8th International Symposium on Imprecise Probability: Theories and Applications, Compiègne, France, 2013

# Dynamic Credal Networks: Introduction and Use in Robustness Analysis

**Matthieu Hourbracq**[1]
matthieu.hourbracq@lip6.fr

**Cédric Baudrit**[2]
cbaudrit@grignon.inra.fr

**Pierre-Henri Wuillemin**[1]
pierre-henri.wuillemin@lip6.fr

**Sébastien Destercke**[3]
sebastien.destercke@hds.utc.fr

## Abstract

Dynamic Bayesian networks (DBN) are handy tools to model complex dynamical systems learned from collected data and expert knowledge. However, expert knowledge may be incomplete, and data may be scarce (this is typically the case in Life Sciences). In such cases, using precise parameters to describe the network does not faithfully account for our lack of information. This is why we propose, in this paper, to extend the notion of DBN to convex sets of probabilities, introducing the notion of dynamic credal networks (DCN). We propose different extensions relying on different independence concepts, briefly discussing the difficulty of extending classical algorithms for each concept. We then apply DCN to perform a robustness analysis of DBN in a real-case study concerning the microbial population growth during a French cheese ripening process.

## 1 Introduction

Dynamic Bayesian networks (DBNs) [36] extend Bayesian networks (BNs) [37, 38] and form a convenient formalism to describe complex dynamical systems. They also extend the well-known Hidden Markov Models (HMMs) [40] by representing the hidden state and the observation in terms of several random variables. The probabilistic and graphical natures of DBNs make them attractive tools to integrate both expert knowledge and data in a single representation. The concept of DBNs makes possible to (i) combine different sources of knowledge; (ii) easily modify the model thanks to its modular nature and (iii) integrate uncertainties. However, one limitation of DBNs lies in the specification of parameters that requires a substantial knowledge that is seldom available. This is particularly the case when experimental data are costly, such as in Life Sciences.

One way to overcome this difficulty is to use *credal sets* [35, 44], i.e., convex sets of probabilities to model the lack of knowledge about the parameters. Applied to Bayesian networks, this idea corresponds to the concept of credal networks (CN) [17, 19], in which each node of the network is associated to a convex set of conditional probabilities (possibly degenerated to a single element). Other approaches such as possibilistic [3] or evidential networks [45] follow the same objective but cannot be interpreted as a proper extension of classical Bayesian Networks.

While the notion of credal network has received much attention in the past years, it is not the case for its dynamic extension. Indeed, the only works dealing with such extension consider specific models related to Markov Processes [22, 26], in which computations on the full dynamic network can be done separately for each time-step. Although such cases are of high interest and can benefit from efficient algorithms, there are many other cases where one will need to perform inferences on a complete network not reducible to a Markov model. This is especially the case in Life and Food Sciences [1, 39], where the modelling of non-linear, multi-scale dynamic processes (maturation processes, evolution of interacting physicochemical phenomena, ...) is often based on qualitative expert knowledge and on limited experimental data. The use of Dynamic Credal networks (DCNs) extending DBNs seems a good way to integrate such heterogeneous and scarce knowledge.

The goal of this paper is two-fold: first to provide in Section 3 a first theoretical and practical discussion of the DBNs extension into DCNs, second to apply in Section 4 the DCNs framework to achieve a robustness analysis of

[1] Laboratoire d'Informatique de Paris VI (UPMC, CNRS UMR7606) 75004 Paris, France
[2] UMR782 Génie et Microbiologie des Procédés Alimentaires. INRA/AgroParisTech, 78850 Thiverval-Grignon, France
[3] Université de Technologie de Compiegne U.M.R. C.N.R.S. 7253 Heudiasyc Centre de recherches de Royallieu F-60205 Compiegne Cedex FRANCE

DBNs in a real-world case study involving the growth of yeast population during the Camembert-type cheese ripening. Preliminary notions are briefly recalled in Section 2.

## 2 Preliminary notions : DBN and CN

### 2.1 Dynamic Bayesian Networks

DBNs are classical Bayesian networks in which nodes $\{X_i(t), i = 1 \dots n\}$, representing (discrete) random variables, are indexed by discrete time $t$. They provide a compact representation of the joint probability distribution $P$ for a finite time interval $[\![1, \tau]\!]$ (we use $[\![i, j]\!]$ to denote the finite set of time indices $\{i, \dots, j\}$) defined as follows:

$$P(\mathbf{X}(1) \dots, \mathbf{X}(\tau)) = \prod_{i=1}^{n} \prod_{t=1}^{\tau} P\left(X_i\left(t\right) \mid \mathbf{U}_i\left(t\right)\right) \quad (1)$$

where $\mathbf{U}_i(.)$ denotes the set of parent nodes of a node $X_i(.)$ and $P\left(X_i\left(t\right) \mid \mathbf{U}_i\left(t\right)\right)$ denotes the conditional probability function associated with the random variable $X_i(t)$ given $\mathbf{U}_i(t)$. $\mathbf{X}(t) = \{X_1(t), \dots, X_n(t)\}$, is called a "slice" and represents the set of all variables indexed by the same time $t$. This joint probability $P(\mathbf{X}(1), \dots, \mathbf{X}(\tau))$ represents the beliefs about possible trajectories of the dynamic process $\mathbf{X}(t)$. DBNs assume the *first-order Markov property* which means that the parents of a variable in time slice $t$ must occur in either slice $t-1$ or $t$ :

$$\mathbf{U}_i(t) \subset \mathbf{X}(t-1) \cup \mathbf{X}(t) \backslash \{X_i(t)\} \quad (2)$$

Moreover, the conditional probabilities are time-invariant (*first-order homogeneous Markov property*):

$$P\left(X_i\left(t\right) \mid \mathbf{U}_i\left(t\right)\right) = P(X_i(2) | \mathbf{U}_i(2))) \ , \forall t \in [\![2, \tau]\!]. \quad (3)$$

To specify a DBN, we need to define the intra-slice topology (within a time slice), the inter-slice topology (between two time slices), as well as the parameters, *i.e* conditional probabilities in Equation (3) for the first two time slices.

