2004 Command and Control Research and Technology Symposium
The Power of Information Age Concepts and Technologies

Title: **Transforming Timed Influence Nets into Time Sliced Bayesian Networks***

Authors: **Sajjad Haider and Abbas K. Zaidi**

Point of Contact: **Sajjad Haider**

Organization: System Architectures Laboratory, George Mason University.

Address: System Architectures Laboratory
C3I Center MSN 4B5
George Mason University
Fairfax, VA 22030

Author Information:

Mr. Sajjad Haider
System Architectures Laboratory.
C3I Center, MSN 4B5
George Mason University
Fairfax, VA 22030-4444
703-993-1725 (v)
703-993-1706 (f)
shaider1@gmu.edu

Dr. Abbas K. Zaidi
System Architectures Laboratory
C3I Center, MSN 4B5
George Mason University
Fairfax, VA 22030-4444
703-993-1714 (v)
703-993-1706 (f)
szaidi2@gmu.edu

Student Paper

* This research was sponsored by Office of Naval Research under Grant No. N00014-03-1-0033 and the Air Force Research Laboratory Information Directorate under Grant No. F30602-01-C-0065.

Transforming Timed Influence Nets into Time Sliced Bayesian Networks*

Sajjad Haider and Abbas K. Zaidi
System Architectures Laboratory
C3I Center MSN 4B5
George Mason University
Fairfax, VA 22030
<shaider1>,<szaidi2>@gmu.edu
Tel: 703 993 1725; 703 993 1774

Abstract

The paper presents an algorithm for transforming Timed Influence Nets (TIN) into Time Sliced Bayesian Networks (TSBN). The advantage of TINs lies in their ability to represent both causal and time-sensitive information in a compact and integrated manner. They are used to help a decision maker model the causal and temporal interdependencies among variables in a system. The TIN formalism offers a suite of analysis tools that can be used by a user to analyze the impact of alternate courses of actions on likely outcomes. An even larger, and more robust suite of analysis tools exists for TSBNs. These algorithms also allow analyses that are not available in the TIN formalism, e.g., provision for incorporating real-time information in the form of evidence regarding certain variables and calculating its impact on the rest of the system. The knowledge acquisition process of TSBNs, however, is intractable for large models. This paper is an attempt to combine the advantages of both modeling paradigms, TIN and TSBN, into a single formalism by providing a mapping from a TIN to a TSBN. The proposed formalism uses the TIN approach for the model building and the TSBN for analysis and evaluation. A system analyst, in this combined approach, interacts with a TIN, and the analysis results obtained on the TSBN are mapped back to the TIN, making the transformation completely hidden to the analyst.

1. Introduction

The easy access to domain-specific information and cost-effective availability of high computational power have changed the way people think about complex decision problems in almost all areas of application, ranging from financial markets to regional and global politics. These decision problems often require modeling of informal, uncertain, and unstructured domains in order for a decision maker to evaluate alternates and available courses of actions. The past few years have witnessed an emergence of several modeling and analysis formalisms that try to address this need. The modeling of

* This research was sponsored by Office of Naval Research under Grant No. N00014-03-1-0033 and the Air Force Research Laboratory Information Directorate under Grant No. F30602-01-C-0065.

an uncertain domain using Probabilistic Belief Networks, or more commonly known as Bayesian Networks (BNs), is considered to be the most used and popular of all such formalisms. The BN approach requires a subject expert to model the parameters of the domain – random variables – as nodes in a network. The arcs (or directed edges) in the network represent the direct dependency relationships between the random variables. The arrows on the edges depict the *direction* of the dependencies. The strengths of these dependencies are captured as conditional probabilities associated with the connected nodes in a network. A complete BN model requires specification of all conditional probabilities prior to its use. The number of conditional probabilities on a node in a BN grows exponentially with the number of inputs to the node. The requirement of specifying an exponentially large number of conditional probabilities presents a, at times insurmountable, modeling challenge. Cheng et al. [1994] developed a formalism, at George Mason University, called CAusal STrength (CAST) logic, as an intuitive, and approximate language to elicit the large number of conditional probabilities from a small set of user-defined parameters. The logic requires only a pair of parameter values for each dependency relationship between any two random variables. The CAST logic is used as a knowledge elicitation interface to an underlying BN. The approach was subsequently named Influence Nets [Rosen and Smith, 1996]. The Influence Nets require a system modeler (or subject expert) to specify the CAST logic parameters instead of the probabilities. The required probabilities are internally generated by the CAST logic with the help of user-defined parameters. The Influence Nets are, therefore, appropriate for modeling situations in which it is difficult to fully specify all conditional probability values and/or the estimates of conditional probabilities are subjective and estimates for the conditional probabilities cannot be obtained from empirical data, e.g., when modeling potential human reactions and beliefs.

Both Bayesian Networks and Influence Nets are designed to capture *static* interdependencies among variables in a system. A situation where the impact of a variable takes some *time* to reach the affected variable(s) cannot be modeled by either of the two approaches. In the last several years, efforts have been made to integrate the notion of time and uncertainty. Wagenhals et al. [Wagenhals et al. 1998] have added a special set of temporal constructs to the basic formalism of Influence Nets. The Influence Nets with these additional temporal constructs are called Timed Influence Nets (TINs). TINs have been experimentally used in the area of Effects Based Operations (EBOs) for evaluating alternate courses of actions and their effectiveness to mission objectives. The provision of time allows for the construction of alternate courses of actions as timed sequences of actions or actionable events represented by nodes in a TIN [Wagenhals and Levis, 2000; Wagenhals and Levis, 2001; Wagenhals et al., 2003].

