Annals of Operations Research manuscript No.
(will be inserted by the editor)

Bayesian Decision Support for Complex Systems with Many Distributed Experts

Manuele Leonelli · **James Q. Smith**

Received: date / Accepted: date

Abstract Complex decision support systems often consist of component modules which, encoding the judgements of panels of domain experts, describe a particular sub-domain of the overall system. Ideally these modules need to be pasted together to provide a comprehensive picture of the whole process. The challenge of building such an integrated system is that, whilst the overall qualitative features are common knowledge to all, the explicit forecasts and their associated uncertainties are only expressed individually by each panel, resulting from its own analysis. The structure of the integrated system therefore needs to facilitate the coherent piecing together of these separate evaluations. If such a system is not available there is a serious danger that this might drive decision makers to incoherent and so indefensible policy choices. In this paper we develop a graphically based framework which embeds a set of conditions, consisting of the agreement usually made in practice of certain probability and utility models, that, if satisfied in a given context, are sufficient to ensure the composite system is truly coherent. Furthermore, we develop new message passing algorithms entailing the transmission of expected utility scores between the panels, that enable the uncertainties within each module to be fully accounted for in the evaluation of the available alternatives in these composite systems.

Keywords Bayesian decision theory · combination of expert judgement · decision support systems · graphical models · uncertainty handling

1 Introduction

Nowadays decision centres are required to make choices in complex and evolving environments, described through multiple and interdependent processes with many associated

Manuele Leonelli
Department of Statistics, The University of Warwick, CV47AL, Coventry, UK
Tel.: +44(0)759 3389567
Fax: +44(0)24 7652 4532
E-mail: m.leonelli@warwick.ac.uk

James Q. Smith
Department of Statistics, The University of Warwick, CV47AL, Coventry, UK
E-mail: j.q.smith@warwick.ac.uk

measurements. The objective of a real time decision making centre is to agree to a sequence of efficacious countermeasures. To achieve this it is usually necessary to integrate opinions and information from an often diverse set of stakeholders, articulating several competing objectives and knowledge over different domains of expertise. A collection of decision support systems (DSSs) can enhance such an integration, not only ensuring that all relevant evidence systematically informs policy making, but also encouraging the decision centre to exhibit an underlying consistency across all its components and to address the problem as a whole.

One domain of application exhibiting such complexity is emergency management, especially ones designed to guide the choice of countermeasures after a nuclear accident. Here decision centres need to address the diverse deleterious outworkings of accidental releases of contaminants, taking into account, for example, the effects on health, the political implications and the environmental consequences of that accident. Each of these issues is likely to be informed by a different panel of domain experts, who are the ones best able to articulate appropriate forecasts, their uncertainties and the evaluation of specific consequences arising directly from these.

Early support systems often consisted of a suite of different *component DSSs*, or *modules*, designed by an appropriate panel of experts. These used a variety of deterministic and stochastic methodologies to guide the estimation and the forecasting of the various quantities relevant to the domain under study (see e.g. Ehrhardt et al, 1993; Ehrhardt, 1997). Fully probabilistic component modules then began to be developed to communicate both the relevant panel's forecasts and their associated uncertainties. For nuclear emergency management the first such probabilistic DSSs tended to focus on the initial consequences of the accident (French et al, 1995; Smith and Papamichail, 1999). Since then more sophisticated probabilistic modules modelling both early and later stages of the accident have been introduced (De and Faria, 2011; Richter et al, 2002; Zheng et al, 2009).

These current component DSSs are designed to function independently, albeit with inputs provided by the outputs of other modules. Nevertheless the challenge is to somehow join together the information from these systems in a *coherent* way, meaning that beliefs expressed within a component module do not contradict those expressed in another. For example, a decision centre aided by a DSS whose modules are non-coherently integrated might be tempted to simply plug in point estimates of the inputs necessary for a receiving module and to ignore the delivered uncertainties associated with the values of these inputs. In this paper these uncertainty measures will correspond to the variance of the mean estimates, but we note here that other descriptions of uncertainty can be implemented in the system (for example belief functions (Shafer, 1976), interval probabilities (Walley, 1996) or higher order moments). It has been known for some time that, even in very simple scenarios, ignoring uncertainties can lead a decision centre into choosing the wrong course of action (see e.g. Leonelli and Smith, 2013b and Section 4). This is because expected utility scores for competing suites of countermeasures often formally depend on these uncertainty measures.

In this paper we develop a framework around which a coherent Bayesian decision analysis can be performed. This reconciles all the panels' delivered uncertainties into a unique formal assessment of the ranking of the decision policies supported by a DSS, thus not suffering from the type of dysfunctionality mentioned above. This is because we are able to deduce formulae to compute formal expected utility scores which appropriately draw together the individual panels' assessments. We then proceed to derive message passing algorithms that enable the centre to evaluate the efficacy of different policies efficiently as well as faithfully. The complete propagation of uncertainty between the modules guaranteed by our algorithms further increases the quality of the modules' outputs since their inputs encode

the faithful representation of other panels' uncertainties. The integrating system so defined therefore ensures that any decision centre is properly supported and provides a *consistent* analysis of the complex composite problem.

In order to provide this formal framework we envisage a real or virtual system manager, or *supraBayesian (SB)*, who is responsible for the aggregation of the individual panels' judgements. A successful combination of the beliefs of the panels needs to give sufficient information for the SB to calculate the expected utility scores associated to different policies. Long ago French (1997) gave a vision for addressing this class of problems using Bayesian decision analysis. However until recently, due to computational and methodological constraints, it has not been feasible to actually implement this vision: now it is. Henceforth, we use the phrase **integrating decision support system (IDSS)** to denote the unifying and integrating framework around which the SB combines component DSSs into a single entity. When sufficient conditions exist for individual panels' beliefs to be formally combined into a coherent whole, we say that the system is *distributed*.

In Section 2 below we discuss technical conditions which are sufficient to ensure we can build a distributed IDSS. There are several advantages that derive from structuring a problem so that the ensuing support is distributed. First, because the responsibility for each aspect of the analysis can be devolved to *appropriate* panels of experts, these are then more likely to deliver better calibrated judgements. The whole system might therefore be expected to be more robust to the misspecification of beliefs (Cooke, 1991). On the other hand, if the system needs to be changed in the light of unexpected developments or unplanned consequences, under suitable conditions, the management of these new developments need only be addressed *autonomously* by the relevant panels. These simply adapt their individual forecasts and the inputs in the light of the new scenario they face. These adjustments can then be folded back into the system to provide a revised output of the relevant modules for other panels to use for their inputs.

Second, the output of a distributed IDSS can produce answers to queries by decision centres about the premises on which it is based and the calculations of its outputs, by directing the query to the relevant panels. A route along which the relevant panels can be queried as to the reasons of their contribution to the expected utility scores is described in Fig. 1, which shows the different levels of support that can be implemented into an IDSS. When queried, generic DSSs are usually able to present justifications for the overall expected utility scores and the ranking of the available policies (third box of the second column from the left of Fig. 1). However IDSSs are able to provide an additional level of support. The system's distributivity permits the IDSS to justify its suggestions in terms of the modules outputs, since expected utilities are functions of these (see the bottom box of the second column from the left of Fig. 1). If these systems did not exhibit this distributivity property, then such devolution may not be possible and so any support would be much less transparent.

Third, distributivity ensures that different decision centres can be given the option of choosing different modules to model the various components of the problem. For example, in nuclear emergency management, different countries often prefer to predict the spread of the contamination using their national agencies' diffusion models.

In the applications we have in mind the input data and the underlying processes supported by an IDSS are intrinsically dynamic and unfold in time. This is in particular the case for emergency management, where new observations are constantly gathered throughout the crisis and decision centres need to make decisions at each time new information is available (Leonelli and Smith, 2013a). For this reason it is necessary to consider classes of models that can accommodate a sequential update of the model parameters and, consequently, of the relevant probabilities. For simplicity here, although this is not strictly necessary, we assume

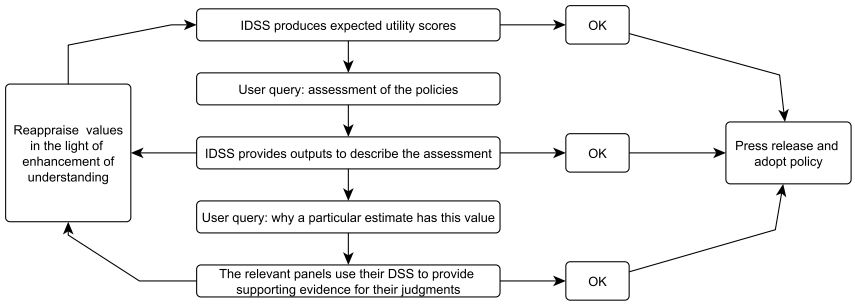

Fig. 1: Description of the possible use of an IDSS for a decision analysis.

that each component module is a dynamic probabilistic model of a type we define later. In addition we suppose that there is an agreed overarching framework of conditional independences common knowledge to all, implying that all panels agree on an overall graphical model specifying the relationships between the component DSSs.

For a full expected utility decision support to be possible, a utility function needs to be specified. It is often reasonable and almost always assumed in practice that this function lies in some family, capturing some form of preferential independence (Abbas, 2010; Fishburn, 1967; Keeney and Raiffa, 1993), which then usually implies some additive or multiplicative factorization. Here we assume that the panels' are able to agree on a unique utility function decomposition lying within a family we define in Section 2, customized to the needs of the network of expert systems. This will enable for the computation of the expected utility scores in a fast distributed fashion.

We envisage that the panels can reach these agreements during *decision conferences*, where a facilitator guides the panels' discussion and exploration of the implications of the possible models (French et al, 2009). During these meetings they also discuss any other necessary probabilistic assumptions needed to be made as, for example, global independence of the various components of the parameter set (Cowell et al, 1999). These behavioural methodologies are well documented and have often been successfully applied in practice in a variety of domains (Ackermann, 1996; Phillips, 1984). The IDSS is then built around this agreed qualitative framework.

Having determined conditions under which an IDSS can be updated dynamically in a modular fashion, in Section 3 we develop message passing algorithms over the network of component DSSs between the panels and the SB. These enable the SB to compute the expected utility scores of potential policies. These also let calculations to be devolved to the relevant panels and hence define the operations of the given distributed system. Our algorithms work similarly to the many others designed for the single agent propagation of probabilities and expected utilities through, respectively, Bayesian networks (BNs) and influence diagrams (IDs) (see e.g. Cowell et al, 1999; Faria and Smith, 1997; Jensen and Nielsen, 2013; Lauritzen, 1992). The subtlety here is to derive conditions to ensure that all calculations can be decomposed into sub-computations that can be performed autonomously by individual panels. We are able to demonstrate that it is surprisingly simple to calculate message passing algorithms based on these individual computations using standard backward induction, albeit in this novel and potentially very complex setting. Distributivity conditions thus imply that the system is able to quickly produce forecasts and expected utilities, en-

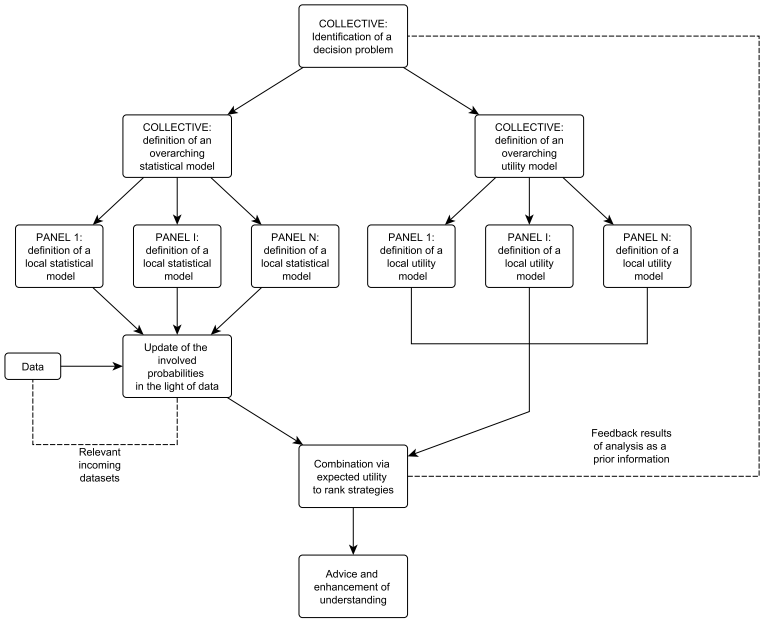

Fig. 2: Structure of a Bayesian decision analysis for a group of distributed experts, generalizing French (1997).

abling decision centres to interrogate the IDSS in real-time. We illustrate these processes in Section 4.

