U_1, the integral in Eq.(12) can be than expressed as:

$$P(D1|D2) = \int_{\Omega_{D2,d1}^{d2}} f(B_1)f(C_1)f(U_1)dB_1 dC_1 dU_1 \quad (13)$$

where $\Omega_{D2,d1}^{d2}$ is the domain that defines the event $D2 = d2$ in the space of B_1 ,C_1 and U_1 given $D1 = d1$. The integral in Eq.(13) appears in the form common to

structural reliability problems and can be easily solved using structural reliability methods, e.g. Monte Carlo methods, Line Sampling, First Order Reliability method etc. [34]. The evaluation of Eq.(13) allows to remove the dependency of D_2 from B_1, C_1 and U_1.

4. Identification and removal of barren nodes. These latter are nodes of any nature which do not have children and do not receive any evidence. They can be removed from the initial network without any consequence for the overall analysis (B_1, C_1 and U_1 in the previous example). The removal of these leads to the structure of the reduced network shown on the right hand side of Fig.6.

The use of reliability methods to reduce the initial network present the further advantage of relaxing the hypothesis of strong independence among non-discrete variables not linked by edges. Well known methods (e.g., Rosenblatt, Nataf transformations etc.) are available in literature to deal with the computation of reliability involving correlated variables. The general approach for addressing this issue is to convert the correlated variables into uncorrelated standard Gaussian variables. This allows hence to take into account in the computation the correlation existent among continuous nodes even if not directly connected by causal links, as far as they are input of the same reliability problem (i.e., have a common child). This is clearly valid also for the computation of continuous variables.

3.3 Inference computation

A CN containing only interval probabilities can be interpreted as an infinite series of traditional BNs sharing the same graph but characterized by different crisp values of conditional probabilities. In a BN the probability of any instantiation $\mathbf{e}$, $P(\mathbf{e})$, is a linear function of any CPTs parameter $p(x|u)$ [8] [9]. This allows to identify a finite number of significant *crisp networks* (as they will referred from now on) each of which is associated with a different combination of the initial interval probability bounds, so with one traditional BN embraced by the CN.

A naive approach for the derivation of exact probability bounds on a query variable of interest for a CN including only crisp and interval probabilities, consists of applying

standard exact inference methods to each of these crisp networks and finally to minimize and maximize the results obtained. For instance, a CN containing n boolean nodes (i.e., with only two complementary outcomes) described through probability intervals can be decomposed in 2^n traditional BNs (or *crisp networks*). Each of these networks shares the same graphical structure of the initial model and is associated with a set of CPTs whose parameters are selected among the bounds of the initial conditional probability intervals, according to the 2^n possible combinations available. This approach is available in the toolbox proposed, but it is largely ineffective and can become easily prohibitive for larger networks due to combinatorial explosion.

In order to overcome this limitation, a further method, previously defined [36], has been included in the toolbox. This is based on the identification of the *crisp networks* (and hence the combinations of probability bounds in input) that actively contribute to the calculation of the query bounds. On the contrary, most of the combinations result in probability values lying within the query interval and are hence ineffective with regards to the aim of the calculation. The method proposed allows to compute the exact bounds of the query probability when no evidence is introduced in the network. In the case of evidence instead, it provides intervals defining the exact location of the true bounds of the query probability. The methodology is limited to the use of boolean variables. The approach can be summed out in the following steps:

1. Firstly, the computation of the upper bounds for the joint probability of the interval probability variables is carried out. This step implies the inference computation of two 'artificial' crisp networks (containing respectively all the upper bounds and all lower bounds of the initial interval parameters) that can be carried out using any traditional exact inference algorithm.

2. The second phase consists in the identification of the bounds combinations which are more likely to maximize and minimize the bounds of the query variable in output. This is realized analysing the joint probability matrix previously obtained, trough the verification of specific conditions [36].

3. Computation of outer bounds of the query variables. The query probability bounds are computed through the marginalization of the joint probability dis-

tribution computed on the 'artificial' networks (step 1). This low-cost calculation results in an outer approximation of the true bounds of the query variables, which are hence included within the probability interval identified.

4. Computation of inner bounds of the query variables. Inference is computed for the query variable on the *crisp networks* identified in step 3, through the use of well-known exact inference algorithms. The results of this analysis coincide with the true bounds of the query probability if no evidence is introduced in the network. Otherwise, the bounds identified are an inner approximation of the true values, hence lying within the true probability interval.

3.4 Sensitivity Analysis

Sensitivity analysis can be performed on the reduced network to identify the uncertainties in input that need to be reduced so that the resulting model satisfies the constraint posed by the user regarding the uncertainty of the output. In other words, given a CN including interval probabilities, the purpose of the computation is to quantify the maximum imprecision level of the input that satisfies the constraint on the uncertainty of the probabilistic query, according to a method previously developed [35]. The methodology adopted focuses on single parameter changes, where only one network parameter is modified in order to satisfy the uncertainty constraint. Hence, each change involves a single probability bound of one node's CPT entry and the co-varying bound that must be changed coherently in order to satisfy the sum-to-one constraint.

In a traditional Bayesian Network, the marginal probability $P(E_1)$ of a generic node state E_1 is a linear function of any parameter $p(X_j|U_l)$ in the network [8]. This makes possible to investigate the global impact of parameter changes, by first bounding the derivative of a query with respect to the parameter under study, then bounding the change in a query due to an arbitrary parameter change [9].

The method adopted is based on the extension of this consideration to the case of CN including only crisp and interval probabilities. On this basis, it can be demonstrated [37] that, given a constraint k on the length of the confidence interval of the query E_1, such that $P(\overline{E_1}) - P(\underline{E_1}) < k$, and identified the *crisp network* m which results in such

bounds, the parameter change $\Delta p_m(X_1|U_l)$ to be applied on the input associated with node X_1 to satisfy the constraint is:

$$\Delta p_m(X_1|U_l) \geq \frac{P(\overline{E_1}) - P(\underline{E_1}) - k}{(P(\underline{E_1}) + k)(\pi_{\mathbf{X}|U_l}^{\underline{E_2}} - \pi_{\mathbf{X}|U_l}^{\overline{E_1}})} \tag{14}$$

where U_l represents the instantiation of the parents of X_1, $P(\overline{E_1})$ and $P(\underline{E_1})$ the bounds of the interval of the query E_1 to be reduced; $\pi_{\mathbf{X}|U_l}^{\underline{E_2}}$ and $\pi_{\mathbf{X}|U_l}^{\overline{E_1}}$ are the derivatives of the query bounds $P(\underline{E_2})$ and $P(\overline{E_1})$ respectively, with respect to the parameter $p_m(X_1|U_l)$. These can be calculated as:

$$\pi_{\mathbf{X}|U_l}^{\underline{E_2}} \equiv \frac{\delta P(\underline{E_2})}{\delta p_m(X_1|U_l)} = \frac{p_m(E_2, X_1, U_l)}{p_m(X_1|U_l)} - \frac{p_m(E_2, X_2, U_l)}{1 - p_m(X_1|U_l)} \tag{15}$$

$$\pi_{\mathbf{X}|U_l}^{\overline{E_1}} \equiv \frac{\partial P(\overline{E_1})}{\partial p_m(\mathbf{X}|U_l)} = \frac{p_m(E_1, X_1, U_l)}{p_m(X_1|U_l)} - \frac{p_m(E_1, X_2, U_l)}{1 - p_m(X_1|U_l)} \tag{16}$$

if $p_m(X_1|U_l) \neq 0$ and $p_m(E_1, X_2, U_l) \neq 0$.

4 Computational Toolbox

The methods described in Section 3 have been implemented into the open computational framework of the Cossan software, i.e. OpenCossan. OpenCossan is a collection of methods and tools under continuous development, coded exploiting the object-oriented MATLAB programming environment [24], hence ensuring programming flexibility and avoiding code duplication. It provides efficient computational methods and allows to define specialized solution sequences including optimisation, uncertainty quantification, reliability based optimisation etc. Furthermore, thanks to the strong flexibility, new reliability methods or optimization algorithms can be easily added.

