Multiagent Bayesian Forecasting of Time Series with Graphical Models

Yang Xiang1, James Smith2 and Jeff Kroes3
1 University of Guelph, Canada 2 University of Warwick, UK

Abstract

Time series are found widely in engineering and science. We study multiagent forecasting in time series, drawing from literature on time series, graphical models, and multiagent systems. Knowledge representation of our agents is based on dynamic multiply sectioned Bayesian networks (DMSBNs), a class of cooperative multiagent graphical models. We propose a method through which agents can perform one-step forecast with exact probabilistic inference. Superior performance of our agents over agents based on dynamic Bayesian networks (DBNs) are demonstrated through experiment.

Introduction

Time series (Brockwell and Davis 1991) are found widely in engineering, science and economics, and allow useful inferences such as forecasting. Time series are traditionally studied under the single agent paradigm, but research under the multiagent paradigm has been seen in recent years, e.g., (Raudys and Zliobaite 2006) and (Kiekintveld et al. 2007).

Graphical models (Pearl 1988; Lauritzen 1996) have become an important tool for analyzing multivariate data. There is now a large literature on time series models which can be depicted by graphs. Some of the earliest models proposed are DBNs (Dean and Kanazawa 1989; Kjaerulf 1992). These graphs code a variety of conditional independence statements both with variables with the same time index and across time. One of the most successful of these is based on the class of vector autoregressive (VAR) models (Brockwell and Davis 1991) led by developments such as (Dahlhaus and Eichler 2003). A second approach adopted by (West and Harrison 1996; Koller and Lerner 2001; Pournara and Wernisch 2004; Queen and Smith 1993) developed state space analogues of these processes. Because of their simplicity and convenient closure properties, this paper focuses on multiagent forecasting models of the first kind.

Under the multiagent paradigm, multiply sectioned Bayesian networks (MBSNs) (Xiang 2002) are proposed as cooperative multiagent graphical models. They are first applied to static domains and have been extended to dynamic domains (An, Xiang, and Cercone 2008).

This paper proposes a technique for cooperative multiagent forecasting with time series based on DMSBNs. For these stochastic graphical models, their time series across (temporal) interface variables share the type of conditional independence structure of VAR models without linearity assumptions. Unlike (Raudys and Zliobaite 2006) where agents are “competing among themselves” for better financial prediction, agents based on DMSBNs are cooperative.

Background

Dynamic Bayesian network

A DBN (Dean and Kanazawa 1989; Kjaerulf 1992) models a dynamic domain over a finite time period. Our formulation follows that of (Xiang 1998).

Definition 1 A DBN of horizon k is a quadruplet $G = (\cup_{i=0}^{k} V_i, \cup_{i=0}^{k} G_i, \cup_{i=0}^{k} F_i, \cup_{i=0}^{k} P_i)$. $V_i$ is a set of variables for time interval $i$. $G_i$ is a DAG whose nodes are labeled by elements of $V_i$. $F_i$ is a set of arcs each directed from a node in $G_{i-1}$ to a node in $G_i$. Each node $v \in \bigcup_{i=0}^{k} V_i$ is conditionally independent of its non-descendants given its parents $\pi(v)$. $P_i$ is a set of probability distributions $P_i = \{P(v|\pi(v))|v \in V_i\}$.

$G$ models a dynamic domain over $k+1$ intervals, each of which is referred to as interval $i$ or time $i$. $V_i$ represents the state of the domain at interval $i$ and $G_i$ models the uncertain dependency among elements of $V_i$. $F_i$ is a set of temporal arcs representing how the domain evolves over time.

Definition 2 In a DBN $G$ of horizon $k$, subset $FI_i = \{x|\exists (x, y) \in F_{i+1}\}$ is the forward interface of $V_i$ ($0 \leq i < k$). Subset $BI_i = \{z|\exists (x, y) \in F_i, z \in \text{fnl}(y) \cap V_i\}$ is the backward interface of $V_i$ ($0 < i \leq k$). Denote $G_i = (V_i, E_i)$, where $E_i$ is the set of arcs, and $D_i = (V_i \cup FI_{i-1}, E_i \cup F_i)$. The pair $S_i = (D_i, P_i)$ is a slice of the DBN and $D_i$ is the structure of $S_i$.

Figure 1: An example DBN.

