Sequential Updating Conditional Probability in Bayesian Networks by Posterior Probability

Yang Xiang*, Michael P. Beddoes* and David Poole†

University of British Columbia, Vancouver, B.C., Canada, V6T 1W5

* Department of Electrical Engineering, yangx@ee.ubc.ca

† Department of Computer Science, poole@cs.ubc.ca

Abstract

The Bayesian network is a powerful knowledge representation formalism; it is also capable of improving its precision through experience. Spiegelhalter *et al.* [1989] proposed a procedure for sequential updating forward conditional probabilities (FCP) in Bayesian networks of diameter 1 with a single parent node. The procedure assumes certainty for each diagnosis which is not practical for many applications. In this paper we present a new algorithm (ALPP) that allows refinement of FCPs based on expert estimates of posterior probability. ALPP applies to any DAG of diameter 1. Fast convergence is achieved. Simulation results compare ALPP with Spiegelhalter's method.

1 Introduction

Much recent research is dedicated to Bayesian belief networks as an inference formalism building expert systems [Pearl 88] [Lauritzen and Spiegelhalter 88] [Heckerman *et al.* 89] [Andersen *et al.* 89]. A Bayesian network is a pair (D, P). D is a directed acyclic graph (DAG) in which the nodes represent generally uncertain variables, and the arcs signify the existence of direct causal influences between the linked variables. P is a probability distribution which quantifies the strengths of these causal influences. P is distributively stored in the network, in the form of FCPs [Pearl 88].

A knowledge based system QUALICON is currently under development, based on Bayesian networks, which can be used in assisting an E.M.G. technician in test quality control during conduction velocity studies [Xiang *et al.* 90]. The system takes qualitative features of recorded compound muscle action potentials as evidences and tries to diagnose the problems in electrode set-up.

Building the system involves two parts. The first is generating the DAG D. This task is easy and natural for the experienced medical staff. The second phase, namely the elicitation of many FCPs, they found much more difficult (and the results were quite imprecise) because, among other causes, the task seems artificial to them. They claim that it would be easier for them to supply a posterior probability (PP) distribution for the possible hypotheses in particular cases. What they give in that case would be more precise since the task is closer to their daily practice. A methodology allowing the system to improve itself through expert's PPs is badly needed.

Spiegelhalter *et al.* [1989] present a procedure for sequential updating conditional probabilities in Bayesian networks decomposed into DAGs of diameter one with a single parent node. The procedure consists of two stages. In the first stage, the FCPs for each link are elicited from the expert. The expert is asked to estimate these probabilities in form of intervals to express the imprecision of the estimation. Then each interval probability is interpreted as an imaginary sample ratio: $p(symptomA|diseaseB) = a/b$, where b is an imaginary patient population with disease B and among these patients a of them show symptom A. In the second stage, *updating* stage, whenever a new patient with disease B comes in, the corresponding sample size b is increased by 1, and the sample size a is increased either by 1 or 0 depending on if the patient shows symptom A. This *updating* stage is the main concern of this paper.

A major problem of this updating approach is the underlying assumption that when updating the link FCP $p(symptomA|diseaseB)$, the system user knows for sure whether the disease B is true or false (thus we will call the procedure $\{0, 1\}$ *distribution learning*). The assumption is not realistic in many applications. A doctor would not always be 100% confident about a diagnosis he made of a patient.

In this paper, a Algorithm of Learning by Poste-

rior Probability (ALPP) is presented which is more general than the $\{0,1\}$ distribution learning. ALPP applies to any DAG with diameter one. The DAG can itself be the whole network or be a subnet of a more sophisticated network as long as the following *DAG-completeness* condition holds: it contains all the incoming links to its child nodes as in the original net. ALPP does not assume 100% accurate posterior judgment. Instead, it utilizes the PPs of each fresh case supplied by the expert to update the FCPs of the network. We show the algorithm converges to the expert's behavior under ideal condition. When ALPP does not converge to the human consultant's posterior judgments, it is an indication of either inadequate network structure or inadequate PPs. The algorithm converges quicker than the $\{0,1\}$ distribution learning equipped with *100% accurate* posterior judgment.

The philosophy which guided this work is described in section 2. Section 3 presents ALPP and section 4 proves its convergence. The performance of the ALPP is demonstrated by simulations given in section 5.

