PROBABILISTIC REASONING

Unit # 6

MILK TEST

Milk from a cow may be infected. To detect whether the milk is infected, you have a test, which may give either a positive or a negative test result. The test is not perfect. It may give a positive result on clean milk as well as a negative result on infected milk.

MILK TEST - BN

Milk from a cow may be infected. To detect whether the milk is infected, you have a test, which may give either a positive or a negative test result. The test is not perfect. It may give a positive result on clean milk as well as a negative result on infected milk.

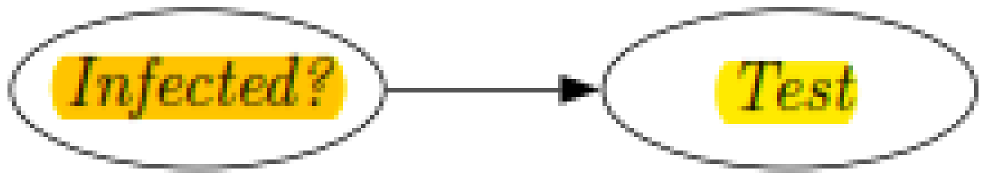


Fig. 3.1. The Bayesian network for the milk test.

MILK TEST TODAY

From one day to another, the state of the milk can change. Cows with infected milk will heal over time, and a clean cow has a risk of having infected milk the next day. Now, imagine that the farmer performs the test each day.

How would your model incorporate information from the last day?

What would happen after a week?

MILK TEST – OVER THE WEEK

After a week, he has not only the current test result but also the six previous test results. For each day of the week , we have a model. These seven models should be connected such that past knowledge can be used for the current conclusion. A natural way would be to let the state of the milk yesterday have an impact on the state today. This yields the below:

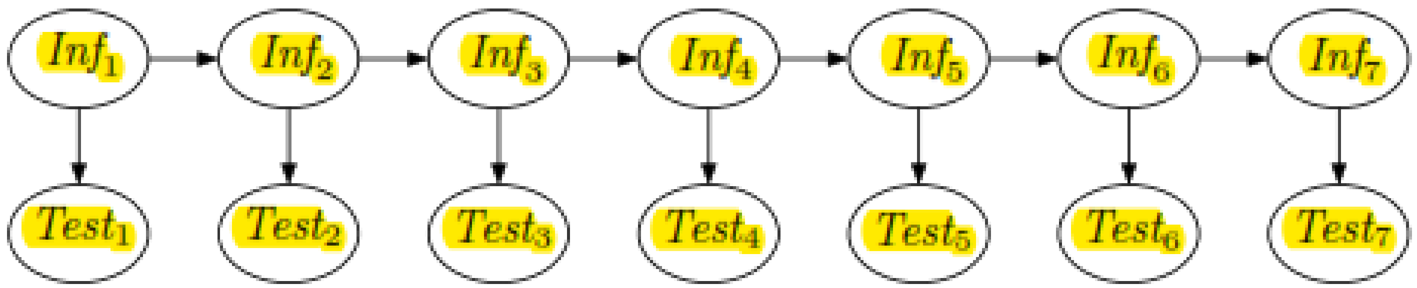


Fig. 3.2. A seven-day model for the milk test.

MILK TEST - MARKOV PROPERTY

Some diseases have a natural span of time. For example, if I have the flu today but was healthy yesterday, then I will most probably have the flu the day after tomorrow. On the other hand, if I have had the flu for four days, then there is a good chance that I will be cured the day after tomorrow.

MILK TEST - MARKOV PROPERTY

Some diseases have a natural span of time. For example, if I have the flu today but was healthy yesterday, then I will most probably have the flu the day after tomorrow. On the other hand, if I have had the flu for four days, then there is a good chance that I will be cured the day after tomorrow.

If the Markov property of Figure 3.2 does not reflect reality, the model should be changed. For example, it may be argued that you also need to go an extra day back, and the model will be like:

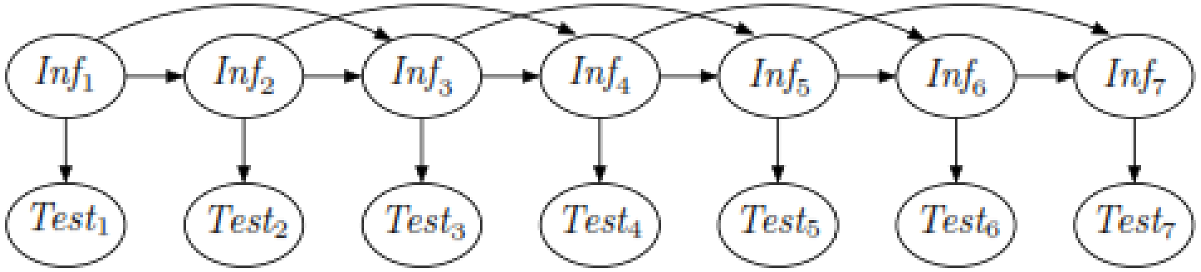


Fig. 3.3. A seven-day model with a two-day memory of infection.

DYNAMIC MODELS

The graph of a probabilistic network is restricted to be a finite acyclic directed graph.

This seems to imply that probabilistic networks as such do not support models with feedback loops or models of dynamic systems changing over time. This is not the case.

DYNAMIC MODELS (CONT'D)

A common approach to representing and solving dynamic models or models with feedback loops is to unroll the dynamic model for the desired number of time steps and treat the resulting network as a static network.

The unrolled static network is then solved using a standard algorithm applying evidence at the appropriate time steps.

DYNAMIC BAYESIAN NETWORK

Dynamic Bayesian networks (DBNs) are an extension of Bayesian networks to model dynamic processes.

A DBN consists of a series of *time slices* that represent the state of all the variables at a certain time, t; a kind of snapshot of the evolving temporal process.

For each temporal slice, a dependency structure between the variables at that time is defined, called the *base network*.

It is usually assumed that this structure is duplicated for all the temporal slices (except the first slice, which can be different).

TEMPORAL LINKS

Additionally, there are edges between variables from different slices, with their directions following the direction of time, defining the *transition network*.

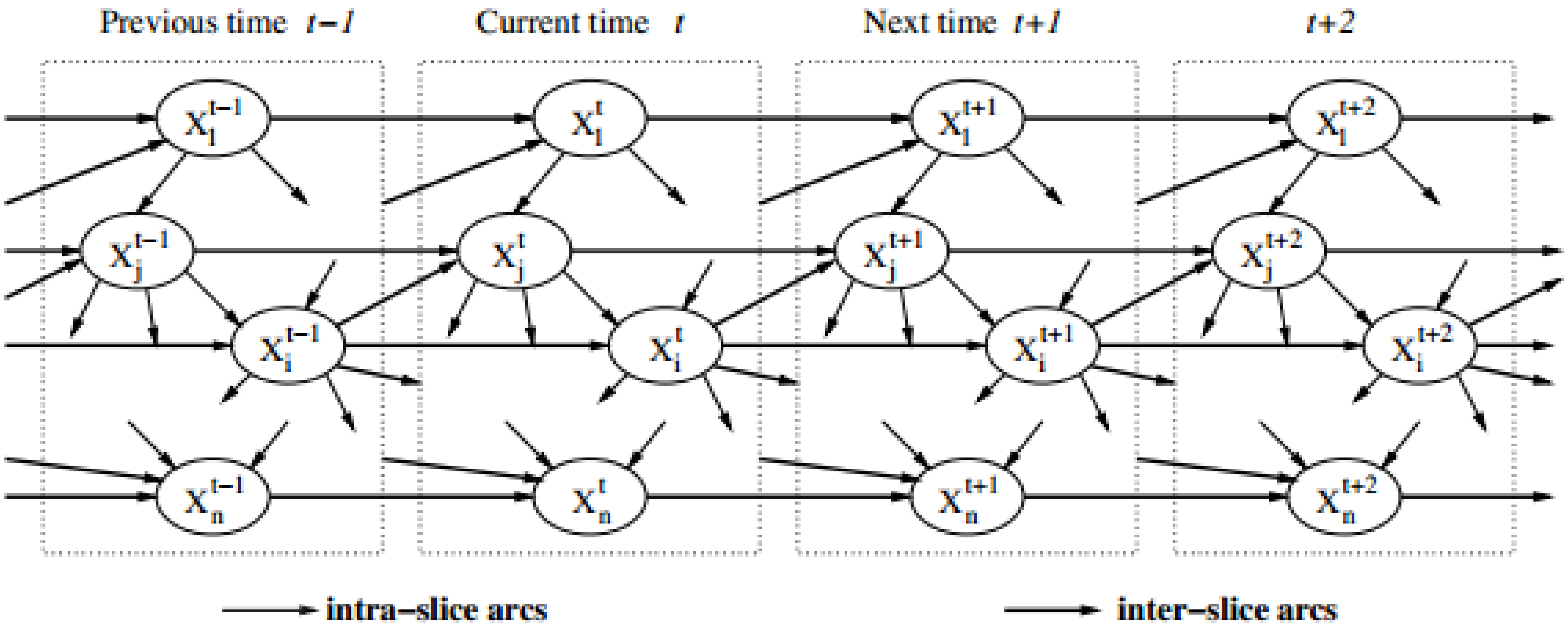
Usually, DBNs are restricted to have directed links between consecutive temporal slices, known as a first-order Markov model; although, in general, this is not necessary.