The TIN approach inherits both the advantages and disadvantages of the Influence Net formalism: it offers an intuitive, and compact, knowledge elicitation interface for modeling purposes, but lacks some important analysis techniques. Currently, the analysis suite of TINs lacks the ability to incorporate the real-time information/evidence coming from different sources during the execution of a previously selected course of action. In a military/political scenario, this new information might come from the surveillance system regarding an adversary's actions. In an economic domain, a new development in a stock market, e.g., bankruptcy filed by some corporation, might be taken into account before making a strategic decision. In any case, this new information results in the revision of a previously held belief about some variables in the system. Haider and Levis [Haider and Levis, 2004] have recently proposed an algorithm to overcome this limitation; however, the approach is applicable for a special class of evidences only.

On a parallel track, scholars in the BN community extended the BN formalism to incorporate a special notion of time in it. The extension, called Time Sliced Bayesian Networks (TSBN) or Dynamic Bayesian Networks [Murphy, 2002], has gained a privileged status among the Artificial Intelligence community as a tool for modeling time and uncertainty. The approach is based on *unrolling* a static BN on a discrete time line with each time *slice* having an instance of a node in the network. The temporal dependencies are modeled with the help of edges across these time *slices*. Several sophisticated techniques for enhancing the capabilities of this approach have been proposed [Hanks et al., 1995; Figueroa and Sucar, 1999; Santos and Young, 1999; Galan and Diez, 2002]. Furthermore, several algorithms have also been proposed to compute the marginal probabilities of the random variables in an efficient manner [Kjaerulff, 1992; Boyen and Koller, 1998; Doucet et al., 2000; Murphy and Weiss, 2001; and Takikawa et al., 2002].

The lack of a comprehensive suite of analysis techniques in the TIN formalism and the recent developments in the field of TSBN bring us to the topic of this paper: The paper is an attempt to combine the advantages of both paradigms, TIN and TSBN, into a single formalism by providing a mapping from a TIN to a TSBN. The proposed formalism uses the TIN approach for the model building and the TSBN for analysis and evaluation. The paper demonstrates that TINs provide a compact and an intuitive way of modeling dynamic domains. A system modeler, therefore, can specify the uncertainties and the temporal constraints, present in a problem domain, in the form of a TIN. Once a TIN is fully specified, it can be converted into a TSBN using the approach presented in this paper. On one hand, the conversion simplifies the intractable task of knowledge elicitation in TSBNs by suggesting the use of TINs as a front end tool; while on the other, the conversion makes it possible to use a variety of analysis algorithms that have been developed for TSBNs.

The rest of the paper is organized as follows: Section 2 provides a technical background of Timed Influence Nets and Time Sliced Bayesian Networks. The algorithm for transforming TIN into TSBN is described in Section 3 with the help of examples. Finally, Section 4 discusses the conclusions and proposes directions for future research.

2. Technical Background

2.1 *Bayesian Networks*

Over the last two decades, Bayesian Networks, (BNs) have become a popular way of modeling uncertainty in several fields of studies [Pearl, 1987; Charniak, 1991; Jensen, 2001; Neapolitan, 2003]. A BN is a Directed Acyclic Graph (DAG) **G = (V, E)**. The nodes or vertices (**V**) in the graph represent random variables while edges (**E)** connecting pairs of variables represent probabilistic dependencies between them. Definitions 2.1- 2.3 present a formal description of BNs and the related terminology.

Definition 2.1 [Neapolitan, 2003]

Given a DAG **G** = (**V**, **E**) and nodes X and Y in **V**, Y is called a *parent* of X if there is an edge from Y to X, Y is called a *descendent* of X and X is called *ancestor* of Y if there is a path from X to Y, and Y is called a *nondescendent* of X if Y is not a descendant of X.

Definition 2.2 [Neapolitan, 2003]

Suppose we have a joint probability distribution **P** of the random variables in some set **V** and a DAG **G** = (**V, E**). We say that (**G**, **P**) satisfies the *Markov Condition* if for each variable X $\in$ **V**, {X} is conditionally independent of the set of all its nondescendent given the set of all its parents.

Definition 2.3

Let **G** = (**V, E**) be a DAG and **P** be the joint probability distribution of **V**. If (**G**, **P**) satisfies the *Markov Condition* then **B** = (**V**, **E**, **P**) is called a *Bayesian Network* and **P** can be written as

$$P(x_1,\ldots\ldots,x_n) = \prod_{i=1}^{n} P(x_i \mid pa(x_i))$$

where *pa(x)* represents the set of all parents of *x* and *x* $\in$ V.