The toolbox is organized in classes, i.e. data structures consisting of data fields and methods together with their interactions and interfaces [29]. Objects (i.e., instances of classes) can be then easily aggregated, forming more complex objects and being processed according to the related methods in order to obtain the output of interest. The numerical implementation consists mainly of two classes: the first, *Node*, provides the basic input of the graphical model. The combination of more Node objects allows
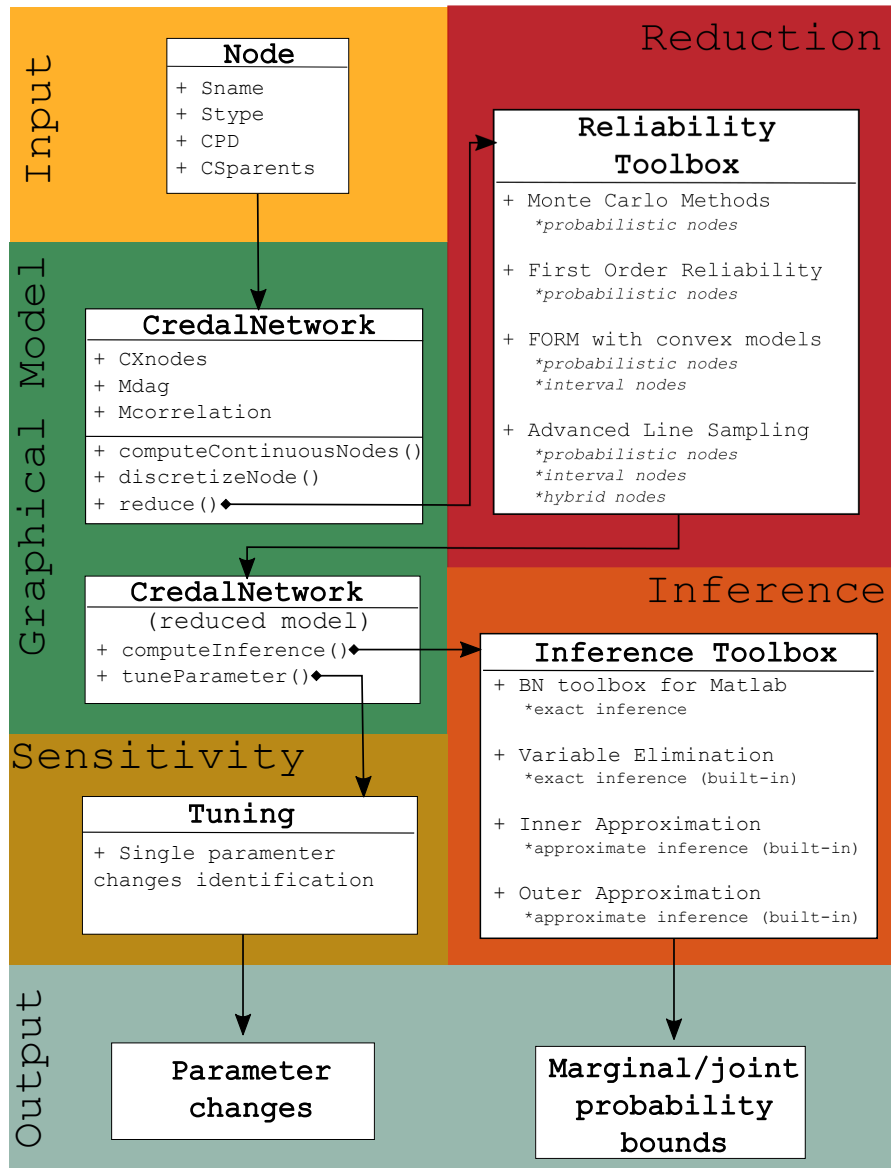
Figure 7: Simplified representation of the computational toolbox

Property	Type	Required	Description
Sname	String	Yes	Name of the node
CSparents	Cell array of strings	Yes	Name of parent nodes
CSchildren	Cell array of strings	No	Name of children nodes
CPD	Cell array	Yes	Conditional Probability Distribution
Nsize	Integer	No	Size of the node
Stype	String	Yes	Type of node, i.e. discrete, probabilistic, bounded or hybrid
Vbounds	Array of doubles	No	Vector of bound values (in case of discrete or discretized nodes)
Evidence	Integer	No	Value of the evidence eventually introduced in the node

Table 2: Main properties of the *Node* object

the construction of *CredalNetwork* objects, defined by their namesake class. This is a general class which includes, as a particular case, Enhanced Bayesian Networks (where all the non-discrete variables are of the probabilistic type [34]), as well as traditional Bayesian Networks (where all the variables are discrete). The combination of the class *Node* with the class *Credal Network* provides the graphical and numerical implementation of the CN models defined by the user.

The reduction of the initial models is realized through the interaction of the classes implemented with reliability methods available in the OpenCossan framework. Finally, the computation of inference in the network can be carried out thanks to the interaction of the tool with the Bayes' Toolbox for MATLAB [26] or using built-in inference algorithms. Fig.7 depicts the main structure of the computational tool presented.

This section aims to give an overall description of the computational tool and methods implemented for the reduction, inference and sensitivity analysis of the initial models. In the following, Sections 4.1 and 4.2 are dedicated to the description of the two main classes of the toolbox and their methods.

4.1 Class *Node*

The basic objects defined by the user in the network design phase belong to the class *Node*. The main properties associated to an object *Node* are shown in Table 2. The standard name convention of OpenCossan has been adopted.

The nature of the Conditional Probability Distribution (CPD) depends on the node type: for discrete nodes characterized by crisp probabilities the CPDs coincide with traditional CPTs; in the case of discrete nodes associated with probability bounds, the CPDs contain probability intervals; likewise, the CPD of probabilistic nodes can contain one or more random variables and, for hybrid nodes, one or more p-boxes. OpenCossan offers a further grade of flexibility providing the option of defining the family of probabilistic distributions from a set of well-known functions specifying the moments, from data available or even from user defined functions. This option can be adopted in the case of probabilistic as well as hybrid nodes. The main methods associated to the class are briefly described in the following. When the node is child of at least one continuous node, according to the initial hypothesis (Section 3) it must be defined as a domain in the outcome space of his non-discrete parents. From a computational point of view, this results in the CPD of the *Node* object to be characterized by one or more scripts, representing the user defined models to be evaluated (through system reliability methods if discrete, through Monte Carlo methods if probabilistic etc.). These are expressed in the form of strings in which the variables involved are named according to the parent nodes' name. Once the analysis for the node to compute is concluded, the initial scripts in the CPD property are substituted with the resulting output: if the initial node was of the probabilistic type, the output will be a probabilistic distributions defined on the data obtained from the Monte Carlo analysis of the model; similarly, if the node is defined as hybrid, the final CPD will include parametric p-boxes; if the node is defined as discrete, the updated CPD will contain the reliability analysis results which are crisp values if the variable involved where only discrete or probabilistic, or probability bounds otherwise.

4.2   Class *CredalNetwork*

The object *CredalNetwork*, which contains the overall model and the related information, is built introducing as input the node objects previously defined. Correlations among continuous variables sharing a child are considered through correlation factors, even if the relative nodes are not directly connected by causal links. This tool can be

Property	Type	Required	Description
$CXnodes$	Cell array	Yes	Cell array of Node objects
$CSnames$	Cell array of strings	No	Cell array of nodes names
$CStypes$	Cell array of strings	No	Cell array of nodes type
$Vsize$	Cell array	Yes	Array of nodes size
$Mdag$	Matrix of integers	No	Directed acyclic graph adjacency matrix
$Cevidence$	Cell array	No	Cell array of evidence values
$CSobserved$	Cell array of strings	No	Cell array of observed nodes names
$Nnodes$	Integer	No	Number of nodes in the network
$Mcorrelation$	Matrix of doubles	No	Correlation matrix

Table 3: Main properties of the *CredalNetwork* object

extremely attractive for those cases in which a certain degree of correlation between two variables is detected but the characterization of the causal mechanisms behind it are unknown, poorly characterized or difficult to model (e.g., the correlation among some weather extremes, such as between extreme precipitations and extreme sea water level [44]). The Nataf transformation is adopted as a default method in the toolbox in order to take into account the variable correlation. These factors are stored in the node object through the use of a correlation matrix: this way it is possible to consider the correlation among random variables or among interval variables (in the form of convex sets) involved in the same reliability analysis as mentioned in previous sections.

Apart form the correlation matrix and the node objects, other properties which characterize the CN object can either be defined by the user or extrapolated from the input node objects. Overall, the main property of an CN object are defined in Table 3. The methods of the class *CredalNetwork* can be divided in four categories according to the nature of the task to be carried out: methods dedicated to the modelling of the network, involved in the reduction procedure, in the inference computation or in the sensitivity analysis computation. Not included in this classification is the method dedicated to the visualization of the graphical model, based on the use of the biograph toolbox for MATLAB.

Modelling The methods of this category have the capability of directly modify the graphical structure (and subsequently the associated numerical values) of the network.

Figure 8: Graphical representation of the discretization procedure: on the left hand side the initial structure, on the right hand side the structure resulting from the discretization of the node A_2

Hence, this includes

- *barrenNodes*: method for the identification and removal of barren nodes (namely those which do not receive any inference and have no children and so do not give any contribution towards the computation of the model)

- *addNodes*: method for the inclusion of new nodes (e.g., those newly defined from the reduction procedure)

- *removeNodes*: method for the elimination of nodes from the network (e.g., probabilistic, hybrid and bounded nodes after the reduction procedure)

- *discretizeNode*: method for the dicretization of continuous nodes

The latter, plays an essential role when a non-discrete variable needs to be included in the reduced network. More generally, it is an important tool of modelling strategy (e.g., to preserve causal links in the reduced network, to take into account the inference introduced in continuous nodes).