DBN REPRESENTATION

A compact specification of a DBN must include:

- Node names
- Intra-slice arcs
- Temporal (inter-slice) arcs
- CPTs for the first time slice t0 (when there are no parents from a previous time)
- CPTs for t + 1 slice (when parents may be from t or t + 1 time-slices)

GENERAL STRUCTURE OF A DBN



IMPORTANT CONCEPTS

Let i be the current time step, then

- *smoothing* is the process of querying about the state of the system at a previous time step $j < i$ given evidence about the system at time i,
- *filtering* is the process of querying about the state of the system at the current time step, and
- *prediction* is the process of querying about the state of the system at a future time step $j > i$.

IMPORTANT CONCEPTS (CONT'D)

- A dynamic Bayesian network is *stationary when the transition probability* distributions are invariant between time steps.
- A dynamic Bayesian network is first-order Markovian when the variables at time step i+1 are d-separated from the variables at time step i −1 given the variables at time step i.
- When a system is stationary and Markovian, the state of the system at time i + 1 only depends on its state at time i, and the probabilistic dependence relations are the same for all i.
- The Markovian property implies that arcs between time slices only go from one time slice to the subsequent time slice.

BASIC DBN

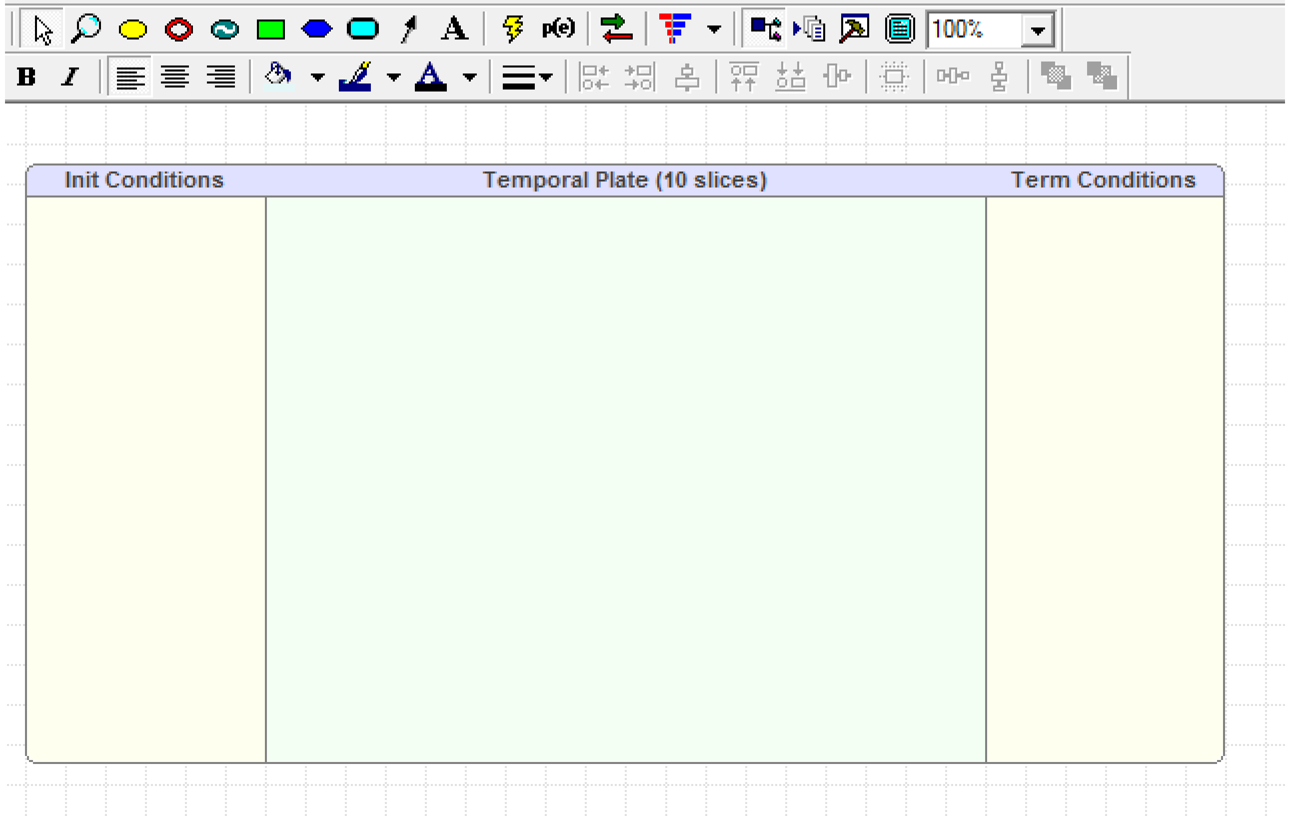
Most of the DBNs considered in practice satisfy the following conditions:

- First-order Markov model. The state variables at time t depend only on the state variables at time $t - 1$ (and other variables at time t).
- Stationary process. The structure and parameters of the model do not change over time.