2 Learning from Posterior Distributions

The spirit of $\{0,1\}$ distribution learning is to improve the precision of probability elicited from the human expert by learning from available data. Now the question is what do we really have in medical practice in addition to patients' symptoms? It may be possible, in some medical domain, that diagnoses can be confirmed with certainty. But this is not commonplace. A successful treatment is not always an indication of correct diagnosis. A disease can be cured by a patient's internal immunity or by a drug with wide disease spectrum. One subtlety of medical diagnosis comes from the unconfirmability for each individual patient case.

For most medical domains, the available data beside patients' symptoms are physician's subjective PPs of possible diseases. They are not distributions with values from $\{0,1\}$, but rather distributions from $[0,1]$[1]. The diagnoses appearing in patients' files are typically not the diagnoses that have been concluded definitely; they are only the top ranking diseases with physician's subjective PP omitted. The assumption of $\{0,1\}$ posterior disease distribution may, naively, be interpreted as an approximation to $[0,1]$ distribution with 1 substituting top ranking PP, and 0 substituting the rest. This approximation loses useful information. Thus a way of learning directly from $[0,1]$ posterior distribution seems more natural and anticipates better performance.

In dealing with *learning* problem in a Bayesian network setting, three "agents" are concerned: the real world (D_r, P_r), the human expert (D_e, P_e), and our artificial system (D_s, P_s). It is assumed that all 3 can be modeled by Bayesian networks. As the building of an expert system involves specifying both the topology of D and probability distribution P, the improvement can also be separated into the two aspects. For the purpose of this paper, D_r, D_e, and D_s are assumed identical, leaving to be improved only the accuracy of quantitative assignment of P_s.

An expert system based on Bayesian networks usually directs its arcs from disease (hypothesis) nodes to symptom (evidence) nodes, encoding quantitative knowledge with priors of diseases and FCPs of symptoms given diseases [Shachter and Heckerman 87, Henrion 88]. These probabilities are usually elicited from human experts.

A question which arises is whether PP is any better in quality compared to priors and FCPs also supplied by the human expert. In our cooperation with medical staff, it is found that the causal network is a natural model to view the domain, however, the task of estimating FCPs is more artificial than natural to them. Forming posterior judgments is their daily practice. An expert is an expert in that he/she is skilled at making diagnosis (posterior judgement), not necessarily skilled at estimating FCPs. It is the expert's posterior judgment that is the behavior we want our expert system to simulate[2].

If we believe that the human expert carries a mental Bayesian network and PPs are produced by the network, it is postulated that the FCPs the expert articulates, which consists of P_s of our system, could be a distorted version of those in P_e. Also, P_e may differ from P_r in general. Thus, 4 categories of probabilities are distinguished: P_r, P_e, P_s, and the PPs produced by P_e (written as p_e). Our access to only P_s and $p_e(hypotheses|evidence)$ is assumed. We want to use the latter to improve P_s such that the system's behavior will approach that of expert.

How can PP be utilized in our updating? The basic idea is: instead of updating imaginary sample sizes by 1 or 0, increase them by the measure of certainty of the corresponding diseases. The expert's PP is just such a measure. Formal treatment is given in the following

[1] Note that $\{0,1\}$ denotes a set containing only elements 0 and 1, and $[0,1]$ is a domain of real numbers between 0 and 1 inclusive.

[2] We are not arguing against the usual way of encoding numerical knowledge from diseases to symptoms. The advantages of it, like simplicity in network structure, clarity of underlying causal dependency, etc. are well known.

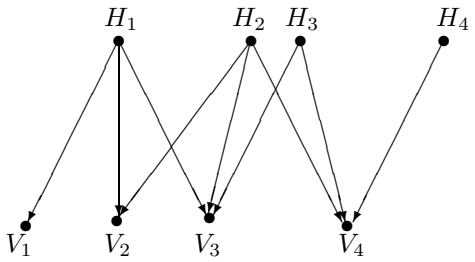

Figure 1: An example of D(1)

section.