The joint probability distribution (jpd) of the domain over $k+1$ intervals is the product of distributions in all slices.
Fig. 1 shows a DBN where \( V_1 = \{a_1, b_1, c_1, d_1, e_1, f_1\} \), \( E_1 = \{(a_1, b_1), (b_1, d_1), (c_1, e_1), (d_1, e_1), (e_1, f_1)\} \), \( F_1 = \{(a_0, b_1), (f_0, f_1)\} \). \( F_{V_1} = \{a_1, f_1\} \), and \( BI_{V_1} = \{a_1, b_1, c_1, f_1\} \). Note that subscripts are used to index temporal distribution of variables and dependency structures.

**Multiply Sectioned Bayesian Networks**

An MSBN models a domain, typically spatially distributed among a set of agents. The domain dependency is captured by a set of (overlapping) graphs, defined below and illustrated in Fig. 2.

**Definition 3** Let \( G^i = (V^i, E^i) \) (\( i = 0, 1 \)) be two graphs. \( G^0 \) and \( G^1 \) are **graph-consistent** if subgraphs of \( G^0 \) and \( G^1 \) spanned by \( V^0 \cap V^1 \) (keeping nodes in \( V^0 \cap V^1 \) and arcs among them only) are identical. Given two graph-consistent graphs \( G^i = (V^i, E^i) \) (\( i = 0, 1 \)), the graph \( G = (V^0 \cup V^1, E^0 \cup E^1) \) is the **union** of \( G^0 \) and \( G^1 \), denoted by \( G = G^0 \cup G^1 \). Given a graph \( G = (V, E) \), a decomposition of \( V \) into \( V^0 \) and \( V^1 \) such that \( V^0 \cup V^1 = V \) and \( V^0 \cap V^1 \neq \emptyset \), and subgraphs \( G^i = (V^i) \) such that \( i = 0, 1 \), \( G^i \) is said to be **sectioned** into \( G^0 \) and \( G^1 \).

![Figure 2: Illustration of graph union and section.](image)

To ensure exact probabilistic inference, distributed graphical models need to satisfy the following conditions. Def. 4 specifies how domain variables are distributed.

**Definition 4** Let \( G = (V, E) \) be a connected graph sectioned into subgraphs \( \{G^i = (V^i, E^i)\} \). Let the subgraphs be organized into an undirected tree \( \Psi \) where each node is uniquely labeled by a \( G^i \) and each link between \( G^i \) and \( G^m \) is labeled by the non-empty interface \( V^k \cap V^m \) such that for each \( G^i \) and \( G^j \) in \( \Psi \) and each \( G^k \) on the path between \( G^i \) and \( G^j \), \( V^i \cap V^k \subset V^j \). Then \( \Psi \) is a **hypertree** over \( G \). Each \( G^i \) is a **hypernode** and each interface is a **hyperlink**. A pair of hypernodes connected by a hyperlink is said to be **adjacent**.

![Figure 3: A trivial MSBN with hypertree \( G_1 - G_0 - G_2 \).](image)

Fig. 3 shows three subgraphs which section a graph \( G \) (not shown). A corresponding hypertree has the topology \( G_1 - G_0 - G_2 \). Def. 5 specifies what variables agent interfaces contain. This condition ensures conditional independence given the interface.

**Definition 5** Let \( G \) be a directed graph sectioned into subgraphs \( \{G^i\} \) such that a hypertree over \( G \) exists. A node \( x \) (whose parent set in \( G \), possibly empty, is denoted \( \pi(x) \)) contained in more than one subgraph is a **d-sepnode** if there exists at least one subgraph that contains \( \pi(x) \). An interface \( I \) is a **d-sepset** if every \( x \in I \) is a d-sepnode.

For the above hypertree related to Fig. 3, it has two identical d-sets \( \{a, b, c\} \). Each d-sepnode is shown with a dashed circle. Def. 6 combines the above definitions to specify the dependence structure of an MSBN.

**Definition 6** A **hypertree MSDAG** \( G = \bigcup G^i \) where each \( G^i \) is a DAG, is a connected DAG such that (1) there exists a hypertree \( \Psi \) over \( G \), and (2) each hyperlink in \( \Psi \) is a d-sepset.

Def. 7 defines an MSBN and specifies its associated probability distributions, which is illustrated in Fig. 3.