2 The Integrating Decision Support System

To formally define the network of component DSSs constituting an IDSS we first need to introduce a theoretical framework based on a set of structural assumptions that can guarantee its distributivity. Our exposition broadly follows the structure of the diagram in Fig. 2. However in addition we also assume the following. First, all the panels agree on a overarching (dynamic) graphical statistical model representing the relationships existing between the quantities they agree to include in the analysis. Second, they are able to jointly determine a decision space describing the available actions that can be taken after the observation of a specific subset of the variables. Third, we assume they share a utility factorization which lies within a customized family of utilities. All these assumptions are defined formally below.

Thus, let $\{\boldsymbol{Y}(t)\}_{t\in[T]}$, $[T]=\{1,2,\ldots,T\}$, be a multivariate time series with finite horizon T partitioned into n multivariate time series $\{\boldsymbol{Y}_i(t)\}_{t\in[T]}$, with $i\in[n]$. For $A\subseteq[n]$, let $\boldsymbol{Y}_A(t)^{\mathrm{T}}=\left(\boldsymbol{Y}_i(t)^{\mathrm{T}}\right)_{i\in A}$, $\boldsymbol{Y}(t)=\boldsymbol{Y}_{[n]}(t)$ and $\boldsymbol{Y}_A^t=(\boldsymbol{Y}_A(1)^{\mathrm{T}},\ldots,\boldsymbol{Y}_A(t)^{\mathrm{T}})^{\mathrm{T}}$. A random vector $\boldsymbol{Y}_A(t)$ takes values in $\mathscr{Y}_A=\times_{i\in A}\mathscr{Y}_i$, where $\mathscr{Y}_i$ is the space associated to $\boldsymbol{Y}_i(t)$. For each $t\in[T-1]$, the vector $(\boldsymbol{Y}_A(t)^{\mathrm{T}},\boldsymbol{Y}_A(t+1)^{\mathrm{T}})$ takes values in $\mathscr{Y}_A\times\mathscr{Y}_A$. Further let $\Pi_i\subseteq[i-1]$ and $Q_i=[i-1]\setminus\Pi_i$.

Lower case letters denote generic instantiations of random vectors, e.g. $\boldsymbol{y}_A(t)$ and $\boldsymbol{y}_A^t$ are realizations of $\boldsymbol{Y}_A(t)$ and $\boldsymbol{Y}_A^t$ respectively. Each individual time series $\{\boldsymbol{Y}_i(t)\}_{t\in[T]}$ is

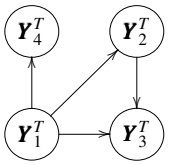

Fig. 3: Example of a DAG of a DDM depicting relationships between processes, not variables.

overseen by a panel of experts, that we denote G_i, and includes all the variables associated to the i-th DSS.

Finally, the term *collective*, represented by the SB, will be used to refer to representatives of each panel, relevant stakeholders and interested parties in the support system. We assume here that the collective is jointly responsible for the definition of the necessary overarching probabilistic, preferential and decision structures. The structure within each individual module will on the other hand be agreed by the members of the relevant panel only.

2.1 The integrating system

2.1.1 The probabilistic integrating structure.

The overall statistical model the collective needs to agree upon is, for the purposes of this paper, a dynamic graphical Bayesian model customized to the needs of multi-expert systems, here called distributed dynamic model (DDM). Graphical models provide a faithful picture of the relationships existing between the main features of the problem, which can be discussed and explored by the collective (for an introduction to graph theory, see e.g. Chapter 4 of Cowell et al, 1999). In a DDM these relationships are depicted by a directed acyclic graph (DAG), whose vertices are $\boldsymbol{Y}_i^T$, $i \in [n]$. Here we assume that the vector $\boldsymbol{Y}^T$ includes all the variables the collective is planning to take into account during the analysis. We are now ready to formally define the DDM model class.

Definition 1 A **DDM** for the time series $\{\boldsymbol{Y}(t)\}_{t\in[T]}$ consists of :

- $n-1$ conditional independence statements for each time point t of the form

$$\boldsymbol{Y}_i(t) \perp\!\!\!\perp \boldsymbol{Y}^t_{Q_i} \mid \boldsymbol{Y}^t_{\Pi_i}, \boldsymbol{Y}^{t-1}_i;$$

- a DAG $\mathscr{G}$ with vertex set $V(\mathscr{G}) = \{\boldsymbol{Y}_1^T, \ldots, \boldsymbol{Y}_n^T\}$ and edge set $E(\mathscr{G})$ including an element $(\boldsymbol{Y}_j^T, \boldsymbol{Y}_i^T)$ if $j \in \Pi_i$, $i \in [n]$.

We use the common notation $\boldsymbol{Y}_i(t) \perp\!\!\!\perp \boldsymbol{Y}^t_{Q_i} \mid \boldsymbol{Y}^t_{\Pi_i}, \boldsymbol{Y}^{t-1}_i$ (Dawid, 1979) to read that the vector $\boldsymbol{Y}_i(t)$ is independent of $\boldsymbol{Y}^t_{Q_i}$ given $\boldsymbol{Y}^t_{\Pi_i}$ and $\boldsymbol{Y}^{t-1}_i$, so that the only information to infer $\boldsymbol{Y}_i(t)$ from $\boldsymbol{Y}^t_{Q_i}$, $\boldsymbol{Y}^t_{\Pi_i}$ and $\boldsymbol{Y}^{t-1}_i$ is from $\boldsymbol{Y}^t_{\Pi_i}$ and $\boldsymbol{Y}^{t-1}_i$. Note that the sets Π_i and Q_i do not depend on the time index t since the vertices of the DAG of the DDM are the time series vector overseen by different panels.

An example of a DAG associated to a DDM is presented in Fig. 3. This network is used throughout the paper to illustrate various features of our methodology. Such a DAG can be thought of as specifying relevant relationships across the components of different time

series. This is in contrast to the more common BN, whose DAG represents relationships between single variables. It is important to note that statements embodied within this DAG are *qualitative* in nature and so in particular can more easily provide the framework for a common knowledge base (see e.g. Smith, 1996, for a discussion).

Since the vertices of the underlying graph are time series, the topology of the DAG does not change through time. Therefore each time slice $\boldsymbol{Y}(t)$ of a DDM, conditionally on the past, can also be described graphically by a DAG with the same topology. This topology remains constant as time progresses. On the other hand the associated probabilities are allowed to be dynamically updated through time, using for example a dynamic linear model (DLM) approach (West and Harrison, 1997). Each time slice DAG has vertex set equal to $\{\boldsymbol{Y}_1(t),\ldots,\boldsymbol{Y}_n(t)\}$ and edges $(\boldsymbol{Y}_j(t),\boldsymbol{Y}_i(t))$ if $j \in \Pi_i$, $i,j \in [n]$. The vertex $\boldsymbol{Y}_j(t)$ is then usually called a *parent* of $\boldsymbol{Y}_i(t)$, whilst $\boldsymbol{Y}_i(t)$ is called a *child* of $\boldsymbol{Y}_j(t)$. We call this DAG the **time slice DAG** of the DDM. Recall that a vertex with no children is usually called *leaf* of the DAG. Let $Le(\mathscr{G})$ be the index of the leaf vertices of $\mathscr{G}$.

Associated to any time slice DAG is a partial order over the vector $\boldsymbol{Y}(t)$. Here an element $\boldsymbol{Y}_i(t)$ is called a *descendant* of $\boldsymbol{Y}_j(t)$ and $\boldsymbol{Y}_j(t)$ is called an *ancestor* of $\boldsymbol{Y}_i(t)$, if there is a directed path from $\boldsymbol{Y}_j(t)$ to $\boldsymbol{Y}_i(t)$. A directed path is a sequence of vertices for which any two consecutive elements of the sequence, $\boldsymbol{Y}_k(t)$ and $\boldsymbol{Y}_l(t)$ say, are such that $(\boldsymbol{Y}_k(t),\boldsymbol{Y}_l(t))$ is an edge of the DAG. An undirected path is similarly defined as a sequence of vertices for which any two consecutive elements, $\boldsymbol{Y}_k(t)$ and $\boldsymbol{Y}_l(t)$ say, are such that either $(\boldsymbol{Y}_k(t),\boldsymbol{Y}_l(t))$ or $(\boldsymbol{Y}_l(t),\boldsymbol{Y}_k(t))$ is an edge of the DAG. We denote by $A_i' \subseteq [i-1]$ the set consisting of the indices of the ancestors of $\boldsymbol{Y}_i(t)$ and, calling $A_i = A_i' \cup \{i\}$, the *ancestral set* of $\boldsymbol{Y}_i(t)$ is $\{\boldsymbol{Y}_i(t) : i \in \{A_i' \cup \{i\}\}\}$. We denote with D_i the set of the indices of the descendants of $\boldsymbol{Y}_i(t)$. We further call *ancestral components* of a DAG $\mathscr{G}$ those $\boldsymbol{Y}_{A_i}(t)$ such that $i \in Le(\mathscr{G})$. As an example consider the time slice DAG at time t associated to the DDM in Fig. 3. For example $(\boldsymbol{Y}_1(t),\boldsymbol{Y}_2(t),\boldsymbol{Y}_3(t))$ is a directed path in the DAG in Fig. 3, whilst $(\boldsymbol{Y}_3(t),\boldsymbol{Y}_1(t),\boldsymbol{Y}_2(t))$ is an undirected path. For this network $A_3 = \{1,2,3\}$, $D_1 = \{2,3,4\}$ and there are two ancestral components: $\boldsymbol{Y}_{[3]}(t)$ and $\boldsymbol{Y}_{\{1,4\}}(t)$. In the following this partial order becomes fundamental since it induces a partial order over the available decision spaces and guides the algorithms we define below.

Just as for other Bayesian graphical models in the literature, the DDM can be associated with a factorization of the probability density function, which depends on the topology of the associated DAG. Specifically, as a direct consequence of the conditional independence structure associated with a DDM, we have the following result.

Proposition 1 *The joint probability density function f associated to a DDM for the time series $\{\boldsymbol{Y}(t)\}_{t\in[T]} = \{\boldsymbol{Y}_1(t),\ldots,\boldsymbol{Y}_n(t)\}_{t\in[T]}$ can be written as*

$$f\left(\boldsymbol{y}^T\right) = \prod_{t=1}^{T}\prod_{i=1}^{n} f_{t,i}\left(\boldsymbol{y}_i(t) \mid \boldsymbol{y}_i^{t-1}, \boldsymbol{y}_{\Pi_i}^{t}\right).$$

For the purpose of the collective's specification of the overarching probability model, it is only relevant that the probability density can be *qualitatively* written as a product of the terms $f_{t,i}$. The actual algebraic form of these terms and the *quantitative* specification of the associated parameters is agreed, as we specify below and as usual in practice, by the members of the relevant panel only. In this sense the algorithms we derive below are built on the agreed *qualitative* framework within the collective common knowledge base and are driven by the topology of the agreed DAG.

We note here that the DDM model class is very large. Particular instances of the DDM have been extensively studied in the literature. For example, it can be shown that the MDM

(Queen and Smith, 1993), the dynamic chain graph model (Anacleto and Queen, 2013) and of course non-dynamic models such as the BN (Cowell et al, 1999; Pearl, 1988), with appropriate global independence assumptions, can all be seen as special cases of the DDM. Consequently, all these classes of models can be used to embellish the qualitative structure of a DDM with explicit probabilistic specifications.

Before introducing an assumption concerning the collective's probabilistic agreement, we need to define a few more terms from graph theory. A time slice DAG $\mathcal{G}$ is said to be *decomposable* if for any $\boldsymbol{Y}_i(t) \in V(\mathcal{G})$, $(\boldsymbol{Y}_j(t), \boldsymbol{Y}_k(t)) \in E(\mathcal{G})$ if $j,k \in \Pi_i$, $j < k < i < n$. We also say that a DAG is *connected* if every two vertices are connected by an undirected path. A *subgraph* $\mathcal{G}'$ of a DAG $\mathcal{G}$ is a DAG such that $V(\mathcal{G}') \subseteq V(\mathcal{G})$ and $E(\mathcal{G}') \subseteq E(\mathcal{G})$. The subgraph $\mathcal{G}'$ *induced* by a subset of the vertex set of $\mathcal{G}$, V' say, is one such that $V(\mathcal{G}') = V'$ and $E(\mathcal{G}') = E(\mathcal{G}) \cap \{V(\mathcal{G}') \times V(\mathcal{G}')\}$. Consider the time slice DAG associated to the DDM in Fig. 3. This is decomposable since $(\boldsymbol{Y}_1(t), \boldsymbol{Y}_2(t)) \in E(\mathcal{G})$. It is also connected since every two vertices are linked by an undirected path. Finally consider the set $\{\boldsymbol{Y}_2(t), \boldsymbol{Y}_3(t), \boldsymbol{Y}_4(t)\}$. The subgraph of Fig. 3 induced by this set has vertex set $\{\boldsymbol{Y}_2(t), \boldsymbol{Y}_3(t), \boldsymbol{Y}_4(t)\}$ and its edge set is equal to $\{(\boldsymbol{Y}_2(t), \boldsymbol{Y}_3(t))\}$.

We are now ready to make the following assumption.