3 The Algorithm for Learning by Posterior Probability (ALPP)

The following notation is used:

$D(1)$ DAGs of diameter 1 (The *diameter* is the length of the longest directed path in the DAG. An example of D(1) is given in Figure 1.);

$(D(1), P)$ Bayesian net with diameter 1 and underlying distribution P;

$H_i \in \{h_{i1}, \ldots, h_{in_i}\}$ the ith parent variable in $D(1)$ with possible values h_{i1} through h_{in_i};

$V_j \in \{v_{j1}, \ldots, v_{jm_j}\}$ the jth child variable in $D(1)$ with possible values v_{j1} through v_{jm_j};

$\mathbf{v}_{jl}$ value conjunction of all the children variables in $D(1)$ with V_j's value being v_{jl};

$b_{k_1 k_2 \ldots k_n}$ the imaginary sample size for joint event $h_{1k_1} \& h_{2k_2} \& \ldots \& h_{nk_n}$ being true;

$a_{l_j k_1 k_2 \ldots k_n}$ the imaginary sample size for joint event $v_{jl_j} \& h_{1k_1} \& \ldots \& h_{nk_n}$ being true;

$\delta^c_{l_j}$ impulse function which equals 1 if for the cth fresh case V_j equals v_{jl_j}, and equals 0 otherwise (superscripts denote the orders of fresh cases);

$p_r(), p_e(), p_s()$ probabilities contained or generated by $(D_r(1), P_r)$, $(D_e(1), P_e)$ and $(D_s(1), P_s)$ respectively.

A Bayesian net $(D(1), P)$ [3] is considered where the underlying distribution is composed via

$$p(h_{1k_1} \& \ldots \& h_{Nk_N} \& v_{1l_1} \& \ldots \& v_{Ml_M}) = \prod_{i=1}^{N} p(h_{ik_i}) \prod_{j=1}^{M} p(v_{jl_j} | \widehat{\mathbf{h}}_j)$$

[3] Whether it is a subnet or a net by itself is irrelevant.

where $\widehat{\mathbf{h}}_j$ is the conjunction of those values h_{ik_i} such that H_i is a parent variable of V_j and $h_{ik_i} \in \{h_{1k_1}, \ldots, h_{Nk_N}\}$.

Each of the FCPs is internally represented in the system as a ratio of 2 imaginary sample sizes. For child node V_1 having its parent nodes H_1, ..., H_Q $(Q \geq 1)$, a corresponding FCP is

$$p^c_s(v_{1l_1} | h_{1k_1} \& \ldots \& h_{Qk_Q}) = a^c_{l_1 k_1 \ldots k_Q} / b^c_{k_1 \ldots k_Q}$$

where the superscript c signifies the cth updating. Only the real numbers $a^c_{l_1 k_1 \ldots k_Q}$ and $b^c_{k_1 \ldots k_Q}$ are stored. The prior probabilities for V_1's parents can be derived as

$$p^c_s(h_{ik_i}) = \frac{\sum_{k_1, \ldots, k_{i-1}, k_{i+1}, \ldots, k_Q} b^c_{k_1 \ldots k_Q}}{\sum_{k_1, \ldots, k_Q} b^c_{k_1 \ldots k_Q}}$$

For a $(D(1), P)$ with M children and with all variables binary, the number of numbers to be stored in this way is upper bounded by

$$B = 2 \sum_{i=1}^{M} 2^{\beta_i}$$

where β_i is the number of *incoming* arcs to child node i. Storage saving can be achieved when different child nodes share a common set of parents.

Updating P is done one child node at a time through updating as and bs associated with the node as illustrated above. Once the as and bs are updated, the updated FCPs and priors can be derived. The order in which child nodes are selected for updating is irrelevant.

Without losing generality, we describe the updating with respect to above mentioned child node V_1. For the cth fresh case where $\mathbf{v}^c$ is the symptoms observed, the expert provides the PP distribution $p_e(h_{1k_1} \& \ldots \& h_{Nk_N} | \mathbf{v}^c)$. This is transformed into

$$p_e(h_{1k_1} \& \ldots \& h_{Qk_Q} | \mathbf{v}^c) = \sum_{h_{Q+1}, \ldots, h_N} p_e(h_{1k_1} \& \ldots \& h_{Nk_N} | \mathbf{v}^c)$$

The sample sizes are updated by

$$a^c_{l_1 k_1 \ldots k_Q} = a^{c-1}_{l_1 k_1 \ldots k_Q} + \delta^c_{l_1} p_e(h_{1k_1} \& \ldots \& h_{Qk_Q} | \mathbf{v}^c)$$

$$b^c_{k_1 \ldots k_Q} = b^{c-1}_{k_1 \ldots k_Q} + p_e(h_{1k_1} \& \ldots \& h_{Qk_Q} | \mathbf{v}^c).$$

4 Convergence of the algorithm

An expert is called *perfect* if $(D_e(1), P_e)$ is identical to $(D_r(1), P_r)$.