**Definition 7** An **MSBN** \( M \) is a triplet \( M = (V, G, P) \). \( V = \bigcup V^i \) is the domain where each \( V^i \) is a set of variables, called a **subdomain**. \( G = \bigcup G^i \) (a hypertree MSDAG) is the **structure** where nodes of each DAG \( G^i \) are labeled by elements of \( V^i \). Each node \( x \in V \) is conditionally independent of its non-descendants given its parents \( \pi(x) \) in \( G \). \( P = \bigcup P^i \) is a collection of probability distributions, where \( P^i = \{P(x | \pi(x)) | x \in V^i \} \), subject to the following condition: For each \( x \), exactly one of its occurrences (in a \( G^i \) containing \( \{x\} \cup \pi(x) \)) is associated with \( P(x | \pi(x)) \), and each occurrence in other DAGs is associated with a constant (uniform) distribution.

Each triplet \( S^i = (V^i, G^i, P^i) \) is called a **subnet** of \( M \). Two subnets \( S^i \) and \( S^j \) are **adjacent** if \( G^i \) and \( G^j \) are adjacent on the hypertree.

Note that if a variable \( x \) occurs in \( G^i \) and \( G^j \) (\( i \neq j \)), \( x \)'s parents \( \pi(x) \) in \( G^i \) may differ from its parents \( \pi(x) \) in \( G^j \). Note also that superscripts are used to index spatial distribution of variables and dependency structures.

For exact, distributed inference, each subnet is compiled into a local junction tree (JT), where each cluster is associated with a potential. The MSBN is thus compiled into a linked junction forest (LJF). Operation **UnifyBelief** allows an agent to bring potentials in its local JT into consistency. Operation **CommunicateBelief** allows potentials in all agents to reach global consistency. The full posteriors can then be retrieved from the relevant potentials. Due to space, readers are referred to (Xiang 2002) for details.

**Dynamic Multiply Sectioned Bayesian Networks**

A DMSBN models a domain that is both spatially distributed and temporally evolving. In the following definition, subscripts are used to index temporal evolution and superscripts are used to index spatial distribution.

**Definition 8** A DMSBN \( DM \) of horizon \( k \) is a quadruplet \( G = (\bigcup_{i=0}^{k} V_i, \bigcup_{i=0}^{k} G_i, \bigcup_{i=1}^{k} F_i, \bigcup_{i=0}^{k} P_i) \).

\( V_i = \bigcup_{j=0}^{i} V^j_i \) is the **domain** for time interval \( i \), where \( V^j_i \) is a **subdomain** for time \( i \). \( G_i = \bigcup_{j=0}^{i} G^j_i \) (a hypertree MSDAG) is the **structure** for time \( i \), where nodes of
each DAG $G_i^j = (V_i^j, E_i^j)$ are labeled by elements of $V_i^j$. $F_i = \bigcup F_i^j$ is a collection of temporal arcs, where $F_i^j$ is a set of arcs each directed from a node in $G_i^{j-1}$ to a node in $G_i^j$. Each node $v \in \bigcup_{k=0}^{i} V_i^j$ is conditionally independent of its non-descendants given its parents $\pi(v)$ in $\bigcup_{k=0}^{i} G_i^j$. $P_i = \bigcup P_i^j$ is a collection of probability distributions, where $P_i^j = \{P(x|\pi(x))|x \in V_i^j\}$, subject to the following condition: for each $x \in V_i$, exactly one of its occurrences (in a $G_i^j$ containing $\{x\} \cup \pi(x)$) is associated with $P(x|\pi(x))$, and each occurrence in other DAGs for time $i$ is associated with a constant distribution.

Let $\mathcal{A} = \{a_n | a_m \in S_B, a_n \in M_S, a_S \in B_N\}$. The set of probability distributions (one per node) in the occurrences at time $i$ is a DBN for each $\mathcal{A}$, and its isomorphic counterpart is a DBN and MSBN. Proposition 1 does so relative to DMSBN. We establish the fundamental relations between DMSBN, time-invariant DMSBNs. Proposition 1 states the following: Although in each subnet $S_i^j$ of $M_i$, $G_i^j$ is enlarged into $\hat{G}_i^j$ with $F_i^{j-1}$ and $F_i^j$, the temporal arcs $F_i^j$ do not introduce direct connection between $G_i^j$ and $G_i^k$ for all $k \neq j$. Hence, whenever $G_i = \bigcup G_i^j$ is a hypertree MSDAG, $\hat{G}_i = \bigcup \hat{G}_i^j$ is also a hypertree MSDAG. □