GENIE DEMO FOR TEMPORAL MODELS

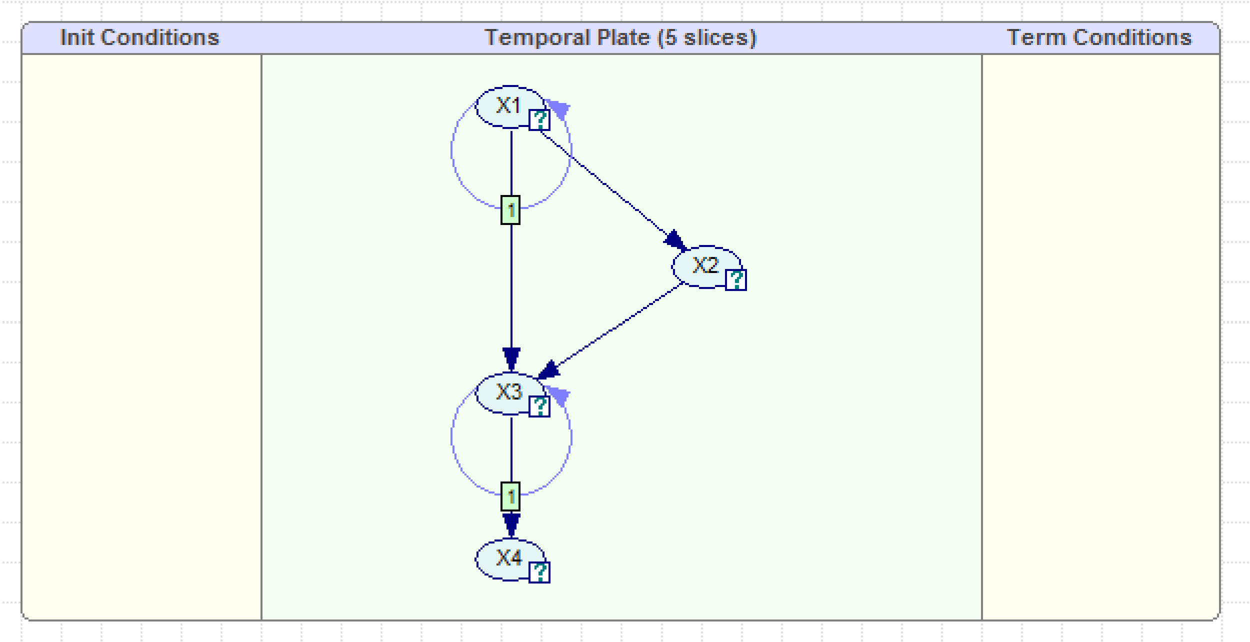
GENIE DEMO (CONT'D)

You can bring the "Temporal Plate" by clicking Network -> Enable Temporal Plate



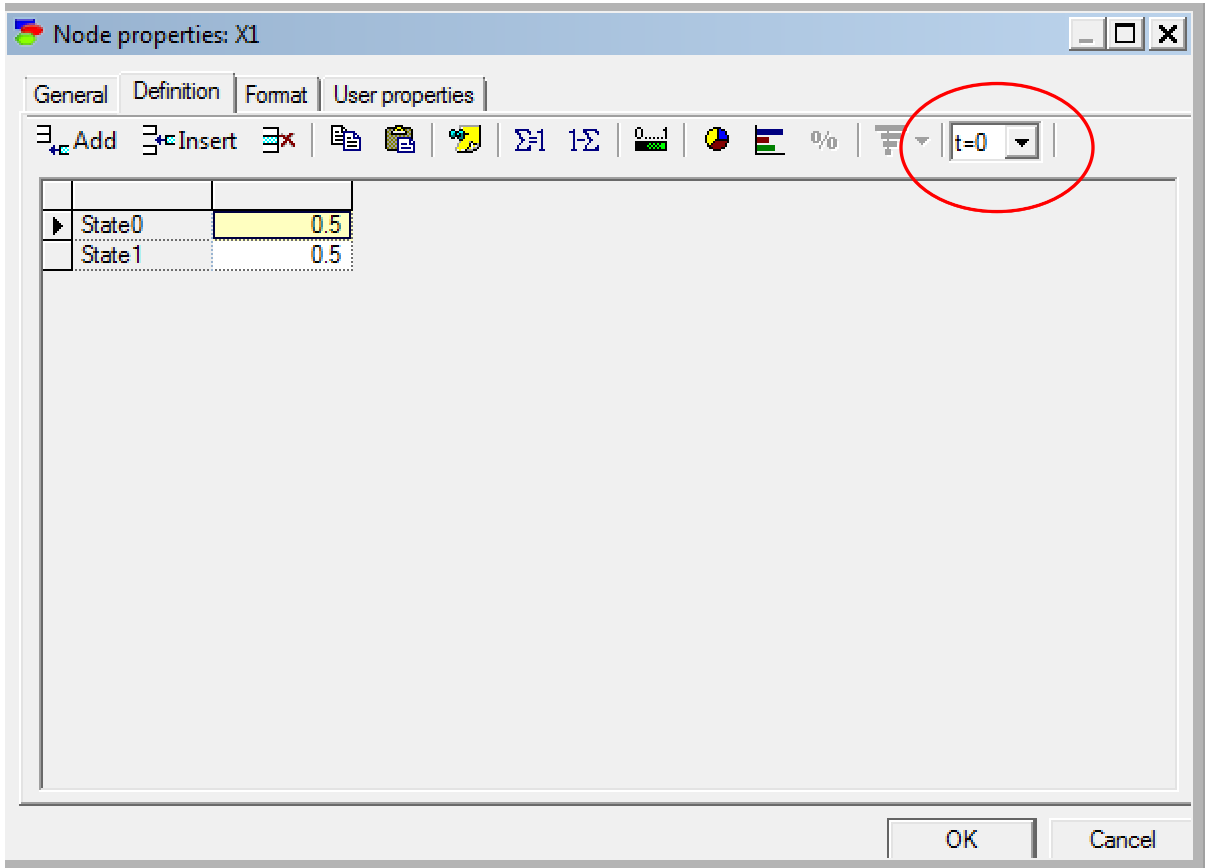
GENIE DEMO (CONT'D)

Dependence of a node on its previous state is modeled by drawing a self-loop.



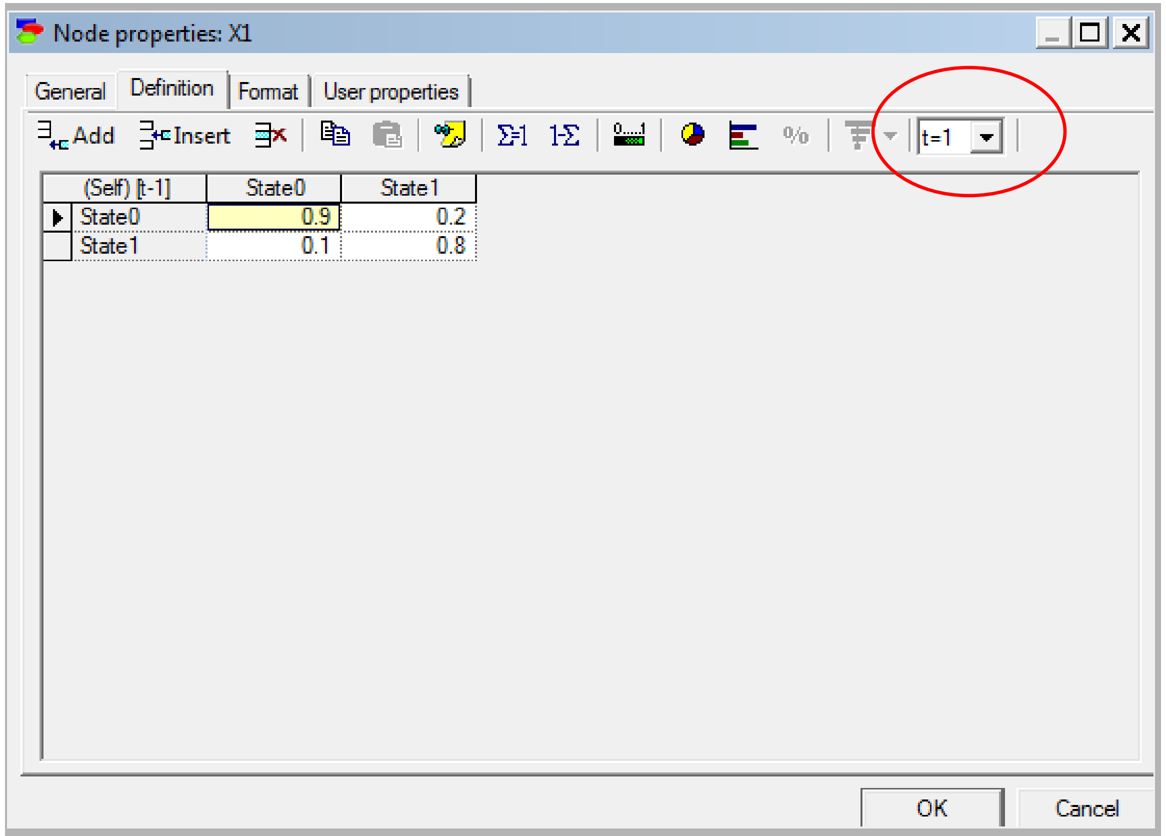
GENIE DEMO (CONT'D)

Prior probability for Node X1 is defined in the usual manner. The only difference is the presence of t=0.