Without losing generality, consider the updating with respect to V_1 described in last section.
(1) Priors. Let $\{\mathbf{v}(1), \mathbf{v}(2), \ldots\}$ be the set of all possible conjuncts of evidence. Let $u(t)$ be the number of times at which event $\mathbf{v}(t)$ is true in c cases; and $\sum_t u(t) = c$. From the prior updating formula of ALPP,

$$\begin{aligned}
\lim_{c\to\infty} p_s^c(h_{ik_i}) &= \lim_{c\to\infty}\left(\frac{\sum_{k_1,\ldots,k_{i-1},k_{i+1},\ldots,k_Q} b^0_{k_1\ldots k_Q} + \sum_{k_1,\ldots,k_{i-1},k_{i+1},\ldots,k_Q}\sum_{x=1}^{c} p_e(h_{1k_1}\&\ldots\&h_{Qk_Q}|\mathbf{v}^x)}{c+\sum_{k_1,\ldots,k_Q} b^0_{k_1\ldots k_Q}}\right)\\
&= \lim_{c\to\infty}\frac{1}{c}\left(\sum_{k_1,\ldots,j_{k-1},j_{k+1},\ldots,k_Q}\sum_t p_e(h_{1k_1}\&\ldots\&h_{Qk_Q}|\mathbf{v}(t))u(t)\right)\\
&= \sum_{k_1,\ldots,k_{i-1},k_{i+1},\ldots,k_Q}\sum_t p_e(h_{1k_1}\&\ldots\&h_{Qk_Q}|\mathbf{v}(t))p_r(\mathbf{v}(t))\\
&= \sum_{k_1,\ldots,k_{i-1},k_{i+1},\ldots,k_Q}\sum_t p_e(h_{1k_1}\&\ldots\&h_{Qk_Q}|\mathbf{v}(t))p_e(\mathbf{v}(t)) \qquad \text{(perfect expert)}\\
&= \sum_{k_1,\ldots,k_{i-1},k_{i+1},\ldots,k_Q} p_e(h_{1k_1}\&\ldots\&h_{Qk_Q}) = p_e(h_{ik_i})
\end{aligned}$$

(2) FCPs. Let $u_{1l_1}(t)$ be the number of times at which event $\mathbf{v}_{1l_1}(t)$ is true in c cases. Following ALPP, we have

$$\begin{aligned}
\lim_{c\to\infty} p_s^c(v_{1l_1}|h_{1k_1}\&\ldots\&h_{Qk_Q}) &= \lim_{c\to\infty}\frac{a^0_{l_1k_1\ldots k_Q} + \sum_{x=1}^{c}\delta^x_{l_1} p_e(h_{1k_1}\&\ldots\&h_{Qk_Q}|\mathbf{v}^x)}{b^0_{k_1\ldots k_Q} + \sum_{y=1}^{c} p_e(h_{1k_1}\&\ldots\&h_{Qk_Q}|\mathbf{v}^y)}\\
&= \lim_{c\to\infty}\frac{\frac{1}{c}\sum_{x=1}^{c}\delta^x_{l_1} p_e(h_{1k_1}\&\ldots\&h_{Qk_Q}|\mathbf{v}^x)}{\frac{1}{c}\sum_{y=1}^{c} p_e(h_{1k_1}\&\ldots\&h_{Qk_Q}|\mathbf{v}^y)}\\
&= \frac{\lim_{c\to\infty}\frac{1}{c}\sum_t p_e(h_{1k_1}\&\ldots\&h_{Qk_Q}|\mathbf{v}_{1l_1}(t))u_{1l_1}(t)}{\lim_{c\to\infty}\frac{1}{c}\sum_z p_e(h_{1k_1}\&\ldots\&h_{Qk_Q}|\mathbf{v}(z))u(z)}\\
&= \frac{\sum_t p_e(h_{1k_1}\&\ldots\&h_{Qk_Q}|\mathbf{v}_{1l_1}(t))p_r(\mathbf{v}_{1l_1}(t))}{\sum_z p_e(h_{1k_1}\&\ldots\&h_{Qk_Q}|\mathbf{v}(z))p_r(\mathbf{v}(z))}\\
&= \frac{\sum_t p_e(h_{1k_1}\&\ldots\&h_{Qk_Q}|\mathbf{v}_{1l_1}(t))p_e(\mathbf{v}_{1l_1}(t))}{\sum_z p_e(h_{1k_1}\&\ldots\&h_{Qk_Q}|\mathbf{v}(z))p_e(\mathbf{v}(z))} \qquad \text{(perfect expert)}\\
&= \frac{p_e(h_{1k_1}\&\ldots\&h_{Qk_Q}\&v_{1l_1})}{p_e(h_{1k_1}\&\ldots\&h_{Qk_Q})} = p_e(v_{1l_1}|h_{1k_1}\&\ldots\&h_{Qk_Q})
\end{aligned}$$