Note that for each $x \in F_i^{j-1}$ in the subnet $S_i^j$, it has no parent in $S_i^j$ and is assigned a constant distribution in Def. 8. Hence, $P(F_i^{j-1})$ as defined by $S_i^j$ is a constant distribution as well. More precisely, the following marginalization

$$\sum_{\hat{V}_i \cap F_i^{j-1}} \prod_{j} P_i^j \quad (\text{where } \hat{V}_i = \bigcup \hat{V}_i^j)$$

is a constant distribution. We summarize this in the following Lemma, which is needed in our later analysis.

**Lemma 1** Let DM be a DMSBN of horizon $k$ and $M_i$ be a slice of DM for time $i > 0$. Then, in each subnet, the distribution over forward interface $F_i^{j-1}$

$$P(F_i^{j-1}) = \sum_{\hat{V}_i \cap F_i^{j-1}} \prod_{j} P_i^j \quad (\text{where } \hat{V}_i = \bigcup \hat{V}_i^j)$$

is a constant distribution.

**Multiagent Forecasting**

We consider a dynamic problem domain that can be represented as a DMSBN. The domain is populated by a set of agents. Each agent $A^j$ is in charge of the subdomain $V_i^j$ and has the access of subnet $S_i^j$ for $i = 0, 1, \ldots, k$. At any time $i$, subdomains are organized into a hypertree and we refer to each interface on the hypertree as an agent interface at time $i$. Variables contained in agent interfaces are public.

We assume that the knowledge of $A^j$ over $V_i^j$ is proprietary. Hence, variables in $V_i^j$ that are not contained in any agent interface of $A^j$ are private variables of $A^j$. The dependency structure among them as well as numerical parameters that quantify the structure are also private to $A^j$. As a result, a centralized representation of the domain is not feasible.

On the other hand, agents share a common interest that motivates them to cooperate truthfully within the limit of their privacy. That is, any message exchanged regarding public variables must be consistent with the true belief of the sending agent. No messages regarding private variables will be communicated.

We make the interface observability assumption: At time $i$, all variables in each agent interface are observed by the two corresponding agents. In addition, each agent $A^j$ may carry out additional observations over its subdomain $V_i^j$. The task of agents is to forecast the state of the domain at time $i + 1$ based on all observations obtained up to time $i$.

The above generalizes a number of cooperative situations such as the following in a supply chain:

**Forecasting in a supply Chain** In order to meet needs of production operations for workers (to be hired or laid-off),
equipment (to be purchased or reconfigured), and materials (to be ordered and shipped), arrangements often must be made in advance. Forecast allows such needs to be anticipated so that necessary arrangements are made in time.

As an example from the equipment perspective, manufacturing of a particular part, device or component requires equipment setup and reconfiguration. Per-part cost is reduced if set up is performed for a large batch of parts to be manufactured. Constant switching between manufacturing of different parts increases per-part cost and should be avoided. On the other hand, maintaining a large inventory over an extended period is also costly. Hence, accurate prediction of short-term demand allows optimal planning of the manufacturing process.

In a supply chain, a demand (from a consumer) of a given component (produced by one manufacturer) generates a demand of parts (likely produced by several other manufacturers) that the component is composed of. This interdependency among suppliers makes isolated forecasting by individual manufacturers less accurate. A cooperative forecasting is advantageous here as it benefits from knowledge and observations of all agents over their individual subdomains. Better forecasting will allow better planning and more cost-effective operation by all suppliers.

Fig. 4 illustrates such a multiagent system of three agents over two time intervals. Spatial dependences are along the horizontal direction and temporal evolution is along the vertical direction. For each supplier, availability of skilled workers, adequate equipment, and material (or component) ordered constrains the level of production, which in turn determines the amount of supply produced and influences the unit cost. Availability of skilled workers influences the workers’ wage, which in turn affects the unit cost. The unit cost is also affected by the sale price of the material from the next supplier down the chain. The amount of supply and the order incoming from the next supplier up the chain determine the inventory left and affect the unit sale price.