In this paper, we consider that $\mathbf{X}_i(t)$ are all discrete variables. Faced with continuous variables $X_i$, these ones will be discretized. Let $P^t_{ijk}$ be the probability that $X_i(t) = k$, given that its parents have instantiation $j$, *i.e.*

$$P^t_{ijk} = P\left(X_i(t) = k \mid \mathbf{U}_i(t) = j\right), \quad (4)$$

for $i \in \{1, \dots, n\}$, $j \in \{1, \dots, c_i\}$ where $c_i$ is the number of distinct configurations of $\mathbf{U}_i(t)$ and $k \in \{1, \dots, r_i\}$ where $r_i$ is the number of values that node $i$ can take.

#### 2.1.1 Parameter learning and local elicitation

The techniques for learning DBNs are generally extensions of the techniques for learning BNs. Different methods exist to learn the structure or the parameters from substantial and/or incomplete data (for overviews, we refer readers to [5, 34]). In our case, we consider that the topology is given (e.g., learned from expert knowledge).

The most commonly used and simplest method which will be used in this paper is to estimate $P^t_{ijk}$ by the occurrence rate of the event $(X_i(t) = k, \mathbf{U}_i(t) = j)$ in a training database :

$$P^t_{ijk} = N^t_{ijk} / \textstyle\sum_k N^t_{ijk} \quad (5)$$

where $N^t_{ijk}$ denotes the number of times where the event $(X_i(t) = k, \mathbf{U}_i(t) = j)$ occurs in database. As we assume the first-order homogeneous Markov property (3), $P^t_{ijk}$ does not depend on time and we can rewrite

$$\forall t' \in [\![2, \tau]\!], \ P^{t'}_{ijk} = \frac{\sum_t N^t_{ijk}}{\sum_t \sum_k N^t_{ijk}} \quad (6)$$

In the case where $N^t_{ijk} = 0$ for all $k$, the uniform distribution is traditionally used as it maximizes the Shannon entropy and corresponds to the Laplace indifference principle.

A practical methodology able to incrementally build and update model parameters from heterogeneous information has been developed in [2] on the basis of Dirichlet model. From a given network structure, it consists in using a priori Dirichlet distributions which are then updated through Bayesian inference by expressing new pieces of information into a frequentist form. This method also integrates the confidence level on the different sources of information.

#### 2.1.2 Knowledge propagation - inference

The use of DBNs consists in "query" expressed as conditional probabilities. The most common task we wish to solve is to estimate the marginal probabilities

$$P\left(\mathbf{X}_Q(t') | \{\mathbf{X}_E(t), \forall t \in [\![1, \tau]\!]\}\right), \forall t' \in [\![1, \tau']\!] \quad (7)$$

where $\mathbf{X}_Q$ is a set of query variables, and $\mathbf{X}_E$ is a set of evidence variables. Inference consists in computing the probability of each state of a variable when we know the state of other variables. In general, DBN inference is performed using recursive operators and Bayes' theorem that update the belief state of the DBN as new observations become available [36]. Due to the natural time ordering of the modelled process, $\mathbf{U}_i(t)$ will usually be observed before $X_i(t)$ and that may help with the sequential updating of the conditional probabilities as well as with the preservation of the original conditional independence structure.

### 2.2 Credal Networks and strong extension

A credal network (CN) [17, 19] is an extension of BNs where imprecision is introduced in probabilities by means of credal sets [35]. CNs specify a closed convex set $K(\mathbf{X})$ of multivariate probability mass functions over the whole

set of variables $\mathbf{X}$. Under the *strong extension* [17] hypothesis, the joint credal set $K(\mathbf{X})$ over $\mathbf{X}$ may be formulated as:

$$K(\mathbf{X}) = CH\left\{P(\mathbf{X}) : P(\mathbf{X}) = \prod_{i=1}^{n} P_i, P_i \in K_i\right\} \quad (8)$$

where $CH$ denotes the convex hull, $P_i = P(X_i \mid \mathbf{U}_i)$ and $K_i = K(X_i \mid \mathbf{U}_i)$ is the closed convex set of probability mass function for the random variable $X_i$ given $\mathbf{U}_i$. In practice, it is sufficient to focus on the extreme points $ext[K(X_i \mid \mathbf{U}_i)]$ of $K(X_i \mid \mathbf{U}_i)$ in Eq. (8). In our experiments, we will limit ourselves to credal sets specified by means of probability intervals [25], that is for all $i = 1 \dots n$ and $j = 1 \dots c_i$:

$$K_{ij} = CH\left\{\begin{array}{l} P_{ij} : P_{ijk} \in [\underline{P}_{ijk}, \overline{P}_{ijk}] \subseteq [0,1],\ \forall k \\ \sum_k P_{ijk} = 1 \end{array}\right\} \quad (9)$$

This model has the advantage of creating a small number of extreme points provided additional constraints : $\forall k\ :\ \overline{P}_{ijk} - \underline{P}_{ijk} = \{0, \epsilon\}$ and $\epsilon = 1 - \sum_k \underline{P}_{ijk}$ is a constant. For such a linear-vacuous mixture, the number of vertices of $K(X_i \mid \mathbf{U}_i = j)$ is precisely the cardinality of $X_i$ – assuming there is no modality $k$ for which $\overline{P}_{ijk} = \underline{P}_{ijk}$ in which case $\mid X_i \mid$ is an upper bound – each vertex corresponding to the selection of a modality $k$ for which $P(X_i = k \mid \mathbf{U}_i = j) = \overline{P}_{ijk}$ and therefor $\forall k' \neq k : P\left(X_i = k' \mid \mathbf{U}_i = j\right) = \underline{P}_{ijk'}$.

Inferences on a credal network comes down to assess lower and upper probabilities, that is search bounds of $P(\mathbf{X}_Q|\mathbf{X}_E)$ within $K(\mathbf{X})$ (under the strong extension hypothesis) for some values of $\mathbf{X}_Q$.