Figure 2: Proof of Theorem 1

Theorem 1 *Let a Bayesian network $(D_s(1), P_s)$ be supported by a perfect expert equipped with $(D_e(1), P_e)$. No matter what initial state P_s is in, it will converge to P_e by ALPP.*

The proof is given in figure 2.

A *perfect* expert is never available. We need to know the behavior of ALPP when supported by an imperfect expert. This leads to the following theorem.

Theorem 2 *Let p_s^c be any resultant probability in $(D_s(1), P_s)$ after c updating by ALPP. p_s^c converges to a continuous function of P_e.*[4]

Proof:

(1) Continuity of priors.

Following the proof of theorem 1, the prior $p_s^c(h_{ik_i})$ converges to

$$f = \sum_{k_1,\ldots,k_{i-1},k_{i+1},\ldots,k_Q} \sum_t p_e(h_{1k_1}\&\ldots\&h_{Qk_Q}|\mathbf{v}(t))\frac{u(t)}{c}$$

where $p_e(h_{1k_1}\&\ldots\&h_{Qk_Q}|\mathbf{v}(t))$ is an elementary function of P_e, and so does f. Therefore, $p_s^c(h_{ik_i})$ converges to a continuous function of P_e.

(2) Continuity of FCP.

From theorem 1, $p_s^c(v_{1l_1}|h_{1k_1}\&\ldots\&h_{Qk_Q})$ converges to

$$f = \frac{\sum_t p_e(h_{1k_1}\&\ldots\&h_{Qk_Q}|\mathbf{v}_{1l_1}(t))\frac{u_{1l_1}(t)}{c}}{\sum_z p_e(h_{1k_1}\&\ldots\&h_{Qk_Q}|\mathbf{v}(z))\frac{u(z)}{c}}$$

where $p_e(h_{1k_1}\&\ldots\&h_{Qk_Q}|\mathbf{v}(z))$ is an elementary function of P_e.

□

Theorem 2, together with Theorem 1, says that when the discrepancy between P_e and P_r is small, the discrepancy between P_s and P_r (P_e as well) will be small after enough learning trials. The specific form of the discrepancy is left open.

The absolute value of PPs is not really important in many applications but the posterior ordering of diseases be. A set of PPs defines such a posterior ordering. We say a 100% *behavior match* between (D, P_1) and (D, P_2) if for any possible set of symptoms the two give the same ordering. The minimum difference between successive PPs of (D, P_1) defines a threshold. Unless the maximum difference between corresponding PPs from 2 (D, P)s exceeds the threshold, 100% behavior match is guaranteed. Thus as long as the discrepancy between P_e and P_r is within some $(D_r(1), P_r)$ dependent threshold, a 100% match between the behavior of P_s and that of P_e is anticipated.

[4]By 'X is a function of P_e', we mean that X takes probability variables in P_e as its independent variables which in turn themselves have [0,1] as their domain.

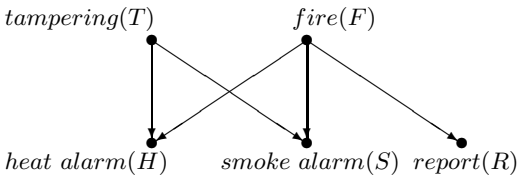

Figure 3: Fire alarm example

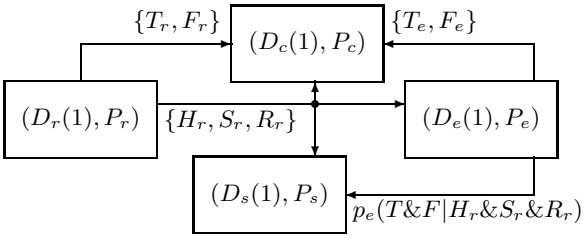

Figure 4: Simulation set-up

5 Simulation results

Several simulations were run using the example in Figure 3. It is a revised version of the smoke-alarm example in [Poole and Neufeld 88]. Here *heat alarm, smoke alarm* and *report* are used as evidences for estimating the likelihood of *tampering* and *fire*. Each variable, denoted by uppercase letters, takes binary values. For example, F has value f or $\overline{f}$ which signify the event *fire* being *true* or *false*.