![Figure 4: A DMSBN based multiagent system.](image)

Temporally, the current availability of workers, the current workers’ wage level, and the current availability of adequate equipment are closely dependent on their status in the previous time interval. Inventory left from the previous time interval affects both the current level of supply and the order of material from the next supplier down the chain.

**Forecasting Algorithms**

Forecasting proceeds as follows: At time \( i = 0 \), agents communicate through the MSBN \( M_0 \) to acquire prior for their respective subdomains. So each agent \( A^j \) acquires a prior \( P(V_0^j) \) for \( i = 0 \).

Then each agent \( A^j \) acquires observations \( obs_0^j \) and updates its belief about its subdomain \( V_0^j \) to get a posterior \( P(V_0^j|obs_0^j) \) for \( i = 0 \). Due to dependent agent interface and interface observability, this step can be performed at each agent’s local JT without communication. After this, the MSBN \( M_1 \) is loaded into agents. The subnet for \( V_0^j \) is separated from the subnet for \( V_1^j \) through the forward interface, and a prior over the temporal interface is defined from a marginalization of \( P(V_0^j|obs_0^j) \). Through the MSBN \( M_i \), agents communicate and forecast for \( i = 1 \). That is, each agent \( A^j \) obtains the prior \( P(V_1^j|obs_0^j) \) for \( i = 1 \), where \( obs_0 \) includes observations at \( i = 0 \) by all agents.

From then on at each \( i \), each agent \( A^j \) acquires observations \( obs_i^j \) and updates its belief about its subdomain \( V_i^j \) to get the posterior \( P(V_i^j|obs_0^j, ..., obs_i^j) \). It is performed at each agent’s local JT without communication. After this, the MSBN \( M_{i+1} \) is loaded into agents. The subnet for \( V_i^j \) is separated from the subnet for \( V_{i+1}^j \) through a temporal interface, and a prior over the interface is obtained from marginalization of \( P(V_i^j|obs_0^j, ..., obs_i^j) \). Using \( M_{i+1} \) with the priors, agents communicate and forecast for \( i + 1 \). Each agent \( A^j \) obtains the prior \( P(V_{i+1}^j|obs_0, ..., obs_i) \) for \( i + 1 \).

The above is enabled through a compilation of the DMSBN. Its subnets for each time \( i \) are compiled into an LIF and reused for each time instance. The compilation is similar to that for MSBNs, except that for each subnet of time \( i \), \( FI_{i-1} \) is contained in a cluster in the local JT and so is \( FI_i \). We denote the local JT of agent \( A^j \) compiled from its subnet \( S_i^j \) by \( T_i^j \).

We assume that no forecast is made for the interval \( i = 0 \). The following diagram illustrates agent activities and their timing. The first line shows a sequence of time intervals each bounded by a pair of vertical bars. In the second line, the label \( obs_0 \) refers to local observation made during interval 0, and the label \( forecast_i \) refers to forecasting on interval 1. The observation and forecasting activities are grouped into two algorithms **InitialObservation** and **Forecast** specified below. The third line illustrates which activities in the 2nd line are included in the execution of each algorithm.

```
| interval0 | interval1 | interval2 | ...
obs0  forecast1  obs  forecast2  obs2  forecast3 ...
< Init > < Forecast > < Forecast > ...
```
Algorithm 1 (InitialObservation) At start of interval 0, each agent \( A_i \) does the following:
1. load local JT \( T_0^i \) into memory;
2. enter local observations from interval 0;
3. perform \text{UnifyBelief}.

Algorithm 2 (Forecast) At end of interval \( i \geq 0 \), each agent \( A_i \) does the following:
1. retrieve potential \( B(FI_0^i) \) from its local JT \( T_0^i \);
2. replace \( T_0^i \) by \( T_{i+1}^j \) in memory;
3. find a cluster \( Q \) in \( T_{i+1}^j \) such that \( Q \supseteq FI_0^i \);
4. update potential \( B(Q) \) into \( B'(Q) = B(Q) * B(FI_0^i) \);
5. respond to call on \text{CommunicateBelief};
6. answer forecasting queries on interval \( i+1 \);

During interval \( i+1 \), \( A_i \) does the following:
8. enter local observations from interval \( i+1 \);
9. perform \text{UnifyBelief}.

