Loopy Propagation: the Convergence Error in Markov Networks

Janneke H. Bolt
Institute of Information and Computing Sciences, Utrecht University
P.O. Box 80.089, 3508 TB Utrecht, The Netherlands
janneke@cs.uu.nl

Abstract

Loopy propagation provides for approximate reasoning with Bayesian networks. In previous research, we distinguished between two different types of error in the probabilities yielded by the algorithm; the cycling error and the convergence error. Other researchers analysed an equivalent algorithm for pairwise Markov networks. For such networks with just a simple loop, a relationship between the exact and the approximate probabilities was established. In their research, there appeared to be no equivalent for the convergence error, however. In this paper, we indicate that the convergence error in a Bayesian network is converted to a cycling error in the equivalent Markov network. Furthermore, we show that the prior convergence error in Markov networks is characterised by the fact that the previously mentioned relationship between the exact and the approximate probabilities cannot be derived for the loop node in which this error occurs.

1 Introduction

A Bayesian network uniquely defines a joint probability distribution and as such provides for computing any probability of interest over its variables. Reasoning with a Bayesian network, more specifically, amounts to computing (posterior) probability distributions for the variables involved. For networks without any topological restrictions, this reasoning task is known to be NP-hard (Cooper 1990). For networks with specific restricted topologies, however, efficient algorithms are available, such as Pearl's propagation algorithm for singly connected networks. Also the task of computing approximate probabilities with guaranteed error bounds is NP-hard in general (Dagum and Luby 1993). Although their results are not guaranteed to lie within given error bounds, various approximation algorithms are available that yield good results on many real-life networks. One of these algorithms is the *loopy-propagation algorithm.* The basic idea of this algorithm is to apply Pearl's propagation algorithm to a Bayesian network regardless of its topological structure. While the algorithm results in exact probability distributions for a singly connected network, it yields approximate probabilities for the variables of a multiply connected network. Good approximation performance has been reported for this algorithm (Murphy et al 1999).

In (Bolt and van der Gaag 2004), we studied the performance of the loopy-propagation from a theoretical point of view and argued that two types of error may arise in the approximate probabilities yielded by the algorithm: the *cycling error* and the *convergence error.* A cycling error arises when messages are being passed on within a loop repetitively and old information is mistaken for new by the variables involved. A convergence error arises when messages that originate from dependent variables are combined as if they were independent.

Many other researchers have addressed the performance of the loopy-propagation algorithm. Weiss and his co-workers, more specifically, investigated its performance by studying the application of an equivalent algorithm on pairwise Markov networks (Weiss 2000, and Weiss and Freeman 2001). Their use of Markov networks for this purpose was motivated by the relatively easier analysis of these networks and justified by the observation that any Bayesian network can be converted into an equivalent pairwise Markov network. Weiss (2000) derived an analytical relationship between the exact and the computed probabilities for the loop nodes in a network including a single loop. In the analysis of loopy propagation in Markov networks, however, no distinction between different error types was made, and on first sight there is no equivalent for the convergence error. In this paper we investigate this difference in results; we do so by constructing the simplest situation in which a convergence error may occur, and analysing the equivalent Markov network. We find that the convergence error in the Bayesian Markov network is converted to a cycling error in the equivalent Markov network. Furthermore, we find that the prior convergence error in Markov networks is characterised by the fact that the relationship between the exact and the approximate probabilities, as established by Weiss, cannot be derived for the loop node in which this error occurs.

2 Bayesian Networks

A *Bayesian network* is a model of a joint probability distribution Pr over a set of stochastic variables $\mathbf{V}$, consisting of a directed acyclic graph and a set of conditional probability distributions[1]. Each variable A is represented by a node A in the network's digraph[2]. (Conditional) independency between the variables is captured by the digraph's set of arcs according to the d-separation criterion (Pearl 1988). The strength of the probabilistic relationships between the variables is captured by the conditional probability distributions $\Pr(A \mid \mathbf{p}(\mathbf{A}))$, where $\mathbf{p}(\mathbf{A})$ denotes the instantiations of the parents of A. The joint probability distribution is presented by

$$\Pr(\mathbf{V}) = \prod_{A \in \mathbf{V}} \Pr(A \mid \mathbf{p}(\mathbf{A}))$$

For the scope of this paper we assume all variables of a Bayesian network to be binary. We will often write a for $A = a_1$ and $\bar{a}$ for $A = a_2$. Fig. 1 depicts a small binary Bayesian network.