Listing 1: Definition of the network in Fig.8a and its discretization

```
% DEFINITION OF THE NETWORKS NODE
XA1=Node('Sname','A1','CPD',CPD_A1,'Stype','discrete',...
    'Nsize',2, 'Lroot', true);
XA2=Node('Sname','A2','CPD',CPD_A2,'Stype','probabilistic',...
    'Nsize',2, 'CSparents',{'A1'});
XA3=Node('Sname','A3','CPD',CPD_A3,'Stype','discrete',...
    'Nsize',2, 'CSparents',{'A2'});
XA4=Node('Sname','A4','CPD',CPD_A4,'Stype','probabilistic',...
    'Nsize',2, 'CSparents',{'A2'});
% BUILT THE NETWORK
XCN=CredalNetwork('Sdescription','Example of CN in Fig.3a',...
        'CXnodes',{XA1,XA2,XA3,XA4});
% GRAPHICAL REPRESENTATION (Fig.3a)
XCN.makeGraph
% DISCRETIZATION OF THE NETWORK
XCNdiscretized=XCN.discretizeNode('Snodename','A2');
% GRAPHICAL REPRESENTATION (Fig.3b)
XCNdiscretized.makeGraph
```

The procedure suggested by Straub and Kiureghiam [34] has been implemented computationally and further extended to the discretization of interval variables. It consists in the substitution of the initial node to discretize (A_2 in the example in Fig.8a) with two nodes: the first generally discrete (referred as primary node from now on), which inherits the parents of the initial node ($A_{2discrete}$ in Fig.8b) and the second continuous ($A_{2continuous}$ in Fig.8b), referred as secondary node. This latter results to be a child of the first one and inherits the children of the initial node (A_3 and A_4 in Fig.8). The related code is shown in Listing 1.

Any discretization process implies splitting the initial continuous domain into more sub-domains (which correspond to the outcome states of the discretized node), each of which is defined by bounds. These bounds can be either defined by the user or computed on the basis of the initial domain. In this latter case, the default setting consists of splitting the initial domain into five sub-domains of equivalent length. If the initial domain is unbounded, a numerical approximation is assumed for the external bounds, assumed to

be equal to -10^{22} and 10^{22}. With regards to the discretization of probabilistic nodes, such as that shown in the example of Fig.8, the probability measures stored in the CPD associated with the primary node ($A_{2discrete}$) are computed as the cumulative probability values over the sub-domains defined. Conversely, in the case of bounded nodes, the sub-domains obtained according to the bounds provided are assumed to be equally possible. Finally, if the node to discretize is hybrid, the primary node will be a bounded node and its CPD will contain interval probabilities associated to each state.

In the cases mentioned, with the only exception of hybrid nodes, the information carried by the initial continuous variables is not lost but rearranged along the new sub-domains associated with the secondary node: each outcome state of the primary node is associated to a random variable stored in the CPD of the secondary node, which in the new set-up results to be its child. The distributions stored in the secondary nodes are built from the distribution of the initial node subject to discretization and can be considered sections of this latter over the specific sub-domains of discretization. This way neither the parents of the node to discretize neither its children are significantly affected by the discretization procedure and their CPDs remain unchanged.

Reduction   The main method included in this group is *reduce2BN*, which provides the reduction of the initial network according to the procedure described in Section 3.2. This is accomplished trough the use of a wide range of secondary methods which are invoked within *reduce2BN* and hence remain hidden to the user:

- *computeContinuousNodes*. This provides the computation of probabilistic, bounded and hybrid nodes which are children of non-discrete nodes and hence initially characterized as domains in the outcome space of their parents. According to the nature of the node to compute and to the parent variables involved, different approaches can be adopted:

  - probabilistic nodes: Monte Carlo methods are used to collect and process samples from the parent nodes distributions for the construction of the resulting probabilistic distributions. As default, a sample 10^6 large is adopted for such purpose, aiming to find a balance between the accuracy of the anal-

31

ysis and its computational cost.

- bounded nodes: a random search is carried out in the parents domain in order to find the bounds of the new interval variables.

- hybrid nodes: the current implementation of the toolbox does not provide the computation of hybrid nodes: since the algorithms here presented are limited to the use of parametric p-boxes no methods for the computation of p-boxes as a result of operations involving p-boxes or probabilistic and interval variables are available. Nevertheless, hybrid nodes can be modelled as children of probabilistic nodes and interval variables as far as these latter refer to uncertain parameters of the probabilistic distributions stored in the CPD of the first.

Due to the limitations of the current implementation in terms of hybrid nodes computation, any continuous node child of at least one hybrid node is computed as a bounded node (e.g., through random search), hence loosing part of the information available.

- *probabilisticSRM*. It provides the computation of discrete nodes whose parent nodes are characterized exclusively by probabilistic distributions or crisp probability values. In this case, probabilistic models are built for each combination of the parents instantiation and the analysis can be carried out adopting approximate methods (e.g., First Order Reliability Method) or simulation based methods, such as a range of Monte Carlo, Line Sampling and Importance Sampling.

- *hybridSRM*. This method provides the computation of discrete nodes whose parent nodes are associated with both probabilistic and non-probabilistic variables. If interval and random variables are involved, the entries of the new discrete CPT can be computed using either the Advanced Line Sampling method (default option) or the generalized FORM, briefly described in Section 2.3. If hybrid nodes are involved, the only option currently available relies on the adoption of the Advanced Line Sampling method.

The selection of the approach adopted in the computation, when more than one is

Node Type	Parent Nodes	Method	New CPD
Discrete	Probabilistic Discrete	Monte Carlo or FORM	Crisp values
Discrete	Probabilistic Bounded Discrete	FORM with convex sets	Probability upper bounds
Discrete	Probabilistic Bounded Discrete	Advanced Line Sampling	Probability bounds
Discrete	Probabilistic Hybrid Bounded Discrete	Advanced Line Sampling	Probability bounds
*Bounded**	Probabilistic Hybrid Bounded Discrete	Random Search	Interval(s)
*Probabilistic**	Probabilistic Discrete	Monte Carlo	Probabilistic Distribution(s)

*to be removed after children computation

Table 4: Methods for the reduction of the initial CN according to the nature of the nodes involved in the reliability analysis

suitable, is driven by the user choice of focusing on near-real time output rather than more accurate results or vice versa. Table 4 summarizes the methods available for the computation of nodes according to their nature and that of their parents.

Inference The main method of this group is called *computeInference*, and allows the user to select among three main degrees of accuracy for the inference computation of the reduced network:

- option *ApproximateCoarse*: invokes the computation of the outer approximation of true bounds, as described in Section 3.3, which is associated with the lowest computational cost among the methods available.

- option *ApproximateFine*: invokes the computation of the inner approximation of true bounds, as described in Section 3.3. When no evidence is introduced in the network, the computation results in the identification of the true bounds of the query probability.

- options *Exact* or *BNT*. Both the options refer to the adoption of a combinatorial approach for exact inference computation. While the first option refers to the use of the built-in inference algorithm (i.e., Variable Elimination), the second allows the interaction of the toolbox with the Bayes' toolbox for MATLAB [26]. In this case, the user can further specify the preferred exact inference algorithm among those included in the software (e.g., Variable Elimination, Junction Tree, etc.). This option ensures the maximum accuracy of the analysis.

Sensitivity analysis   The method *tuneParameter* allows to identify the suitable parameters changes which verify the constraint imposed by the user, according to the methodology described in Section 3.4. The method requires the definition of a constraint on the length of the confidence bounds (i.e., upper limit to the uncertainty affecting the output of interest) of a target node. Candidate nodes can be also defined, in order to restrict the search for possible changes only to specific variables. If this information is not entered in the analysis, the method scans each conditional probability interval in input in order to find the possible network modifications which would ensure the respect of the query constraint. The results provide an intuitive understanding of when parameter changes do or do not matter in terms of query robustness. The main computational cost is related to the computation of inference for the query variable of interest and the subsequent identification of the relevant *crisp networks*.

4.3   Limits of the current implementation

The current implementation of the toolbox presents some limitations. First, as mentioned in Section 3, the algorithms available do not provide the computation of non parametric p-boxes. This restricts the models implemented to the use of parametric p-boxes for the characterization of user-defined hybrid nodes. Moreover, the method available for the sensitivity analysis is currently restricted to the tuning of single network parameters and relies on exact inference computation, which can result high expensive for not-elementary CNs. Nevertheless, the limitations of the current toolbox are compensated by the high flexibility of the computational implementation, which can easily integrate new and more efficient methods when available. Further research is being
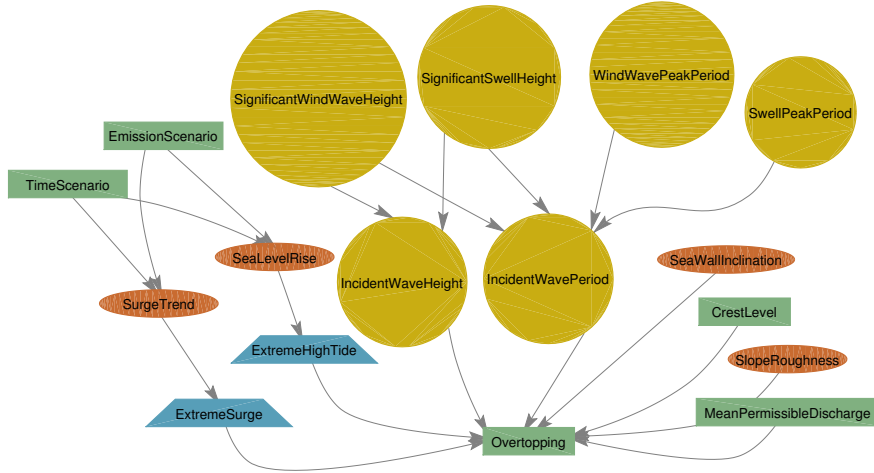
Figure 9: Overview of the CN for sea wave overtopping risk assessment

carried out to overcome the mentioned restrictions and enhance the current implementation.