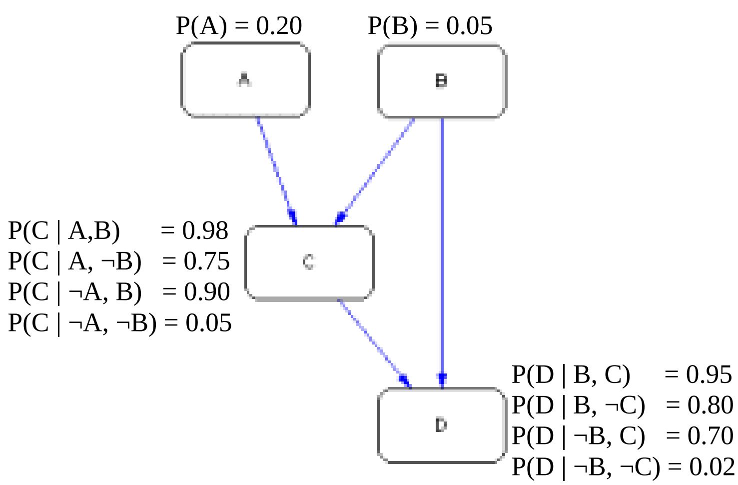

Figure 1: A Sample Bayesian Network

Figure 1 shows an example of a BN having four binary variables, namely, A, B, C, and D. The text in the figure shows the prior and conditional probabilities associated with the root and non-root nodes, respectively. The joint distribution of all the variables is computed as the product of these probabilities. For example, P (A, ¬B, C, D) is computed as

$$P(A, \neg B, C, D) = P(D \mid \neg B, C)\, P(C \mid A, \neg B)\, P(A)\, P(\neg B)$$

The other values in the joint distribution can be computed in a similar fashion. Once the computation of joint distribution is completed, it can be used to determine the marginal probabilities of the variables of interest. Several algorithms have been developed for various graphical structures of BN that compute the marginal probabilities in an efficient way by *propagating* probabilities without first calculating all the joint distribution values.

The random variables in a BN could be either discrete or continuous. Binary random variables are considered a special case of discrete random variables. The approach in this sequel, assumes that all the variables in a BN have binary states. The presented approach can be extended to more general cases.

2.2 *Time Sliced Bayesian Networks*

A TSBN works by discretizing time and creating instances of variables in a BN for each point in the time interval under consideration. The process starts with the identification of static cause and effect relationships among the variables and then by repeating the same structure for multiple time slices. Links are drawn between variables having temporal dependencies. Suppose, in the model of Figure 1, the probabilities of node A and B at time t depend upon their probabilities at time t-1. Then, the probabilities of A and B at time 1 are influenced by their respective probabilities at time 0; the probabilities at time 2 are influenced by the probabilities at time 1, and so on. These temporal dependencies can be captured in a TSBN as shown in Figure 2.

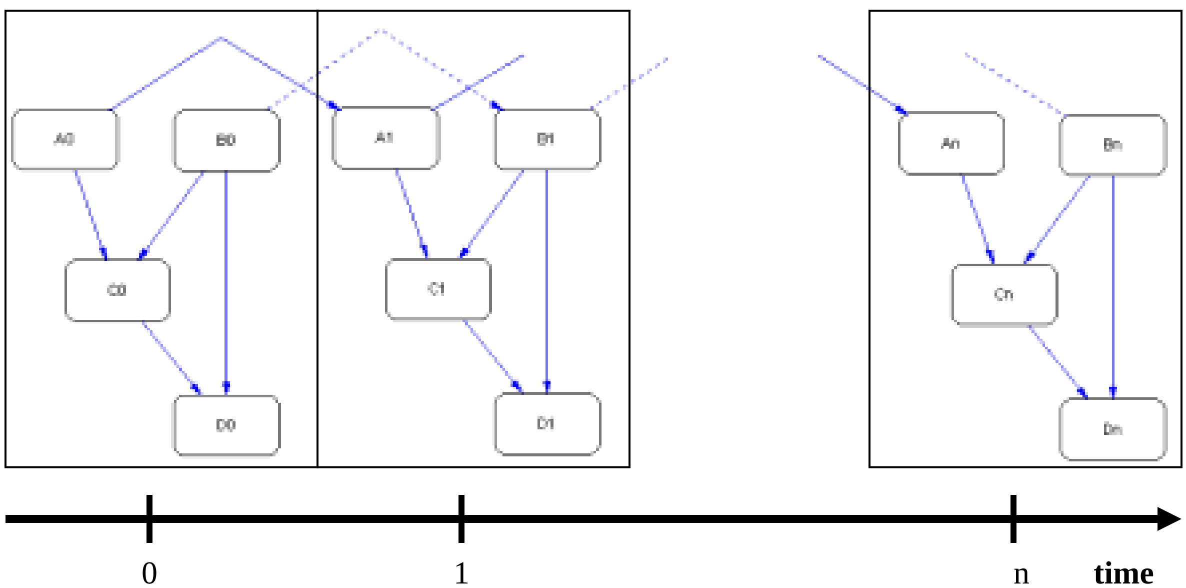

Figure 2: A Time Sliced Bayesian Network

Depending upon the situation, the state of a variable at time slice t may depend upon the states of influencing variables in preceding time slices ranging from t-1 to 0, i.e., the conditional probability of X at time t depends upon the set of influencing variable (the parents of X) at time slices ({t-1} or {t-1, t-2} or ….or {t-1,t-2,….,t-k}). In this context, a TSBN can be seen as an order ‘k’ *Markov Chain*. Typically, the TSBNs are built as order one Markov Chain, i.e., the future is conditionally independent of the past given the present. One more assumption that simplifies the specification of a TSBN is that the changes in the state of variables are caused by *stationary processes*. In other words, it is assumed that the conditional probabilities do not change over time, i.e.,

$$P(x_t \mid pa(x_t)) = P(x_{t-1} \mid pa\ (x_{t-1})) \quad t = 1, \ldots, n$$

In the sequel, a TSBN is assumed to have stationary conditional probabilities and of order one Markov process, unless stated otherwise.