# 3 Dynamical Credal Networks (DCNs) : definitions and algorithms

This section introduces the notion of Dynamic Credal Networks (DCNs) and discusses their features.

## 3.1 Definition of Dynamic Credal Networks (DCNs)

A dynamic credal network is a DBN where conditional probabilities $P(X_i(t) \mid \mathbf{U}_i(t))$ (noted $P_i^t$) are replaced by credal sets $K(X_i(t) \mid \mathbf{U}_i(t))$ (noted $K_i^t$). We assume the same *first-order Markov property* (2) as in DBNs (parents only originate from same or previous time slice) and Eq. (3) becomes

$$K(X_i(t) \mid \mathbf{U}_i(t)) = K(X_i(2) \mid \mathbf{U}_i(2)),\ \forall t \in [\![2, \tau]\!]. \quad (10)$$

Therefore, specifying a DCN requires the same effort as a DBN but allows the user to provide conditional credal sets rather than probabilities if these latter cannot be reliably estimated (from data and/or experts).

## 3.2 Independence in DCN

When working with probability sets rather than precise probabilities, the notion of stochastic independence can be extended in several ways [15]. Within graphical models, the most commonly used extension is *strong independence*, that induces the strong extension defined in Eq. 8. It can be interpreted as a robust model of a precise yet ill-known BN.

This is in contrast with the notions of epistemic irrelevance and independence whose semantic as belief models is clearer. However, these notions encounters severe computational difficulties [18], limiting their practical interest. Recent results show that for particular models such as Hidden Markov ones, efficient algorithms can be used [22], however they remain intractable for the kind of models considered in this paper. This is why we focus on extending the notion of strong extension to dynamic schemes.

The most straightforward extension is to simply apply strong independence to the whole network, i.e.,

$$K(\mathbf{X})_{st} = CH\left\{P(\mathbf{X}) : P(\mathbf{X}) = \prod_{i=1}^{n}\prod_{t=1}^{\tau} P_i^t, P_i^t \in K_i^t\right\} \quad (11)$$

where $\mathbf{X} = (\mathbf{X}(1), \dots, \mathbf{X}(\tau))$. We call this extension the *dynamic strong extension* and it is worth noting $P_i^t \neq P_i^{t'}$ is valid, $t, t' \in [\![2, \tau]\!]$.

However, when stepping to dynamic models, Condition (10) allows us to use the notion of *repetitive independence*. This condition states that if two variables $X, Y$ have the same set of possible outcomes, that is $\Omega_X = \Omega_Y$, and governed by the same probability distribution belonging to $K(X)$, then the joint credal set $K(X, Y)$ is :

$$K(X, Y) = CH\{P(X)P(X) \mid P(X) \in K(X)\}. \quad (12)$$

Adapting this notion of independence to DCN, so that probabilities of each time slice are assumed to be identical, leads to a second extension, i.e.,

$$K(\mathbf{X})_{rp} = CH\left\{\begin{array}{l} P(\mathbf{X}) : P(\mathbf{X}) = \prod_{i=1}^{n}\prod_{t=1}^{\tau} P_i^t, \\ P_i^2 \in K_i^2 \text{ and } P_i^t = P_i^2\ \forall t \in [\![2, \tau]\!] \end{array}\right\} \quad (13)$$

that we call the *dynamic repetitive extension*. We have $K(\mathbf{X})_{rp} \subseteq K(\mathbf{X})_{st}$, as $K(\mathbf{X})_{rp}$ is more constrained. In practice, the strong extension assumes that the dynamic network is ill-defined and that its behaviour can change between time slices, while the repetitive extension assumes that we seek a precise classical DBN who is partially known.

Next sections investigate the differences between these two extensions. In particular, we will see that some algorithms extend more easily to one extension than to another.

### 3.3 Inference algorithms in DCN

(D)CNs can be queried as (D)BNs were in Section 2.1.2 to get information about the state of a variable given evidence about other variables, with respect to the network *extension*. However, the use of credal sets makes the updating problem much harder, as it becomes an optimization problem. As such, the computation of the lower bound on $P(X_Q \mid \mathbf{X}_E)$ requires to minimize a quotient containing polynomials :

$$\underline{P}(X_Q \mid \mathbf{X}_E) = \min \left\{ \frac{\sum_{X_i \in \mathbf{X} \setminus X_Q \cup \mathbf{X}_E} \prod_{i=1}^{n} \prod_{t=1}^{\tau} P_i^t}{\sum_{X_i \in \mathbf{X} \setminus \mathbf{X}_E} \prod_{i=1}^{n} \prod_{t=1}^{\tau} P_i^t}, P \in K_\omega(\mathbf{X}) \right\} \tag{14}$$

with $P : P(\mathbf{X}) \in K_\omega(\mathbf{X})$ belonging to the *dynamic strong extension* ($\omega = st$) or *dynamic repetitive extension* ($\omega = rp$) of the network. An upper bound can be obtained by maximizing (14). It is known that such a minimum (or maximum) is obtained at a vertex of the *dynamic strong/repetitive extension*.

Depending on (1) the structure of network, (2) the number of modality of variables and (3) the chosen extension (strong/repetitive), the updating problem will be more or less complex to solve. Because inferences are already hard in static credal networks, little work has been done on DCNs (except for special cases already mentioned). By unrolling a two-time slice network over $T$ time steps, the number of possible vertex combinations goes from $\prod_{i,t=0} |ext[K_i^t]| \prod_{i,t=1} |ext[K_i^t]|$ (with $|ext[K_i^t]|$ the number of vertices of $K_i^t$) in the case of repetitive independence, to $\prod_{i,t=0} |ext[K_i^t]| \prod_{i,t=1} |ext[K_i^t]|^{T-1}$ in the case of strong independence. Given the potential number of vertices, approximate algorithms seem more appropriate regarding DCNs.

Many algorithms, exact and approximate, have been proposed to deal with CN. Some are generalizations of well known (D)BNs algorithms. Among the approximate algorithms, there are those that compute inner bounds, i.e. bounds that are enclosed by the exact ones, outer bounds, which enclose the exact ones, and those that perform randomly.