5 Numerical Example

This section is dedicated to the description of the implementation and analysis of a simple CN model carried out using the computational toolbox proposed in this study. The network analysed is a modified version of a model previously implemented for the sea wave overtopping hazard quantification [35]. The model takes into consideration the effect of sea level rise and future surge trends on the probability of sea wave overtopping. In the following, the network structure is briefly explained followed by a detailed description of the data adopted for the case study and the results obtained.

5.1 Model Implementation

The network shown in Fig.9, consists of 17 nodes. As shown in Listing 2, a matrix (*Mcorrelation*) and a cell array (*CNodes*) are initially defined in order to collect the correlation factors and the *Node* objects respectively. Table 7 sums up the characteristics of the model nodes.

Table 5: Storylines of the SRES emission scenarios considered

Emission Scenario	Social Change	Economic Change	Technological change
A1B	Global population that peaks in mid-century then declines	Very rapid economic growth	Balanced progress across all resources and technologies from energy supply to end use, as well as balanced land-use changes
A1f1	Global population that peaks in mid-century then declines	Very rapid economic growth	Fossil fuel-intensive future, with significant rise of global carbon dioxide emissions

Listing 2: Definition of Initial Variables

```
%% INITIALISE VARIABLES

Nnodes          = 17;               % Nodes in the net

Mcorrelation    = eye(Nnodes);      % Correlation Matrix

CNodes          = cell(1,Nnodes);   % Cellarray of Node objects

n               = 0;                % Node index
```

The discrete nodes *TimeScenario* and *EmissionScenario* allow to model different time periods and emission scenarios, respectively. Each of these is based on several assumptions about future emissions of greenhouse gases and other pollutants from human activities, according to the UKCP09 classification. The analysis of different time scenarios and possible emission trends allows to make projections of UK climate change over the next century. SRES provide a set of comprehensive global narratives, or storylines, that define local, regional and global socio-economic driving forces of change such as economy, population, technology, energy and agriculture — key determinants of the future emissions pathway [27].

In order to test the computational algorithms implemented, the model has been applied to a case study similar to that proposed by Reis et al. [32] and refers to a hypothetical structure sited in Liverpool Bay in the south-eastern corner of the Irish Sea. Nine outcome states referring to as many decades between 2020 to 2100 have been adopted

Table 6: Characteristics of the distributions adopted for the probabilistic nodes of the network

Node	Type	Mean	STD	Lower Limit	Correlation Factor
$WindWavePeakPeriod$	Weibull	6.4	1.15	4.2	0.6 ($WindWaveHeight$)
$SwellPeakPeriod$	Weibull	13.12	3.51	0	−0.1 ($SwellHeight$)
$WindWaveHeight$	Weibull	1.2	0.7	0.45	0.6 ($WindWavePeakPeriod$) 0.2 ($SwellHeight$)
$SwellHeight$	Weibull	0.342	0.221	0	−0.1 ($SwellPeakPeriod$) 0.2 0.2 ($WindWaveHeight$)

for the node $TimeScenario$. Similarly, two different emission scenarios have been considered, namely $A1B$, generally known as medium emission scenario, and $A1f1$, which assumes a high emission trend. The assumptions behind these scenarios are described in Table 5. Moreover, the $A1B$ scenario has been assumed to be potentially more likely, with probability bounds $[0.6, 0.9]$, while the high emission scenario has been associated with a probability interval of $[0.1, 0.4]$. All this info is stored in the $Node$ objects $TimeScenario$ and $EmissionScenario$ as shown in Listing 3.

Listing 3: Definition of scenarios nodes

```
%% SCENARIOS NODES
%01-TimeScenario: [2020:10y:2100]
n=n+1;
CPDXTimeScenario    = cell(1,9);
CPDXTimeScenario(:) = {1/9};
CNodes{n}           = Node('Sdescription','Time scenarios',...
    'Sname','TimeScenario','Lroot',true,...
    'CPD',CPDXTimeScenario,'Nsize',9,'Stype','discrete');
%02-EmissionScenario: 1=medium(A1B), 2=high(A1f1)
n=n+1;
CPDXEmissionScenarioLow       = cell(1,2);
CPDXEmissionScenarioUp        = cell(1,2);
CPDXEmissionScenarioLow(1,1)  = {0.6};
CPDXEmissionScenarioLow(1,2)  = {0.1};
CPDXEmissionScenarioUp(1,1)   = {0.9};
CPDXEmissionScenarioUp(1,2)   = {0.4};
CNodes{n}       = Node('Sdescription','Emission Scenarios',...
    'Sname','EmissionScenario','Lroot',true,'Lpbounds',true,'CPD',...
    {CPDXEmissionScenarioLow,CPDXEmissionScenarioUp},...
    'Nsize',2,'Stype','discrete');
```

WindWavePeakPeriod, *SwellPeakPeriod*, *SignificantWindWaveHeight*, *SignificantSwell-Height*, *SurgeLevel* are the roots of the network and represented by probabilistic distributions. This is justified by the large record dataset usually available for sea conditions and their variability, which can be well described by random variables. Nevertheless, it is possible to substitute probabilistic nodes with bounded or hybrid ones if the data available are not suitable for the construction of precise probabilistic distributions.

To each combination of the outcomes of the nodes *TimeScenario* and *EmissionScenario* corresponds a specific sea level rise interval (stored in the nodes *SeaLevelRise*) and *SurgeTrend*), which takes into consideration the uncertainty associated with the predictions available adopting as lower and upper bound the 5%ile and 95%ile of the projected values respectively. The nodes *SeaLevelRise* and *SurgeTrend* are combined with the baseline distributions for the extreme tide and extreme surge: the interval
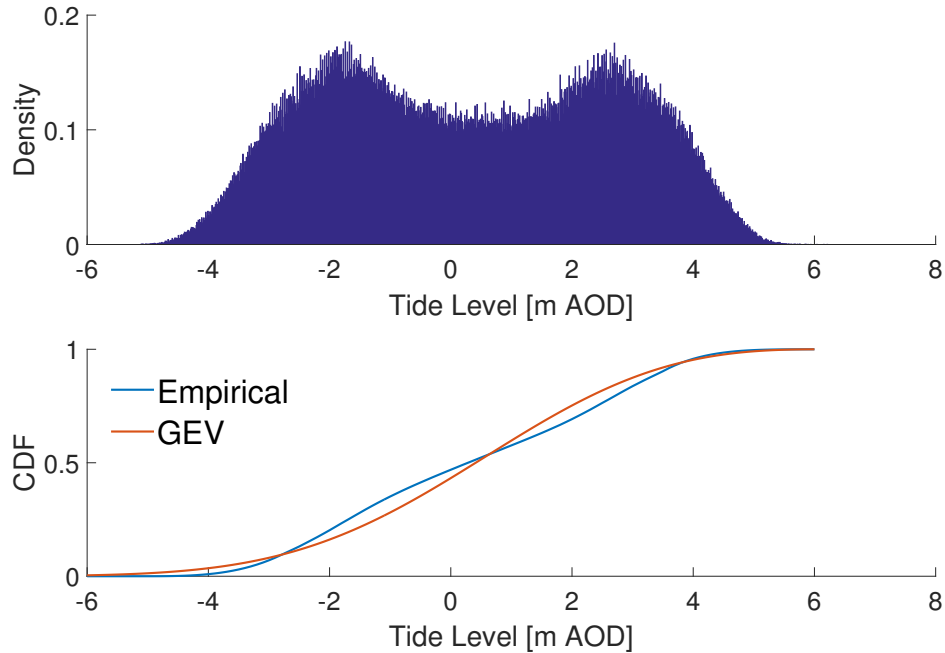
Figure 10: Density of data available for the tide level in Liverpool Bay and related generalized extreme value distribution (compared to the empirical CDF) adopted in the model as baseline for the node *ExtremeTide*

values stored in these nodes are assumed as means for the distributions of the hybrid nodes *ExtremeTide* and *ExtremeSurge* adopted in the analysis.

The probability distribution of extreme tide level at Liverpool has been implemented on the basis of the data provided by the the British Oceanographic Data Centre and combined with the sea level rise projections obtaining a parametric p-box. Fig. 10 shows the empirical probability distribution for the data and the relative p-boxes assumed to represent the nowadays distribution for each emission scenario considered. The related code implemented is shown in Listing 4.