Structural Assumption 1 (Probabilistic consensus) *The collective agrees to:*

- *describe the predictive factorization of $\boldsymbol{Y}^T$ by a DDM, whose DAG is connected and decomposable;*
- *assume that the elements of $\{\boldsymbol{Y}(t)\}_{t\in[T]}$ are observed according to the order defined by the following rules:*
  - *$\boldsymbol{Y}_i(t_1)$ is observed before $\boldsymbol{Y}_j(t_2)$ if $t_1 < t_2$, for $i, j \in [n]$;*
  - *$\boldsymbol{Y}_j(t)$ is observed before $\boldsymbol{Y}_i(t)$ if $j \in A_i$.*

The requirement that the graph is decomposable is simply a technical one, similar to those used in junction trees propagation algorithms (Lauritzen, 1992). Note that any DAG can be converted into a decomposable one which then gives a valid (albeit inefficient) representation of the underlying processes (see e.g. Smith, 2010, for an explicit description of this embedding). In this sense this assumption is not too fierce. In particular it ensures that no new dependencies are introduced in the IDSS through the backward induction steps we define below. This is because the vertices in an ancestral component are totally ordered for decomposable DAGs, whilst this does not hold in the non-decomposable case. These total orders are specified by father-son relationships between vertices. We call a vertex $\boldsymbol{Y}_i(t)$ the *son* of $\boldsymbol{Y}_j(t)$, if $\Pi_i \subseteq \{\{j\} \cup \Pi_j\}$ and we say that $\boldsymbol{Y}_j(t)$ is the unique *father* of $\boldsymbol{Y}_i(t)$, whose index is F_i. Furthermore, we let S_j be the set of the indices of the sons of $\boldsymbol{Y}_j(t)$. For example, $\boldsymbol{Y}_2(t)$ is the father of $\boldsymbol{Y}_3(t)$ in the time slice DAG of the DDM in Fig. 3, whilst $\boldsymbol{Y}_1(t)$ has two sons, $\boldsymbol{Y}_2(t)$ and $\boldsymbol{Y}_4(t)$.

We can also assume without any loss of generality that the network is connected, since if this were not the case, then, when the structural assumptions we introduce below hold, the overall problem could be decomposed into smaller and independent ones that could be treated separately.

However, more critical, especially in emergency management where it can be commonly violated, is the assumption, as expressed in the second bullet, that it is possible to observe all the quantities the collective planned to observe in the order they happen. It has long been known that when the delivery of some of the data is delayed, the underlying DAG of the DDM does not in general remain constant through time (Queen and Smith, 1993) and previously uncorrelated time series overseen by different panels could then become highly

correlated, thus breaking the distributivity of the system and the validity of our message passing algorithms. For the purposes of this paper we assume that the receipt of information is never delayed. We briefly discuss two potential practical ways of addressing violations of this assumption in the discussion.

2.1.2 The decision space.

As we specify below, the structure of the decision space the collective shares assumes that a potential decision centre has the possibility of intervening after having observed any variable in the system. For $i \in [n]$ and $t \in [T]$, let $\mathscr{D}_i(t)$ be the decision space available after having observed $\boldsymbol{Y}_i(t)$ and $\mathscr{D}(0)$ be the decision space associated to an initial decision. We also let $\mathscr{D}_A(t) = (\mathscr{D}_i(t))_{i\in A}^{\mathrm{T}}$, $\mathscr{D}_A^t = (\mathscr{D}(0), \mathscr{D}_A(1), \ldots, \mathscr{D}_A(t))^{\mathrm{T}}$ and $\mathscr{D}^t = \mathscr{D}_{[n]}^t$, for $A \subseteq [n]$. We also denote with $d_A(t)$ and d_A^t generic elements of $\mathscr{D}_A(t)$ and $\mathscr{D}_A^t$ respectively.

We next make the following assumption.

Structural Assumption 2 (Structure consensus) *The collective agrees:*

- *the specification of the decision spaces $\mathscr{D}(0)$ and $\mathscr{D}_i(t)$, $i \in [n]$, $t \in [T]$, defining the acts a decision centre might take;*
- *to assume that the choice of a decision $d_i(t_2) \in \mathscr{D}_i(t_2)$ is not constrained by a decision $d_j(t_1) \in \mathscr{D}_j(t_1)$ if $j \notin A_i'$ in the DAG of the DDM, $j < i$, $t_1 \leq t_2$;*
- *to commit to a decision $d_i(t) \in \mathscr{D}_i(t)$ only after having observed the value of $\boldsymbol{Y}_{A_i}(t)$ and $\boldsymbol{Y}^{t-1}$, and having already committed to decisions $d_{A_i}(t)$ and d^{t-1};*
- *that the underlying DDM remains valid under any policy choice open to the centre.*

Structural assumption 2 guarantees that the graphical framework of the IDSS remains unaffected after a decision is taken, so that the system provides a coherent picture of the problem throughout the unfolding of events and actions in a particular incident. This is because under the assumption above the topology of the time slice DAGs does not change. So the algorithms we define below are still able to compute coherent expected utility scores through message passing. Of course we can still allow for the possibility that the probability judgments *within* that structure might change in response to a decision - they usually do.

To illustrate the assumption above we consider the diagram of the time slice DAG at time t of our network, reported in Fig. 4, which includes four decision spaces: $\mathscr{D}_1(t), \ldots, \mathscr{D}_4(t)$. This network is in reduced form (see e.g. Smith, 2010), meaning that if $(\mathscr{D}_i(t), \mathscr{D}_k(t))$, $(\mathscr{D}_i(t), \mathscr{D}_j(t)) \in E(\mathscr{G})$, $i < j < k$, then in its reduced version the edge $(\mathscr{D}_i(t), \mathscr{D}_j(t))$ is omitted. Furthermore, in reduced form networks edges of the type $(\boldsymbol{Y}_i(t), \mathscr{D}_j(t))$ are included only if $i = j$. Note that this is not a simple ID since Structural Assumption 2 does not guarantee that the decision spaces are totally ordered, although for convenience their indexing respects the total order of natural numbers, just as for the common BN model. In fact these decision spaces only need to be partially ordered consistently with the DAG of the DDM. Therefore, for example, there is no fixed order in which a decision centre commits to decisions $d_2(t) \in \mathscr{D}_2(t)$ and $d_4(t) \in \mathscr{D}_4(t)$. The constraints associated with this partial order are denoted in Fig. 4 by the absence of an edge between these two decision spaces. A decision centre needs to commit to one of these decisions, $d_i(t) \in \mathscr{D}_i(t)$ say, only after having observed the value of $\boldsymbol{Y}_i(t)$, $i \in [4]$: in our notation only after having observed $\boldsymbol{Y}_{A_i}(t)$ and $\boldsymbol{Y}^{t-1}$ as specified by the third bullet of Structural Assumption 2. Of course a decision $d_i(t) \in \mathscr{D}_i(t)$ is made after having already committed to $d^{t-1} \in \mathscr{D}^{t-1}$. We further assume that the overall decision space is such that $\mathscr{D}_4(t) \times \mathscr{D}_2(t)$, so that in particular these two decision spaces do

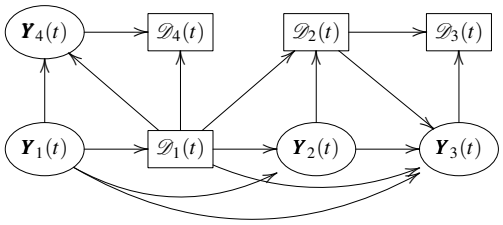

Fig. 4: A time slice DAG of the DDM in Fig. 3 including decision nodes.

not constrain one another. In general decision spaces that are not connected by an edge in the non-reduced representation of the network in Fig. 4 cannot be mutually constrained.

The second assumption about the structure of the decision problem concerns a set of irrelevance statements.

Structural Assumption 3 (Decisions' irrelevance) *The collective agrees that*

$$f_{t,i}\left(\boldsymbol{y}_i(t) \mid d^T, \boldsymbol{y}^t_{\Pi_i}, \boldsymbol{y}^{t-1}_i\right) = f_{t,i}\left(\boldsymbol{y}_i(t) \mid d^t_{A'_i}, \boldsymbol{y}^t_{\Pi_i}, \boldsymbol{y}^{t-1}_i\right), \tag{1}$$

for $i \in [n]$ and $t \in [T]$.

Equation (1) states that a random vector $\boldsymbol{Y}_i(t)$ does not functionally depend on the decisions that are not included in $\mathscr{D}^t_{A'_i}$. This assumption is a very weak one. For example the *sufficiency theorem* (Smith,1989a; Smith, 1989b) guarantees that a decision centre can always find *one* Bayes optimal decision based on a decision rule which respects these statements. We further note here that within each time slice this assumption is an instance of the *causal consistency lemma* of Cowell et al (1999), but applied to this more general setting. The lemma guarantees that decisions can have a direct influence only on variables that are yet to be observed. More generally here Structural Assumption 3 implies the lemma holds for partially ordered decisions and decision spaces that are not simply product spaces.

Return now to the example above. Since there is no fixed order between $\boldsymbol{Y}_2(t)$ and $\boldsymbol{Y}_4(t)$, Structural Assumption 3 demands that $\boldsymbol{Y}_4(t)$ does not functionally depend on $d_2(t) \in \mathscr{D}_2(t)$. Similarly, we require that $\boldsymbol{Y}_2(t)$ does not functionally depend on $d_4(t) \in \mathscr{D}_4(t)$. This can be noted in the diagram of Fig. 4 by the absence of edges between these nodes.

These are the irrelevances the collective *needs* to be ready to assume. Of course they might believe that some decisions do not have any direct effect to additional variables and thus assume further irrelevances. For example they might believe that the initial decision space $\mathscr{D}(0)$ is irrelevant for the outcomes of the variables at the first time point, $\boldsymbol{Y}(1)$. Such additional assumptions do not affect the validity of the architecture of the DAG.

2.1.3 The utility integrating structure.

The last overarching agreement the collective needs to find concerns the utility factorization. We suppose the time series with index in $U \subseteq [n]$ to be the attributes of the decision problem. For $i \in U$, define $\boldsymbol{r}_i = \boldsymbol{r}_i(\boldsymbol{y}^T_i, d^T_{A_i})$ to be a function of both $\boldsymbol{y}^T_i$ and $d^T_{A_i}$ and let $\boldsymbol{r}_A = (\boldsymbol{r}_i)_{i \in A}$ for $A \subseteq [n]$. Note that each vertex $\boldsymbol{Y}^t_i$ of the DAG of the DDM, for $i \in U$, can be uniquely associated with an $\boldsymbol{r}_i$. This notation will turn out to be very useful and concise to depict utility independent statements between time series under the responsibility of different panels. We assume here that a set of assumptions implying the existence of a utility function u over $\boldsymbol{r}_U$

is appropriate (see e.g. French and Rious Insua, 2000). For simplicity we also assume that $i \in U$ for all $i \in Le(\mathcal{G})$.

The utility function describes the preferential structure of the collective. When there are more than one or two attributes, a faithful elicitation of such a function is difficult, unless certain preferential independence conditions are imposed. An added problem in the multi-expert setting we study here is that joint utility elicitations across different panels in a single integrating decision conference are only rarely possible (see e.g. Chapter 11 of French et al, 2009). So for example it is typically possible to elicit the scores associated with the overall weighting of one attribute over another, for example as expressed by the *criterion weights* of multi-attribute independent utilities. But other more detailed elicitations, for example the appropriate forms of the marginal utility functions, are better delegated to those closest to understanding the consequences of such attributes (for an illustration of why this is so see Von Winterfeldt and Edwards, 1986). However, for this type of delegation to be formally justified it is first necessary to assume that the collective is prepared to entertain certain sets of preferential independences in order to be able to elicit, through individual panels' assessments, a joint utility function.

Here we define a multi-attribute utility factorization compatible with the DAG of the multi-expert DDM we defined above. Specifically, this first assumes that the time series with index in U belonging to different ancestral components are generalized additive independent (GAI) (see e.g. Braziunas and Boutilier, 2006). The utility independence structure within each of this these components is then assumed to be described by a member of a certain class of *utility diagrams* (Abbas, 2010) representing conditional utility independence statements (Keeney and Raiffa, 1993).

Recall that a utility diagram is a bidirected graph whose vertex set corresponds to the sets of attributes, in our notation $\{\boldsymbol{r}_i\colon i \in U\}$, and its edge set is defined by the following rules:

- a bidirected edge between two vertices does not assert any independence relation;
- a directed edge from a vertex to another asserts that the first is utility independent of the second given all the other vertices;
- the absence of an edge between two vertices asserts that the first is utility independent of the second and vice versa, given all the other vertices.