#### 3.3.1 Exact inference algorithms

The 2U algorithm [27] performs an exact rapid inference in the case of binary tree-shaped (D)CNs with the assumption of *strong independence*.

The CCM transformation [9] turns a (D)CN into a (D)BN by adding transparent nodes before performing an Maximum A Posteriori (MAP) estimation over the latter to find the best combination of vertices. It has the same complexity as credal network inference, that is $NP^{PP}Complete$, and performs poorly with separately specified credal networks such as the one we used during our trials (because of the sheer number of vertices).

Optimization techniques such as branch and bound over local vertices of credal sets [21, 7] are also well suited to medium-sized networks and can be stopped at any time to give an approximate answer.

Other algorithms are based on a variable elimination scheme from (D)BNs, such as Separable Variable Evaluation [20, 42] which keeps the separately specified credal sets as separated as possible during propagation, and can be mapped to an integer or a multi-linear program [24, 23].

#### 3.3.2 Approximate inference algorithms

Regarding binary and DAG-shaped (DAG : Directed Acyclic Graph) credal networks, algorithm L2U (Loopy 2U) [32] (similar to LBP (Loopy Belief Propagation) [46]) produces either inner or outer approximations, and its efficiency is mainly due to the bounded cardinality of variables and in lesser extent to ignoring loops.

Another way to handle credal sets complexity is to represent them by simpler means. Variational methods [31, 30] choose a family of functions to approximate the exact combination of credal sets to decrease computational costs. Those functions are optimized according to some criteria until convergence and the inference is then realized in the network with the original credal sets replaced by the new found functions.

The A\R(+)(+) algorithm [21] uses interval probability arithmetic to approximate credal sets in a propagation scheme in tree-shaped networks (with the use of some additional constraints limiting the information loss in its enhanced version). The intervals produced are outer bounds of the real ones. Although those algorithms are fast in medium-sized network, they either produce too many approximations or are too complex to work with DCNs.

Another popular family of approximate algorithms producing inner bounds is based on Monte-Carlo sampling [29]. Several methods have been proposed to better guide the search (simulated annealing [6], genetic algorithms [8]) among the vertices of the (conditional) local credal sets, but they require some tuning for more accurate results, otherwise they can lead to poor approximations.

Although there exist several inference algorithms, none allows to do inference, in a realistic and practical way, on networks capable of representing global complex system of Life Sciences especially in Food Sciences. Indeed, networks is composed of a large number of interacting variables capable of describing the behaviour of microscopic scales (as micro organism) involving macroscopic view (as the evolution of sensory properties). In further inferences,

we used a simple Monte-Carlo sampling algorithm [29] which has the advantage as point of reference, as it applies with the same easiness to *dynamic repetitive* and *strong extensions* (with a faster convergence for *dynamic repetitive extension*).

# 4 DCN for Robustness in DBN

In this section, we apply the concept of DCN to perform a robustness analysis of a learned precise DBN (both repetitive and strong independence concepts well correspond to this idea). We first recall some elements about robustness in classical BN before proceeding to our study.

## 4.1 Robustness in BN

Roughly speaking, a robustness analysis is the study of the behaviour of a model given small perturbations in its parameters. Robustness in Bayesian network is commonly addressed using sensitivity analysis where the main concern is to analyse the relationships between local network parameters and global conclusions drawn based on the BN. Sensitivity analysis has been largely studied by many researchers [10, 4, 14, 11]. We propose here a small survey of the main approaches.

The most common case of sensitivity analysis in BN is the study of single-parameter influence [14, 33]. In a BN, a parameter is a number in the CPT : $p(x_i|u)$ where $x_i$ is a possible value for a random variable $X$ and $u$ is a possible instantiation of the parents of $X$ in the BN. In this framework, a perturbation $\epsilon$ consists in modifying $p(X|u)$ into

$$p(x_j|u)[\epsilon] = \begin{cases} \epsilon & \text{where } i = j \\ \frac{p(x_j|u)\cdot(1-\epsilon)}{1-p(x_i|u)} & \text{otherwise.} \end{cases} \quad (15)$$

Under covariation conditions, inferred posterior distribution of any variable in the BN then takes the form of a quotient of two linear functions: $\frac{c_1\cdot\epsilon+c_2}{c_3\cdot\epsilon+c_4}$. Efficient algorithms have been proposed to assess the values of the $c_i$ [43]. This kind of study can further be generalized to $n$-way sensitivity analyses where $n$ is the number of parameters. It has been applied for DBNs in [13]. However, the results are often difficult to interpret [33].

Testing the sensitivity of the results of an inference can be more globally performed in a different manner. Soft evidence (i.e. uncertain evidence) is a way to disturb global behaviour of the BN using (local) belief revision [41, 12]. However, even if the specification of the perturbations is different, this methods still faces the same difficulty to interpret the results when multiple local changes are performed [11].

Sensitivity analysis in BN proposes tools to analytically follow the change in posterior distributions as a function of the parameters (or the beliefs) in local CPTs. As attractive as it might be, this is not exactly what it is asked in robustness analysis. Indeed, the effects of numerous small perturbations is not easy to be estimated with such analysis (using derivative of sensitivity expressions for instance). One would like to obtain a set of possible distributions for the posterior as a result. [16] describes such an approach but with a framework (epistemic independence) difficult to use in the context of large and complex systems such as DBNs. The next section extends and implements this approach by using DCN as a dedicated tool for specification of sets of complex distributions.

## 4.2 DCN as a robustness analysis tool

In this paper, we propose a robustness analysis that consists in perturbing the precise DBN by means of conditional credal sets $K^t_{ij|\epsilon}$=$K_\epsilon(X_i(t)|\mathbf{U}(t) = j, \epsilon)$ such that for all $i = 1 \dots n$, $j = 1 \dots c_i$ and $\epsilon \in [0, 1]$:

$$K^t_{ij|\epsilon} = \left\{ \begin{array}{l} P^t_{ijk} \in [(1-\epsilon)P^t_{ijk}, (1-\epsilon)P^t_{ijk} + \epsilon], \\ \sum_k P^t_{ijk} = 1 \end{array} \right\} \quad (16)$$

The parameter $\epsilon$ may be understood as a perturbation coefficient: the higher it is, the more imprecise $K^t_{ij|\epsilon}$ becomes.