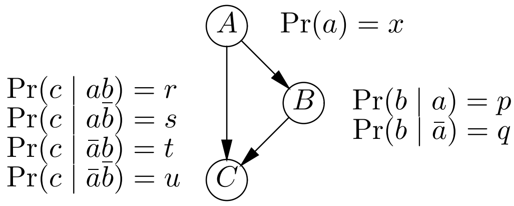

Figure 1: An example Bayesian network.

A multiply connected network includes one or more loops. We say that a loop is simple if none of its nodes are shared by another loop. A node that has two or more incoming arcs on a loop will be called a *convergence node* of this loop. Node C is the only convergence node in the network from Fig. 1. Pearl's propagation algorithm (Pearl 1988) was designed for exact inference with singly connected Bayesian networks. The term *loopy propagation* used throughout the literature, refers to the application of this algorithm to networks with loops.

3 The Convergence Error in Bayesian Networks

When applied to a singly connected Bayesian network, Pearl's propagation algorithm results in exact probabilities. When applied to a multiply connected network, however, the computed probabilities may include errors. In previous work we distinguished between two different types of error (Bolt and van der Gaag 2004).

The first type of error arises when messages are being passed on in a loop repetitively and old information is mistaken for new by the variables involved. The error that thus arises will be termed a *cycling error*. A cycling error can only occur if for each convergence node of a loop either the node itself or one of its descendents is observed. The second type of error originates from the combination of causal messages by the

[1]Variables are denoted by upper-case letters (A), and their values by indexed lower-case letters (a_i); sets of variables by bold-face upper-case letters ($\mathbf{A}$) and their instantiations by bold-face lower-case letters ($\mathbf{a}$). The upper-case letter is also used to indicate the whole range of values of a variable or a set of variables.

[2]The terms node and variable will be used interchangeably.

convergence node of a loop. A convergence node combines the messages from its parents as if the parents are independent. They may be dependent, however, and by assuming independence, a *convergence error* may be introduced. A convergence error may already emerge in a network in its prior state. In the sequel we will denote the probabilities that result upon loopy propagation with $\widetilde{\Pr}$ to distinguish them from the exact probabilities which are denoted by Pr.

Moreover, we studied the prior convergence error. Below, we apply our analysis to the example network from Figure 1. For the network in its prior state, the loopy-propagation algorithm establishes

$$\widetilde{\Pr}(c) = \sum_{A,B} \Pr(c \mid AB) \cdot \Pr(A) \cdot \Pr(B)$$

as probability for node C. Nodes A and B, however, may be dependent and the exact probability $\Pr(c)$ equals

$$\Pr(c) = \sum_{A,B} \Pr(c \mid AB) \cdot \Pr(B \mid A) \cdot \Pr(A)$$

The difference between the exact and approximate probabilities is

$$\Pr(c) - \widetilde{\Pr}(c) = x \cdot y \cdot z$$

where

$$\begin{aligned} x &= \Pr(c \mid ab) - \Pr(c \mid a\bar{b}) - \Pr(c \mid \bar{a}b) + \Pr(c \mid \bar{a}\bar{b}) \\ y &= \Pr(b \mid a) - \Pr(b \mid \bar{a}) \\ z &= \Pr(a) - \Pr(a)^2 \end{aligned}$$

The factors that govern the size of the prior convergence error in the network from Figure 1, are illustrated in Figure 2; for the construction of this figure we used the following probabilities: $r = 1$, $s = 0$, $t = 0$, $u = 1$, $p = 0.4$, $q = 0.1$ and $x = 0.5$. The line segment captures the exact probability $\Pr(c)$ as a function of $\Pr(a)$; note that each specific $\Pr(a)$ corresponds with a specific $\Pr(b)$. The surface captures $\widetilde{\Pr}(c)$ as a function of $\Pr(a)$ and $\Pr(b)$. The convergence error equals the distance between the point on the line segment that matches the probability $\Pr(a)$ from the network and its orthogonal projection on the surface. For $\Pr(a) = 0.5$, more specifically, the difference between $\Pr(c)$ and $\widetilde{\Pr}(c)$ is indicated by the vertical dotted line segment and equals $0.65 - 0.5 = 0.15$. Informally speaking:

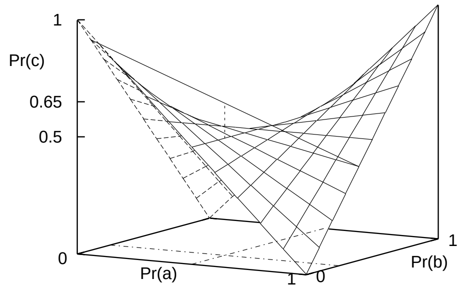

Figure 2: The probability of c as a function of $\Pr(a)$ and $\Pr(b)$, assuming independence of the parents A and B of C (surface), and as a function of $\Pr(a)$ (line segment).

- the more curved the surface is, the larger the distance between a point on the line segment and its projection on the surface can be; the curvature of the surface is reflected by the factor x;
- the distance between a point on the segment and its projection on the surface depends on the orientation of the line segment; the orientation of the line segment is reflected by the factor y;
- the distance between a point on the line segment and its projection on the surface depends its position on the line segment; this position is reflected by the factor z.

We recall that the convergence error originates from combining messages from dependent nodes as if they were independent. The factors y and z now in essence capture the degree of dependence between the nodes A and B; the factor x indicates to which extent this dependence can affect the computed probabilities.

4 Markov Networks

Like a Bayesian network, a Markov network uniquely defines a joint probability distribution over a set of statistical variables $\mathbf{V}$. The variables are represented by the nodes of an undirected graph and (conditional) independency between the variables is captured by the graph's set of edges; a variable is (conditionally) independent of every other variable given its Markov blanket. The strength of the probabilistic relationships is captured by clique potentials. Cliques C are subsets of nodes that are completely connected; $\cup C = \mathbf{V}$. For each clique, a potential function $\psi_C(\mathbf{A}_C)$ is given that assigns a non-negative real number to each configuration of the nodes $\mathbf{A}$ of C. The joint probability is presented by:

$$\Pr(\mathbf{V}) = 1/Z \cdot \prod_C \psi_C(\mathbf{A}_C)$$

where $Z = \sum_{\mathbf{V}} \prod_C \psi_C(\mathbf{A}_C)$ is a normalising factor, ensuring that $\sum_{\mathbf{V}} \Pr(\mathbf{V}) = 1$. A pairwise Markov network is a Markov network with cliques of maximal two nodes.

For pairwise Markov networks, an algorithm can be specified that is functionally equivalent to Pearl's propagation algorithm (Weiss 2000). In this algorithm, in each time step, every node sends a probability vector to each of its neighbours. The probability distribution of a node is obtained by combining the steady state values of the messages from its neighbours.

In a pairwise Markov network, the transition matrices M^{AB} and M^{BA} can be associated with any edge between nodes A and B.

$$M^{AB}_{ji} = \psi(A = a_i, B = b_j)$$

Note that matrix M^{BA} equals M^{AB^T}.

Example 1 Suppose we have a Markov network with two binary nodes A and B, and suppose that for this network the potentials $\psi(ab) = p$, $\psi(a\bar{b}) = q$, $\psi(\bar{a}b) = r$ and $\psi(\bar{a}\bar{b}) = s$ are specified, as in Fig. 3 . We then associate the transition matrix $M^{AB} = \begin{bmatrix} p & r \\ q & s \end{bmatrix}$ with the link from A to B and its transpose $M^{BA} = \begin{bmatrix} p & q \\ r & s \end{bmatrix}$ with the link from B to A. □

$\psi(ab) = p$
$\psi(a\bar{b}) = q$
$\psi(\bar{a}b) = r$
$\psi(\bar{a}\bar{b}) = s$

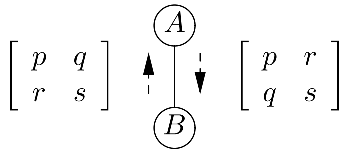

Figure 3: An example pairwise Markov network and its transition matrices.