Definition 2 Let $\mathcal{G}$ be the DAG of a DDM of a time series $\{\boldsymbol{Y}(t)\}_{t\in[T]}$ and relabel a vertex $\boldsymbol{Y}^T(i)$ of $\mathcal{G}$ with $\boldsymbol{r}_i$, $i \in [n]$. Let $U \subseteq [n]$ be the index set of the attributes of the decision problem. We say that a utility function u is in the class $\mathbb{U}^{\mathcal{G}}$ of utilities **compatible** to the graph $\mathcal{G}$, if:

- the ancestral components of the subgraph $\mathcal{G}'$ of $\mathcal{G}$ induced by $\{\boldsymbol{r}_i : i \in U\}$ are GAI;
- the independence structure associated to an ancestral component of $\mathcal{G}'$, A_i say, can be described by a subgraph of the utility diagram coinciding with the subgraph of $\mathcal{G}'$ induced by $\{\boldsymbol{r}_j : j \in A_i\}$;

In order to illustrate the class of compatible utility factorizations, consider the DAG in Fig. 3 and let $U = \{2,3,4\}$. Recall that $\{4\}$ and $\{2,3\}$ are in different ancestral components and therefore in the class of compatible utility factorizations $\boldsymbol{r}_4$ and $\{\boldsymbol{r}_2, \boldsymbol{r}_3\}$ are GAI. The two utility diagrams derived following the procedure in Definition 2 would then correspond to the following two networks: one including only the vertex $\boldsymbol{r}_4$, the other with vertices $\boldsymbol{r}_2$ and $\boldsymbol{r}_3$ and an edge from $\boldsymbol{r}_2$ to $\boldsymbol{r}_3$.

The class of compatible utilities to a graph $\mathcal{G}$ can be equivalently described by a certain multiattribute factorization as formalized by the following proposition. For $A \in [n]$, let $\boldsymbol{R}_A^{*0}$

denote the set of all possible instantiations $\boldsymbol{r}_A^{*0}$ such that every $\boldsymbol{r}_i$, $i \in A$, is fixed either at the best, $\boldsymbol{r}_i^*$, or the worst value, $\boldsymbol{r}_i^0$. We also let $u(\boldsymbol{r}_i \mid \boldsymbol{r}_{C_i})$ be a normalized conditional utility function over $\boldsymbol{r}_i$ given $\boldsymbol{r}_{C_i}$, where $C_i \subseteq \{[n] \setminus \{i\}\}$ (Abbas, 2010; Keeney and Raiffa, 1993).

Proposition 2 *A utility function u over $\boldsymbol{r}_U$ compatible with the graph $\mathscr{G}$ can be written as*

$$u^{\mathscr{G}}(\boldsymbol{r}_U) = \sum_{i \in Le(\mathscr{G})} u_i^{\mathscr{G}}(\boldsymbol{r}_{A_i}) = \sum_{i \in Le(\mathscr{G})} \sum_{\boldsymbol{r}_{A_i}^{*0} \in \boldsymbol{R}_{A_i}^{*0}} u_i^{\mathscr{G}}\left(\boldsymbol{r}_{A_i}^{*0}\right) \prod_{j \in A_i} g_j\left(\boldsymbol{r}_j \mid \boldsymbol{r}_{\Pi_j}^{*0}, \boldsymbol{r}_{A_i \setminus \{\Pi_j \cup j\}}^{0}\right), \quad (2)$$

where

$$g_j\left(\boldsymbol{r}_j \mid \boldsymbol{r}_{\Pi_j}^{*0}, \boldsymbol{r}_{A_i \setminus \{\Pi_j \cup j\}}^{0}\right) = \begin{cases} u\left(\boldsymbol{r}_j \mid \boldsymbol{r}_{\Pi_j}^{*0}, \boldsymbol{r}_{A_i \setminus \{\Pi_j \cup j\}}^{0}\right), & \text{if } \boldsymbol{r}_j = \boldsymbol{r}_j^* \text{ in } u\left(\boldsymbol{r}_{A_i}^{*0}\right), \\ 1 - u\left(\boldsymbol{r}_j \mid \boldsymbol{r}_{\Pi_j}^{*0}, \boldsymbol{r}_{A_i \setminus \{\Pi_j \cup j\}}^{0}\right), & \text{otherwise.} \end{cases}$$

This result easily follows by first writing the overall utility as a linear combination of the utilities over the ancestral components (as assured by the GAI independence condition), then applying the expansion theorem of Abbas (2010) in each of these components in increasing order over their indices and finally impose the conditional utility independence structure.

Each of the utilities $u_i^{\mathscr{G}}$ in equation (2) is the product of terms $u_i^{\mathscr{G}}(\boldsymbol{r}_{A_i}^{*0})$, corresponding to criterion weights (French and Rios Insua, 2000) and $g_j\left(\boldsymbol{r}_j \mid \boldsymbol{r}_{\Pi_j}^{*0}, \boldsymbol{r}_{A_i \setminus \{\Pi_j \cup j\}}^{0}\right)$ which is a function of $\boldsymbol{r}_j$ only since the attributes $\boldsymbol{r}_{A_i \setminus j}$ are fixed to a certain instantiation. Thus a compatible utility factorization is a function of the criterion weights and utilities whose arguments are overseen by individual panels only. We can therefore envisage that the collective can agree on using one of these compatible factorizations and to delegate the elicitation of the form of the functions g_j, $j \in U$, to the most informed panels. The functional form of compatible utilities in equation (2) guarantees that the computation of expected utilities scores can be achieved in a distributed fashion through the message passing algorithms we define below.

We now make the following assumption.

Structural Assumption 4 (Preferential consensus) *The collective is able to identify an agreed compatible multi-attribute utility decomposition over $\boldsymbol{r}_U$ within the class $\mathbb{U}_{\mathscr{G}}$ and to elicit the common criterion weights $u_i^{\mathscr{G}}(\boldsymbol{r}_{A_i}^{*0})$, $i \in Le(\mathscr{G})$.*

2.1.4 The IDSS expected utility.

Under the assumptions introduced above, which specify the qualitative structure of the decision problem, the expected utility function factorizes into separate factors of the beliefs that particular individual panels can provide themselves. To show this, for $t \in [T-1]$, let

$$\bar{u}^T\left(\boldsymbol{y}^{T-1}, d^T\right) \triangleq \int_{\mathscr{Y}_{[n]}} u^{\mathscr{G}}(\boldsymbol{r}_U)\, f\left(\boldsymbol{y}(T) \mid \boldsymbol{y}^{T-1}, d^T\right) \mathrm{d}\boldsymbol{y}(T),$$

$$\bar{u}^{t-1}\left(\boldsymbol{y}^{t-1}, d^T\right) \triangleq \int_{\mathscr{Y}_{[n]}} \bar{u}^t\left(\boldsymbol{y}^t, d^T\right) f\left(\boldsymbol{y}(t) \mid \boldsymbol{y}^{t-1}, d^t\right) \mathrm{d}\boldsymbol{y}(t).$$

These two terms correspond to the expected utility scores after marginalization steps have been performed over all the variables with time index bigger or equal than t in the algorithms we define below.

We now show that any function $\bar{u}^t$, $t \in [T]$, can be deduced recursively as a function of the individual panels' statements.

Theorem 1 *Under Structural Assumptions 1-4, $\bar{u}^t$, for $t \in [T]$, can be written as*

$$\bar{u}^t = \int_{\mathscr{Y}_1} \tilde{u}_{t,1}\left(\mathbf{y}_1^t, d^T\right) f_{t,1}\left(\mathbf{y}_1(t) \mid \mathbf{y}_1^{t-1}, d_1^{t-1}\right) \mathrm{d}\mathbf{y}_1(t), \tag{3}$$

where

$$\tilde{u}_{t,i}\left(\mathbf{y}_{A_i}^t, d^T\right) \triangleq \begin{cases} \sum_{\boldsymbol{r}_{A_i}^{*0} \in \boldsymbol{R}_{A_i}^{*0}} u_i^{\mathscr{G}}\left(\boldsymbol{r}_{A_i}^{*0}\right) \prod_{j \in A_i} g_j\left(\boldsymbol{r}_j \mid \boldsymbol{r}_{D_j}^{*0}, \boldsymbol{r}_{\Pi_j}^{*0}\right), & i \in Le(\mathscr{G}),\ t = T \\ \hat{u}_{t+1,i}\left(\mathbf{y}_{A_i}^t, d^T\right), & i \in Le(\mathscr{G}),\ t \neq T \\ \sum_{j \in S_i} \bar{u}_{t,j}\left(\mathbf{y}_{A'_j}^t, \mathbf{y}_j^{t-1}, d^T\right), & \textit{otherwise}, \end{cases} \tag{4}$$

$$\bar{u}_{t,i}\left(\mathbf{y}_{A'_j}^t, \mathbf{y}_j^{t-1}, d^T\right) \triangleq \int_{\mathscr{Y}_i} \tilde{u}_{t,i}\left(\mathbf{y}_{A_i}^t, d^T\right) f_{t,i}\left(\mathbf{y}_i(t) \mid d_{A'_i}^t, \mathbf{y}_{\Pi_i}^t, \mathbf{y}_i^{t-1}\right) \mathrm{d}\mathbf{y}_i(t), \tag{5}$$

and $\hat{u}_{t,i}$ is uniquely defined as the function for which

$$\bar{u}^{t+1} = \sum_{i \in Le(\mathscr{G})} \hat{u}_{t,i}(\mathbf{y}_{A_i}^t, d^T), \tag{6}$$

The proof of this theorem is provided in Appendix A.

We note here that again the actual algebraic form of the terms in equations (3)-(6) is not fundamental to the construction of a coherent distributed IDSS. This form depends on the individual panels' agreements concerning the quantities under their particular jurisdiction. Importantly, however, any $\bar{u}^t$ can be written as a function of these terms, whatever they are. Its computation, as we show in the following section, can therefore be obtained through a message passing algorithm, guided by the topology of the DAG of the DDM, between each individual panel and the SB.

The quantities appearing in the theorem above are fundamental to later developments of this paper. So we now discuss their interpretation. The definition of $\tilde{u}_{t,i}$ in equation (4) depends on whether or not $\boldsymbol{Y}_i(t)$ is a leaf vertex of the time slice DAG $\mathscr{G}$. In the former case, for $t = T$, this corresponds to the utility function over the appropriate ancestral component, whilst if $t \neq T$ this is simply equal to $\hat{u}_{t,i}$. If $i \notin Le(\mathscr{G})$ then equation (4) consists of the sum of the incoming messages $\bar{u}_{t,i}$ received by Panel G_i in our algorithms below. Equation (5) defines $\bar{u}_{t,i}$ which consists of the result of a marginalization of $\tilde{u}_{t,i}$ with respect to the conditional density function $f_{t,i}$. Finally, the theorem asserts that $\bar{u}^{t+1}$ can be uniquely written as a linear combination of the functions $\hat{u}_{t,i}$, for $i \in Le(\mathscr{G})$. Throughout this paper, for ease of notation, we use the convention of writing the arguments of the utility functions u and g in terms of the attributes $\boldsymbol{r}$, whilst for the other functions, e.g. $\tilde{u}_{t,i}$, the arguments will be written in terms of the random variables $\boldsymbol{Y}^T$ and the decisions d^T.

2.2 The component DSSs

It is often recommended that the evaluation of both the conditional utilities and the conditional probabilities should be delegated to groups of individuals best able to compare the efficacy and the likelihood of different value of that attribute (see e.g. Von Neumann and Morgenstern, 1947; Von Winterfeldt and Edwards, 1986). We therefore assume the following.

Structural Assumption 5 (Individual panel consensus) *Every expert within a panel G_i agrees on a probabilistic model for the associated component DSS, $f_{t,i}$, $i \in [n]$, $t \in [T]$, as a function of its inputs. In addition every expert in G_i shares a marginal utility function over $\boldsymbol{r}_i$, if $i \in U$.*

It is possible to encourage the experts within a panel to come to these agreements in a variety of ways, appropriate depending on the context, for example through a facilitated Bayesian decision conference or by following a *Delphi Protocol* (see e.g. French et al, 2009). Similarly, the probabilistic individual agreement might consist of following certain pooling axioms (see e.g. French, 2011; Faria and Smith, 1997; Wisse et al, 2008) or by using agreed software on expert inputs, for example a probabilistic emulator (see e.g. O' Hagan, 2006).

3 The message passing algorithms

The structure of the IDSS has been fully defined. We now proceed to discuss the computation of the expected utilities through message passing. We first introduce an algorithm which includes partial optimization steps to deduce an optimal expected utility score. We then consider two special cases of this algorithm. The first does not include optimization steps and computes the expected utility score of a specific policy, whilst the second works over a non-dynamic network of expert systems.