### 4.2.1 Choosing $\epsilon$

The perturbation should depend on the quantity of data used to learn the DBNs as well as on the strength of the intended perturbation. While the strength of the perturbation should be the same over all the network, the number of data used may differ significantly in different places. We propose, to pick the $\epsilon$ used for a given (conditional) probability, to use a function $\psi(n, \beta) : \mathbb{N} \times [0, 1] \to [0, 1]$ where $n$ corresponds to the quantity of data for learning each $P^t_{ij}$ (that is $n = N^t_{ij}$ in our case) and $\beta$ the strength of the perturbation, and to take $\epsilon = \psi(N^t_{ij}, \beta)$ to perturb the conditional probabilities $P^t_{ij}$ of the network. The mapping $\psi$ should satisfy the following constraints:

- $\psi(n, 0) = 0$ and $\psi(n, 1) = 1$
- $\psi$ is decreasing in $n$
- $\psi$ is increasing in $\beta$

The first conditions ensure that no perturbation will keep $P^t_{ij}$ unchanged, while a full perturbation will make the network completely imprecise (this condition may be relaxed into requiring only that $\psi(0, 1) = 1$). The two other conditions ensure that a higher perturbation will induce more imprecision (for a given data set), while more data will result in less imprecision (for a given perturbation). We may also require that $\psi(0, \beta) = 1$ for any $\beta > 0$, that is no data means full imprecision (unless no perturbation is applied), and that $\lim_{n\to\infty} \psi(n, \beta) = 0$ for any $\beta$ , that is the perturbation tends to the null perturbation as data accumulates.

The following function satisfies the conditions:

$$\psi(n, \beta) = \beta^{f(n)} \tag{17}$$

where $f(n)$ is an increasing function of $n$. The natural logarithmic operator $ln$ satisfies these properties and we use $f(n) = \ln(n + 1)$.

#### 4.2.2 Keeping the constraint

Note that if $P^t_{ijk} = 0$ because it corresponds to an hard constraint in the network, it should be kept to 0 even when perturbing the whole network by making it imprecise (only non constraint probabilities should be made imprecise). We will see in the next section that preserving such (physical) constraints indeed play a very important role to ensure the good behaviour of the prediction dynamics.

### 4.3 Experiments on real-life case study

To illustrate our approach on a real case, we have focused on a typical French product, namely the process of the Camembert-type soft mould cheese ripening that is still ill known and complicated to control [28]. During the ripening process, cheese represents an ecosystem and a bioreactor where relationships exist between microbiological, physicochemical and organoleptic changes which depend on environmental conditions. Despite the number of areas involved in cheese research, available knowledge of the cheese ripening process remains fragmented and pervaded with uncertainty. None of the approaches or investigations carried out up to now makes it possible to provide an explicit overview of the causal structure of associations between the underlying variables and an objective interpretation of the cheese ripening process. From operational and scientific knowledge, the structure of a dynamic Bayesian network providing a qualitative representation of the coupled dynamics of micro-organism behaviour with their substrate consumptions influenced by temperature and involving the sensory changes of cheese during ripening has been defined [1]. Figure 1 displays a sub-section of the DBN structure providing a representation of the coupled dynamics of a yeast behaviour (*Kluyveromyces marxianus* concentration ($Km$)) with their substrate consumptions (lactose concentration (lo) influenced by temperature (T). We attempt to estimate the lower and upper mean time evolution

$$\begin{array}{l} \underline{X}_{Q|E,\epsilon}(t) = \min_{P \in K_\epsilon(\mathbf{X})} E_P(X_Q(t)|X_E(t), \forall t) \\ \overline{X}_{Q|E,\epsilon}(t) = \max_{P \in K_\epsilon(\mathbf{X})} E_P(X_Q(t)|X_E(t), \forall t) \end{array} \tag{18}$$

(where $E_P(X_Q(t)|X_E(t))$ denotes the mean time evolution of $X_Q$ given $X_E$) under some perturbation. The initial precise model has been learned by integrating (1) experimental trials; (2) simulated database stemming from existing partial mechanistic models; (3) expert rules based to the conservation laws of microbial activities.

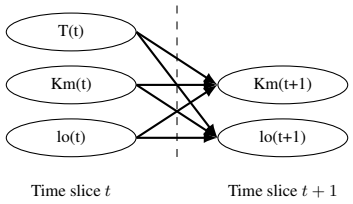


**Figure 1:** Dynamic Bayesian network representing the coupled dynamics $Km$ growth versus $lo$ consumptions influenced by temperature during the cheese ripening process.

In our experiment, a simple Monte-Carlo sampling algorithm over vertices is used to draw inference. The reasons for using such an algorithm are that (1) producing exact inference is too costly, even for small DCN with few time steps (here, 3 variables over 14 time steps), (2) it provides satisfactory bounds that are guaranteed to be inside exact ones and (3) it is sufficient in the present case, as our primary objective is not algorithmic efficiency.

#### 4.3.1 Forward propagation

Forward propagation consists in trying to estimate

$$\begin{array}{c} Km(t)|\{Km(1), lo(1), T(1), \ldots, T(\tau)\} \\ lo(t)|\{Km(1), lo(1), T(1), \ldots, T(\tau)\} \end{array} \tag{19}$$

for all $t \in [1, \tau]$, using Eq. (18) to test the robustness of predictions. All temperatures are constant ($T(1) = \ldots = T(\tau) = 12^o$C) and $\tau$ corresponds to the day before the wrapping of cheeses, namely $\tau = 14$ .

The Monte-Carlo sampling is stopped when lower and upper expectation bounds were not improved in the last 4000 samplings. In all our results about forward propagation, we have not observed differences between the dynamic strong and repetitive extension and we currently investigate whether it is always true in the case of forward propagation.