2.3 *Timed Influence Nets*

As mentioned earlier, Influence Nets simplify the intractable task of eliciting Conditional Probability Tables (CPTs) from subject experts, especially when a node in the net has many parents. They use CAST Logic as an interface for eliciting CPTs. The logic has its origin in ‘Noisy-OR’ approach [Agosta, 1991; Drudzel and Henrion, 1993; Heckerman and Breese, 1996]. The CAST logic not only simplifies the elicitation of CPTs, but it also provides a mechanism to obtain information from various experts and then combine their individual assessments in a mathematical manner. The exact details of the CAST logic algorithm are beyond the scope of this paper. The interested reader should refer to Chang et al. [1994] and Rosen and Smith [1996].

Timed Influence Nets extend the capabilities of Influence Nets by allowing the provision of specifying several types of temporal information. These types can be broadly classified into two categories. One is related to the delays present in a problem domain while the other is related to the actionable events. The delays present in the domain represent the amount of time it takes for knowledge about a change, in the status of any variable, to be propagated to the node that is affected by that change. In TINs, this phenomenon is modeled by associating delays to arcs and nodes. The delay on an arc represents the communication delay, while a delay on a node represents the information processing delay. The second type of temporal information in TINs is associated with the actions taken in a course of action. Wagenhals et al. [Wagenhals et al., 2003] have called this type an *input scenario*. It describes the time at which the actions are taken and the intervals during which these actions are maintained. Actions in this context refer to the random variables that are modeled as root nodes in the corresponding TIN. In Bayesian literature, these actions could correspond to having the evidence on the root nodes. It is assumed that the actions occur instantaneously. Because of the dynamic nature of the problem, it is possible that the state of an action is changed during a course of actions. Thus, an action can be true during a particular time interval and false in another. Furthermore, these actions can be repeated an arbitrary number of times. It should be mentioned that causal strengths in TINs do not change over time. It is, therefore, assumed that like TSBNs, the changes in the state of variable in TIN are caused by stationary processes. The following items characterizes a TIN:

1. A set of random variables that makes up the nodes of a TIN. All the variables in the TIN have binary states.
2. A set of directed links that connect pairs of nodes.
3. Each link has associated with it a pair of CAST Logic parameters that shows the causal strength of the link (usually denoted as g and h values).
4. Each non-root node has an associated CAST Logic parameter (denoted as baseline probability), while a prior probability is associated with each root node.
5. Each link has a corresponding delay d (where $d \geq 0$) that represents the communication delay.
6. Each node has a corresponding delay e (where $e \geq 0$) that represents the information processing delay.
7. A pair (p, t) for each root node, where p is a list of real numbers representing probability values. For each probability value, a corresponding time interval is defined in t. In general, (p, t) is defined as

$$([p_1, p_2, \ldots, p_n], [[t_{11}, t_{12}], [t_{21}, t_{22}], \ldots, [t_{n1}, t_{n2}]] \text{ where } t_{i1} < t_{i2} \text{ and } t_{ij} > 0 \quad \forall\, i = 1, 2, \ldots, n \text{ and } j = 1, 2$$

Formally, a TIN is described by either of the following definitions (Definitions 2.4a, b, c).

Definition 2.4a

A *Timed Influence Net* is a tuple (**V, E, C, B, $\mathbf{D_E}$, $\mathbf{D_V}$, A**) where
V: set of Nodes,
E: set of Edges,
C represents causal strengths: **E** → { (**h**, **g**) such that $-1 \leq$ **h**, **g** ≤ 1 },
B represents Baseline / Prior probability: **V** → [0,1],
$\mathbf{D_E}$ represents Delays on Edges: **E** → $\mathbb{N}$,
$\mathbf{D_V}$ represents Delays on Nodes: **V** → $\mathbb{N}$, and
A (input scenario) represents the probabilities associated with the state of actions and the time associated with them.

$$\mathbf{A}: \mathbf{R} \rightarrow \{([p_1, p_2, \ldots, p_n], [[t_{11}, t_{12}], [t_{21}, t_{22}], \ldots, [t_{n1}, t_{n2}]]) \text{ such that } p_i = [0, 1], \; t_{ij} \rightarrow \mathbb{Z} \text{ and } t_{i1} \leq t_{i2}, \forall\, i = 1, 2, \ldots, n \text{ and } j = 1, 2 \text{ where } \mathbf{R} \subset \mathbf{V} \}$$

Definition 2.4a can be further simplified by reducing some of the elements in the tuple. The elements **C, B** in the tuple are used to approximate conditional probabilities, which in turn, are used to represent the joint distribution **P** of the random variables in **V**.