Listing 4: Definition of nodes associated to extreme tide levels

```matlab
%% EXTREME TIDE LEVEL
n=n+1;
%Load Data from external sources
Mmedium = importdata('MediumEmissionScenario.txt');
Mhight = importdata('HighEmissionScenario.txt');
Vyears=30:10:110; %2020:10:2100;
CPDXSeaLevelRise    = cell(9,2,1);
for itime=1:9
    % lowerBound=5%ile  upperBound=95%ile
    CPDXSeaLevelRise(itime,1,1)=...
    {Interval('lowerBound',Mmedium(Mmedium(:,1)==Vyears(itime),2),...
    'upperBound',Mmedium(Mmedium(:,1)==Vyears(itime),4))};
    CPDXSeaLevelRise(itime,2,1)=...
    {Interval('lowerBound',Mhight(Mhight(:,1)==Vyears(itime),2),...
    'upperBound',Mhight(Mhight(:,1)==Vyears(itime),4))};
end
CNodes{n}      = Node('Sdescription','Emission Scenarios',...
    'Sname','SeaLevelRise','CPD',CPDXSeaLevelRise,....
    'CSparents',{'TimeScenario','EmissionScenario'},...
    'Nsize',1,'Stype','bounded');
%04- TideLevel
n=n+1;
CPDXTideLevel{1,1}=RandomVariable('Sdistribution','gev',....
'par1',-0.3591,'par2',2.3791,'par3',-0.4063);
CNodes{n}=Node('Sdescription','Surge contribution to water level',...
    'Sname','ExtremeHighTide','CSparents',{'SeaLevelRise'},...
    'CPD',CPDXTideLevel,'Nsize',1,'Stype','hybrid',...
    'Cmapping',{'sealevelrise','extremehightide','mean'});
```

A similar procedure has been followed for the implementation of the node *ExtremeSurge*: the baseline for extreme surge events has been combined with the UKCP09 projections regarding the surge trend for the scenarios considered. Unfortunately only predictions regarding the medium emission scenario (A1B1) are available, while most studies aiming to quantify future surge trends have resulted in high model disagreement for the geographical area of interest with regards to the other emission scenario considered [39] [22].

To take into account our ignorance, a symmetric interval centred in the baseline mean has been considered for the low and high emission scenarios in the representation of the node *ExtremeSurge*. See Listing 5 for details about the related computational code.

Listing 5: Definition of nodes associated to extreme surge levels

```
%05-ExtremeSurgeTrend > 5%and95%ile of 10y return period
n=n+1;
CPDXSurgeLevel=cell(9,2,1);
for itime=1:9
    CPDXSurgeLevel(itime,1,1)=...
    {Interval('lowerBound',0.019 -0.123*(2+((itime-1)*10))*0.001,...
    'upperBound',0.019 +0.728*(31+((itime-1)*10))*0.001)};
    CPDXSurgeLevel(itime,2,1)=...
    {Interval('lowerBound',0.019 -0.08,...
    'upperBound',0.019+0.08)};
end
CNodes{n}=Node('Sdescription','Surge contribution to water level',...
    'Sname','SurgeTrend','CPD',CPDXSurgeLevel,...
    'CSparents',{'TimeScenario','EmissionScenario'},...
    'Nsize',1,'Stype','bounded');
%06- ExtremeSurge
n=n+1;
CPDXSurge{1,1}=RandomVariable('Sdistribution','Gumbel',...
'mean',0.019,'std',0.192);
CNodes{n}=Node('Sdescription','Surge contribution to water level',...
    'Sname','ExtremeSurge','CSparents',{'SurgeTrend'},...
    'CPD',CPDXSurge,'Nsize',1,'Stype','hybrid',...
    'Cmapping',{'surgetrend','extremesurge','mean'});
```

The values adopted for the probabilistic user-defined nodes of the model are coherent with those provided by Reis et al. [32] and are summarized in Table 6. A linear coefficient of 0.6 has been adopted to take into account the correlation between the peak period and the significant wave height of wind sea waves. Moreover, lower limits have been considered for the distributions of these variables to account for calm conditions (i.e., assuming that no wave overtopping occurs when there are no locally generated waves). A correlation factor of 0.2 has been considered to represent the correlation

Table 7: Characteristic of the nodes involved in the analysis

Node	Type	Size
SeaWallInclination	Bounded	1
CrestLevel	Discrete	1
SlopeRoughness	Bounded	1
WindWavePeakPeriod	Probabilistic	1
SignificantWindWaveHeight	Probabilistic	1
SwellPeakPeriod	Probabilistic	1
SignificantSwellHeight	Probabilistic	1
IncidentWaveHeight	Probabilistic	1
IncidentWavePeriod	Probabilistic	1
SurgeLevel	Hybrid	1
TideLevel	Hybrid	1
MeanPermissibleDischarge	Discrete	1
Overtopping	Probabilistic	1
TimeScenario	Discrete	9
EmissionScenario	Discrete	3
SeaLevelRise	Bounded	1
SurgeTrend	Bounded	1

between the wind-sea and the swell [32]. For simplification purposes no near shore wave transformation model has been adopted. Furthermore, all the waves are supposed to approach the seawall normally. Listings 6 and 7 show the sections of the computational code defining the wind waves and swell waves nodes, respectively.

The overall behaviour of the sea waves, resulting from the combination of wind waves and swell characteristics, is represented by the nodes *IncidentSignificantHeight* and *IncidentPeakPeriod*. These are computed according to the model suggested in Reis et al. [32], which is explicitly implemented within the network, as shown in Listing 8.

The bounded roots of the network refer to the slope characteristics such as the angle at which the seawall front slope is built (*SeaWallInclination*) and its roughness (*SlopeRoughness*); indeed, intervals allow to model construction tolerances. The node (*CrestLevel*) refers to the seawall characteristic and it is also an input of the model (i.e., defined by the user) together with the *MeanPermissibleDischarge* and the root nodes previously mentioned.

Listing 6: Definition of nodes associated with wind waves

```
%% SEA WAVE CONDITIONS
%07- WINDWAVEPeakPeriod
n=n+1;
CPDXWavePeakPeriod{1}=RandomVariable('Sdistribution','Weibull',...
    'Parameter1',6.868,'Parameter2',6.511,'lowerBound',4.2);
CNodes{n}=Node('Sdescription','Wind waves peak period',...
    'Sname','WindWavePeakPeriod','Lroot',true,'CPD',CPDXWavePeakPeriod,...
    'Nsize',1,'Stype','probabilistic','censoring',4.2);
%08- WINDWAVESignificantWaveHeight
n=n+1;
CPDXSignificantWindWaveHeight{1}=
        RandomVariable('Sdistribution','Weibull',...
    'Parameter1',1.348,'Parameter2',1.771,'lowerBound',0.45);
CNodes{n}=Node('Sdescription','Wind waves significant height',...
    'Sname','SignificantWindWaveHeight','Lroot',...
    true,'CPD',CPDXSignificantWindWaveHeight,'Nsize',1,...
    'Stype','probabilistic','censoring',0.45);
Mcorrelation(n,n-1)=0.6;
Mcorrelation(n-1,n)=0.6;%correlation between Hw and Tw
```

The wall is assumed characterized by a slope roughness between 0.7 and 0.8 over a range $[0, 1]$ and the seawall slope is supposed to be gentle, with an inclination angle between 0.4 and 0.6 rad. Due to possible failures or wearing of the structure, the probability for the maximum height of the wall to remain equal to 10 m AOD along with all time scenarios is assumed to fall in the interval $[0.3, 0.5]$. A second more conservative outcome refer to a maximum height guaranteed to be not lower than 8 m and is associated with probability value between 0.5 and 0.7.

The discrete node *Overtopping* is associated with the probability of overcoming the *MeanPermissibleDischarge*, which represents the mean permissible overtopping discharge per unit length of seawall and has been considered equal to 0.1 l/s. The definition of these two nodes within the toolbox is shown in Listing 10. Finally, once all the nodes of the model are defined, the object of the class *CredalNetwork* can be implemented (see Listing 11).