Note that for each \( x \in FI_{i-1}^j \) in the subnet \( S_i^j \), it has no parent in \( S_i^j \) and is assigned a constant distribution. Hence, \( B(FI_{i-1}^j) \) in \( T_i^j \) is a constant distribution immediately after the local JT is loaded into memory.

\text{CommunicateBelief} is called upon an arbitrary agent during each interval.

Theorem 1 After execution of \text{InitialObservation} at each agent, followed by \text{Forecast} from interval 0 to \( i-1 \), followed by the first 7 lines of \text{Forecast} at end of interval \( i \), the answers from each agent to forecasting queries on interval \( i+1 \) are exact.

Proof: We prove this by induction on time intervals.

During \text{InitialObservation}, the LJF loaded in line 1 is globally consistent. In line 2, observations are entered at each agent. As agent interfaces are d-sepset and due to interface observability assumption, each (enlarged) subdomain \( V_{i+1}^j \) is conditionally independent on each other subdomain \( V_{i+1}^j \) where \( k \neq j \), given observations on an agent interface between them. Therefore, line 3 is equivalent to \text{CommunicateBelief} without actual communication. After line 3, not only each local JT \( T_0^i \) is locally consistent, but also the LJF at interval 0 is globally consistent. From Theorem 8.12 in (Xiang 2002), and the fact that \( FI_0^i \) is contained in a single cluster in \( T_0^i \), \( B(FI_0^i) \) retrieved from a unique cluster in \( T_0^i \) is exact: That is,

\[ B(FI_0^i) = \text{const} * P(FI_0^i|obs_0^i) = \text{const} * P(FI_0^i|obs_0), \]

where ‘const’ is a constant, \( obs_0^i \) is the local observation by \( A_i \) at \( i = 0 \), and \( obs_0 \) includes observations at \( i = 0 \) by all agents.

For the base case \( i = 0 \), we need only to consider one execution of the first 7 lines of \text{Forecast} at each agent. Based on the above argument, \( B(FI_0^i) \) retrieved at line 1 is exact.

At line 2, the LJF for \( i = 1 \) is loaded. From Lemma 1, marginalization of \( B(Q) \) to \( FI_0^i \) is a constant distribution. Therefore, before line 5 is executed, \( B(Q) = \text{const} * P(Q \setminus FI_0^i) \), and the potential associated with local JT \( T_1^i \) is \( B(V_1^i) = \text{const} * P(V_1^i|FI_1^i) \). After line 5 is executed, the potential over \( Q \) becomes

\[ B'(Q) = \text{const} * P(Q \setminus FI_0^i) * P(FI_0^i|obs_0^i) \]
\[ = \text{const} * P(Q|obs_0^i). \]

This implies that the potential over \( T_1^i \) becomes

\[ B'(V_1^i) = \text{const} * P(V_1^i|FI_1^i) * P(FI_1^i|obs_0^i) \]
\[ = \text{const} * P(V_1^i|obs_0^i). \]

That is, the potential over \( T_1^i \) has been conditioned on observation \( obs_0^i \). This, however, makes the LJF for \( i = 1 \) inconsistent. After line 6, from Theorem 8.12 in (Xiang 2002), the LJF for \( i = 1 \) is again globally consistent and \( B'(V_1^i) = \text{const} * P(V_1^i|obs_0^i) \). Hence, forecasting on \( i = 1 \) at line 7 is exact. This concludes the proof for the base case.

Assume that the theorem holds when \( i = m \). That is, when line 7 of \text{Forecast} is executed at end of interval \( m \), the LJF for \( i = m+1 \) is globally consistent and, for each \( A_i \),

\[ B'(V_{m+1}^i) = \text{const} * P(V_{m+1}^i|obs_0^i, \ldots, obs_m^i). \]

Hence, forecast on \( i = m+1 \) is exact.