The propagation algorithm now is defined as follows. The message from A to B equals $M^{AB} \cdot v$ after normalisation, where v is the vector that results from the component wise multiplication of all message vectors sent to A except for the message vector sent by B. The procedure is initialised with all message vectors set to (1,1,...,1). Observed nodes do not receive messages and they always transmit a vector with 1 for the observed value and zero for all other values. The probability distribution for a node, is obtained by combining all incoming messages, again by component wise multiplication and normalisation.

5 Converting a Bayesian Network into a Pairwise Markov Network

In this section, the conversion of a Bayesian network into an equivalent pairwise Markov network is described (Weiss 2000). In the conversion of the Bayesian network into a Markov network, for any node with multiple parents, an auxiliary node is constructed into which the common parents are clustered. This auxiliary node is connected to the child and its parents and the original arcs between child and parents are removed. Furthermore, all arc directions in the network are dropped. The clusters are all pairs of connected nodes. For a cluster with an auxiliary node and a former parent node, the potential is set to 1 if the nodes have a similar value for the former parent node and to 0 otherwise. For the other clusters, the potentials are equal to the conditional probabilities of the former child given the former parent. Furthermore, the prior probability of a former root node is incorporated by multiplication into one of the potentials of the clusters in which it takes part.

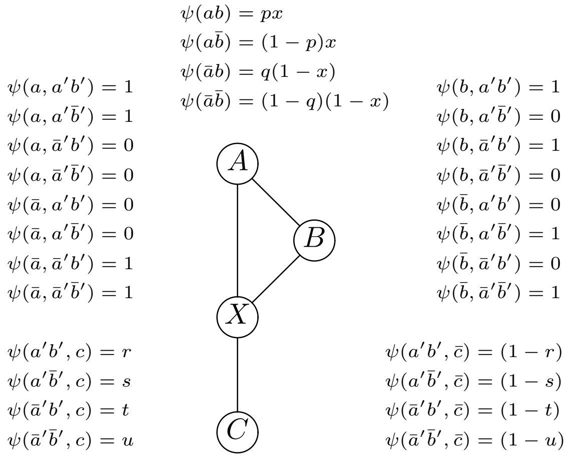

Figure 4: A pairwise Markov network that represents the same joint probability distribution as the Bayesian network from Figure 1.

Example 2 The Bayesian network from Figure 1 can be converted into the pairwise Markov network from Figure 4 with clusters AB, AX, BX and XC. Node X is composed of A' and B' and has the values $a'b'$, $a'\bar{b}'$, $\bar{a}'b'$ and $\bar{a}'\bar{b}'$. Given that the prior probability of root node A is incorporated in the potential of cluster AB, the network has the following potentials: $\psi(AB) = \Pr(B \mid A) \cdot \Pr(A)$; $\psi(XC) = \Pr(C \mid AB)$: $\psi(AX) = 1$ if $A' = A$ and 0 otherwise and; $\psi(BX) = 1$ if $B' = B$ and 0 otherwise.

6 The Analysis of Loopy Propagation in Markov Networks

Weiss (2000) analysed the performance of the loopy-propagation algorithm for Markov networks with a single loop and related the approximate probabilities found for the nodes in the loop to their exact probabilities. He noted that in the application of the algorithm messages will cycle in the loop and errors will emerge as a result of the double counting of information. The main idea of his analysis is that for a node in the loop, two reflexive matrices can be derived; one for the messages cycling clockwise and one for the messages cycling counterclockwise. The probability distribution computed by the loopy-propagation algorithm for the loop node in the steady state, now can be inferred from the principal eigenvectors of the reflexive matrices plus the other incoming vectors. Subsequently, he showed that the reflexive matrices also include the exact probability distribution and used those two observations to derive an analytical relationship between the approximated and the exact probabilities.

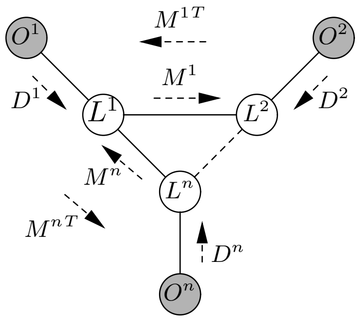

Figure 5: An example Markov network with just one loop.