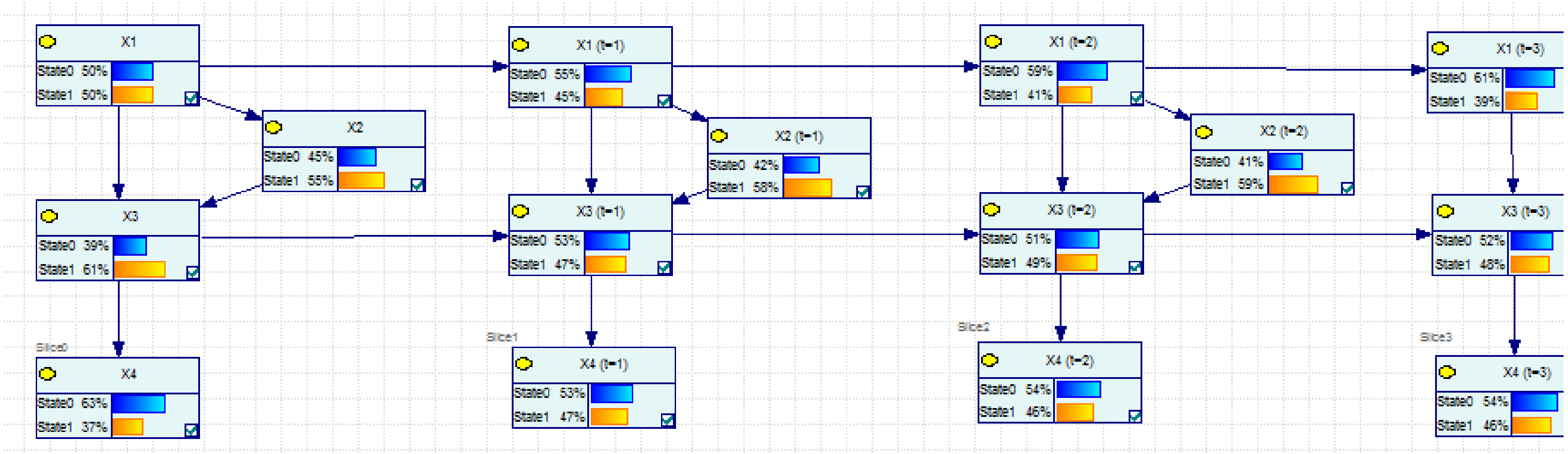
GENIE DEMO (CONT'D)

Temporal dependence is captured by specifying the transitional probabilities after setting t=1



GENIE DEMO (CONT'D)

DBN can be unrolled by selecting Network → Unroll option.



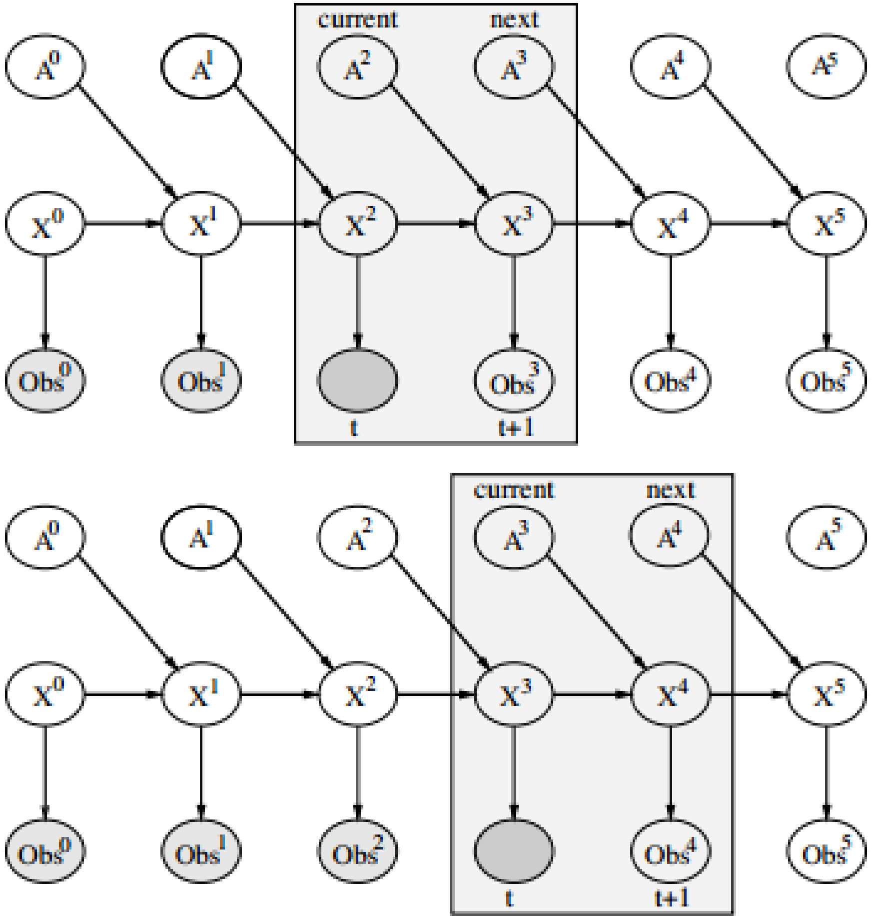
INFERENCE

In most cases, the DBN is not extended far into the future. Instead, a fixed size, sliding "window" of time slices is maintained.

As the reasoning process moves forward with time, one older time slice is dropped off the DBN, while another is added.

This use of a fixed window means that every time we move the window along, the previous evidence received is no longer directly available. Instead, it is summarized taking the current belief for (root) nodes, and making these distributions the new priors.

SLIDING WINDOW



APPLICATION OF DBN IN ANTI-MONEY LAUNDERING

APPLICATION OF DBN IN ANTI-MONEY LAUNDERING

Suspicious activity reporting (SAR) has been a crucial part of anti-money laundering (AML) systems.

Financial transactions are considered suspicious when they deviate from the regular behavior of their customers.

Money launderers pay special attention to keep their transactions as normal as possible to disguise their illicit nature.

Source: Saleha Raza and Sajjad Haider, Suspicious Activity Reporting using Dynamic Bayesian Networks, *Proceedia Computer Science*, 3, 2011, pp. 987-991.

APPLICATION (CONT'D)

An approach, called **SARDBN (Suspicious Activity Reporting using Dynamic Bayesian Network)**, was developed that employed a combination of clustering and dynamic Bayesian network (DBN) to identify anomalies in sequence of transactions.

SARDBN applied DBN to capture patterns in a customer's monthly transactional sequences as well as to compute an anomaly index

APPLICATION (CONT'D)

The first step was to form clusters of customers that exhibit similar transactional pattern.

The similarity in the transactional behavior was assessed by a customer's average monthly credit and debit amounts, frequency of credit and debit transactions, and delay in two consecutive transactions.

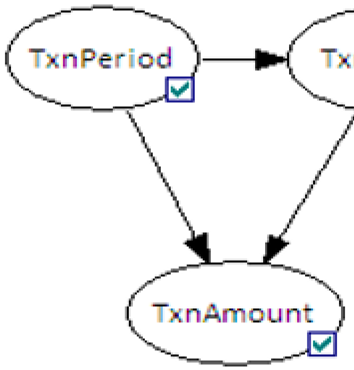
APPLICATION (CONT'D)

Once clustering was performed, the next step was to learn the parameters of a DBN for each cluster.