We consider interval \( i = m+1 \). First, each agent completes lines 8 and 9 with respect to the LJF for interval \( m+1 \). Due to d-sepset agent interfaces and interface observability, each subdomain \( V_{m+1}^i \) is conditionally independent on each other subdomain \( V_k^i \) where \( k \neq j \), given observations on an agent interface between them at intervals \( i = 0, 1, \ldots, m, m+1 \). Therefore, line 9 is equivalent to \text{CommunicateBelief}. After line 9, the LJF for interval \( m+1 \) is globally consistent, and \( B(FI_{m+1}^i) \) retrieved from \( T_{m+1}^i \) in line 1 during next execution of \text{Forecast} satisfies

\[ B(FI_{m+1}^i) = \text{const} * P(FI_{m+1}^i|obs_0^i, \ldots, obs_m^i, obs_{m+1}^i). \]

At line 2, the LJF for \( i = m+2 \) is loaded by agents. After line 5, the potential over \( T_{m+2}^i \) becomes

\[ B'(V_{m+2}^i) = \text{const} * P(V_{m+2}^i|obs_0^i, \ldots, obs_m^i, obs_{m+1}^i), \]

and the LJF for \( i = m+2 \) is inconsistent. After line 6, the LJF is again globally consistent and \( B'(V_{m+2}^i) = \text{const} * P(V_{m+2}^i|obs_0^i, \ldots, obs_m^i, obs_{m+1}^i) \). Hence, forecast at line 7 on \( i = m+2 \) is exact. \( \square \)

Experiments

The 3-agent supply chain DMSBN in Fig. 4 and the equivalent centralized DBN are implemented using WebWeaverIV. Each batch of experiment is conducted on a group of ten scenarios each of horizon 7, simulated from the DBN. For each scenario, five forecasting sessions \((S_1, \ldots, S_5)\) may be run. \( S_2, S_4 \) and \( S_5 \) are run using the DMSBN. In \( S_2, \)
only variables in agent interfaces are observed as assumed by interface observability. Additional local observations are made in $S_1$. In $S_5$, the agent interface and forward interface are observed. In $S_1$ and $S_3$, 3 agents are run independently (using DBNs) without communication. In $S_1$, agents’ observations are identical to those in $S_2$. In $S_3$, they are identical to those in $S_4$.

A common probabilistic inference session combines deductive and abductive inference. Consider a directed path $x \rightarrow ... \rightarrow y \rightarrow ... \rightarrow z$. If the posterior on $y$ is needed, then the observation of $x$ drives deductive inference and the observation of $z$ drives abductive inference. Intuitively, forecasting is similar to deductive inference in the temporal direction, without the assistance of the abductive counterpart. As a result, the accuracy of forecasting is heavily dependent on the causal strength between present events and future events. To take this dependency into account, we conducted experiment at different levels of causal strength:

Let $v$ be a variable in the DMSBN associated with $P(v|\pi(v))$. For each instantiation $\pi(v)$ of $\pi(v)$, denote $M(v|\pi(v)) = \max_{v'} P(v|\pi(v))$, where $\max$ is over all possible values of $v$. $M(v|\pi(v))$ is a simple indicator of the causal strength. The closer it is to 1, the more predictable the value of $v$ when $\pi(v)$ is true. To set the level of causal strength for a DMSBN, a parameter $t \in (0, 1)$ is specified, and for each variable $v$, $M(v|\pi(v))$ is lower-bounded by $t$.

**Table 1: Forecasting accuracy with causal strength $t = 0.93$**

<table>
<thead>
<tr>
<th>Scenario</th>
<th>$S_1$</th>
<th>$S_2$</th>
<th>$S_3$</th>
<th>$S_4$</th>
<th>$S_5$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.65</td>
<td>0.65</td>
<td>0.71</td>
<td>0.81</td>
<td>0.88</td>
</tr>
<tr>
<td>2</td>
<td>0.69</td>
<td>0.82</td>
<td>0.67</td>
<td>0.79</td>
<td>0.81</td>
</tr>
<tr>
<td>3</td>
<td>0.64</td>
<td>0.68</td>
<td>0.71</td>
<td>0.88</td>
<td>0.90</td>
</tr>
<tr>
<td>4</td>
<td>0.55</td>
<td>0.54</td>
<td>0.59</td>
<td>0.63</td>
<td>0.73</td>
</tr>
<tr>
<td>5</td>
<td>0.60</td>
<td>0.65</td>
<td>0.71</td>
<td>0.95</td>
<td>0.96</td>
</tr>
<tr>
<td>6</td>
<td>0.60</td>
<td>0.76</td>
<td>0.67</td>
<td>0.87</td>
<td>0.87</td>
</tr>
<tr>
<td>7</td>
<td>0.51</td>
<td>0.65</td>
<td>0.63</td>
<td>0.77</td>
<td>0.81</td>
</tr>
<tr>
<td>8</td>
<td>0.51</td>
<td>0.55</td>
<td>0.54</td>
<td>0.67</td>
<td>0.68</td>
</tr>
<tr>
<td>9</td>
<td>0.72</td>
<td>0.85</td>
<td>0.71</td>
<td>0.83</td>
<td>0.83</td>
</tr>
<tr>
<td>10</td>
<td>0.63</td>
<td>0.62</td>
<td>0.73</td>
<td>0.85</td>
<td>0.85</td>
</tr>
<tr>
<td><strong>mean</strong></td>
<td>0.61</td>
<td>0.68</td>
<td>0.66</td>
<td>0.81</td>
<td>0.83</td>
</tr>
</tbody>
</table>