3.1 The collective optimal expected utility algorithm

In contrast to the quantities defined in equations (3)-(6), which compute the expected utility score of a *particular policy*, we now include additional optimization steps to the algorithms. These enable us to identify an optimal policy. In fact we can use exactly the same architecture of message passing in this case. For this slight generalization we need to first define a new quantity, $u^*_{t,i}$, which accounts for optimizations over decision spaces. Let

$$u^*_{t,i}\left(\boldsymbol{y}^t_{A_i}, d^{t-1}, d_{A'_i}(t)\right) \triangleq \max_{\mathscr{D}_i(t)} \tilde{u}_{t,i}\left(\boldsymbol{y}^t_{A_i}, d^{t-1}, d_{A_i}(t)\right). \tag{7}$$

This function is an optimized version, over the decision space $\mathscr{D}_i(t)$, of $\tilde{u}_{t,i}$. We also let $\bar{u}^*_{t,i}$ be the result of the marginalization of $u^*_{t,i}$. Specifically,

$$\bar{u}^*_{t,i}\left(\boldsymbol{y}^t_{A'_i}, \boldsymbol{y}^{t-1}_i, d^{t-1}, d_{A'_i}(t)\right) \triangleq \int_{\mathscr{Y}_i} u^*_{t,i}\left(\boldsymbol{y}^t_{A_i}, d^{t-1}, d_{A'_i}(t)\right) f_{t,i}\left(\boldsymbol{y}_i(t) \mid d^t_{A'_i}, \boldsymbol{y}^t_{\Pi_i}, \boldsymbol{y}^{t-1}_i\right) \mathrm{d}\boldsymbol{y}_i(t). \tag{8}$$

Before illustrating the algorithm using the network of Fig. 3, we introduce a new notation which is also used in the formal algorithms below. We let G_i : or SB : denote the entity that is responsible for the corresponding operation, while we represent with $\longrightarrow G_i$ or $\longrightarrow SB$ the fact that panel G_i and the SB, respectively, receives the value of an appropriate function. So for example $G_i : \tilde{u}_{t,i} \longrightarrow SB$ denotes that panel G_i computes the function $\tilde{u}_{t,i}$ and communicates its value to the SB.

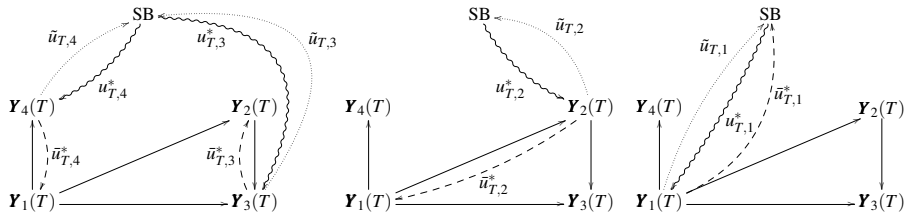

Fig. 5: Optimal expected utility algorithm over the last time slice of the network in Fig. 3.

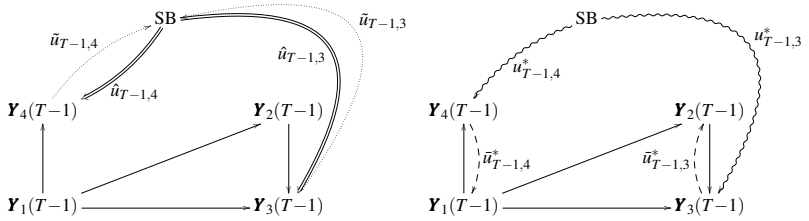

Fig. 6: Optimal expected utility algorithm over the time slice DAG at time $T-1$ of the network in Fig. 3.

3.1.1 An illustrative example.

The algorithm starts from the leaves of the last (time T) time slice DAG and assumes that each panel overseeing a leaf of the time slice DAG, $\boldsymbol{Y}_i(T)$ say, has been provided with the term $u_i^{\mathscr{G}}$. Panels $G_i : \tilde{u}_{T,i} \longrightarrow SB$, for $i = 3,4$. Note that in this case $\tilde{u}_{T,i}$ simply corresponds to $u_i^{\mathscr{G}}$. This step is represented by the dotted arrows on the left network of Fig. 5 from $\boldsymbol{Y}_3(T)$ and $\boldsymbol{Y}_4(T)$ to the SB. Then $SB : u^*_{T,i} \longrightarrow G_i$ as in equation (7) for $i = 3,4$. This has been depicted by the curly arrows in the left network of Fig. 5. At this stage $G_4 : \bar{u}^*_{T,4} \longrightarrow G_1$ and $G_3 : \bar{u}^*_{T,3} \longrightarrow G_2$, since $\boldsymbol{Y}_2(T)$ is the father of $\boldsymbol{Y}_3(T)$ and $\boldsymbol{Y}_1(T)$ is the father of $\boldsymbol{Y}_4(T)$. These two operations are described by the dashed arrows on the left network of Fig. 5.

Now $G_2 : \tilde{u}_{T,2} \longrightarrow SB$, where $\tilde{u}_{T,2} = \bar{u}^*_{T,3}$ since $\boldsymbol{Y}_2(T)$ has only one son. Then as before $SB : u^*_{T,2} \longrightarrow G_2$ and $G_2 : \bar{u}^*_{T,2} \longrightarrow G_1$, since $\boldsymbol{Y}_1(T)$ is the father of $\boldsymbol{Y}_2(T)$. The whole process is depicted by the network in the middle of Fig. 5, where, as before, a dotted arrow is associated to $\tilde{u}_{T,i}$, a curly arrow to $u^*_{T,i}$ and a dashed one to $\bar{u}^*_{T,i}$.

Because $\boldsymbol{Y}_1(T)$ is the father of both $\boldsymbol{Y}_2(T)$ and $\boldsymbol{Y}_4(T)$, now $G_1 : \tilde{u}_{T,1} \longrightarrow SB$, by simply adding $\bar{u}^*_{T,2}$ and $\bar{u}^*_{T,4}$, received from panels G_2 and G_4 respectively. Panel G_1 then repeats the same procedure as the other panels, with the only difference that $\bar{u}^*_{T,1} \longrightarrow SB$ and not to another panel, since it oversees the unique root (i.e. a vertex with no parents) of the DAG. This is depicted by the dashed arrow in the right network of Fig. 5.

Theorem 1 states that $\bar{u}^*_{T,1} \equiv \bar{u}^T$ is equal to the sum of the terms $\hat{u}_{T-1,i}$, $i \in Le(\mathscr{G})$. So, if i is the index of a leaf vertex, $SB : \hat{u}_{T-1,i} \longrightarrow G_i$. This is denoted in the left network of Fig. 6 by the double arrows. Panels $G_i : \tilde{u}_{T-1,i} \longrightarrow SB$, $i \in Le(\mathscr{G})$, where $\tilde{u}_{T-1,i} = \hat{u}_{T-1,i}$. From this stage on, the message passing algorithm copies the calculations and the actions of the previous time slice. So the arrows in Fig. 6 match the ones on the left network of Fig. 5. The algorithm repeats the same sequence depicted by the dashed, curly, dotted and double arrows in Fig. 5 and 6, until it reaches the root vertex of the first time slice. When this happens the SB, after receiving $\bar{u}^*_{1,1}$ from panel G_1, computes a final optimization step

Algorithm 3.1: THE COLLECTIVE OPTIMAL EXPECTED UTILITY($\boldsymbol{u}, \boldsymbol{f}, \mathscr{G}$)

for $t \leftarrow T$ **downto** 1 (1)
do
  for $i \leftarrow n$ **downto** 1 (2)
  do
    if $i \in Le(\mathscr{G})$ (3)
    do
      if $t = T$ (4)
      $\{G_i : \tilde{u}_{t,i} = u_i^{\mathscr{G}} \longrightarrow SB$ (5)
      else $G_i : \tilde{u}_{t,i} = \hat{u}_{t,i} \longrightarrow SB$ (6)
    else $G_i : \tilde{u}_{t,i} = \sum_{j \in S_i} \bar{u}^*_{t,j} \longrightarrow SB$ (7)
    $SB : u^*_{t,i} = \max_{\mathscr{D}_i(t)} \tilde{u}_{t,i} \longrightarrow G_i$ (8)
    if $(i \neq 1)$ (9)
    then $\{G_i : \bar{u}^*_{t,i} = \int_{\mathscr{D}_i} u^*_{t,i} f_{t,i} \mathrm{d}\mathbf{y}_i(t) \longrightarrow G_{F_i}$ (10)
    else
      $G_i : \bar{u}^*_{t,i} = \int_{\mathscr{D}_i} u^*_{t,i} f_{t,i} \mathrm{d}\mathbf{y}_i(t) \longrightarrow SB$ (11)
      if $t \neq 1$ (12)
      then
        for each $j \in Le(\mathscr{G})$ (13)
        do $\{SB:$ computes $\hat{u}_{t-1,j} \longrightarrow G_j$ (14)
      else $SB : u^*_0 = \max_{\mathscr{D}(0)} \bar{u}^*_{t,i}$ (15)

over the decision space $\mathscr{D}(0)$. The algorithm has now been completed and can return the expected utility score of the optimal sequence of decisions.

3.1.2 The algorithm.

Having described the algorithm on the running example, we now introduce it for a generic DDM and in particular for more realistic scenarios. Specifically, this algorithm takes as inputs the utilities $u_i^{\mathscr{G}}$ associated to the ancestral components of $\mathscr{G}$, denoted as $\boldsymbol{u}$, all the conditional density functions $f_{t,i}$, denoted as $\boldsymbol{f}$, and all the information concerning the DAG $\mathscr{G}$. A formal definition of the algorithm can be found in Algorithm 3.1, which is called henceforth **the collective optimal expected utility algorithm**. For simplicity we have left implicit the arguments of various quantities the panels and the SB communicate to each other.

Theorem 2 *Under Structural Assumptions 1-5, Algorithm 3.1 produces an optimal expected utility score resulting from a unique Bayesian probability model, informed only by the individual judgments delivered by the panels.*

The proof of this theorem is provided in Appendix B.

We note here that often the utility function is a polynomial function in its attributes. When this is so, its expectation is a polynomial in which the variables are, in the continuous case, low order moments. This can dramatically simplify the message passing algorithm for computing the optimal policy as we illustrate below. As a result the IDSS often needs as inputs only a few low order moments to work coherently. Even in rather complex domains this in turn means that we can expect the algorithms defined above to be almost instantaneous if each component module can produce the values of these uncertainties under various policy choices efficiently. A study of the polynomial structure of expected utilities in the rather more complex discrete domain is presented in Leonelli et al (2015).

Algorithm 3.2: NON-DYNAMIC COLLECTIVE OPTIMAL EXPECTED UTILITY($\boldsymbol{u}, \boldsymbol{f}, \mathscr{G}$)

for $i \leftarrow n$ **downto** 1

do
- **if** $i \in Le(\mathscr{G})$
  - $\{G_i : \tilde{u}_i = u_i^{\mathscr{G}} \longrightarrow SB$
  - **else** $G_i : \tilde{u}_i = \sum_{j \in S_i} \bar{u}_j^* \longrightarrow SB$
- $SB : u_i^* = \max_{\mathscr{D}_i} \tilde{u}_i \longrightarrow G_i$
- **if** $(i \neq 1)$
  - **then** $\{G_i : \bar{u}_i^* = \int_{\mathscr{Y}_i} u_i^* f_i \mathrm{d}\mathbf{y}_i \longrightarrow G_{F_i}$
  - **else** $\begin{cases} G_i : \bar{u}_i^* = \int_{\mathscr{Y}_i} u_i^* f_i \mathrm{d}\mathbf{y}_i \longrightarrow SB \\ SB : u_0^* = \max_{\mathscr{D}_l} \bar{u}_i^* \end{cases}$

3.2 Two variations of Algorithm 3.1

3.2.1 The score associated to a generic policy.

Algorithm 3.1 provides an operational guideline of how to compute the score associated to an optimal policy. Recall however that the aim of a DSS is not only to identify the decisions with highest expected utilities, but also to provide explanations and the reasoning behind the outputs it provides (French et al, 2009). It is therefore also relevant to compute the expected utility score associated with any policy that might be adopted. These scores then allow decision centres to compare the different available options in more detail, possibly following the route described in Fig. 1. To accommodate this feature a simple variant of Algorithm 3.1 not including any optimization steps can be defined. This would simply correspond to the collective optimal expected utility algorithm without lines (8) and (15), and with the policy of interest as an additional input.

3.2.2 The non-dynamic case.

In some domains it can be more appropriate to model decision problems using a non-dynamic probabilistic model, as for example a BN. Within the IDSS framework, this is possible by simply adapting Algorithm 3.1 to the non-dynamic case. Algorithm 3.2, which we call henceforth **non-dynamic optimal expected utility** algorithm, shows how this can be done. Note that we adapt the notation to the non-dynamic case by dropping the dependence on the time-varying index. Although this notation is new, it is self-explanatory and follows straightforwardly from the dynamic one. It is easy to notice that this algorithm works, since the last time slice of a dynamic DDM alone can be thought of as a non dynamic network. For this time slice, Theorem 2 guarantees the algorithm computes exact expected utility scores.

In the non-dynamic case it is easier to highlight the relationships between our algorithms and standard backward induction evaluation of IDs. Consider the time slice DAG of Fig. 4, which can be related to a non-dynamic problem. The conditions imposed by Structural Assumptions 1-5 guarantee that a standard evaluation can be performed over each ancestral set of the DAG. Furthermore, as stated by Theorem 1, the evaluation of each of these ancestral components can be distributed to the different panels since the expected utility factorizes accordingly.