Definition 2.4b

Given a TIN (**V, E, C, B, $\mathbf{D_E}$, $\mathbf{D_V}$, A**), the elements **C**, **B** can be replaced by **P** that represents the joint distribution of the variables in **V**.The transformation is done by CAST Logic.

$$\text{TIN} = (\mathbf{V, E, C, B, D_E, D_V, A}) \rightarrow \text{TIN} = (\mathbf{V, E, P, D_E, D_V, A})$$

The elements $\mathbf{D_E}$ **and** $\mathbf{D_V}$ in the Definitions 2.4a and 2.4b represent the delays associated with the edges and nodes, respectively. The delay associated with a node can be remodeled by adding it to the delays on its incoming arcs and removing it from the corresponding node. For example, consider the TIN shown in Figure 3 (a).

(a) **(b)**

Figure 3: Reassignment of Node Delays

The delays on the links between A-C and B-C are 2 and 1 time units, respectively. The delay associated with the node C is 2 time units. Figure 3(b) shows the equivalent net with the node delay transformed to the edge delays. This transformation yield the following definition for TINs.

Definition 2.4c

Given a TIN = (**V**, **E**, **P, $\mathbf{D_E}$, $\mathbf{D_V}$, A**), the elements $\mathbf{D_E}$ and $\mathbf{D_V}$ can be mapped into an equivalent **D** that represents the transformed delays associated with the edges **E** in a TIN.

$$\text{TIN} = (\mathbf{V}, \mathbf{E}, \mathbf{P}, \mathbf{D_E}, \mathbf{D_V}, \mathbf{A}) \rightarrow \text{TIN} = (\mathbf{V}, \mathbf{E}, \mathbf{P}, \mathbf{D}, \mathbf{A})$$

3. Transformation from TINs to TSBNs

The existing algorithms for TINs propagate the influence of actions in the forward direction only, i.e., the probabilities are propagated from source (input nodes) to sink (or target nodes) through intermediate nodes. This presents an analysis and computational limitation of TINs for situations where observations, regarding states of non-root nodes in a TIN, arrive during the execution of a selected course of action. An approximation algorithm [Haider and Levis, 2004] has been proposed for incorporating such observations. The algorithm, however, puts certain restrictions on the timing of input evidence, thus making it impractical for some cases. One way of overcoming this limitation is to transform a TIN into a TSBN. The exact details of the transformation algorithm are presented in this section.

3.1 The Algorithm

The transformation algorithm first determines the required number of time slices by taking the maximum length of the paths that exist between the root nodes and the target node. More slices are added later when a course of action is selected. The additional slices are determined by looking at the largest time stamp associated with the actions in the selected course of action. Later, the connections between the nodes are established based on the time delays associated with the arcs that connect two nodes in the TIN. The subsequent indices of a root node (representing actionable events) are also connected except for the time when an action is taken. The exact algorithm is presented in Table 1.

Table 1: The Transformation Algorithm

Given TIN = (**V, E, P, D, A**)

1. Find the maximum path length between the root nodes and target nodes, i.e.,

   M = $\max_{i,j}[P_{i,j}]$ where

   $P_{i,j}$: path between nodes i and j such that i, k $\in$ V and $\neg\exists$ (k, i) $\in$ E

2. Construct a TSBN (**V1, E1, P1**) where

   V1: $\forall$ $\nu \in$ **V** add ν_i to **V1** where i = 0, 1, …, M

   = {ν_i | $\nu \in$ **V**, i = 0, 1, …., M}

   E1 = {(x_i, y_j) | i = max (0, j – D(x, y)); x, y $\in$ **V** and i, j = 0, 1, …., M}

   P1: **P** when indices are ignored

   For example, P(y_i | x_i) = P(y | x) when x, y $\in$ **V** and x_i, y_j $\in$ **V1**.

This step draws the nodes in the TSBN for M time slices. The connections are drawn between the non-root nodes and their parents. The following step is required once an input scenario is determined.

3. Let S = maximum time stamp associated with the root nodes as provided by the input scenario:
   (a) Add S additional time slices in the TSBN obtained in the previous step by following the procedure outlined in Step 2.
   (b) The resultant network is the modified TSBN (**V1, E1, P1**) where
   V1 = {ν_i | $\nu \in$ **V**, i = 0, 1, …., M+S}
   E1 = {(x_i, y_j) | i = max (0, j – D(x, y)); x, y $\in$ **V** and i, j = 0, 1, …., M+S }
   P1: **P** when indices are ignored
   (c) Let **R1** = Set of Root Nodes where **R1** $\subset$ **V1**. $\forall$ r $\in$ **R1** connect r_{t-1} to r_t where t = 1, 2, …, M+S, unless t is the time at which the variable is set to a state.

3.2 Application of the Transformation Algorithm

This section illustrates the transformation algorithm with the help of an example. The example model is shown in Figure 4. The TIN in the figure shows a hypothetical crisis that arose on a piece of land, belonging to a peaceful country, but annexed by a hostile country B. The objective of building this model is to explore the possibility of a peaceful solution of the crisis, or, in other words, the objective is to determine the probability that country B would agree to withdraw its forces based upon certain actions taken by the international community. There are four nodes in the TIN, namely, A, B, C, and D. The description of these nodes is shown in the figure. The text besides the links represents the delays associated with them. For instance, the link between B and C has a delay of 1 time unit while the link between C and D has a delay of 3 time units, and so on. The steps involved in the transformation algorithm are shown in Figure 5.