Figure 2 displays the upper and lower mean time evolutions of $Km$ and $lo$ for different perturbation levels where parameter learning have only been carried out from six experimental trials. We may observe that (1) the precise inferences of $Km$ seem rather biased towards a rapid growth (line corresponding to $\beta = 0$ close to upper expectations); (2) $Km$ may decrease (a physically impossible phenomena) even for relatively small perturbations ($\beta = 0.6$ and mean perturbation level $\epsilon = 0.124$) due to the absence of constraints based on conservation laws.

Figure 3 displays the upper and lower mean time evolutions of $km$ and $lo$ when constraints, based on conservation laws, are added. The effect of adding or preserving the constraints is obvious in the perturbed results. However, we may remark that the precise network is almost unchanged when constraints are added. This means that

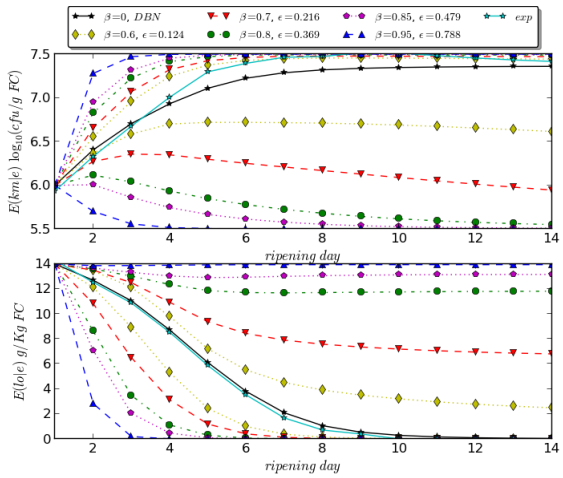


**Figure 2:** Upper and Lower mean evolutions of $Km$ and $lo$ according to different $\beta$ values for forward propagation, without constraints. $\epsilon$ =mean contamination level

constraints play a secondary role when network parameters are well-estimated, however the comparison of Figures 2 and 3 shows that preserving them in case of bad estimation ensures more robustness in the inferences.

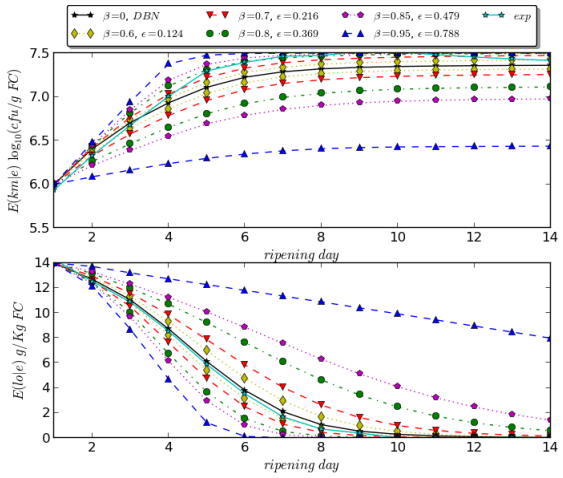


**Figure 3:** Upper and Lower mean evolutions of $Km$ and $lo$ according to different $\beta$ values for forward propagation, with constraints. $\epsilon$ =mean contamination level

### 4.3.2 Forward-backward propagation

Forward-backward propagation consists in trying to estimate

$$Km(t)|\{Km(1), lo(1), Km(\tau), lo(\tau), T(1), \ldots, T(\tau)\} \quad \text{and} \quad lo(t)|\{Km(1), lo(1), Km(\tau), lo(\tau), T(1), \ldots, T(\tau)\} \tag{20}$$

for all $t \in [1, \tau]$, using Eq. (18) in order to test the robustness of predictions. Monte-Carlo sampling was done as in the previous experiment.

Figure 4 displays the upper and lower mean time evolutions of $km$ and $lo$ for different perturbations with the *dynamic repetitive extension*, while Figure 5 displays the same results for the *dynamic strong extension* without the preservation of constraints.

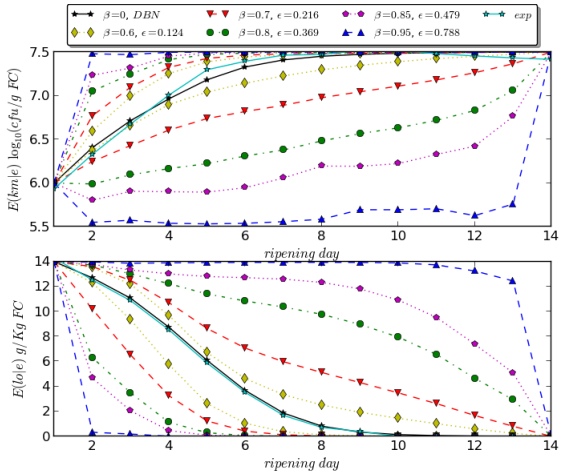


**Figure 4:** Upper and Lower mean evolutions of $Km$ and $lo$ according to different $\beta$ values and *dynamic repetitive extension*.

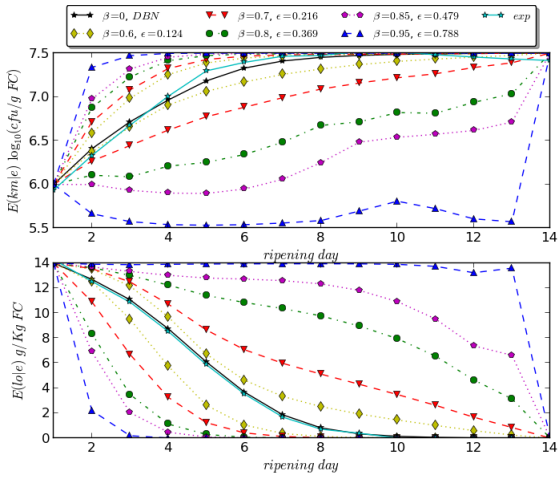


**Figure 5:** Upper and Lower mean evolutions of $Km$ and $lo$ according to different $\beta$ values and *dynamic strong extension*.