Listing 7: Definition of nodes associated with swell waves

```matlab
%09- SWELLWAVEPeakPeriod
n=n+1;
CPDXSwellPeakPeriod{1}=RandomVariable('Sdistribution','Weibull',...
    'Parameter1',1.443e+01,'Parameter2',4.215e+00);
CNodes{n}=Node('Sdescription','Swell peak period',...
    'Sname','SwellPeakPeriod','Lroot',true,'CPD',CPDXSwellPeakPeriod,...
    'Nsize',1,'Stype','probabilistic');
%10- SWELLWAVESignificantWaveHeight
n=n+1;
CPDXSignificantSwellHeight{1}=RandomVariable('Sdistribution','Weibull',...
    'Parameter1',3.811e-01,'Parameter2',1.583e+00);
CNodes{n}=Node('Sdescription','Swell significant height',...
    'Sname','SignificantSwellHeight','Lroot',true,...
    'CPD',CPDXSignificantSwellHeight,'Nsize',1,'Stype','probabilistic');
Mcorrelation(n,n-1)=-0.1;
Mcorrelation(n-1,n)=-0.1;  %correlation between Hs and Ts
Mcorrelation(n,n-2)=0.2;
Mcorrelation(n-2,n)=0.2;    %correlation between Hs and Hw
```

Listing 8: Definition of nodes associated with combined sea waves

```matlab
%11- COMBINED INCIDENT SIGNIFICANT HEIGHT
n=n+1;
CPDIncidentWaveHeight(1,1,1)=...
        {'sqrt((TableInput.significantswellheight.^2)+...
        (TableInput.significantwindwaveheight.^2))'};
CNodes{n}=Node('Sdescription','Combined incident significant height',...
    'Sname','IncidentWaveHeight','CPD',CPDIncidentWaveHeight,...
    'CSparents',{'SignificantSwellHeight','SignificantWindWaveHeight'},...
    'Nsize',1,'Stype','probabilistic');
%12- COMBINED INCIDENT PEAK PERIOD
n=n+1;
CPDXIncidentWavePeriod(1,1,1,1,1)=
        {'(((TableInput.significantswellheight.^2)./...
        ((TableInput.significantswellheight.^2)+...
        (TableInput.swellpeakperiod.^4))).*...
        (TableInput.significantswellheight.^2)+...
        ((TableInput.significantwindwaveheight.^2)./...
        ((TableInput.significantswellheight.^2)+...
        (TableInput.significantwindwaveheight.^2))).*...
        (TableInput.windwavepeakperiod.^4)).^1/4'};
CNodes{n}=Node('Sdescription','Combined incident peak period',...
    'Sname','IncidentWavePeriod','CPD',CPDXIncidentWavePeriod,...
    'CSparents',{'SignificantSwellHeight','SignificantWindWaveHeight',...
    'SwellPeakPeriod','WindWavePeakPeriod'},'Nsize',1,...
    'Stype','probabilistic');
```

Listing 9: Definition of nodes associated with the structure

```matlab
%% SEA WALL CHARACTERISTICS
%13- Sea Wall Inclination
n=n+1;
CPD_alpha(1,1)={Interval('lowerBound',0.4,'upperBound',0.6)};
CNodes{n}=Node('Sdescription','Inclination of the sea wall',...
    'Sname','SeaWallInclination','Lroot',true,'CPD',CPD_alpha,...
    'Nsize',1,'Stype','bounded');
upperCOT=cot(CPD_alpha{1,1}.lowerBound);
lowerCOT=cot(CPD_alpha{1,1}.upperBound);
%14- CrestLevel state#1= 8m, state#2=10m
n=n+1;
CPDXCrestLevelLow         = cell(1,2);
CPDXCrestLevelUp          = cell(1,2);
CPDXCrestLevelLow(1,1)    = {0.5};
CPDXCrestLevelLow(1,2)    = {0.3};
CPDXCrestLevelUp(1,1)     = {0.7};
CPDXCrestLevelUp(1,2)     = {0.5};
CNodes{n}        = Node('Sdescription','Sea wall crest level',...
    'Sname','CrestLevel','Lroot',true,...
    'CPD',{CPDXCrestLevelLow CPDXCrestLevelUp},'Nsize',2,...
    'Stype','discrete','Lpbounds',true); %,'Value',10
%15- Roughness
n=n+1;
CPD_r(1,1)={Interval('lowerBound',0.7,'upperBound',0.8)};
CNodes{n}        = Node('Sdescription','Sea wall Roughness',...
    'Sname','SlopeRoughness','Lroot',true,...
    'CPD',CPD_r,'Nsize',1,'Stype','bounded');
```

Listing 10: Definition of nodes involved in the overtopping model

```
%% EXTERNAL MODELS
%16- MeanPermissibleOvertoppingDischarge
n=n+1;
CNodes{n}=Node('Sdescription','Parameter C of the model',...
    'Sname','MeanPermissibleDischarge','Lroot',true,...
     'CPD',{1},'Nsize',1,'Stype','discrete','Vvalues',0.1);
%17- Overtopping
n=n+1;
CPDXOvertopping(1,1,1,1,1,1,1,1,1)={[]};
CPDXOvertopping(1,1,1,1,1,1,1,1,2)=
        {'TableInput.meanpermissibledischarge-...
        ComputeOvertoppingRateEBNJournal(TableInput.incidentwaveheight,...
        TableInput.incidentwaveperiod,(TableInput.extremehightide+...
        TableInput.extremesurge),8,TableInput.seawallinclination,...
         TableInput.sloperoughness)'};
CPDXOvertopping(1,1,1,1,1,1,2,1,1)={[]};
CPDXOvertopping(1,1,1,1,1,1,2,1,2)=
    {'TableInput.meanpermissibledischarge-
    ComputeOvertoppingRateEBNJournal(TableInput.incidentwaveheight,...
    TableInput.incidentwaveperiod,(TableInput.extremehightide+...
    TableInput.extremesurge),10,TableInput.seawallinclination,...
    TableInput.sloperoughness)'};
Cparent={'MeanPermissibleDischarge','ExtremeHighTide',...
        'ExtremeSurge','IncidentWaveHeight','IncidentWavePeriod',...
        'SlopeRoughness','CrestLevel','SeaWallInclination'};
MImpDir=[-[1 1 1 1 1 1];1 1 1 1 1 1];
CNodes{n}       = Node('Sdescription','Overtopping event',...
    'Sname','Overtopping','CSparents',Cparent, 'CPD',...
    CPDXOvertopping,'Nsize',2,'Stype','discrete',..
    'MimportantDirection',MImpDir,'Lboolean',true);
```
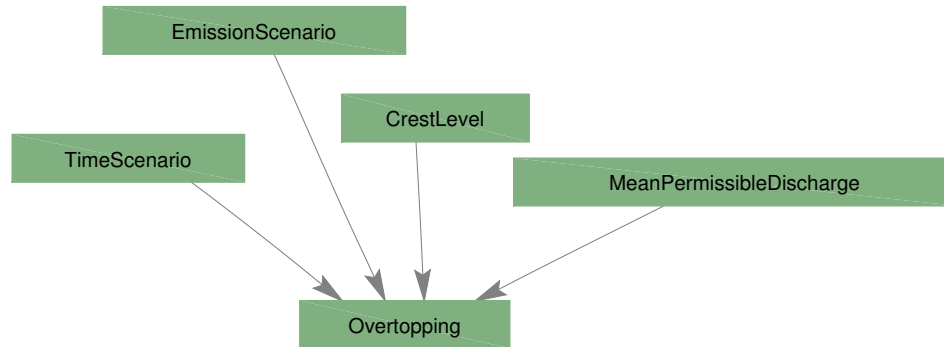
47

Figure 11: Overview of the reduced network

Listing 11: Definition of the Credal Network object

```matlab
%% BUILD CN
XCN=CredalNetwork('Sdescription','Overtopping risk assessment',...
        'CXnodes',CNodes,'Mcorrelation',Mcorrelation);
```

5.2 Analysis

The reduction of the initial network (9) has been invoked as shown in Listing 12, and was performed adopting the Adaptive Line Sampling method. The resulting reduced network (*XCNreduced* in the computational code) is shown in Fig.11. It consists of only 5 nodes, 3 of which (i.e., *EmissionScenario*, *CrestLevel*, *Overtopping*) are associated with interval probabilities. The remaining nodes, *TimeScenario* and *MeanPermissibleDischarge*, are characterized by crisp probabilities.

Listing 12: CN reduction

```matlab
XALS=AdaptiveLineSampling('Nlines',20,'NmaxPoints',14);
XCNreduced=XCN.reduce2BN('Cxsimulation',{XALS});
```

A complete analysis has been carried out on the model proposed, starting from the inference computation of the network. This was carried out focusing on the marginal probability of the event *Overtopping* and adopting both exact and approximate algorithms, comparing the results obtained by the three approaches described in Section
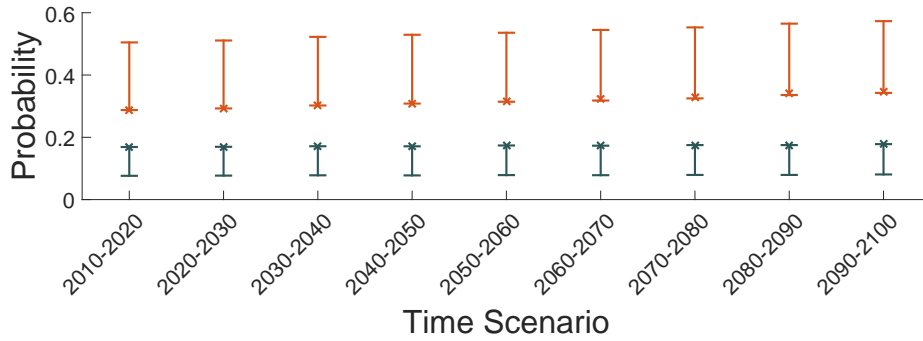
48

Figure 12: Exact upper (square) and lower (circle) bounds and related inner and outer approximations (error bars) for the probability of occurrence of overtopping for each time scenario analysed

3.3. The related part of the code is presented in Listing 13.

Listing 13: Inference analysis

```
XCNreduced.computeInference('CSmarginal',{'Overtopping'},...
        'Lbnt',true);
XCNreduced.computeInference('CSmarginal',{'Overtopping'},..
        'LapproximateInner',true);
XCNreduced.computeInference('CSmarginal',{'Overtopping'},...
        'LapproximateOuter',true);
```

Regardless the scenario of reference, hence not introducing any evidence, the marginal probability of overtopping occurrence results bound by the interval $[0.173, 0.316]$, according to the exact inference analysis. The output obtained through the adoption of the inner approximation algorithm perfectly matches these values, while the outer approximation results equal to 0.0783 and 0.538, respectively for the lower and upper bounds. These estimates refer to the overall scenarios considered, providing a general value for the probability of sea wave overtopping over the entire century considered. Introducing evidence in the node *TimeScenario*, it is instead possible to visualize the trend of the risk over more specific time domains.