More in detail, Weiss considered a Markov network with a single loop with n nodes $L^1 ... L^n$ and with connected to each node in the loop, an observed node $O^1 ... O^n$ as shown in Figure 5. During propagation, a node O^i will constantly send the same message into the loop. This vector is one of the columns of the transition matrix $M^{O^i L^i}$ In order to enable the incorporation of this message into the reflexive matrices, this vector is transformed into a diagonal matrix D^i, with the vector elements on the diagonal. For example, suppose that $M^{O^i L^i} = \begin{bmatrix} p & r \\ q & s \end{bmatrix}$ and suppose that the observation $O^i = o^i_1$ is made, then $D^i = \begin{bmatrix} p & 0 \\ 0 & q \end{bmatrix}$. Furthermore, M^1 is the transition matrix for the message from L^1 to L^2 and M^{1^T} the transition matrix for the message from L^2 to L^1 etc. The reflexive matrix C for the transition of a counterclockwise message from node L^1 back to itself is defined as $M^{1^T} D^2 ... M^{n-1^T} D^n M^{n^T} D^1$. The message that L^2 sends to L^1 in the steady state now is in the direction of the principal eigenvector of C. The reflexive matrix C^2 for the transition of a clockwise message from node L^1 back to itself is defined as $M^n D^n M^{n-1} D^{n-1} ... M^1 D^1$. The message that node L^n sends to L^1 in the steady state is in the direction of the principal eigenvector of C^2. Component wise multiplication of the two principal eigenvectors and the message from O^1 to L^1, and normalisation of the resulting vector, yields a vector of which the components equal the approximated values for L^1 in the steady state. Furthermore, Weiss proved that the elements on the diagonals of the reflexive matrices equal the correct probabilities of the relevant value of L_1 and the evidence, for example, $C_{1,1}$ equals $\Pr(l_1^1, \mathbf{o})$. Subsequently, he related the exact probabilities for a node A in the loop to its approximate probabilities by

$$\Pr(a_i) = \frac{\lambda_1 \widetilde{\Pr}(a_i) + \sum_{j=2} P_{ij} \lambda_j P_{ji}^{-1}}{\sum_j \lambda_j} \tag{1}$$

in which P is a matrix that is composed of the eigenvectors of C, with the principal eigenvector in the first column, and $\lambda_1 ... \lambda_j$ are the eigenvalues of the reflexive matrices, with λ_1 the maximum eigenvalue. We note that from this formula it follows that correct probabilities will be found if λ_1 equals 1 and all other eigenvalues equal 0.

In the above analysis, all nodes O^i are considered to be observed. Note that given unobserved nodes outside the loop, the analysis is essentially the same. In that case a transition matrix $M^{O^iL^i} = \begin{bmatrix} p & r \\ q & s \end{bmatrix}$ will result in the diagonal matrix $D^i = \begin{bmatrix} p+r & 0 \\ 0 & q+s \end{bmatrix}$.

7 The Convergence Error in Markov Networks

As discussed in Section 3 in Bayesian networks, a distinction could be made between the cycling error and the convergence error. In the previous section it appeared that for Markov network such a distinction does not exist. All errors result from the cycling of information and, on first sight, there is no equivalent for the convergence error. However, any Bayesian network can be converted into an equivalent pairwise Markov network on which an algorithm equivalent to the loopy-propagation algorithm can be used. In this section, we investigate this apparent incompatibility of results and indicate how the convergence error yet is embedded in the analysis of loopy propagation in Markov networks. We do so by constructing the simplest situation in which a convergence error may occur, that is, the Bayesian network from Figure 1 in its prior state, and analysing this situation in the equivalent Markov network. The focus thereby is on the node that replaces the convergence node in the loop. We then argue that the results have a more general validity.

Consider the Bayesian network from Figure 1. In its prior state, there is no cycling of information, and exact probabilities will be found for nodes A and B. In node C, however, a convergence error may emerge. The network can be converted into the pairwise Markov network from Figure 4. For this network we find the transition matrices $M^{XA} = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}$, $M^{XB} = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix}$, $M^{XC} = \begin{bmatrix} r & s & t & u \\ 1-r & 1-s & 1-t & 1-u \end{bmatrix}$, $M^{AB} = \begin{bmatrix} px & q(1-x) \\ (1-p)x & (1-p)(1-x) \end{bmatrix}$ and their transposes. In the prior state of the network, C will send the message $M^{CX} \cdot (1,1) = (1,1,1,1)$ to X. In order to enable the incorporation of this message into the reflexive loop matrices it is transformed into D^{CX} which, in this case, is the 4x4-identity matrix.