Step 1: In this step, the maximum path length between the root nodes (A and B) and the target node (D) is determined. The path A-C-D has the maximum length of 5. Thus, M is set to 5.

Step 2: This step draws the nodes in the TIN for M time slices in the corresponding TSBN. The step is shown in Figure 5(a). After drawing the nodes for 5 time slices, the connections between the nodes are drawn. The delays on the arcs in the TIN determine the indices of the connected nodes in the corresponding TSBN. For instance, the delay between B and D is 2. The connections between instances of B and instances of D are determined as shown below:

D5 is connected to B3 as max (0, 5-2) = 3
D4 is connected to B2 as max (0, 4-2) = 2
⋮
D0 is connected to B0 as max (0, 0-2) = 0

Similarly, the connections between instances of C and instances of D are determined as follows:

D5 is connected to C2 as max (0, 5-3) = 2
⋮
D0 is connected to C0 as max (0, 0-3) = 0

The process is shown in Figures 5(b) and 5(c).

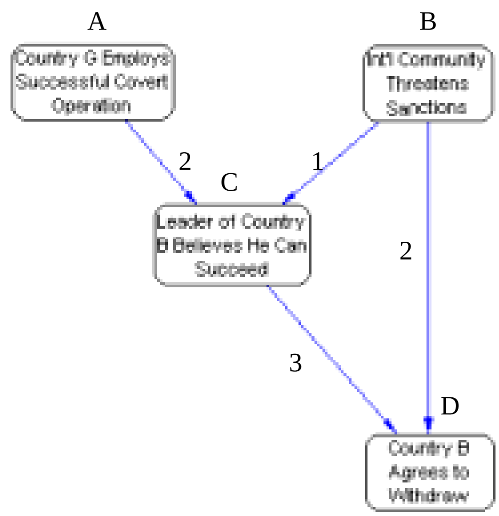

Figure 4: A Simple Timed Influence Net

Step 3: Let the input scenario is given. This step adds additional nodes and links in the TSBN based upon the selected course of actions provided as an input scenario. Suppose in the input scenario, action A is taken at time 2 while action B is taken at time 1. The maximum time stamp associated with the actionable nodes is 2, therefore this step adds two more slices to the TSBN and connects the parents and children as described in the previous step. Furthermore, the connections are added between the nodes representing actionable events as explained in Step 3(c) of the algorithm. For instance A0 and A1 are connected but since A is taken at time 2 therefore there is no connection between A1 and A2. The connections, however, are drawn between A2 and A3 and between A3 and A4, etc. Similarly, as B is taken at time 1 therefore connections are made between B1 and B2, B2 and B3, etc. while B0 is not connected to B1. The final TSBN is shown in Figure 5(d).

Once a TSBN is obtained from the corresponding TIN, the task of real time execution monitoring can be accomplished by entering observations in the model that arrive during the execution of the selected course of action. Suppose in the model of Figure 5(d), the evidence regarding variable D is received. The evidence states that D happens to be true at time 4. In the figure, all the indices of D equal to or greater than 4 are set to true. Thus D4, D5, D6, and D7 in Figure 5(d) become the evidence nodes. This new information revises the belief about the state of other non-actionable variables (non-root nodes) in the problem domain. For instance, this information would change the initial belief about C at time 1, 2, and onwards. If the time associated with the new information is greater than the number of slices drawn in the TSBN then more slices could be added to it. For example, if the new information says that D occur at time 9 then the system modeler can add few more slices with the help of step 2 of the transformation algorithm in order to observe the impact of this new information on C at time 6 and onwards.

It can be noticed in Figure 5(d) that variables in the TIN only depend upon the previous states of their parents and do not depend upon their own previous states. This is due to the fact that there is no link from node X_{t-1} to node X_t, where X belongs to the set of non-root nodes. There can be situations in which a variable's state at time t may depend upon its own state at a previous time stamp. In TSBNs, this issue is addressed by adding a link between different instances of the same variable at different time slices. The process is shown in Figure 6 where the links in bold show the connection between (C_{t-1}, C_t) and (D_{t-1}, D_t). In TINs this requirement can be modeled by adding self-loops to such nodes, as shown in Figure 6. This self-loop represents the dependency of the state of a variable at time t on its previous state.

4. Conclusions

The paper presented a transformation algorithm for converting Timed Influence Nets into Time Sliced Bayesian Networks. The transformation provides the equivalence that exists between TINs and a class of TSBNs. Furthermore, the approach suggested in the paper delivers the advantages of both modeling paradigms to a system modeler. On one hand, it simplifies the knowledge elicitation process of TSBNs by suggesting TINs as a front end tool for modeling time and uncertainty; while on the other, it enhances the current capabilities of the TINs by providing the modeler ability to enter evidence that arrives during plan execution. In other words, the approach suggests that TINs be used for model building and course of action selection process, and TSBNs for execution monitoring of the selected course of actions. The task of inference in TSBNs, however, is computationally intractable. Thus, there is a tradeoff between the available approximate and exact algorithms in terms of accuracy and the time to compute probability of the variable of interest. The future research would focus on determining a set of inference algorithms (exact or approximate) that works better with the class of TSBNs that are obtained from TINs as a result of the transformation.