As shown in Fig.12 and from the results in Table 8, both the upper and lower bounds of the probability of overtopping occurrence generally tend to grow in time regardless the inference computation method adopted. Also the length of the intervals grows along the time domain (e.g., from a minimum of 0.1186 in the first scenario to 0.1695 in the

Table 8: Probability intervals for the marginal probability of overtopping occurrence associated with the time scenario of reference

Scenario	Exact	Inner	Outer
$2010 - 2020$	$[0.1689, 0.2875]$	$[0.1689, 0.2875]$	$[0.0764, 0.5047]$
$2020 - 2030$	$[0.1695, 0.2928]$	$[0.1695, 0.2928]$	$[0.0770, 0.5108]$
$2030 - 2040$	$[0.1713, 0.3022]$	$[0.1713, 0.3022]$	$[0.0780, 0.5225]$
$2040 - 2050$	$[0.1711, 0.3083]$	$[0.1711, 0.3083]$	$[0.0777, 0.5292]$
$2050 - 2060$	$[0.1737, 0.3143]$	$[0.1737, 0.3143]$	$[0.0786, 0.5357]$
$2060 - 2070$	$[0.1731, 0.3219]$	$[0.1731, 0.3181]$	$[0.0784, 0.5448]$
$2070 - 2080$	$[0.1750, 0.3293]$	$[0.1750, 0.3249]$	$[0.0792, 0.5529]$
$2080 - 2090$	$[0.1751, 0.3402]$	$[0.1751, 0.3358]$	$[0.0792, 0.5650]$
$2090 - 2100$	$[0.1781, 0.3476]$	$[0.1781, 0.3426]$	$[0.0807, 0.5730]$

last, referring to the exact intervals) coherently with the increasing uncertainty of the climate projections. In all the scenarios considered, the inner approximation values matches the estimate of the exact probability bounds, while the outer approximation provides only a coarse estimation of these, in particular with regards to the upper values of the probability. In spite of this, the information provided by this analysis, in particular when coupled with that provide by the inner approximation, can be extremely useful in case of complex networks where an exact estimate would require a much larger computational power.

Fig.13 shows the results according not only to the time scenario but also to the two different emission scenarios considered. The probability bounds results slightly higher for the high emission scenario, as expected. Moreover, the length of the intervals is generally larger for the high emission scenario, highlighting an higher degree of uncertainty related to the projections associated with this assumption. Finally the sensitivity analysis of the network has been carried out, according to the methodology previously described (see Listing 14).

Listing 14: Inference analysis

```
XCNreduced.tuneParameter('Starget','Overtopping',...
'query',InitialUncertainty*0.80);
```

Regardless the scenario of reference (i.e., without introducing any evidence in the network), the possible parameter changes which can reduce the length of the exact probability interval for the overtopping event to 80% of its initial value (i.e., a length of the
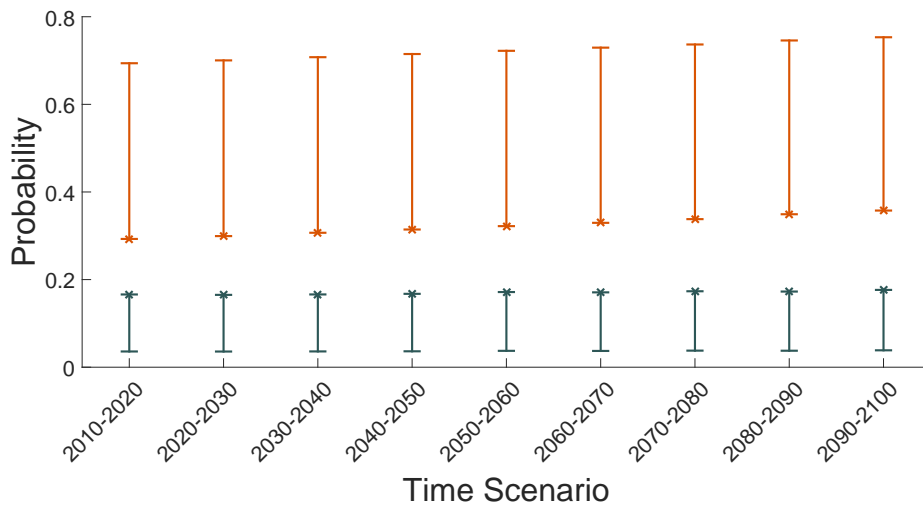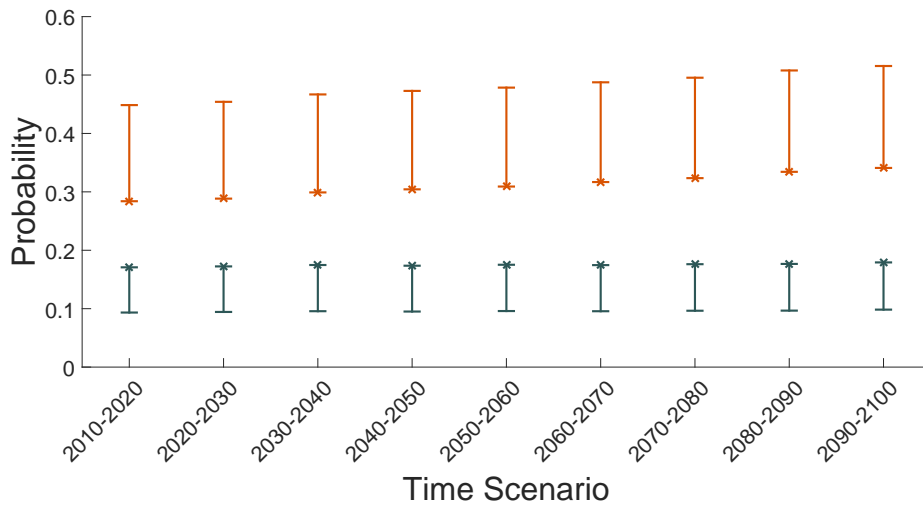
Figure 13: Exact and approximate probability bounds for the different time periods and for the medium and high emission scenarios

interval equal to 0.1145) involve the node *CrestLevel*. Two solutions are possible in this case, as shown in Table 9: the first involves the reduction of the upper bound regarding the probability of the maximum height of the structure to reach 8 m of 21.92% of its initial value (i.e., reducing the probability interval associated with the event of 76.39% of its initial value); in this case the marginal probability interval associated with the event *Overtopping* would be $[0.173, 0.287]$. In the second case, the upper bound associated to the maximum height of the structure to reach 10 m is reduced of 38.46% (i.e., 95% of the initial uncertainty affecting the parameter), resulting in a marginal probability for the event *Overtopping* of $[0.201, 0.316]$. Hence, even if both options guarantee the same

Table 9: Parameter changes for reducing *Overtopping* marginal probability uncertainty to 80%

Parameter (τ)	Absolute Change	Relative Change
$p(\overline{CrestLevel=8})$	-0.15346	-21.92%
$p(\overline{CrestLevel=8})$	-0.1923	-38.46%

reduction of uncertainty in output, they are not equivalent in terms of reliability of the structure: the first option implies a reduction of the overall probability of overtopping while the second the increase. In order to enhance the accuracy of the information in output, reducing the imprecision affecting the marginal probability obtained, more precautions could be taken in order to increase the confidence in the robustness of the structure (hence its capability of keep a maximum height of 10 m for all the time domain considered), decreasing the probability of overtopping occurrence as well as the uncertainty affecting its estimate.