The simulation set-up is illustrated in Figure 4. Logical sampling [Henrion 88] was used in the real world model $(D_r(1), P_r)$ to generate scenarios $\{T_r, F_r, H_r, S_r, R_r\}$. The observed evidences $\{H_r, S_r, R_r\}$ were feed into $(D_e(1), P_e)$. Posterior distributions $p_e(T\&F|H_r\&S_r\&R_r)$ was made by the expert model and were forwarded to update system model $(D_s(1), P_s)$.

To compare the performance between ALPP and $\{0, 1\}$ distribution learning, a Control model $(D_c(1), P_c)$ was constructed in the set-up. It had the same DAG structure and initial probability distribution as $(D_s(1), P_s)$ but was updated by $\{0, 1\}$ distribution learning.[5] Two different sets of diagnoses were utilized in different simulation runs by $(D_c(1), P_c)$ for the purpose of comparison. In simulation 1, 2 and 3 to

[5]Here we have extended $\{0, 1\}$ distribution learning to D(1).

be described below, the top diagnosis $\{T_e, F_e\}$ made by $(D_e(1), P_e)$ was used. In simulation 4, the scenario $\{T_r, F_r\}$ was used. The former simulated the situation where posterior judgments could not be fully justified. The latter simulated the case where such justification was indeed available.

For all the simulations let P_r be the following distribution

$p(h\|f\&t)$	0.50	$p(s\|f\&t)$	0.60
$p(h\|f\&\overline{t})$	0.90	$p(s\|f\&\overline{t})$	0.92
$p(h\|\overline{f}\&t)$	0.85	$p(s\|\overline{f}\&t)$	0.75
$p(h\|\overline{f}\&\overline{t})$	0.11	$p(s\|\overline{f}\&\overline{t})$	0.09
$p(r\|f)$	0.70	$p(f)$	0.25
$p(r\|\overline{f})$	0.06	$p(t)$	0.20

and let P_s and P_c be an identical distribution with maximal error relative to P_r being 0.3. The initial imaginary sample size for each joint event $F\&T$ is set to 1. Such setting is mainly for the purpose of demonstrating the convergence of ALPP under poor initial condition. The distribution error should generally be smaller and initial sample sizes be much larger in case of practical application where the convergence will be a slowly evolving process.

	$(D_e(1), P_e)$	$(D_s(1), P_s)$		$(D_c(1), P_c)$	
trial No.	diag. rate	behv. mat. rate	max. err. S-E	behv. mat. rate	max. err. C-E
0			0.30		0.30
1∼25	68%	60%	0.14	48%	0.21
26∼50	76%	96%	0.10	12%	0.25
51∼100	80%	100%	0.06	36%	0.27
101∼200	76%	100%	0.03	33%	0.28

Table 1: Simulation 1 summary

Simulation 1 was run with P_e being the same as P_r which assumed a perfect expert. The results are depicted in Table 1. The diagnostic rate of $(D_e(1), P_e)$ is defined as A/N where N is the base number of trials and A is the number of trials where the top diagnosis agrees with $\{T_r, F_r\}$ simulated by $(D_r(1), P_r)$. The behavior matching rate of $(D_s(1), P_s)$ relative to $(D_e(1), P_e)$ is defined as B/N where B is the number of trials in which $(D_s(1), P_s)$'s diagnoses give the same ordering as $(D_e(1), P_e)$'s do. The behavior matching rate of $(D_c(1), P_c)$ to $(D_e(1), P_e)$ is similarly defined.

The results show convergence of probability values in P_s to those in P_e (maximum error(S-E) $\rightarrow$ 0). The behavior matching rate of $(D_s(1), P_s)$ increases along with the convergence of probabilities and finally $(D_s(1), P_s)$ achieved exactly the same behavior as that of $(D_e(1), P_e)$. An interesting phenomenon is that, despite $P_e = P_r$, the diagnostic rate of $(D_e(1), P_e)$ was only 76% in the total 200 trials. Though the rate is dependent of the particular (D, P), it is expected to be less than 100% in general. In terms of medical diagnosis, this is because some disease may manifest through unlikely symptoms, making other diseases more likely. In an uncertain world with limited evidence, mistakes in diagnoses are unavoidable. More importantly, P_s converged to P_e under the guidance of this 76% correct diagnoses while P_c did not. The maximum error of P_c remained about the same throughout the 200 trials and the behavior matching rate of $(D_c(1), P_c)$ was low. Similar performance of $(D_c(1), P_c)$ was seen in the next 2 simulations. This shows that under the circumstances where good experts are available but confirmations to diagnoses are not available, ALPP is robust while {0,1} distribution learning will be misled by the errors in diagnoses. This is not surprising since the assumption underlying {0,1} distribution learning is violated. We will gain more insight into this from the results of simulation 4 below.