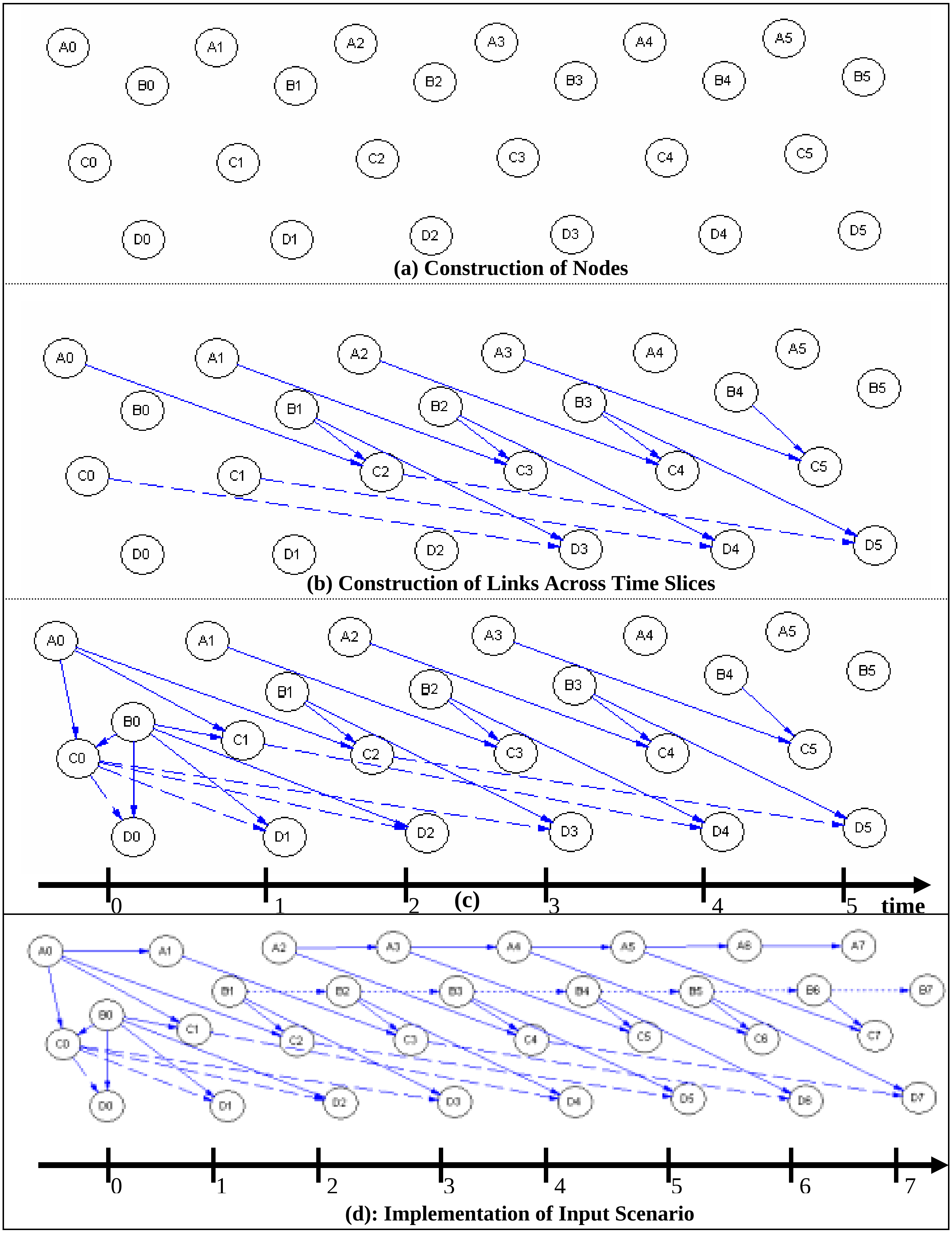

Figure 5: Steps of the Transformation Process

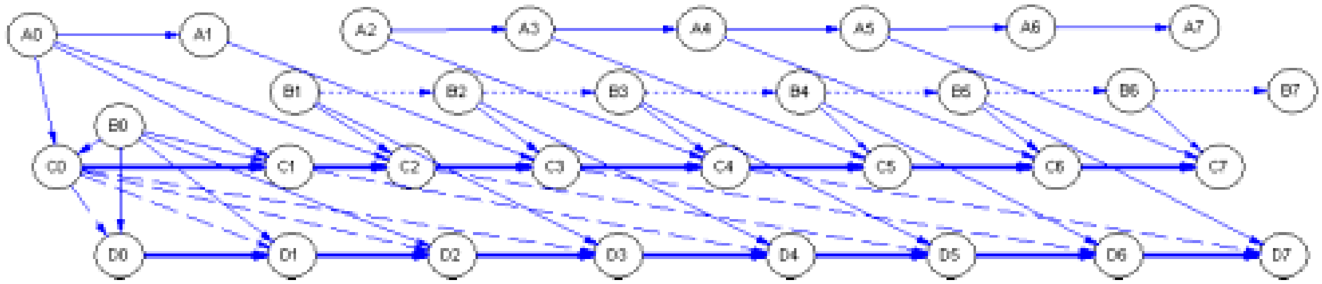

Figure 6: TSBN of Figure 5(d) with Dependencies Among Instances of Non-root Nodes