4 The IDSS in practice: illustration of the typical recursions

The previous section formally presented message passing algorithms for IDSSs. These work in general and for any network of expert systems that respect the structural assumptions of Section 2. We now illustrate how the different panels of experts can communicate with each other through an IDSS using the network of Fig. 3 concerning the policies after an accidental release of contaminants at a nuclear power plant. We show here the typical recursions of an IDSS in a continuous and dynamic agreed structure.

Many applications we have in mind have a geographical structure, in the sense that many of the values of the required variables are recorded at several locations in an area of interest. This is for example the case in a nuclear emergency, where levels of contamination are collected at many different locations in the surroundings of a power plant. Thus, the processes IDSSs will usually deal with are high dimensional. However, the associated utilities are usually low dimensional and can consequently be evaluated transparently. Note that if the impacts of the countermeasures need to be considered at a regional level, it is straightforward to implement these into an IDSS framework. Panels then simply need to provide different scores for the different regions of interest.

Because for real problems the number of equations required to define the problem scales up to an extent where the outworkings of the algorithm are obscured, the example below illustrates how a geographic component can be included into the analysis in the simplest possible case. However in much larger scenarios the calculations are still very feasible and just as straightforward to calculate as in this example because everything is distributed and in closed form. Furthermore, each of the unknown quantities are in practice numbers, rather than algebraic entities, provided by the component DSS and so quick to integrate within the composite system.

4.1 A multiregression dynamic model for a nuclear emergency

The network in Fig. 3 gives our representation of the possible policies after an accidental release of contaminants at a nuclear power plant. Let $\{\boldsymbol{Y}_1(t)\}$ be a time series that computes the contamination in a certain area, $\{\boldsymbol{Y}_2(t)\}$ describe the intake of radioactive elements in the population of the area, $\{\boldsymbol{Y}_3(t)\}$ measure the effects on health on the population and $\{\boldsymbol{Y}_4(t)\}$ rank the political disruption in the area consequently to the accident. The topology of this network implies that, conditional on the past, the political disruption in the area is independent of the human intake and the deleterious effects on health given that the amount of contamination has been observed. Four different panels of experts have jurisdiction over one of these time series, where G_i is responsible for $\{\boldsymbol{Y}_i(t)\}$. Assume further that each vector $\boldsymbol{Y}_i(t) = (Y_i^1(t), \ldots, Y_i^r(t))^{\mathrm{T}}$, $i \in [4]$, $t \in [T]$, is such that $Y_i^l(t)$ is a univariate continuous random variable observed at location $l \in [r]$. The locations are the same for all the time series and do not change through time.

To keep this illustration simple we consider here a simple linear MDM (Queen and Smith, 1993) over a finite time horizon T equal to 2. Specifically, for $i = 2, \ldots, 4$, $t = 1, 2$ and $l \in [r]$, we let

$$Y_i^l(t) = \sum_{j \in \Pi_i} \theta_{ji}^l(t) Y_j^l(t) + v_i^l(t), \qquad \text{and} \qquad \theta_{ji}^l(2) = \theta_{ji}^l(1) + w_{ji}^l(2). \tag{9}$$

Equation (9) implicitly makes the simplifying assumption that the processes at different locations are independent of each other. We are further assuming that the intercepts are equal

to zero and that a simple steady state DLM (West and Harrison, 1997) has been assumed for the root vertex, so that

$$Y_1^l(t) = \theta_{11}^l(t) + v_1^l(t), \quad \text{and} \quad \theta_{11}^l(2) = \theta_{11}^l(1) + w_{11}^l(2). \tag{10}$$

The errors $v_i^l(t)$, $w_{ji}^l(2)$ are assumed by the collective to be mutually independent of each other following a Gaussian distribution with mean zero and known variance $V_i^l(t)$ and $W_{ji}^l(2)$ respectively. Assume further that each panel has provided prior information about the parameter vector at time $t = 1$, such that a_{ji}^l is the mean prior estimate of $\theta_{ji}^l(1)$, whilst its variance is elicited to be c_{ji}^l. Note that each of these parameters and parameter estimates are possibly a function of the available decisions. Here in order not to make the notation too heavy, we do not explicitly label this dependence. However it is important to remember that these values might be different for each available policy.

An important result concerning linear MDMs, and more generally MDMs, is that the predictive densities $f(\boldsymbol{y}(t) \mid \boldsymbol{y}^{t-1})$, $t \in [T]$, also enjoy a factorization which respects the topology of the associated graph (Queen and Smith, 1993). When the errors are assumed to be normal with unknown variances, then these predictive densities can be written in closed form as products of multivariate T-distributions and represent an instance of the factorization of a DDM. Thus, Algorithm 3.1 can be directly applied to this class of models once these predictive distributions are provided by the individual panels. Note that, because of the distributivity of the system, panels can also provide the reasoning behind the value choices for their parameters, since these will be independent to the ones of other panels.

Now assume that the collective has agreed on a linear utility factorization over the attributes, $u = \{2,3,4\}$ and that each panel has individually agreed to model every individual conditional utility function as a cubic. Let $r_i^l(t) = y_i^l(t)$ and assume that decisions are not arguments of the utility function. We let

$$u^{\mathscr{G}}(\cdot) = \sum_{i=2}^{4}\sum_{t=1}^{2}\sum_{l=1}^{r} -\gamma_i(t) y_i^l(t)^3, \tag{11}$$

where $\gamma_i(t) > 0$. It is easy to deduce by comparing equations (2) and (11) that this factorization is a member of the class of compatible utilities. Furthermore the utility function, for each attribute of the decision problem, is assumed to be the same for the r geographical locations and that the overall score of an attribute is equal to the sum of the scores of each region for that attribute. The cubic utility function is a member of the family of constant relative risk aversion utilities, used in the literature to model risk aversion (see Wakker, 2008).

Now that the IDSS has been fully defined for this example, we can show how the algorithm works symbolically when the overarching structure is the linear MDM. Recall that the third non-central moment of a Gaussian distribution with mean μ and variance σ^2 is equal to $\mu^3 + 3\mu\sigma^2$. Since this is a function of the first two moments (as for any high-order moment of a Gaussian), the algorithm consists of a sequential use of the tower property for the first two conditional moments. Specifically, recall that for any two random variables X and Y,

$$\mathbb{E}(X) = \mathbb{E}(\mathbb{E}(X|Y)), \qquad \mathbb{V}(X) = \mathbb{V}(\mathbb{E}(X|Y)) + \mathbb{E}(\mathbb{V}(X|Y)) \tag{12}$$

where $\mathbb{E}$ and $\mathbb{V}$ denote respectively the expectation and variance operators.

Suppose the IDSS needs to identify the expected utility score associated to a specific policy. Equation (11) can be written as $u^{\mathscr{G}}(\cdot) = u_4^{\mathscr{G}}(\boldsymbol{r}_{A_4}) + u_3^{\mathscr{G}}(\boldsymbol{r}_{A_3})$, where

$$u_3^{\mathscr{G}}(\boldsymbol{r}_{A_3}) = \sum_{t=1}^{2}\sum_{l=1}^{r}(-\gamma_2(t) y_2^l(t)^3 - \gamma_3(t) y_3^l(t)^3), \qquad u_4^{\mathscr{G}}(\boldsymbol{r}_{A_4}) = \sum_{t=1}^{2}\sum_{l=1}^{r} -\gamma_4(t) y_4^l(t)^3. \tag{13}$$

$c_2(1) = -\sum_{l=1}^{r}\left(\sum_{i=2}^{4}\gamma_i(1)y_i^l(1)^3 + 3\gamma_3(2)\mathbb{E}\left(\theta_{13}^l(2)^2\theta_{23}^l(2)\theta_{12}^l(2)Y_1^l(2)^3\right)\right)$
$c_2(2) = -\gamma_3(2)\sum_{l=1}^{r}\left(3V_3^l(2)\mathbb{E}\left((\theta_{13}^l(2)+\theta_{23}^l(2)\theta_{12}^l(2))Y_1^l(2)\right) + \mathbb{E}\left((\theta_{13}^l(2)Y_1^l(2))^3\right)\right)$
$c_2(3) = -\sum_{l=1}^{r}\left(\gamma_2(2)+\gamma_3(2)\mathbb{E}\left(\theta_{23}^l(2)^3\right)\right)\left(\mathbb{E}\left(\theta_{12}^l(2)^3Y_1^l(2)^3\right) + 3V_2^l(2)\mathbb{E}\left(\theta_{12}^l(2)Y_1^l(2)\right)\right)$
$c_2(4) = -3\gamma_3(2)\sum_{l=1}^{r}\mathbb{E}\left(\theta_{13}^l(2)\theta_{23}^l(2)^2Y_1^l(2)\right)\left(\mathbb{E}\left(\theta_{12}^l(2)^2Y_1^l(2)^2\right) + \mathbb{V}\left(\theta_{12}^l(2)Y_1^l(2)\right) + V_2^l(2)\right)$

Table 1: Definition of the terms $c_2(i)$ in $\bar{u}_{2,2}$.

The algorithm starts with panel G_4 computing $\bar{u}_{2,4} = \mathbb{E}(u_4^{\mathscr{G}}(\cdot))$ through a sequential use of the two identities in equation (12). Specifically, from equation (13), we have that

$$\bar{u}_{2,4}(\cdot) = \mathbb{E}(\mathbb{E}(u_4^{\mathscr{G}}(\cdot) \mid \cdot)) = -\sum_{l=1}^{r}\left(\gamma_4(1)y_4^l(1)^3 + \gamma_4(2)\mathbb{E}(\mathbb{E}(Y_4^l(t)^3 \mid \cdot))\right), \tag{14}$$

where, from the properties of Gaussian random variables,

$$\mathbb{E}(\mathbb{E}(Y_4^l(t)^3 \mid \cdot)) = \mathbb{E}(\theta_{14}^l(2)^3Y_1^l(2)^3) + 3V_4^l(2)\mathbb{E}(\theta_{14}^l(2)Y_1^l(2)), \tag{15}$$

since $\mathbb{E}(Y_4^l(t) \mid \cdot) = \theta_{14}^l(2)Y_1^l(2)$ and $\mathbb{V}(Y_4^l(t) \mid \cdot) = V_4^l(t)$. Conversely, panel G_3 computes $\bar{u}_{2,3} = \mathbb{E}(u_3^{\mathscr{G}}(\cdot))$ as

$$\bar{u}_{2,3} = -\sum_{l=1}^{r}\left(\gamma_3(1)y_3^l(1)^3 + \gamma_2(1)y_2^l(1)^3 + \gamma_2(2)y_2^l(2)^3 + \gamma_3(2)\mathbb{E}(\mathbb{E}(Y_3^l(2)^3 \mid \cdot))\right), \tag{16}$$

where $\mathbb{E}(\mathbb{E}(Y_3^l(2)^3 \mid \cdot)) = \mathbb{E}(\mathbb{E}(Y_3^l(2) \mid \cdot)^3) + 3V_3^l(2)\mathbb{E}(\mathbb{E}(Y_3^l(2) \mid \cdot))$ and

$$\mathbb{E}(Y_3^l(2) \mid \cdot) = \theta_{13}^l(2)Y_1^l(2) + \theta_{23}^l(2)Y_2^l(2), \tag{17}$$

$$\mathbb{E}(Y_3^l(2) \mid \cdot)^3 = \sum_{i=1,2}(\theta_{i3}^l(2)Y_i^l(2))^3 + 3\sum_{\substack{j+k=3\\ j\neq k\in\mathbb{Z}_{\geq 1}}}(\theta_{j3}^l(2)Y_j^l(2))^2\theta_{k3}^l(2)Y_k^l(2). \tag{18}$$

Now, equations (16)-(18) are functions of $Y_2^l(2)$ and $Y_2^l(1)$ only and can therefore $\longrightarrow G_2$. Then panel $G_2 : \bar{u}_{2,2}$ which can be written as the sum of the terms $c_2(i)$, $i \in [4]$, defined in Table 1, by noting that $\mathbb{E}(Y_2^l(2)) = \mathbb{E}(\theta_{12}^l(2)Y_1^l(2))$,

$$\begin{aligned}
\mathbb{E}(Y_2^l(2)^2) &= \mathbb{V}(\mathbb{E}(Y_2^l(2) \mid \cdot)) + \mathbb{E}(\mathbb{V}(Y_2^l(2) \mid \cdot) + \mathbb{E}(Y_2^l(2))^2 \\
&= \mathbb{V}(\theta_{12}^l(2)Y_1^l(2)) + V_2^l(2) + \mathbb{E}(\theta_{12}^l(2)Y_1^l(2))^2, \\
\mathbb{E}(Y_2^l(2)^3) &= \mathbb{E}(\mathbb{E}(Y_2^l(2)^3 \mid \cdot)) = \mathbb{E}(\theta_{12}^l(2)^3Y_1^l(2)^3) + 3V_2^l(2)\mathbb{E}(\theta_{12}^l(2)Y_1^l(2)),
\end{aligned}$$

and substituting these expressions into equations (17)-(18).