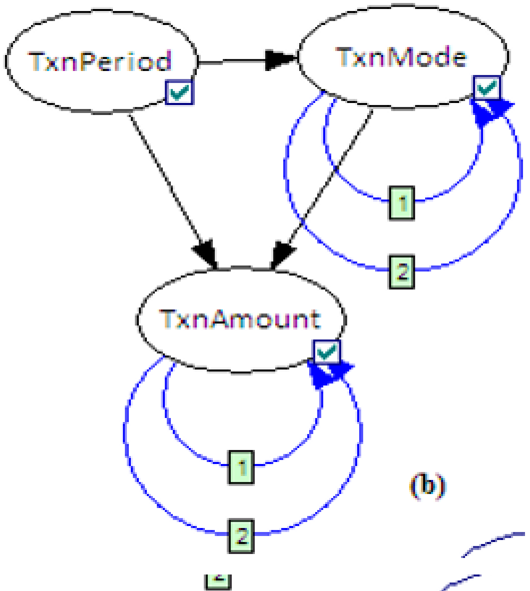
The structure of these DBNs can vary in different scenarios and is devised with the aid of subject matter experts.

For this study, the three variables under consideration are transaction amount (TxnAmount), mode of transaction (TxnMode) (e.g. cheque withdrawal/deposit, ATM withdrawal, salary transfer, POS payment, bank draft), and period of transaction (TxnPeriod) (i.e. start/middle/end of month).

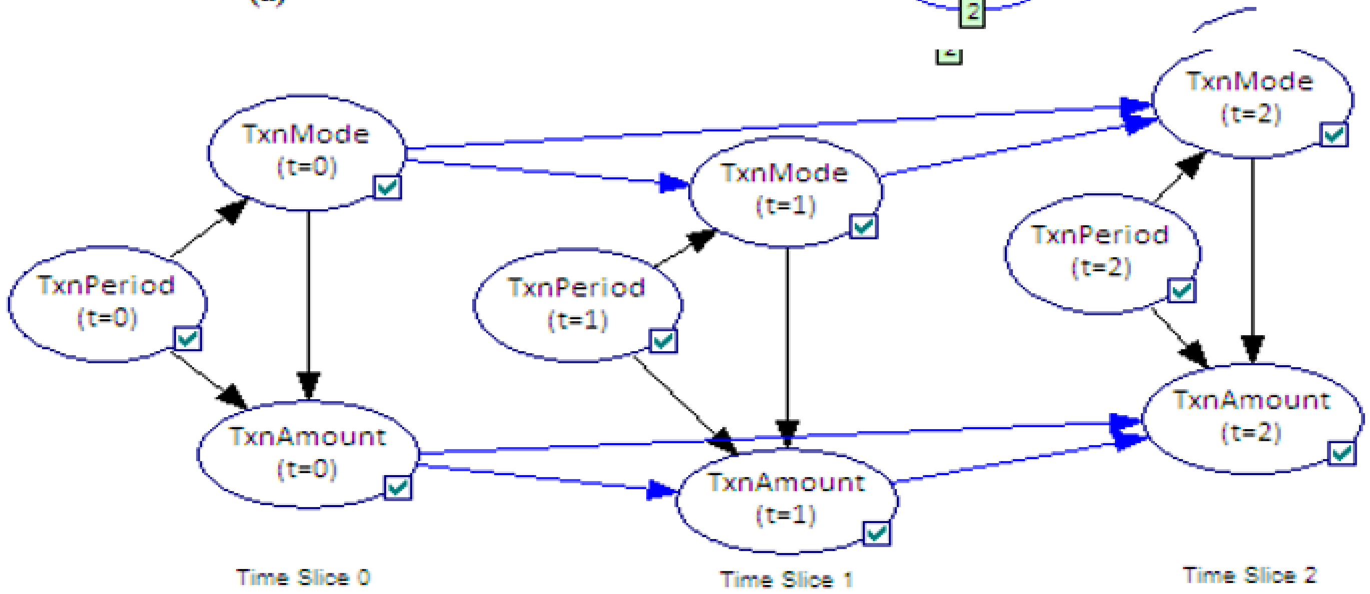
SARDBN - MODEL



(a)



(b)



SARDBN - INSTANTIATION

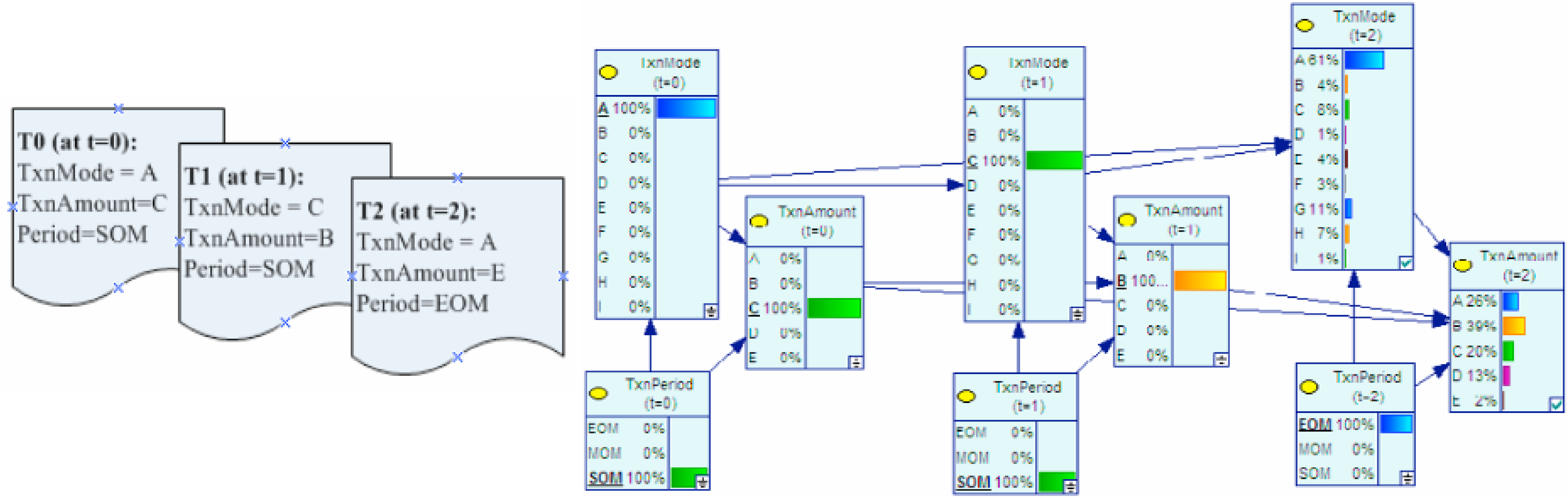


Fig. 3 - Instantiated DBN for a sequence of three transactions

SARDBN – ANOMALY SCORING

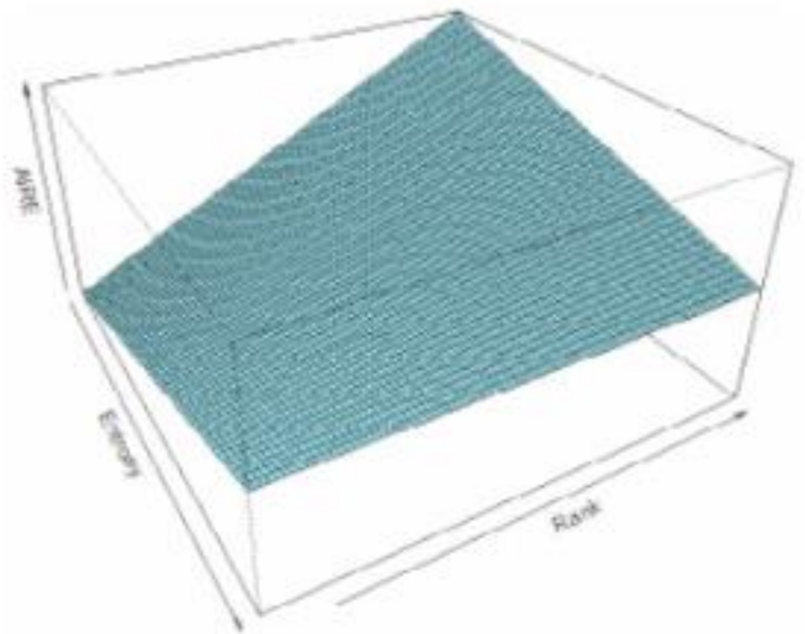


Fig. 2 – The surface showing AIRE as function of rank and entropy

SARDBN - RESULTS

	Correctly Predicted TxnMode		Correctly Predicted TxnAmount	
Cluster	1st Order DBN	2nd Order DBN	1st Order DBN	2nd Order DBN
1	62 %	71%	73%	79%
2	89 %	90%	94%	95%
3	64 %	51%	67%	69%
4	81 %	84%	79%	80%