We first evaluate the performance of the loopy propagation algorithm for the regular loop node A. This node has the following reflexive matrices for its clockwise and counterclockwise messages respectively:

$$M^{\circlearrowright A} = M^{XA} \cdot D^{CX} \cdot M^{BX} \cdot M^{AB} = \begin{bmatrix} x & 1-x \\ x & 1-x \end{bmatrix}$$

with eigenvalues 1 and 0 and principal eigenvector (1,1) and

$$M^{\circlearrowleft A} = M^{BA} \cdot M^{XB} \cdot D^{CX} \cdot M^{AX} = \begin{bmatrix} x & x \\ 1-x & 1-x \end{bmatrix}$$

with eigenvalues 1 and 0 and principal eigenvector $(x, 1-x)$. Note that the correct probabilities for node A indeed are found on the diagonal of the reflexive matrices. Furthermore, $\lambda_1 = 1$ and $\lambda_2 = 0$ and therefore correct approximations are expected. We indeed find that the approximations $(1 \cdot x, 1 \cdot (1-x))$ equal the exact probabilities. Note also that, as expected, the messages from node A back to itself do not change any more after the first cycle. As in the Bayesian network, for node A no cycling of information occurs in the Markov network. For node B a similar evaluation can be made.

We now turn to the convergence node C. In the Bayesian network in its prior state a convergence error may emerge in this node. In the conversion of the Bayesian network into the Markov network, the convergence node C is placed outside the loop and the auxiliary node X is added. For X, the following reflexive matrices are computed for the clockwise and counterclockwise messages from node X back to itself respectively:

$$M^{\circlearrowright X} = M^{BX} \cdot M^{AB} \cdot M^{XA} \cdot D^{CX} =$$
$$\begin{bmatrix} px & px & q(1-x) & q(1-x) \\ (1-p)x & (1-p)x & (1-q)(1-x) & (1-q)(1-x) \\ px & px & q(1-x) & q(1-x) \\ (1-p)x & (1-p)x & (1-q)(1-x) & (1-q)(1-x) \end{bmatrix}$$

with eigenvalues 1, 0, 0, 0; principal eigenvector $((px+q(1-x))/((1-p)x+(1-q)(1-x)), 1, (px+q(1-x))/((1-p)x+(1-q)(1-x)), 1)$ and other eigenvectors $(0,0,-1,1)$, $(-1,1,0,0)$ and $(0,0,0,0)$.

$$M^{\circlearrowleft X} = M^{AX} \cdot M^{BA} \cdot M^{XB} \cdot M^{CX} =$$
$$\begin{bmatrix} px & (1-p)x & px & (1-p)x \\ px & (1-p)x & px & (1-p)x \\ q(1-x) & (1-q)(1-x) & q(1-x) & (1-q)(1-x) \\ q(1-x) & (1-q)(1-x) & q(1-x) & (1-q)(1-x) \end{bmatrix}$$

with eigenvalues 1, 0, 0, 0; principal eigenvector $(x/(1-x), x/(1-x), 1, 1)$ and other eigenvectors $(0,-1,0,1)$, $(-1,0,1,0)$ and $(0,0,0,0)$.

On the diagonal of the reflexive matrices of X we find the probabilities $\Pr(AB)$. As the correct probabilities for a loop node are found on the diagonal of its reflexive matrices, these probabilities can be considered to be the exact probabilities for node X. The normalised vector of the component wise multiplication of the principal eigenvectors of the two reflexive matrices of X equals the vector with the normalised probabilities $\Pr(A) \cdot \Pr(B)$. Likewise, these probabilities can be considered to be the approximate probabilities for node X.

A first observation is that λ_1 equals 1 and the other eigenvalues equal 0, but the exact and the approximate probabilities of node X may differ. This is not consistent with Equation 1. The explanation is that for node X, the matrix P, is singular and therefore, the matrix P^{-1}, which is needed in the derivation of the relationship between the exact and approximate probabilities, does not exist. Equation 1, thus isn't valid for the auxiliary node X. We note furthermore that the messages from node X back to itself may still change after the first cycle. We therefore find that, although in the Bayesian network there is no cycling of information, in the Markov network, for node X information may cycle, resulting in errors computed for its probabilities.