**Table 2: Forecasting accuracy with causal strength $t = 0.80$**

<table>
<thead>
<tr>
<th>Scenario</th>
<th>$S_1$</th>
<th>$S_2$</th>
<th>$S_3$</th>
<th>$S_4$</th>
<th>$S_5$</th>
</tr>
</thead>
<tbody>
<tr>
<td>11</td>
<td>0.41</td>
<td>0.50</td>
<td>0.49</td>
<td>0.62</td>
<td>0.55</td>
</tr>
<tr>
<td>12</td>
<td>0.69</td>
<td>0.72</td>
<td>0.73</td>
<td>0.83</td>
<td>0.83</td>
</tr>
<tr>
<td>13</td>
<td>0.55</td>
<td>0.58</td>
<td>0.65</td>
<td>0.71</td>
<td>0.68</td>
</tr>
<tr>
<td>14</td>
<td>0.60</td>
<td>0.76</td>
<td>0.59</td>
<td>0.76</td>
<td>0.79</td>
</tr>
<tr>
<td>15</td>
<td>0.53</td>
<td>0.54</td>
<td>0.59</td>
<td>0.67</td>
<td>0.68</td>
</tr>
<tr>
<td>16</td>
<td>0.35</td>
<td>0.41</td>
<td>0.40</td>
<td>0.51</td>
<td>0.56</td>
</tr>
<tr>
<td>17</td>
<td>0.58</td>
<td>0.58</td>
<td>0.67</td>
<td>0.71</td>
<td>0.71</td>
</tr>
<tr>
<td>18</td>
<td>0.64</td>
<td>0.63</td>
<td>0.72</td>
<td>0.79</td>
<td>0.79</td>
</tr>
<tr>
<td>19</td>
<td>0.68</td>
<td>0.73</td>
<td>0.76</td>
<td>0.94</td>
<td>0.94</td>
</tr>
<tr>
<td>20</td>
<td>0.55</td>
<td>0.72</td>
<td>0.62</td>
<td>0.81</td>
<td>0.85</td>
</tr>
<tr>
<td><strong>mean</strong></td>
<td>0.56</td>
<td>0.62</td>
<td>0.62</td>
<td>0.73</td>
<td>0.74</td>
</tr>
</tbody>
</table>

We simulated three groups of scenarios (10 each). $G_1, G_2, G_3$, with strength parameter 0.93, 0.8, 0.7, respectively. For each scenario in $G_1$ and $G_2$, sessions $S_1$,...,$S_5$ are run. For each scenario (of horizon 7), six forecasting are made. The accuracy over 13 variables (distributed among agents) in each forecasting is recorded. Tables 1 and 2 show the average accuracy over $13 \times 6 = 78$ variables.

By comparing results between $S_1$ and $S_2$, and between $S_2$ and $S_4$, it can be seen that DMSBN agents have more accurate forecasting than DBN agents. By comparing results between $S_1$ and $S_3$, and between $S_2$, $S_4$ and $S_5$, it can be seen that more observations result in more accurate forecasts by both DBN and DMSBN agents.

In addition, we run session $S_5$ for each scenario in $G_3$, and the average accuracy over 10 scenarios is 0.53. From the average accuracies of $S_5$ in $G_1$, $G_2$ and $G_3$, i.e., 0.83, 0.74 and 0.53, respectively, it is clear that stronger causal strength in the environment results in more accurate forecasting.

**Acknowledgements**

We acknowledge the financial support from Discovery Grant, NSERC, Canada, to the first author, and funding from EPSRC CRiSM Initiative through the second author.

**References**