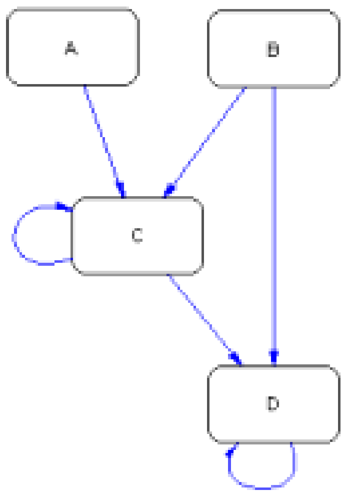

Figure 7: Timed Influence Net with Self-Loop

References

Agosta, J. M, "Conditional Inter-Causally Independent Node Distributions, a Property of Noisy-OR Models", In Proceedings of the 7th Conference on Uncertainty in Artificial Intelligence, 1991.

Boyen, X. and Koller, D. Tractable inference for complex stochastic processes. In Proc. of the Conf. on Uncertainty in AI, 1998.

Chang, K. C., Lehner, P. E., Levis, A. H., Zaidi, S. A. K., and Zhao, X., "On Causal Influence Logic", Technical Report, George Mason University, Center of Excellence for C3I, Dec. 1994.

Chariak, E., "Bayesian Network without Tears", AI Magzine, Winter 1991.

Doucet, A. de Freitas, N., Murphy, K. and Russell, S., "Rao-Blackwellised Particle Filtering for Dynamic Bayesian Networks", In Proceedings of the 16th Conference on Uncertainty in Artificial Intelligence, 2000.

Drudzel, M. J., and Henrion, M., "Intercausal Reasoning with Uninstantiated Ancestor Nodes", In the Proceedings of the 9th Conference on Uncertainty in Artificial Intelligence, 1993.

Figueroa, G. A., and Sucar, L. E., "A Temporal Bayesian Network for Diagnosis and Prediction", In Proceedings of the 15th Conference on Uncertainty in Artificial Intelligence, 1999.

Galan, S. F., and Diez, F. J., "Networks of Probabilistic Events in Discrete Time", International Journal of Approximate Reasoning, 30, pp. 181-202, 2002.

Haider, S. and Levis, A. H., “An Approximate Technique for Belief Revision in Timed Influence Nets”, Proceedings of Command and Control Research and Technology Symposium, 2004 (To Appear).

Hanks, S., Madigan, D., and Gavrin, J., “Probabilistic Temporal Reasoning with Endogenous Change”, In Proceedings of 11th Conference on Uncertainty in Artificial Intelligence, 1995.

Heckerman, D., and Breese, J. S., “Causal Independence for Probability Assessment and Inference Using Bayesian Networks”, IEEE Transactions on Systems, Man, and Cybernatics – Part A: Systems and Humans, Vol. 26, No. 6, Nov. 1996.

Jensen, F. V., “Bayesian Networks and Decision Graphs”, Springer-Veralg, 2001.

Kjaerulff, U., “A Computational Scheme for Reasoning in Dynamic Probabilistic Networks”, In Proceedings of 8th Conference on Uncertainty in Artificial Intelligence, 1992.

Murphy, K and Weiss Y., “The Factored Frontier Algorithm for Approximate Inference in DBNs”, In Proceedings of 17th Conference on Uncertainty in Artificial Intelligence, 2001.

Murphy, K., “Dynamic Bayesian Networks: Representation, Inference and Learning”, PhD Thesis, UC Berkley, Jul. 2002.

Neapolitan, R. E., “Learning Bayesian Networks”, Prentice Hall, 2003.

Pearl, J., “Probabilistic Reasoning in Intelligent Systems: Network of Plausible Inference”, Morgan Kaufmann, 1987.

Rosen, J. A., and Smith, W. L., “Influence Net Modeling with Causal Strengths: An Evolutionary Approach, Proceedings of the Command and Control Research and Technology Symposium, Naval Post Graduate School, Monterey CA, Jun. 1996.

Santos Jr. E., and Young, J. D., “Probabilistic Temporal Network: A Unified Frameword for Reasoning with Time and Uncertainty”, International Journal of Approximate Reasoning, 2, 1999.

Takikawa, M., d’Ambrosio, B. and Wright, E., “Real-Time Inference with Large-Scale Temporal Bayes Nets”, In Proceedings of the 18th Conference on Uncertainty in Artificial Intelligence, 2002.

Wagenhals, L. W., Shin, I., and Levis, A. H., “Creating Executable Models of Influence Nets with Coloured Petri Nets,” Int. J. STTT, Spring-Verlag, Vol. 1998, No. 2, 1998.

Wagenhals, L. W. and Levis, “Course of Action Development and Evaluation, Proceedings of Command and Control Research and Technology Symposium, 2000.

Wagenhals, L. W. and Levis, A. H., “Modeling Effects Based Operations in Support of War Games”, 15th International Symposium on Aerospace / Defense Sensing, Internal Society for Optical Engineering, Proceedings of SPIE, Vol. # 4367, 2001.

Wagenhals, L. W., Levis, A. H., and McCrabb, M. B., “Effects-Based Operations; A Historical Perspective of a Way Ahead”, Proceedings of 8th International Command and Control Research and Technology Symposium, 2003.