The probabilities computed by the loopy-propagation algorithm for node C equal the normalised product $M^{XC} \cdot v$, where v is the vector with the approximate probabilities found at node X. It can easily be seen that these approximate probabilities equal the approximate probabilities found in the equivalent Bayesian network. Furthermore we observe that if node X would send its exact probabilities, that is, $\Pr(AB)$, exact probabilities for node C would be computed. In the Markov network we thus may consider the convergence error to be founded in the cycling of information for the auxiliary node X.

In Section 3, a formula for the size of the prior convergence error in the network from figure 1 is given. We there argued that this size is determined by the factors y and z that capture the degree of dependency between the parents of the convergence node and the factor x, that indicates to which extent the dependence between nodes A and B can affect the computed probabilities. In this formula, x is composed of the conditional probabilities of node C. In the analysis in the Markov network we have a similar finding. The effect of the degree of dependence between A and B is reflected in the difference between the exact and the approximate probabilities found for node X. The effect of the conditional probabilities at node C emerges in the transition of the message vector from X to C.

We just considered the small example network from Figure 1. Note, however, that for any prior binary Bayesian networks with just simple loops, the situation for any loop can be 'summarised' to the situation in Figure 1 by marginalisation over the relevant variables. The results with respect to the manifestation of the convergence error by the cycling of information and the invalidity of Equation 1 for the auxiliary node, found for the network from Figure 1, therefore, apply to any prior binary Bayesian networks with just simple loops.[3][4]

8 Discussion

Loopy propagation refers to the application of Pearl's propagation algorithm for exact reasoning with singly connected Bayesian networks to networks with loops. In previous research we identified two different types of error that may arise in the probabilities computed by the algorithm. Cycling errors result from the cycling of information and arise in loop nodes as soon as for each convergence node of the loop, either the node itself, or one of its descendents is observed. Convergence errors result from combining information from dependent nodes as if they were independent and may arise at convergence nodes. This second error type is found both in a network's prior and posterior state. Loopy propagation has also been studied by the analysis of the performance of an equivalent algorithm in pairwise Markov networks with just a simple loop. According to this analysis all errors result from the cyling of information and on first sight there is no equivalent for the convergence error. We investigated how the convergence error yet is embedded in the analysis of loopy propagation in Markov networks. We did so by constructing the simplest situation in which a convergence error may occur, and analysing this situation in the equivalent Markov network. We found that the convergence error in the Bayesian network is converted to a cycling error in the equivalent Markov network. Furthermore, we found that the prior convergence error is characterised by the fact that the relationship between the exact probabilities and the approximate probabilities yielded by loopy propagation, as established by Weiss, can not be derived for the loop node in which this error occurs. We then argued that these results are valid for binary Bayesian network with just simple loops in general.

[3]Given a loop with multiple convergence nodes, in the prior state of the network, the parents of the convergence nodes are independent and effectively no loop is present.

[4]Two loops in sequence may result in incorrect probabilities entering the second loop. The reflexive matrices, however, will have a similar structure as the reflexive matrices derived in this section.

Acknowledgements

This research was partly supported by the Netherlands Organisation for Scientific Research (NWO).

References

J.H. Bolt, L.C. van der Gaag. 2004. The convergence error in loopy propagation. Paper presented at *the International Conference on Advances in Intelligent Systems: Theory and Applications.*

G.F. Cooper. 1990. The computational complexity of probabilistic inference using Bayesian belief networks. *Artificial Intelligence*, 42:393–405.

P. Dagum, M. Luby. 1993. Approximate inference in Bayesian networks is NP hard. *Artificial Intelligence*, 60:141–153.

K. Murphy, Y. Weiss, M. Jordan. 1999. Loopy belief propagation for approximate inference: an empirical study. In *15th Conference on Uncertainty in Artificial Intelligence*, pages 467–475.

J. Pearl. 1988. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference.* Morgan Kaufmann Publishers, Palo Alto.

Y. Weiss. 2000. Correctness of local probability propagation in graphical models with loops. *Neural Computation*, 12:1–41.

Y. Weiss, W.T. Freeman. 2001. Correctness of belief propagation in Gaussian graphical models of arbitrary topology. *Neural Computation*, 13:2173–2200.