Table 3 – Number of Anomalies

Threshold	Number of Anomalies
0.6	917
0.65	338
0.7	143
0.75	65
0.8	27
0.85	4

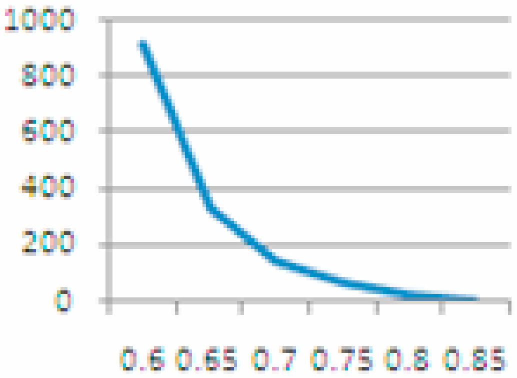


Fig. 4. Anomalies vs Threshold

REFERENCE

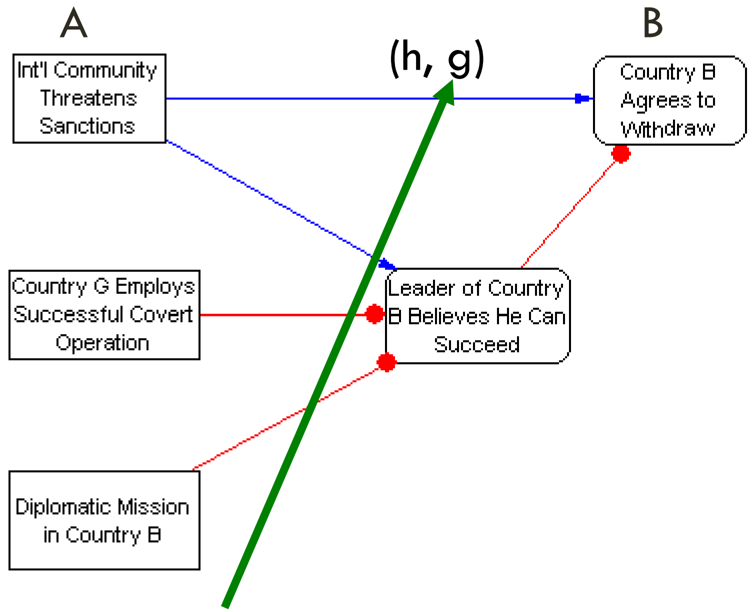
The following slides on Dynamic Influence Nets is taken from the following paper "Sajjad Haider and Alexander H. Levis, Modeling Time-varying Uncertain Situations using Dynamic Influence Nets, *International Journal of Approximate Reasoning*, 49 (2), 2008, pp. 488-502"

INFLUENCE NETS

- A set of random variables that makes up the nodes of an IN. All the variables in the IN have binary states.
- A set of directed links that connect pairs of nodes.
- Each link has associated with it a pair of parameters that shows the causal strength of the link (usually denoted as h and g values).

Positive Impact

Negative Impact



h is Influence of A on B: Analogous to $P(B \mid A)$

g is Influence of $\neg A$ on B: Analogous to $P(B \mid \neg A)$

Root Nodes

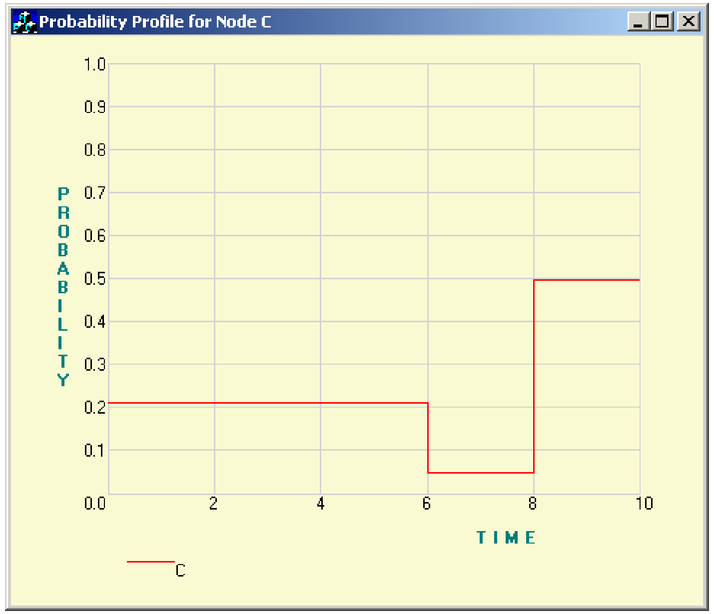
Non-root Nodes

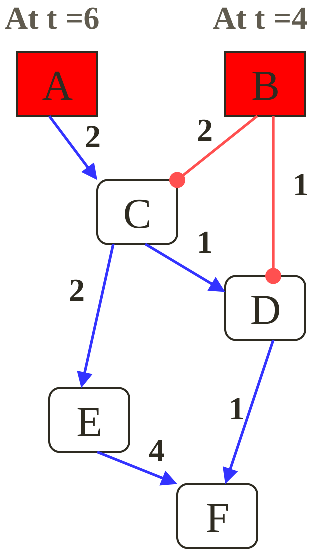
DEMO OF IN IN IBAYES

TIMED INFLUENCE NET

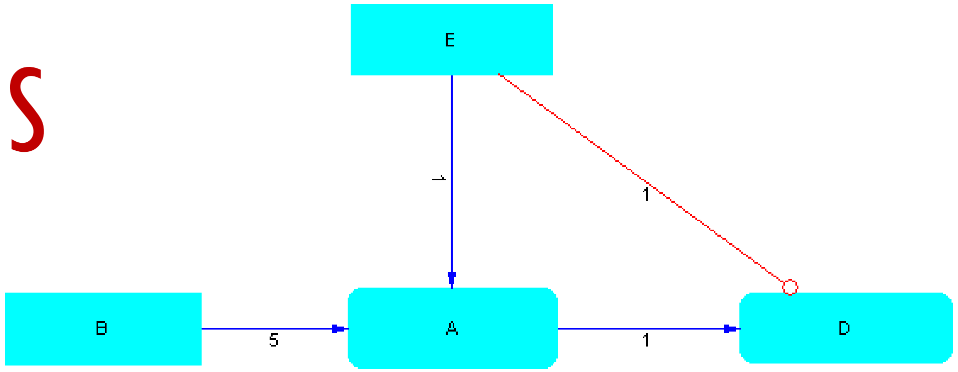
The specification of a TIN require the following additional parameters besides the one required for by an ordinary IN:

- A time delay is associated with each arc.
- A time delay is associated with each node.
- Each actionable event is assigned time stamp(s) at which the decision(s) regarding the state of that action is(are) made