In the case of forward-backward propagation, the results from the two extensions do not coincide in general. However, the bounds obtained with the *dynamic strong extension* are sometimes inside those obtained for the repetitive extension, meaning that the sampling algorithm has not reached optimal bounds (indeed, $K(\mathbf{X})_{rp} \subseteq K(\mathbf{X})_{st}$ by definition). We may also observe that the decreasing of $Km$ is less severe than in forward propagation even for high $\beta$ value because $Km(\tau)$ and $lo(\tau)$ are now evidences. Figure 6 displays the results of forward-backward inference with the *dynamic strong extension* when constraints are preserved. Again, we can see that preserving such constraints has a serious effect on the results precision.

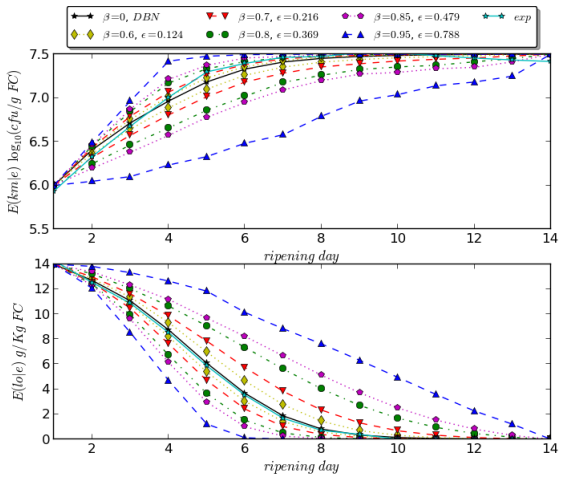


**Figure 6:** Upper and Lower mean evolutions of $Km$ and $lo$ according to different $\beta$ values and *dynamic strong extension*, with constraints.

## 5 Conclusion

There are complex dynamical processes for which no deterministic model describing the complete process exists. In such cases, dynamic Bayesian networks are convenient models that allow to include expert knowledge, data and variable interaction in a single framework. However, they do not allow for a faithful representation of incomplete knowledge or of scarce data, features that are inherent to the complexity of bio-physicochemical phenomena occurring in Food and Life Sciences.

In this paper, we have discussed how DBNs can be extended to include credal sets and cope with such incompleteness and imprecision. We have introduced the concept of dynamic credal networks and have proposed the concepts of dynamic repetitive and strong extensions. While the latter can be seen as a straightforward extension of classical credal networks, the former considers repetitive independence to allow the model to preserve a temporal regularity. Inference algorithms of credal networks may extend better to one case than to the other, depending on their characteristics.

We have proposed to apply such DCN to the problem of robustness analysis, introducing an easy method to perturb a given precise network and performing some experiments on a real-case study concerning microbial population growth. These experiments have shown that including constraints (often provided by expert knowledge) in the network is essential in case of bad estimation of parameters, as they ensure more robustness, while such constraints seem unnecessary in case of good estimation.

We have also observed that in the case of forward propagation (evidences only on nodes without parents), inferences for the strong and repetitive extensions coincided. We are currently investigating under which conditions inferences of strong and repetitive extensions coincide.

In further works, DCNs should enable us to determine the contribution of imprecision and/or incompleteness on the outcomes of a model in order to know if an ambiguous answer is due to a lack of information or due to a random phenomenon. That is, we plan to develop refined sensitivity analysis techniques based on their use. They should thus determine key variables and/or key phenomena for which it will be necessary to acquire more information. Finally, we also plan to investigate their usefulness in determining optimal control commands.

## References

[1] C. Baudrit, M. Sicard, PH Wuillemin, and N. Perrot. Towards a global modelling of the camembert-type cheese ripening process by coupling heterogeneous knowledge with dynamic bayesian networks. *Journal of Food Engineering*, 98(3):283–293, 2010.

[2] C. Baudrit, P.H. Wuillemin, and N. Perrot. Parameter elicitation in probabilistic graphical models for modelling multi-scale food complex systems. *Journal of Food Engineering*, 115(1):1 – 10, 2013.

[3] S. Benferhat and S. Smaoui. Hybrid possibilistic networks. *International journal of approximate reasoning*, 44(3):224–243, 2007.

[4] L.K. Blackmond. Sensitivity analysis for probability assessments in bayesian networks. In *Proceedings of the Ninth international conference on Uncertainty in artificial intelligence*, UAI'93, pages 136–142, San Francisco, CA, USA, 1993. Morgan Kaufmann Publishers Inc.

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

[6] A. Cano, J. Cano, and S. Moral. Convex sets of probabilities propagation by simulated annealing on a tree of cliques. In *In: Proceedings of Fifth International Conference on Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU '94*, pages 4–8, 1994.

[7] A. Cano, M. Gómez, S. Moral, and J. Abellán. Hill-climbing and branch-and-bound algorithms for exact and approximate inference in credal networks. *Int. J. Approx. Reasoning*, 44(3):261–280, 2007.

[8] A. Cano and S. Moral. A genetic algorithm to approximate convex sets of probabilities. In *Proc. of the*

*Int. Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems*, pages 859–864, 1996.

[9] A. Cano and S. Moral. Using probability trees to compute marginals with imprecise probabilities. *International Journal of Approximate Reasoning*, 29(1):1 – 46, 2002.

[10] E. Castillo, R.M. Gutiérrez, and A.S. Hadi. Sensitivity analysis in discrete bayesian networks. *IEEE Transactions on Systems, Man, and Cybernetics*, 27:412–423, 1997.

[11] H. Chan and A. Darwiche. Sensitivity analysis in bayesian networks: From single to multiple parameters. In *Proceedings of the 20'th international Conference on Uncertainty in Artificial Intelligence*, UAI'04, 2004.

[12] H. Chan and A. Darwiche. On the revision of probabilistic beliefs using uncertain evidence. *Artif. Intell.*, 163(1):67–90, March 2005.