All the terms in Table 1, as well as those in equations (14) and (15), are a function of $Y_1^l(2)$, parameters and variables in the first time slice only and can therefore be sent to panel G_1, which sums these two incoming messages. Panel G_1 then applies sequentially the tower rules to compute $\bar{u}_{2,1}$. This corresponds to the sum of the rhs of (14) and the terms in Table 1 where $Y_1^l(2)$ and $Y_1^l(2)^3$ are substituted with $\theta_{11}^l(2)$ and $\theta_{11}^l(2)^3 + 3\theta_{11}^l(2)V_1(2)$ respectively.

The algorithm then considers the first time slice. Because of the very regular structure of the LMDM, the expressions resulting from the first time slice are almost identical to the

ones at the second time point. These follow the same steps illustrated above and inherit the expectations of the parameters at the second time point. Since in an LMDM the parameters are all independent of each other, the expected utility function can be deduced by simply computing the expectation of each of these. Specifically, letting for $i = 1,2$ and $j = 2,3,4$

$$\begin{aligned}
k^l_{11}(2) &= \mathbb{E}(\theta^l_{11}(2)^3) = ((a^l_{11})^3 + 3a^l_{11}(c^l_{11} + V^l_1(2) + W^l_{11}(2))),\\
k^l_{11}(1) &= \mathbb{E}(\theta^l_{11}(1)^3) = ((a^l_{11})^3 + 3a^l_{11}(c^l_{11} + V^l_1(1))),\\
k^l_{ij}(1) &= \mathbb{E}(\theta^l_{ij}(1)^3) = ((a^l_{ij})^3 + 3a^l_{ij}c^l_{ij}),\\
k^l_{ij}(2) &= \mathbb{E}(\theta^l_{ij}(2)^3) = ((a^l_{ij})^3 + 3a^l_{ij}(c^l_{ij} + W^l_{ij}(2))),\\
d^l_{ij}(2) &= \mathbb{E}(\theta^l_{ij}(2)^2) = (a^l_{ij})^2 + c^l_{ij} + W^l_{ij}(2),\\
d^l_{ij}(1) &= \mathbb{E}(\theta^l_{ij}(1)^2) = (a^l_{ij})^2 + c^l_{ij},
\end{aligned}$$

we can write the expected utility function as

$$\bar{u} = \sum_{l=1}^{r}\left(\sum_{t=1}^{2}\left(\sum_{i=2}^{4}\gamma_i(t)\bar{u}^l_i(t) + \gamma_3(t)(\bar{m}^l_3(t) + \bar{n}^l_3(t) + k^l_{23}(t)\bar{u}_2(t))\right)\right) \tag{19}$$

where

$$\begin{aligned}
\bar{u}^l_i(1) &= k^l_{1i}(1)k^l_{11}(1) + 3V^l_i(1)a^l_{1i}a^l_{11}, \qquad \bar{n}^l_3(t) = 3a^l_{13}d^l_{23}(t)(d^l_{12}(t)k^l_{11}(t) + V^l_2(t)a^l_{11}),\\
\bar{u}^l_i(2) &= k^l_{1i}(2)k^l_{11}(2) + 3V^l_i(2)a^l_{1i}a^l_{11}, \qquad \bar{m}^l_3(t) = 3V^l_3(t)a^l_{23}a^l_{12}(a^l_{11} + k^l_{11}(t)d^l_{13}(t)).
\end{aligned}$$

There are a few important points to notice here:

- because of the form of the utility factorization in equation (11), the expected utility consists of the sum of the expected scores at each location l and of the sum, at each of these locations, of the scores associated to the two time slices. This result is a direct consequence of the independence of the processes at different locations. However, it would have been straightforward to embellish the example to allow the different processes to be dependent on one another, by defining a hierarchical model over $\boldsymbol{\theta}(t)^{\mathrm{T}} = (\boldsymbol{\theta}_1(t)^{\mathrm{T}}, \ldots, \boldsymbol{\theta}_4(t)^{\mathrm{T}})$, where $\boldsymbol{\theta}_i(t)$ is the vector including the parameters $\theta_{ji}(t)$ in equations (9) and (10), $i \in [4]$;
- the expected utility in equation (19) is a polynomial, where the unknown quantities are the individual judgements delivered by the panels. This polynomial has for this example degree six and it is not a simple multilinear combination of the unknowns. Note that knowing the shape of the expected utility allows potential decision centres to understand how different factors influence the decision making process;
- the IDSS would have not been able to compute the expected utility of equation (19) if the panels had only delivered the mean estimates of the variables under their jurisdiction. The quantities $V^l_i(t)$, $W^l_{ij}(2)$ and c^l_{ij} represent levels of uncertainty concerning these mean estimates. If we just plug-in the expectations of relevant means and set equal to zero the above measures of uncertainty we obtain a spurious evaluation of an expected utility score corresponding to

$$\bar{u} = \sum_{l=1}^{r}\sum_{t=1}^{2}\left(\sum_{i=2}^{4}\gamma_i(t)(a^l_{11})^3(a^l_{1i})^3 + \gamma_3(t)(a^l_{11})^3(a^l_{23})^2(a^l_{12})^2(3a^l_{13} + a^l_{23}a^l_{12})\right). \tag{20}$$

Equations (20) is way different from (19). A DSS that provides expected utility scores from equation (20) could thus lead decision centres to behave as non expected utility maximizer and put them in danger of adopting indefensible countermeasures (see Leonelli and Smith, 2013b, for another example).

5 Discussion

The implementation of Bayesian methods for a group decision analysis involving the assessment and the aggregation of both probabilistic and utility models in complex systems has often been considered too difficult to be developed because of the unwieldy large number of factors and interactions that needs to be taken into account. In this paper we have considered both a dynamic and a non-dynamic framework in which, from a theoretical point of view at least, it is possible to feasibly deal with such a class of problems. As shown by the above example all calculations are straightforward and scale up, albeit with a large number of moments or probabilities to be computed, stored and transmitted between panels. However, these quantities can be provided by an IDSS. So the large number of computations necessary for coherently evaluating different policies are actually trivial ones and operationally computable in real time. We note that the algorithms we define in this multi-expert system are closely related to the ones already cited for the propagation of probabilities and expected utilities in graphical structures, which have now been successfully implemented in many large applications (Gómez, 2004; Oliver and Smith, 1990). So we can be confident that our methods remain feasible for current and much larger applications.

The critical assumption of our methodology is that the collective observes all the data they planned to collect. Of course it is very common in practice for data about, for example, the development of an accident to arrive non-sequentially. For instance, a van collecting deposition measurements can be delayed in transmitting these, so that readings are available from the last hour but not the current ones. In such a situation the distributivity of the system is broken. We briefly mention two possible practical solutions here:

- Panels can accommodate only a subset of the data into the system that is appropriately time ordered. This gives the basis for a framework to analyze the system. This can then be elaborated by the appropriate communication of extra information that does not satisfy the criteria;
- Methods can be developed where the distributivity property can only be approximatively satisfied. These methods are beyond the scope of this paper. We have some encouraging new results in this area which will be formally reported in Smith et al (2015).

Of course in some applications the lost of distributivity is not critical and non-observable data can be treated using standard statistical missing data techniques (Little and Rubin, 2014).

So it is feasible for an IDSS to support rapid policy evaluations even when drawing together judgments from diverse panels of probabilistic expert systems, provided conditions ensure distributivity (or this is approximately so). Distributivity can be guaranteed if the density associated with the graphical statistical model the collective agrees upon factorizes appropriately. In this paper we chose a specific, although a rather large, class of directed graphical models where this is so. However, other classes of graphical models can also entertain a distributed analysis. For example chain event graphs (Smith and Anderson, 2008) and their dynamic variant (Barclay et al, 2013) might be able to provide in some domains a better representation of the involved uncertainties and consequently a more focused decision making. We plan to extend our results to additional classes of model in future research.

References

1. Abbas A. E. (2010). General decompositions of multiattribute utility functions with partial utility independence. *Journal of MultiCriteria Decision Analysis*, 17, 37-59.

2. Ackermann F. (1996). Participants' perceptions on the role of facilitators using group decision support systems. *Group Decision and Negotiation*. 5. 93-112.
3. Anacleto O. & Queen C. M. (2013). Dynamic chain graph models for high-dimensional time series. Tech. rep., Department of Statistics, Open University.
4. Barclay L. M., Smith J. Q., Thwaites P. A. & Nicholson A. (2013). The dynamic chain event graph. Tech. rep., CRISM, The University of Warwick.
5. Braziunas D. & Boutilier C. (2006). Preference elicitation and generalized additive utility. *Proceedings of the National Conference on Artificial Intelligence*, 21, 1573-1576.
6. Cooke R. M. (1991) *Experts in uncertainty: opinion and subjective probability in science*. New York: Oxford University Press.
7. Cowell R. G., Dawid P. A., Lauritzen S. L. & Spiegelhalter D. J. (1999). *Probabilistic networks and experts systems*. New York: Springer.
8. De S. & Faria A. E. (2011). Dynamic spatial Bayesian models for radioactivity deposition. *Journal of Time Series Analysis*, 32, 607-617.
9. Ehrhardt J. (1997). The RODOS system: decision support for nuclear off-site emergency management in Europe. *Radiation Protection Dosimetry*, 73, 35-40.
10. Ehrhardt J., Pasler-Sauer J., Schule O., Benz G., Rafat M. & Richter J. (1993). Development of RODOS, a comprehensive decision support system for nuclear emergencies in Europe - an overview. *Radiation Protection Dosimetry*, 50, 195-203.
11. Faria A. E. & Smith J. Q. (1997). Conditionally externally Bayesian pooling operators in chain graphs. *The Annals of Statistics*, 25, 1740-1761.
12. Fishburn P. C. (1967). Interdependence and additivity in multivariate, unidimensional expected utility theory. *International Economic Review*, 8, 335-342.
13. French S. (1997). Uncertainty modelling, data assimilation and decision support for management of off-site nuclear emergencies. *Radiation Protection Dosimetry*, 73, 11-15.
14. French S. (2011). Aggregating expert judgement. *Revista de la Real Academia de Ciencias Exactas, Fisicas y Naturales. Serie A. Matematicas*, 105, 181-206.
15. French S. & Rios Insua D. (2000). *Statistical decision theory*, London: Arnold.
16. French S., Ranyard D. C. & Smith J. Q. (1995). Uncertainty in RODOS. Tech. Rep., School of Computer Science, University of Leeds.
17. French S., Maule J., Papamichail N. (2009). *Decision behaviour, analysis and support*. New York: Cambridge University Press.
18. Gómez M. (2004). Real-world applications of influence diagrams. In *Advances in Bayesian Networks* (pp. 161-180). Springer Berlin Heidelberg.
19. Jensen F. V. & Nielsen T. D. (2013). Probabilistic decision graphs for optimization under uncertainty. *Annals of Operations Research*, 204, 223-248.
20. Keeney R. L. & Raiffa H. (1993). *Decision with multiple objectives: preferences and value trade-offs*. New York: Cambridge University Press.
21. Lauritzen S. L. (1992). Propagation of probabilities, means and variances in mixed graphical association models. *Journal of the American Statistical Association*, 87, 1098-1108.
22. Leonelli M. & Smith J. Q. (2013a). Dynamic uncertainty handling for coherent decision making in nuclear emergency response. *Proceedings of the Winter Meeting of the ANS*, (to appear).
23. Leonelli M. & Smith J. Q. (2013b). Using graphical models and multi-attribute utility theory for probabilistic uncertainty handling in large system, with application to the nuclear emergency management. In *Data Engineering Workshops (ICDEW)*, pp. 181-192.
24. Leonelli M., Riccomagno E. & Smith J. Q. (2015). Using computer algebra to symbolically evaluate discrete influence diagrams. In preparation.
25. Little R. J. A. & Rubin D. B. (2014). *Statistical analysis with missing data*. Hoboken: Wiley.
26. O' Hagan A. (2006). Bayesian analysis of computer code outputs: a tutorial. *Reliability Engineering & Safety Systems*, 91, 1290-1300.
27. Oliver R. M. & Smith J. Q. (1990). *Influence diagrams, belief nets and decision analysis*. New York: Wiley.
28. Pearl J. (1988). *Probabilistic reasoning in intelligent systems: networks of plausible inference*. San Francisco: Morgan Kaufmann.
29. Phillips L. D. (1984). A theory of requisite decision models. *Acta Psychologica*, 56, 29-48.
30. Queen C. M. & Smith J. Q. (1993). Multiregression dynamic models. *Journal of the Royal Statistical Society, Series B*, 55, 849-870.
31. Richter K., Gering F. & Müller H. (2002). Data assimilation for assessing radioactive contamination of feed and foodstuffs. In *Development and Application of Computer Techniques to Environmental Studies*, WIT Press, pp. 393-402.
32. Shafer G. (1976). *A mathematical theory of evidence*. Princeton: Princeton University Press.