PROBABILITY PROFILES

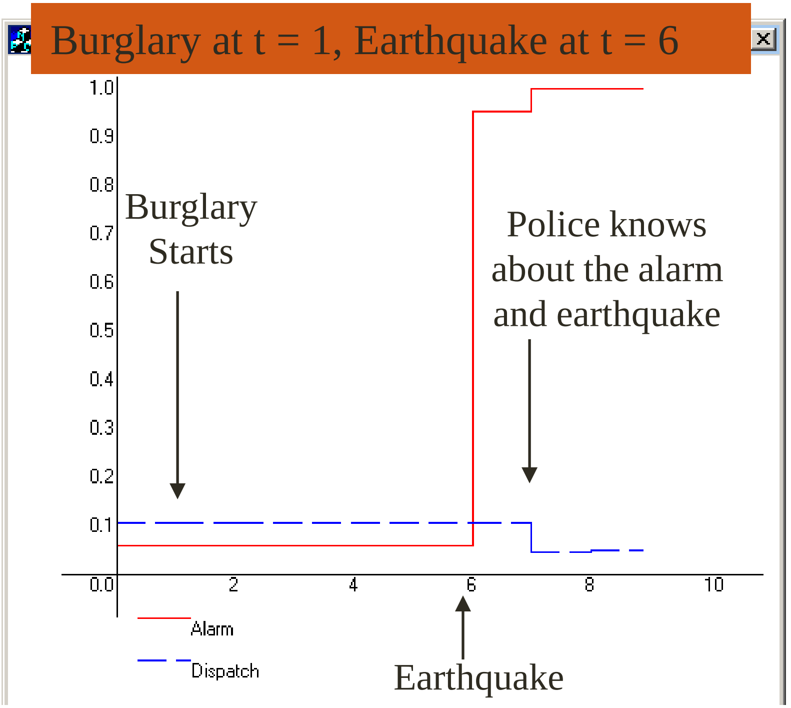


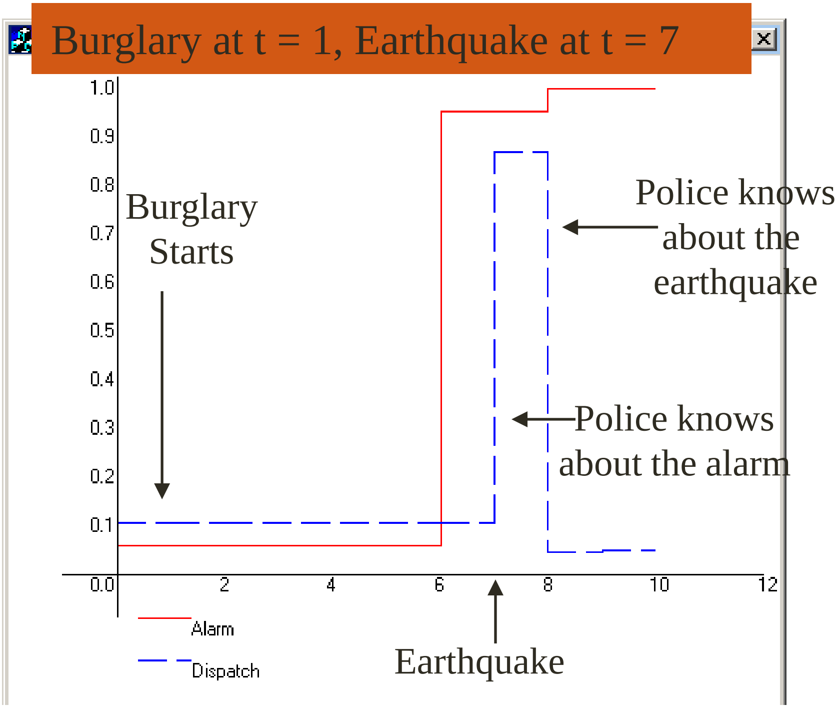
B: Burglary

E: Earthquake

A: Alarm Goes Off

D: Police Dispatch



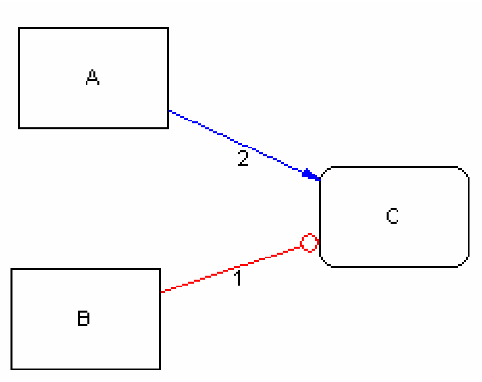


TIME-VARYING INFLUENCE (PERSISTENCE OF INFLUENCE)

Classical TINs only models time-invariant influences. A variation was proposed for modeling time-varying influences

A list of influences along with their time of effect is specified for each arc in a TIN

The proposed scheme can be used to model time-dependent structural changes in a TIN



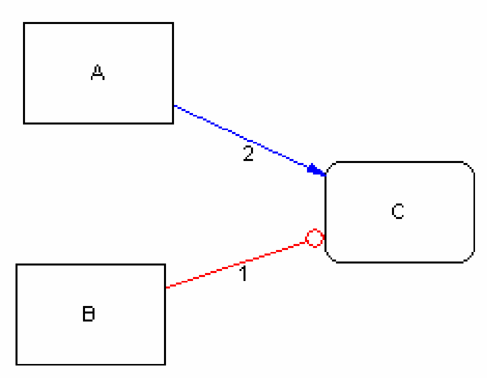
Influence of A on C when information at A is t time units old
Strong: $2 \leq t < 4$
Moderate: $4 \leq t \leq 6$
Low: $t > 6$

Influence of B on C when information at B is t time units old
Strong: $1 \leq t < 3$
Low: $t > 3$

TIME-VARYING INFLUENCE (CONT'D)

$P(A) = 0.05 @ 0$
$= 1.0 @ 4$

$P(B) = 0.1 @ 0$
$= 0.6 @ 7$
$= 1.0 @ 10$



Influence of A on C when information at A is t time units old

Strong: $2 \leq t < 4$

Moderate: $4 \leq t \leq 6$

Low: $t > 6$

Influence of B on C when information at B is t time units old

Strong: $1 \leq t < 3$

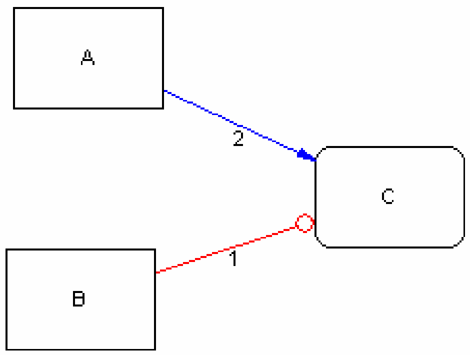
Low: $t > 3$

- C is updated at time: 6, 8, 11
- At time 6: P(A) @ 4 is used, while P(B) @ 0 is used
  - Information coming from A is 2 time units old
  - Information coming from B is 6 time units old
    - C has strong influence of A and low influence of B at time 6
- At time 8: P(A) @ 4 is used, while P(B) @ 7 is used
  - Information coming from A is 4 time units old
  - Information coming from B is 1 time units old
    - C has moderate influence of A and strong influence of B at time 8