An imperfect expert was assumed in simulation 2 (Table 2). The distribution P_e differed from P_r up to 0.05. Because of this error, P_s converges to neither P_e (as shown in Table 2) nor P_r. But the error between P_s and P_e approached a small value (about 0.07) such that after 200 trials the behavior of P_s matched that of P_e perfectly.

	$(D_e(1), P_e)$	$(D_s(1), P_s)$		$(D_c(1), P_c)$	
trial No.	diag. rate	behv. mat. rate	max. err. S-E	behv. mat. rate	max. err. C-E
0			.300		.300
1∼100	84%	82%	.058	32%	.272
101∼200	86%	92%	.122	43%	.287
201∼300	80%	100%	.067	32%	.290
301∼400	83%	100%	.076	36%	.292

Table 2: Simulation 2 summary

If the discrepancy between P_s and P_r is further increased so that the threshold discussed in last section is crossed, $(D_s(1), P_s)$ will no longer converge to $(D_e(1), P_e)$. This is the case in simulation 3 (Table 3) where the maximum error and root mean square error (rms) between P_e and P_r were 0.15 and 0.098 respectively. The rms error was calculated over all the priors and conditional probabilities of P_e and P_r. We introduced rms error for interpretation of simulation 3 because maximum error itself, when not approaching to 0, did not give good indication of the distance between the two.

The simulation shows that the behavior matching

$(D_e(1), P_e)$		$(D_s(1), P_s)$				
trial No.	diag. rate	diag. rate	behv. mat. rate	rms err. S-E	rms err. S-R	max. err. S-E
0				.170	.169	.39
1∼25	80%	84%	20%	.086	.079	.17
26∼75	74%	74%	40%	.071	.068	.11
76∼175	73%	73%	53%	.050	.083	.087
176∼375	79%	79%	46%	.059	.072	.095
376∼475	78%	78%	43%	.061	.071	.119
$(D_e(1), P_e)$		$(D_c(1), P_c)$				
trial No.	diag. rate	diag. rate	behv. mat. rate	rms err. C-E	rms err. C-R	max. err. C-E
0				.170	.169	.39
1∼25	80%	80%	32%	.110	.091	.20
26∼75	74%	74%	38%	.110	.092	.16
76∼175	73%	73%	38%	.098	.092	.15
176∼375	79%	79%	23%	.100	.090	.15
376∼475	78%	78%	26%	.096	.084	.15

Table 3: Simulation 3 summary

rate of P_s and P_e is quite low (43% after 475 trials). Since the diagnostic rate of P_e is also lower (77%), one could ask which one is better. One way of viewing this is to compare the diagnostic rates. It is observed that, among Ps, P_c and P_e, no one is superior than others if *only* top diagnosis is concerned. More careful examination can be obtained by comparison of distances among models. It turns out that the distance (S-E) and distance (S-R) are smaller than the distance (E-R) with corresponding rms errors 0.061, 0.071 and 0.098 respectively.

The above 3 simulation assumed that only the subjective posterior judgments were available. In simulation 4, it was assumed that the correct diagnosis was also accessible. This time, $(D_c(1), P_c)$ was supplied with the scenario generated by $(D_r(1), P_r)$. P_e was the same as P_r.

The results (Table 4) showed that ALPP converged much quicker than {0,1} distribution learning even the latter had access to "true" answers to the diagnostic problem. After 1100 trials, $(D_s(1), P_s)$ reduced its maximum error from $(D_e(1), P_e)$ to 0.041 and matched the latter's behavior perfectly, while $(D_c(1), P_c)$ was still on its way of convergence with its error about 2 times larger and its behavior matching rate 80%.