33. Smith J.Q. (1989a). Influence diagrams for Bayesian decision analysis. *European Journal of Operational Research*, 40, 363-376.
34. Smith J. Q. (1989b). Influence diagrams for statistical modelling. *The Annals of Statistics*, 17, 654-672.
35. Smith J. Q. (1996). Plausible Bayesian games. In *Bayesian Statistics 5*, pp. 387-406.
36. Smith J. Q. (2010). *Bayesian decision analysis: principles and practice*. Cambridge: Cambridge University Press.
37. Smith J. Q. & Anderson P. E. (2008). Conditional independence and chain event graphs. *Artificial Intelligence*, 172, 42-68.
38. Smith J. Q. & Papamichail N. (1999). Fast Bayes and the dynamic junction forest. *Artifical Intelligence*, 107, 99-124.
39. Smith J. Q., Barons M. J. & Leonelli M. (2015). Coherent inference for integrating decision support systems. In preparation.
40. Von Neumann L. J. & Morgenstern O. (1947). *Theory of games and economic behavior*. Princeton: Princeton University Press.
41. Von Winterfeldt D. & Edwards W. (1986). *Decision analysis and behavioral research*. Cambridge: Cambridge University Press.
42. Wakker P. P. (2008). Explaining the characteristics of the power (CRRA) utility family. *Health Economics*, 17, 1329-1344.
43. Walley P. (1996). Measures of uncertainty in expert systems. *Artificial Intelligence*, 83, 1-58.
44. West M. & Harrison J. (1997). *Bayesian forecasting and dynamic models*. New York: Springer.
45. Wisse B., Bedford T. & Quigley J. (2008). Expert judgement combination using moment methods. *Reliability Engineering & System Safety*, 93, 675-686.
46. Zheng D., Leung J. K. C. & Lee B. (2009). Online update of model state and parameters of a Monte Carlo atmospheric dispersion model by using ensemble Kalman filter. *Atmospheric Environment*, 43, 2005-2011.

A Proof of Theorem 1.

We develop this proof via backward induction both through the vertices of the DAG and through time. For the purpose of this proof define for $t = T$

$$\bar{u}^{T,i}(\boldsymbol{y}_1^T,\dots,\boldsymbol{y}_{i-1}^T,\boldsymbol{y}_i^{T-1},\dots,\boldsymbol{y}_n^{T-1},d^T) = \int_{\mathscr{Y}_i}\cdots\int_{\mathscr{Y}_n} u^{\mathscr{G}} f_{T,i}\cdots f_{T,n}\mathrm{d}\mathbf{y}_i(T)\cdots\mathrm{d}\mathbf{y}_n(T), \tag{21}$$

and note that $\bar{u}^{T,1} \equiv \bar{u}^T$.

First, without any loss of generality, fix a policy d^T. Then start the backward induction from $\boldsymbol{Y}_n(T)$, which, by construction, is a leaf of the time slice DAG at time T. For a leaf, $\boldsymbol{Y}_i(T)$ say, it follows from (4) that $\tilde{u}_{T,i} = u_i^{\mathscr{G}}(\boldsymbol{r}_{A_i})$ and note that consequently $u_i^{\mathscr{G}}$ is a function of $\boldsymbol{Y}_n(T)$ only through $\tilde{u}_{T,n}$. Therefore $\tilde{u}_{T,n}$ can then be simply marginalized as in equation (5) to obtain $\bar{u}_{T,n}$. Furthermore

$$\bar{u}^{T,n} = \sum_{i\in\{Le(\mathscr{G})\setminus\{n\}\}} u_i^{\mathscr{G}}(\boldsymbol{r}_{A_i}) + \bar{u}_{T,n}\left(\boldsymbol{y}_{A'_n}^T,\boldsymbol{y}_n^{T-1},d^T\right). \tag{22}$$

Now consider $\boldsymbol{Y}_{n-1}(T)$. The vertex associated with this random vector in the time slice DAG is either the father of $\boldsymbol{Y}_n(T)$ or a leaf of the DAG. In the latter case, since by construction $n-1 \in U$, the exact same method followed for $\boldsymbol{Y}_n(T)$ can be applied to $\boldsymbol{Y}_{n-1}(T)$, and thus

$$\bar{u}^{T,n-1} = \sum_{i\in\{Le(\mathscr{G})\setminus\{n,n-1\}\}} u_i^{\mathscr{G}}(\boldsymbol{r}_{A_i}) + \sum_{j=n-1}^{n} \bar{u}_{T,j}\left(\boldsymbol{y}_{A'_j}^T,\boldsymbol{y}_j^{T-1},d^T\right). \tag{23}$$

If on the other hand $\boldsymbol{Y}_{n-1}(T)$ is the father of $\boldsymbol{Y}_n(T)$, then by construction $\boldsymbol{Y}_{n-1}(T)$ has only one son. Thus from equation (4) $\bar{u}_{T,n} \equiv \bar{u}_{T,n-1}$ and equation (22) is a function of $\boldsymbol{Y}_{n-1}(T)$ only through $\bar{u}_{T,n}$. In order to deduce $\bar{u}^{T,n-1}$ only $\bar{u}_{T,n-1}$ has to be marginalized with respect to $f_{T,n-1}$ and therefore

$$\bar{u}^{T,n-1} = \sum_{i\in\{Le(\mathscr{G})\setminus\{n,n-1\}\}} u_i^{\mathscr{G}}(\boldsymbol{r}_{A_i}) + \bar{u}_{T,n-1}\left(\boldsymbol{y}_{A'_{n-1}}^T,\boldsymbol{y}_{n-1}^{T-1},\boldsymbol{y}_n^{T-1},d^T\right). \tag{24}$$

We can note from equations (23) and (24) that $\bar{u}^{T,n-1}$ consists of the linear combination of two summations: the first over the leaves of the graphs with index j smaller than $n-1$ of utility terms $u_j^{\mathscr{G}}$; the second

over the indices j bigger or equal than $n-1$ of the terms $\bar{u}_{T,j}$ such that the father of $\boldsymbol{Y}_j(T)$ has an index smaller than $n-1$ in the time slice DAG. So for example in equation (23) the second summation is over both n and $n-1$ since the associated vertices are both leaves of the graphs. On the other hand in equation (24) there is no term $\bar{u}_{T,n}$ since its father has index $n-1$. More generally, for $j \in [n]$, $\bar{u}^{T,j}$ can be written as the linear combination of the following two summations:

- the first over the indices i in $Le(\mathscr{G}) \cap [j-1]$ of $u_i^{\mathscr{G}}$;
- the second over the indices k in $B_j = \{k \geq j : F_k < j\}$ of $\bar{u}_{T,k}$, where F_k is the index of the father of $\boldsymbol{Y}_k^{\mathrm{T}}$.

Therefore, for a $j \in [n]$, we have that

$$\bar{u}^{T,j} = \sum_{i \in \{Le(\mathscr{G}) \cap [j-1]\}} u_i^{\mathscr{G}}(\boldsymbol{r}_{A_i}) + \sum_{k \in B_j} \bar{u}_{T,k}\left(\boldsymbol{y}_{A'_k}^T, \boldsymbol{y}_k^{T-1}, \boldsymbol{y}_{Dn_k}^{T-1}, d^T\right), \tag{25}$$

where Dn_k is the set of the indices of the descendants of $\boldsymbol{Y}_k^{\mathrm{T}}$. In particular for $\boldsymbol{Y}_2(T)$ we can write equation (25) as

$$\bar{u}^{T,2} = \sum_{k \in S_1} \bar{u}_{T,k}\left(\boldsymbol{y}_1^T, \boldsymbol{y}_k^{T-1}, \boldsymbol{y}_{S_k}^{T-1}, d^T\right), \tag{26}$$

since, by the connectedness of the time slice DAG, $\boldsymbol{Y}_1(T)$ is the father of all the vertices whose father's index is not $[n] \setminus \{1\}$. It then follows that equation (26) corresponds to $\bar{u}_{T,1}$, as defined in equation (4), and therefore $\bar{u}^T$ can be written as in equation (3). Thus Theorem 1 holds for time T.

Now, since $\boldsymbol{Y}_1(T)$ is the unique root of the time slice DAG, if $i, j \in S_1$, then

$$A'_i \cap A'_j = \{1\}. \tag{27}$$

Suppose that any vertex $\boldsymbol{Y}_j(T)$, for $j \in S_1$, is either connected by a path to one only leaf of the DAG or is a leaf of the graph itself. Because of the identity in equation (27) and because of the algebraic form of equation (26), which consists of a linear combination of the terms $\bar{u}_{T,j}$, for $j \in S_1$, we can deduce that equation (6) holds for the last time slice. Now, consider the case where one vertex $\boldsymbol{Y}_j(T)$ with index in S_1 is connected to more than one leaf. Equation (4) guarantees the existence of a vertex $\boldsymbol{Y}_i(T)$, $i > j$, connected to both $\boldsymbol{Y}_j(T)$ and the above mentioned leaves, such that $\bar{u}_{T,i}$ can be written as a linear combination of terms $\bar{u}_{T,k}$, for which each of these terms is a function of one of the leaves only. It therefore follows that equation (6) also holds in this case.

Therefore equation (6) guarantees that $\bar{u}^{T,1}$ can be written as a linear combination of terms involving only variables in the same ancestral set. Since also the probability factorisation does not change as formalised in Proposition 1, the exact same recursions we explicated at time T can then be followed at time $T-1$ by substituting $u_i^{\mathscr{G}}$ with $\hat{u}_{T-1,i}$, $i \in Le(\mathscr{G})$, in equations (21)-(25) and by changing the time index. This then also holds for any time slice t, $1 \leq t \leq T-1$, since $\bar{u}^{t,1}$ will be again a linear combination of terms $\hat{u}_{t-1,i}$, $i \in Le(\mathscr{G})$, and the probability density function factorizes as in Proposition 1.

B Proof of Theorem 2.

To prove Theorem 2 we proceed as follows:

- We relate the lines of the pseudo-code of Algorithm 3.1 to the equations (3)-(6) of Theorem 1 and their variations which include optimization steps in equations (7) and (8);
- We then show that each panel and the SB have sufficient information to perform the steps of the algorithm they are responsible for;
- We conclude by showing that the optimization steps, which in the algorithm correspond to lines (8) and (15), are able to identify optimal decisions using only combinations of quantities individual panels are able to calculate.

We start with the first two bullets. Line (1) describes the backward induction step over the time index, t, while line (2) does the same over the index of the vertices of the graph, i. Now note that in lines (5)-(7), Panel $G_i : \tilde{u}_{t,i}$ using equation (4). Each panel has enough information to do this, since line (10) guarantees that the scores are communicated to the panels overseeing father vertices and line (14) denotes the fact that the SB transmits $\hat{u}_{t,i}$ to the appropriate panels. The functions $\tilde{u}_{t,i}$ are then sent to the *SB*, who performs an optimization step in line (8) and communicates the result back to the panel. We address the validity of this step below.

Since the $SB : u^*_{t,i} \longrightarrow G_i$, each panel is able to compute $\bar{u}^*_{t,i}$ (lines 10-11) following equation (8). As noted before, if i is not the root of the DAG, $\bar{u}^*_{t,i}$ is sent to the appropriate panel, whilst if $i = 1$, as specified

by the if statement in line (9), $\longrightarrow SB$. For each time slice with time index $t \neq 1$ lines (13)-(14) compute $\hat{u}_{t,i}$, as in equation (6). These are sent to the appropriate panels, which can then continue the backward inductive process from the time slice with a lower time index. If on the other hand $t = 1$, then the expected utility is a function of the initial decision space $\mathscr{D}(0)$ only. The SB can then perform a final optimization step over this space and thus conclude the algorithm (line 15).

We now address the optimization steps. The influence on the scores associated with time slices with index bigger than t of a decision space $\mathscr{D}_i(t)$ are included, by construction, only in the terms $\hat{u}_{t,k}$, where k is either the index of a descendant $\boldsymbol{Y}_k(t)$ of $\boldsymbol{Y}_i(t)$ or $k = i$. Further note that the same decision space $\mathscr{D}_i(t)$ can affect the scores of terms including descendants of $\boldsymbol{Y}_i(t)$ at the same time point. Thus the whole contribution of $\mathscr{D}_i(t)$ is summarized within $\tilde{u}_{t,i}$, as it can be seen by recursively using equations (4) and (5).

Now, as specified by equation (7), the optimization step over $\mathscr{D}_i(t)$ is performed by maximizing $\tilde{u}_{t,i}$, which carries all the information concerning this decision space. More specifically, no other term is an explicit function of $\mathscr{D}_i(t)$ at this stage of the algorithm, as guaranteed by equations (1). Finally, Structural Assumption 2 guarantees that all the elements that appears as arguments of $\tilde{u}_{t,i}$ are observed and therefore known at the time the decision associated to this decision space needs to be made.