NON-STATIONARY CONDITIONAL PROBABILITIES

P(A) = 0.05 @ 0
= 1.0 @ 4

P(B) = 0.1 @ 0
= 0.6 @ 7
= 1.0 @ 10



Influence of A on C when information at A is t time units old

Strong: $2 \leq t < 4$

Moderate: $4 \leq t \leq 6$

Low: $t > 6$

Influence of B on C when information at B is t time units old

Strong: $1 \leq t < 3$

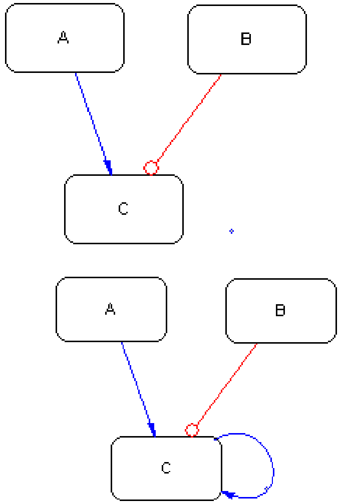
Low: $t > 3$

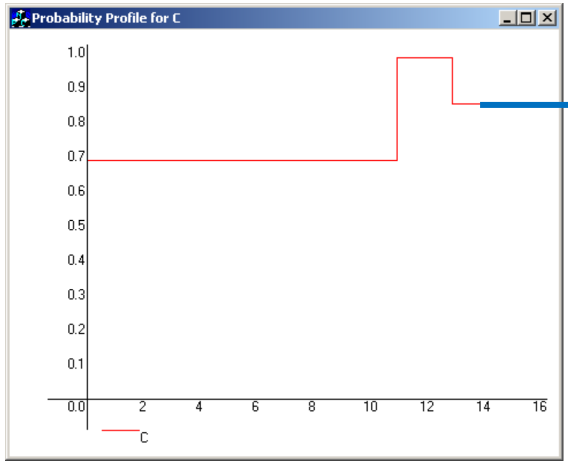
		Time	
Parents Combination	**6**	**8**	**11**
P(C|¬A,¬B)	**0.07**	**0.85**	**0.93**
P(C|¬A, B)	**0.03**	**0.02**	**0.03**
P(C| A,¬B)	**0.97**	**0.98**	**0.97**
P(C| A, B)	**0.93**	**0.15**	**0.07**

ADDING MEMORY TO THE NODES

TINs assumed that the nodes are memoryless and as a result the impact of different sequences of actions on the final probability was not captured.

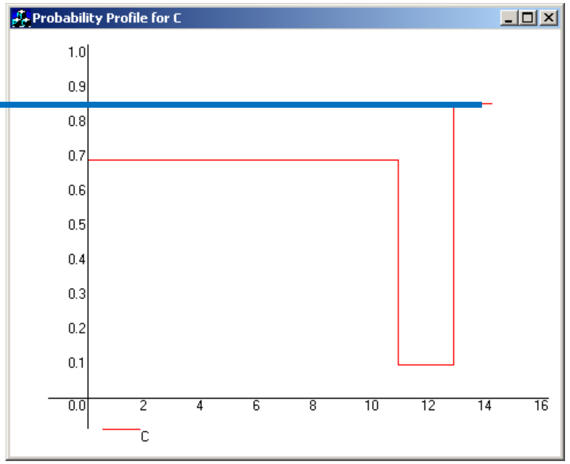
An approach was proposed that adds memory to the nodes in a TIN by adding a self-loop to each node whose current state is dependent on its previous state.





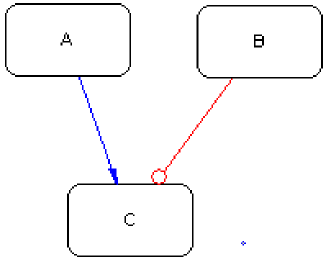
A @ 10, B @ 12

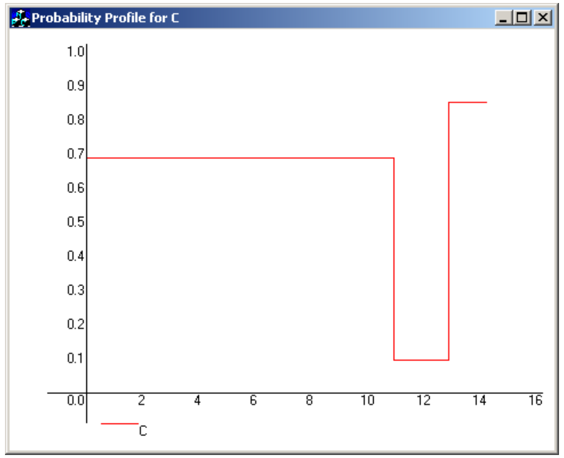
Same Final Probability



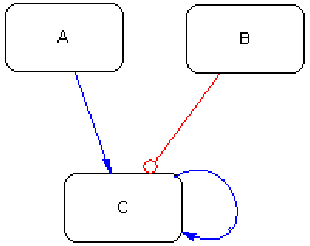
B @ 10, A @ 12

ADDING MEMORY TO THE NODES (CONT'D)

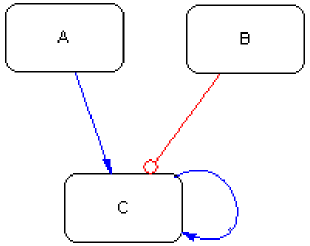




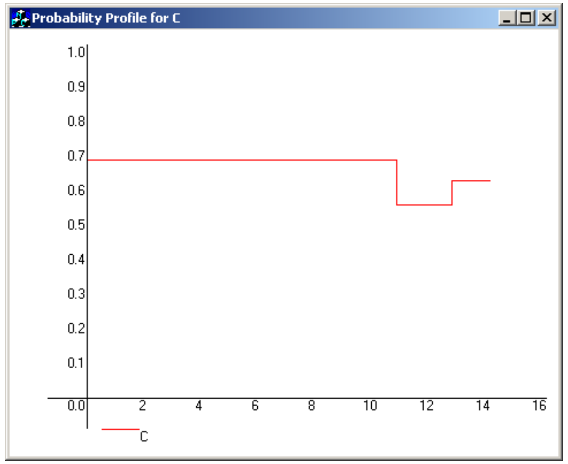
B @ 10, A @ 12

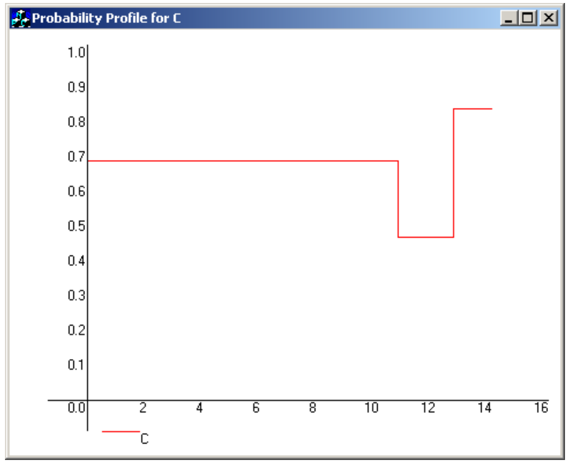


(0.33, -0.33)
Weak Memory



(0.90, -0.90)
Strong Memory



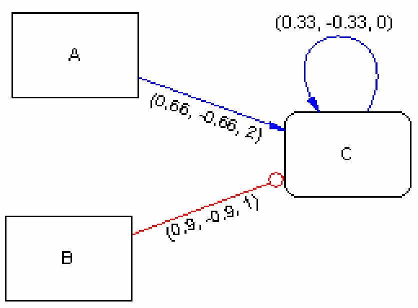


DYNAMIC INFLUENCE NETS

Timed Influence Nets with

- Time-varying influences
- Memory represented by a self-loop

P(A) = 0.05 @ 0
= 1.0 @ 4



P(B) = 0.1 @ 0
= 0.6 @ 7
= 1.0 @ 10

Influence of A on C when information at A is t time units old
Strong: $2 \leq t < 4$
Moderate: $4 \leq t \leq 6$
Low: $t > 6$

Influence of B on C when information at B is t time units old
Strong: $1 \leq t < 3$
Low: $t > 3$

TIN AND DIN DEMO WITH IBAYES