Real world scenarios could be distinguished as being *common* or *exceptional*. An expert with knowledge about the real world tends to catch the common and to ignore the exceptional. Thus the diagnostic rate will never be 100%. This is the best one could do given the limited evidence. The PPs provided by

$(D_e(1), P_e)$		$(D_s(1), P_s)$		$(D_c(1), P_c)$	
trial No.	diag. rate	behv. mat. rate	max. err. S-E	behv. mat. rate	max. err. C-E
0			.300		.300
1∼100	88%	95%	.130	60%	.375
101∼600	78%	98%	.048	72%	.045
601∼1100	78%	93%	.052	61%	.075
1101∼1500	79%	100%	.041	80%	.079
1501∼1700	81%	100%	.025	85%	.093

Table 4: Simulation 4 summary

the expert contain the information about the entire domain, while a scenario contains only the information about this particular scene. Thus, although both $(D_s(1), P_s)$ and $(D_c(1), P_c)$ converged, the former converged quicker. This difference in convergence speed is expected to emerge wherever the diagnosis is difficult and the diagnostic rate of the expert is low.

6 Remarks

An algorithm of learning by PP distribution (ALPP) for sequential updating probability in Bayesian networks is presented. ALPP is based on any DAGs of diameter 1. After a network is constructed through elicitation of expert knowledge (qualitatively the dependency in the domain and quantitatively the FCPs), ALPP can be applied to improve it towards the expert's behavior. Several features of ALPP can be appreciated through the theoretical analysis and simulation results given in the paper.

- ALPP does not assume 100% posterior knowledge about the "true" answer to a diagnostic problem as does the {0,1} distribution learning [Spiegelhalter *et al.* 89]. When only expert's posterior judgments are available, ALPP converges to expert's behavior while {0,1} distribution learning will be misled by unavoidable error made in expert's diagnoses due to the violation of its underlying assumption.

- When both expert's posterior judgments and "true" answers are accessible, ALPP converges faster than {0,1} distribution learning due to the richer information contained in expert's posterior judgments.

- ALPP is tolerant to human consultants who are good but imperfect. When ALPP can not converge after many learning trials, it is an indication of inadequate DAG structure or inadequate posterior judgments.

- Computation of ALPP is simple.
- ALPP offers the possibility of combining the expertise from multiple experts, although this requires further research.

Acknowledgements

This work is supported by Operating Grant 583200 and OPPOO44121 from NSERC. Thanks are also directed to A. Eisen and M. MacNeil for valuable discussions.

References

[Andersen *et al.* 89] S. K. Andersen, K. G. Olesen, F. V. Jensen and F. Jensen, "HUGIN - a shell for building Bayesian belief universes for expert systems," *Proceedings, IJCAI - 89*, 1080-1085, 1989.

[Heckerman *et al.* 89] D. E. Heckerman, E. J. Horvitz and B. N. Nathwani, "Update on the Pathfinder project," *13th Symposium on computer applications in medical care*, 1989.

[Henrion 88] M. Henrion, "Propagating uncertainty in Bayesian networks by probabilistic logic sampling", J. F. Lemmer and L. N. Kanal (Edt), *Uncertainty in Artificial Intelligence 2*, Elsevier Science Publishers, 1988.

[Pearl 88] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*, Morgan Kaufmann.

[Lauritzen and Spiegelhalter 88] S. L. Lauritzen and D. J. Spiegelhalter, "Local computation with probabilities on graphical structures, and their application to expert systems," *J. Roy. Stat. Soc.*, B, 50, 157-244, 1988.

[Poole and Neufeld 88] D. Poole and E. Neufeld, "Sound probabilistic inference in Prolog: an executable specification of influence diagrams," *I SIMPOSIUM INTERNACIONAL DE INTELIGENCIA ARTIFICIAL*, Oct. 1988.

[Shachter and Heckerman 87] R. D. Shachter and D. E. Heckerman, "Thinking backward for knowledge acquisition", *AI Magazine*, 8:55-62, 1987.

[Spiegelhalter *et al.* 89] D. J. Spiegelhalter, R. C. G. Franklin, and K. Bull, "Assessment, criticism and improvement of imprecise subjective probabilities for a medical expert system," Proceedings, Fifth workshop on uncertainty in artificial intelligence, 335-342, Aug. 1989.

[Xiang *et al.* 90] Y. Xiang, A. Eisen, M. MacNeil and M. P. Beddoes, "QUALICON: Artificial intelligence in quality control and diagnosis of neuromuscular disease", to appear in American Academy of Neurology Annual Meeting, May, 1990.