[13] T. Charitos and L.C. Van der Gaag. Sensitivity analysis for threshold decision making with dynamic networks. In *Proceedings of the Twenty-Second Conference Annual Conference on Uncertainty in Artificial Intelligence (UAI-06)*, pages 72–79, Arlington, Virginia, 2006. AUAI Press.

[14] V.M.H Coupe and L.C. Van Der Gaag. Properties of sensitivity analysis of bayesian belief networks. In *Proceedings of the Joint Session of the 6th Prague Symposium of Asymptotic Statistics and the 13th Prague Conference on Information Theory, Statistical Decision Functions and Random Processes, Union of Czech Mathematicians and Physicists*, pages 81–86, 1999.

[15] I. Couso, S. Moral, and P. Walley. A survey of concepts of independence for imprecise probabilities. *Risk Decision and Policy*, 5(2):165–181, 2000.

[16] F.G. Cozman. Robustness analysis of bayesian networks with local convex sets of distributions. In *Proceedings of the Thirteenth conference on Uncertainty in artificial intelligence*, UAI'97, pages 108–115, San Francisco, CA, USA, 1997. Morgan Kaufmann Publishers Inc.

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

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

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

[20] J.C.F. da Rocha and F.G. Cozman. Inference with separately specified sets of probabilities in credal networks. In *Proceedings of the Eighteenth conference on Uncertainty in artificial intelligence*, UAI'02, pages 430–437, San Francisco, CA, USA, 2002. Morgan Kaufmann Publishers Inc.

[21] J.C.F. da Rocha, F.G. Cozmanl, and C.P. de Campos. Inference in polytrees with sets of probabilities. In *Proceedings of the Nineteenth conference on Uncertainty in Artificial Intelligence*, UAI'03, pages 217–224, San Francisco, CA, USA, 2003. Morgan Kaufmann Publishers Inc.

[22] J. De Bock and G. de Cooman. State sequence prediction in imprecise hidden markov models. In *Proceedings of the seventh International Symposium on Imprecise Probabilities: Theory and Applications*, pages 159–168, 2011.

[23] C.P. de Campos and F.G. Cozman. Inference in credal networks using multilinear programming. In *In Proceedings of the 2nd Starting AI Researchers' Symposium*, pages 50–61, 2004.

[24] C.P. de Campos and F.G. Cozman. Inference in credal networks through integer programming. In *In Proceedings of the 5th International Symposium on Imprecise Probability: Theories and Applications*, pages 145–154, 2007.

[25] L.M. De Campos, J.F. Huete, and S. Moral. Probability intervals: a tool for uncertain reasoning. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 2(2):167–196, 1994.

[26] K.V. Delgado, L.N. de Barros, F.G. Cozman, and R. Shirota. Representing and solving factored markov decision processes with imprecise probabilities. In *Proceedings ISIPTA, Durham, United Kingdom*, pages 169–178, 2009.

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

[28] Patrick F Fox. *Cheese - Chemistry, Physics and Microbiology; 3rd ed.* Elsevier, San Diego, CA, 2004.

[29] James E. Gentle. Monte carlo methods. *Encyclopedia of Statistical Sciences*, 2006.

[30] J.S. Ide and F.G. Cozman. Approximate Inference in Credal Networks by Variational Mean Field Methods. In *International Symposium on Imprecise Probabilities and Their Applications*, pages 203–212, 2005.

[31] J.S. Ide and F.G. Cozman. Approximate algorithms for credal networks with binary variables. *International Journal of Approximate Reasoning*, 48(1):275 – 296, 2008. Special Section: Perception Based Data Mining and Decision Support Systems.

[32] J.S. Ide and Cozman F.G. Ipe and l2u: Approximate algorithms for credal networks. In *Proceedings of the second starting AI Researcher Symposium*, pages 118–127. IOS Press, 2004.

[33] J. Kwisthout and L.C. Van der Gaag. The computational complexity of sensitivity analysis and parameter tuning. In David A. McAllester and Petri Myllymäki, editors, *Proceedings of the 24th Conference in Uncertainty in Artificial Intelligence, July 9-12, 2008, Helsinki, Finland*, UAI'08, pages 349–356. AUAI Press, 2008.

[34] S.L. Lauritzen. The em algorithm for graphical association models with missing data. *Computational Statistics & Data Analysis*, 19(2):191–201, 1995.

[35] I. Levi. *The Enterprise of Knowledge: An Essay on Knowledge, Credal Probobility, and Chance*. MIT press, 1983.

[36] K.P. Murphy. *Dynamic Bayesian networks: representation, inference and learning*. PhD thesis, University of California, 2002.

[37] T.D. Nielsen and F.V. Jensen. *Bayesian networks and decision graphs*. Springer, 2007.

[38] J. Pearl. *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan Kaufmann, 1988.

[39] N. Perrot, IC Trelea, C. Baudrit, G. Trystram, and P. Bourgine. Modelling and analysis of complex food systems: State of the art and new trends. *Trends in Food Science & Technology*, 22(6):304–314, 2011.

[40] L.R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. In *Proceedings of the IEEE*, pages 257–286, 1989.

[41] J.C. Richard. *The Logic of Decision*. University Of Chicago Press, 1983.

[42] J.C.F. Rocha and F.G. Cozman. Evidence propagation in credal networks: An exact algorithm based on separately specified sets of probability. In G. Bittencourt and G.L. Ramalho, editors, *Advances in Artificial Intelligence*, volume 2507, pages 376–385. 2002.

[43] L.C. Van der Gaag and U. Kjaerulff. Making sensitivity analysis computationally efficient. In *Proceedings of the 16th Conference on Uncertainty in Artificial Intelligence*, UAI'00, pages 317–325. Morgan Kaufmann Publishers, 2000.

[44] P. Walley. *Statistical reasoning with imprecise probabilities*. Chapman and Hall London, 1991.

[45] H. Xu and P. Smets. Reasoning in evidential networks with conditional belief functions. *International Journal of Approximate Reasoning*, 14(2):155–185, 1996.

[46] J.S. Yedidia, W.T. Freeman, and Y. Weiss. Generalized belief propagation. In *NIPS 13*, pages 689–695. MIT Press, 2000.