6   Conclusions

A novel computational toolbox for the manipulation of Credal Networks and their analysis has been presented. The toolbox integrates well-known traditional, cutting-edge and novel methods and allows to interact with external software (such as the reliability toolbox in OpenCossan and the Bayes' Toolbox for MATLAB) ensuring extreme flexibility in the analysis. Algorithms are provided for the reduction of the initial Credal Networks including probabilistic, discrete, bounded and hybrid nodes to equivalent

simplified models. The reduced networks include only nodes which are associated with crisp probability values and interval probabilities, streamlining the structure of the network. The inference computation can be carried out adopting different methods (both built-in or from third parties) characterized by different degrees of accuracy and, hence, associated to different computational costs. Finally, sensitivity analysis methods allow identifying the best possible strategy, in terms of tuning of single network parameters with regards to the required maximum level of imprecision in output. Thanks to the high flexibility of the computational tool, new methods can be integrated easily in the existent framework, making room for constant improvements of the implementation. A numerical example based on a simple CN model for the risk assessment of sea waves overtopping is analysed in order to fully test the capabilities of the toolbox implemented.

Aknowledgements

References

[1] A. Antonucci, C.P. de Campos, D. Huber, and M. Zaffalon. Approximating credal network inferences by linear programming. In *Proceedings of the 12th European Conference, ECSQARU 2013*, volume 7958 of *Lecture Notes in Computer Science*, pages 13–24, New York, U.S.A., 2013. Springer.

[2] A. Antonucci, A. Salvetti, and M. Zaffalon. Assessing debris flow hazard by credal nets. In *Soft Methodology and Random Information Systems*, pages 125–132. Springer, 2004.

[3] Alessandro Antonucci, Andrea Salvetti, and Marco Zaffalon. Credal networks for hazard assessment of debris flows. *Advanced Methods for Decision Making and Risk Management in Sustainability Science*, pages 237–256, 2007.

[4] M. Beer, S. Ferson, and V. Kreinovich. Imprecise probabilities in engineering analyses. *Mechanical Systems and Signal Processing*, 37(1–2):4 – 29, 2013.

[5] Y. Ben-Haim. A non-probabilistic concept of reliability. *Structural safety*, 14(4):227–245, 1994.

[6] 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*, 71(3):249 – 260, 2001.

[7] A. Cano, F.G. Cozman, T. Lukasiewicz, M. Gómez, S. Moral, and J. Abellán. Reasoning with imprecise probabilities hill-climbing and branch-and-bound algorithms for exact and approximate inference in credal networks. *International Journal of Approximate Reasoning*, 44(3):261 – 280, 2007.

[8] E. Castillo, J.M. Gutiérrez, and A.S. Hadi. Sensitivity analysis in discrete bayesian networks. *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, 27(4):412–423, 1997.

[9] H. Chan. *Sensitivity analysis of probabilistic graphical models*. PhD thesis, University of California, Los Angeles, 2005.

[10] F.G. Cozman. Credal networks. *Artificial Intelligence*, 120(2):199–233, 2000.

[11] F.G. Cozman. Separation properties of sets of probability measures. In *Proceedings of the Sixteenth conference on Uncertainty in artificial intelligence*, pages 107–114. Morgan Kaufmann Publishers Inc., 2000.

[12] F.G. Cozman. Graphical models for imprecise probabilities. *International Journal of Approximate Reasoning*, 39(2):167–184, 2005.

[13] M. de Angelis, E. Patelli, and M. Beer. Advanced line sampling for efficient robust reliability analysis. *Structural Safety*, 52:170–182, 2015.

[14] E. Fagiuoli and M. Zaffalon. 2U: an exact interval propagation algorithm for polytrees with binary variables. *Artificial Intelligence*, 106(1):77–107, 1998.

[15] S. Ferson, V. Kreinovich, L. Ginzburg, D.S. Myers, and K Sentz. Constructing probability boxes and Dempster-Shafer structures. Technical Report SAND2002-4015, Sandia National Laboratories, Albuquerque, NM, January 2003.

[16] R. Ghanem, D. Higdon, and H. Owhadi, editors. *Handbook of Uncertainty Quantification*, chapter COSSAN: A Multidisciplinary Software Suite for Uncertainty Quantification and Risk Management, pages 1–69. Springer, New York, U.S.A., 1 edition, 2017.

[17] M.S. Hamada, A. Wilson, C.S. Reese, and H. Martz. *Bayesian reliability*. Springer Science & Business Media, 2008.

[18] J.M. Hammersley and D.C. Handscomb. *Monte carlo methods*, volume 1. Methuen London, 1964.

[19] A.M. Hasofer and N.C. Lind. Exact and invariant second-moment code format. *Journal of the Engineering Mechanics division*, 100(1):111–121, 1974.

[20] C. Jiang, X. Han, G.Y. Lu, J. Liu, Z. Zhang, and Y.C. Bai. Correlation analysis of non-probabilistic convex model and corresponding structural reliability technique. *Computer Methods in Applied Mechanics and Engineering*, 200(33):2528–2546, 2011.

[21] H.E. Kyburg Jr. Interval-valued probabilities, 1998. SIPTA Documention on Imprecise Probability `http://www.sipta.org/documentation/interval_prob/kyburg.pdf`.

[22] J. Lowe, T. Howard, A. Pardaens, J. Tinker, J. Holt, S. Wakelin, G. Milne, J. Leake, J. Wolf, and K. Horsburgh. *UK Climate Projections science report: Marine and coastal projections*, chapter 4. Met Office, Exeter, U.K., June 2009.

[23] Y. Luo, Z. Kang, and A. Li. Structural reliability assessment based on probability and convex set mixed model. *Computers & Structures*, 87(21):1408–1415, 2009.

[24] MathWorks. MATLAB [computer software]. Natick, MA.

[25] R. Moore and W. Lodwick. Interval analysis and fuzzy set theory. *Fuzzy Sets and Systems*, 135(1):5–9, 2003.

[26] K. Murphy. The Bayes net toolbox for MATLAB. *Computing science and statistics*, 33(2):1024–1034, 2001.

[27] Met Office. UK climate projections, 2013. data available from:`http://ukclimateprojections.metoffice.gov.uk/`.

[28] E. Patelli, M. Broggi, M. de Angelis, and M. Beer. OpenCossan: an efficient open tool for dealing with epistemic and aleatory uncertainties. *Vulnerability, Uncertainty, and Risk: Analysis, Modeling, and Management*, pages 2564–2573, 2014.

[29] E. Patelli, H.M. Panayirci, M. Broggi, B. Goller, P. Beaurepaire, H.J. Pradlwarter, and G. I. Schuëller. General purpose software for efficient uncertainty management of large finite element models. *Finite Elements in Analysis and Design*, 51:31–48, 2012.

[30] J. Pearl and S. Russell. Bayesian networks. Technical Report R-277, UCLA Cognitive Systems Laboratory, November 2000. MIT Press, Cambridge, MA, 157–160.

[31] R. Rackwitz. Reliability analysis: a review and some perspectives. *Structural safety*, 23(4):365–395, 2001.

[32] M.T. Reis, T.S. Hedges, A. Williams, and K. Keating. Specifying seawall crest levels using a probabilistic method. *Proceedings of the ICE-Maritime Engineering*, 159(4):137–145, 2006.

[33] S.J. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach*, volume 25. Prentice-Hall, Egnlewood Cliffs, 1994.

[34] D. Straub and D. Kiureghian. Bayesian network enhanced with structural reliability methods: methodology. *Journal of engineering mechanics*, 136(10):1248–1258, 2010.

[35] S. Tolo, E. Patelli, and M. Beer. Enhanced bayesian network approach to sea wave overtopping hazard quantification. In *Proceedings of the 25th European Safety and Reliability Conference, ESREL*, Zurich, Switzerland, 2015. Eidgenössische Technische Hochschule Zürich.

[36] S. Tolo, E. Patelli, and M. Beer. An inference method for bayesian networks with probability intervals. *Structural Safety*, Submitted on July 2016.

[37] S. Tolo, E. Patelli, and M. Beer. Sensitivity analysis for bayesian networks with interval probabilities. In *Proceedings of the 26th European Safety and Reliability Conference, ESREL, 25th – 29th September 2016*, Glasgow, U.K., to be published.

[38] F. Tonon. Using random set theory to propagate epistemic uncertainty through a mechanical system. *Reliability Engineering & System Safety*, 85(1):169–181, 2004.

[39] M.I. Vousdoukas, E. Voukouvalas, A. Annunziato, A. Giardino, and L. Feyen. Projections of extreme storm surge levels along europe. *Climate Dynamics*, pages 1–20, 2016.

[40] P. Weber, G. Medina-Oliva, C. Simon, and B. Iung. Overview on bayesian networks applications for dependability, risk analysis and maintenance areas. *Engineering Applications of Artificial Intelligence*, 25(4):671–682, 2012.

[41] K. Weichselberger. The theory of interval-probability as a unifying concept for uncertainty. *International Journal of Approximate Reasoning*, 24(2):149–170, 2000.

[42] L.A. Zadeh. Fuzzy sets. *Information and control*, 8(3):338–353, 1965.

[43] M. Zaffalon, K. Wesnes, and O. Petrini. Reliable diagnoses of dementia by the naive credal classifier inferred from incomplete cognitive data. *Artificial Intelligence in Medicine*, 29(1):61–79, 2003.

[44] F. Zheng, S. Westra, and S.A Sisson. Quantifying the dependence between extreme rainfall and storm surge in the coastal zone. *Journal of Hydrology*, 505:172–187, 2013.