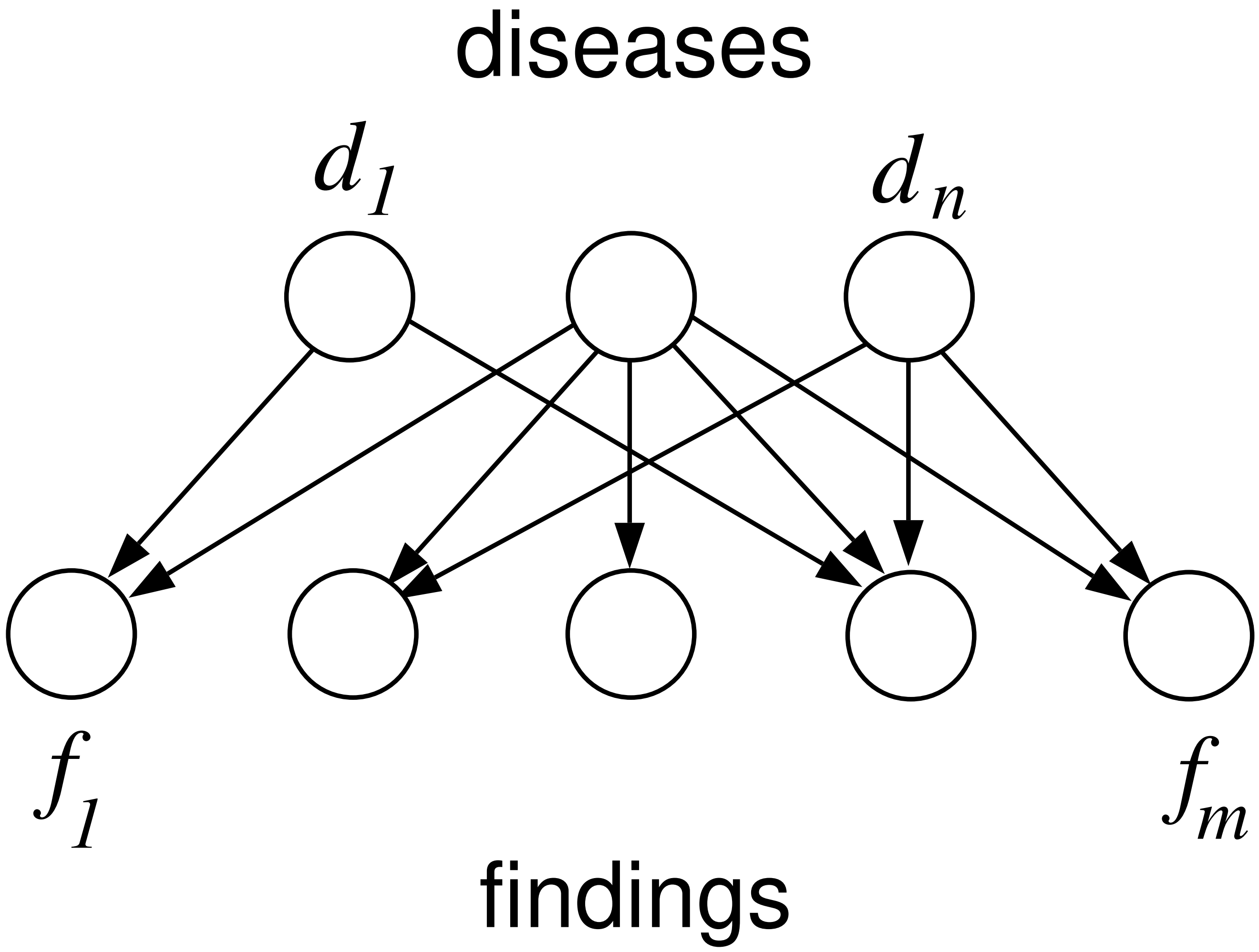

diseases
d_1
d_n
f_1
f_m
findings

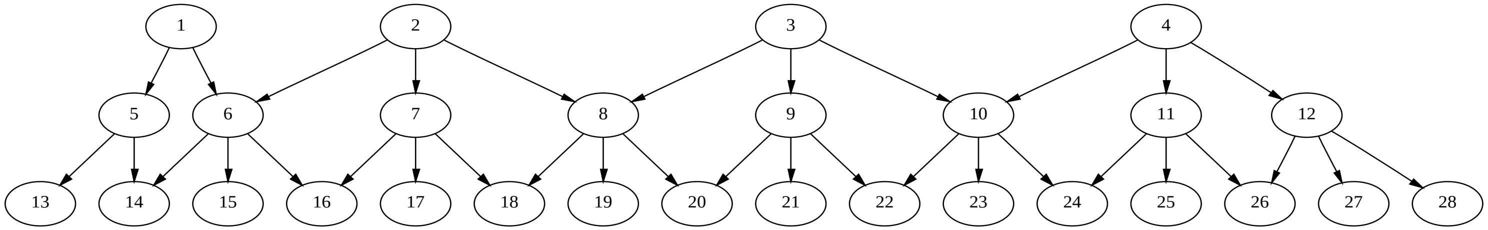
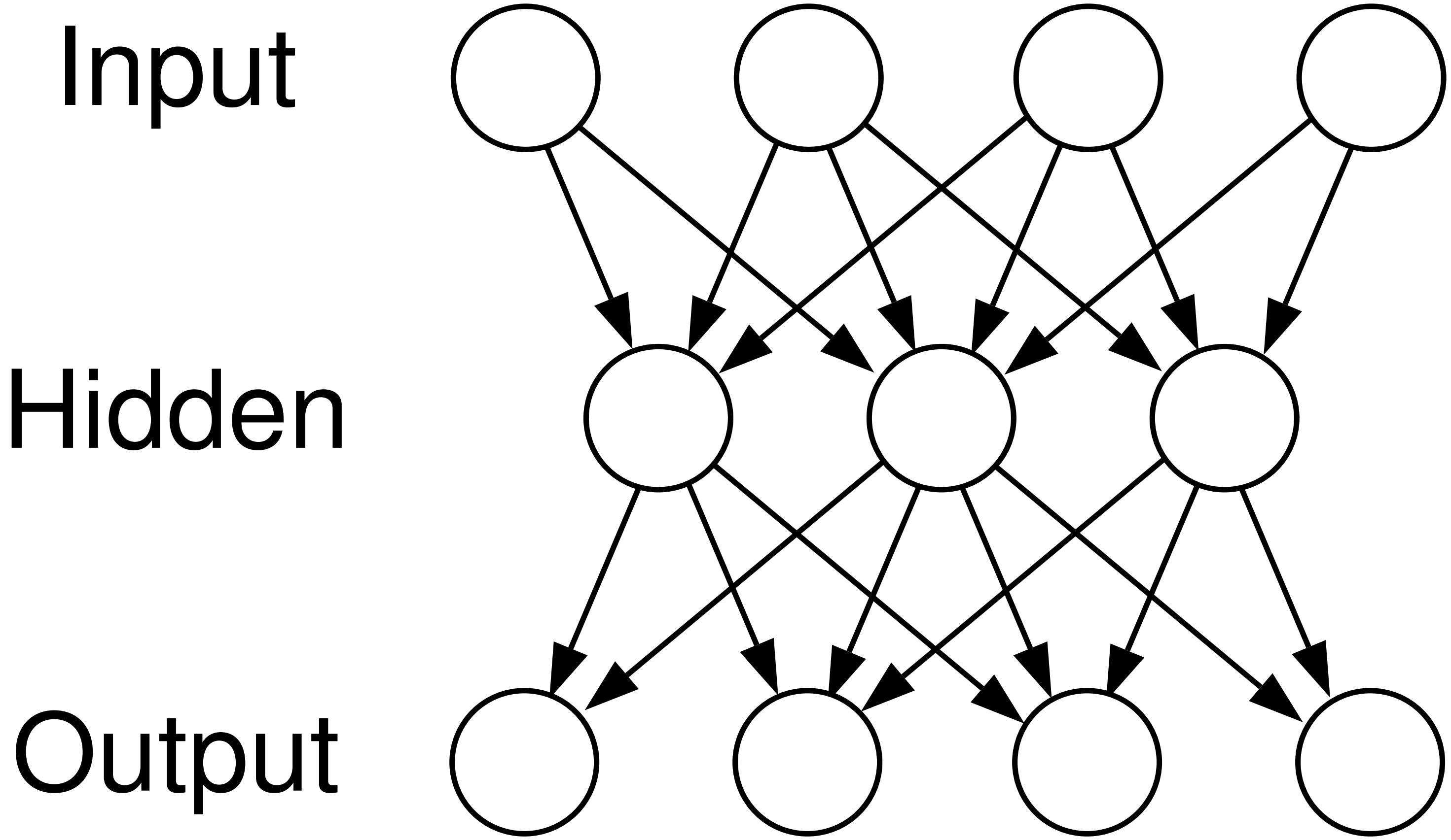
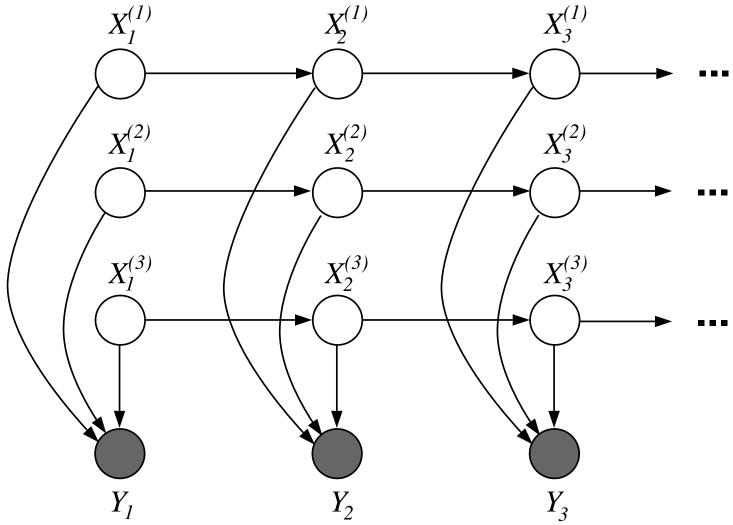
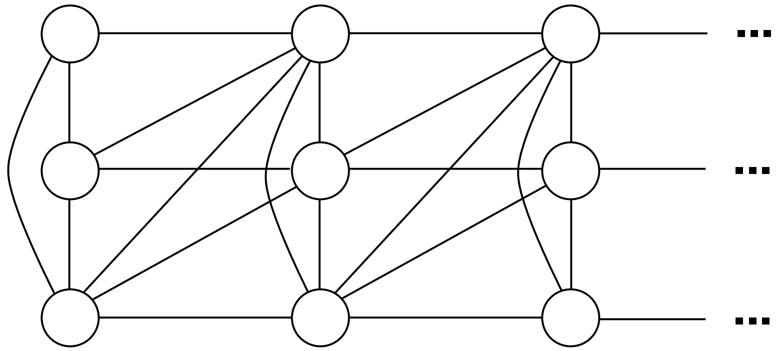


Figure 1: The structure of the PYRAMID network. All nodes are binary and observations appear only on the bottom layer. Such networks occur often in image analysis where the bottom layer would correspond to pixels.


Input
Hidden
Output

Why approximate propagation? - Factorial Hidden Markov Models (e.g. for speech recognition)





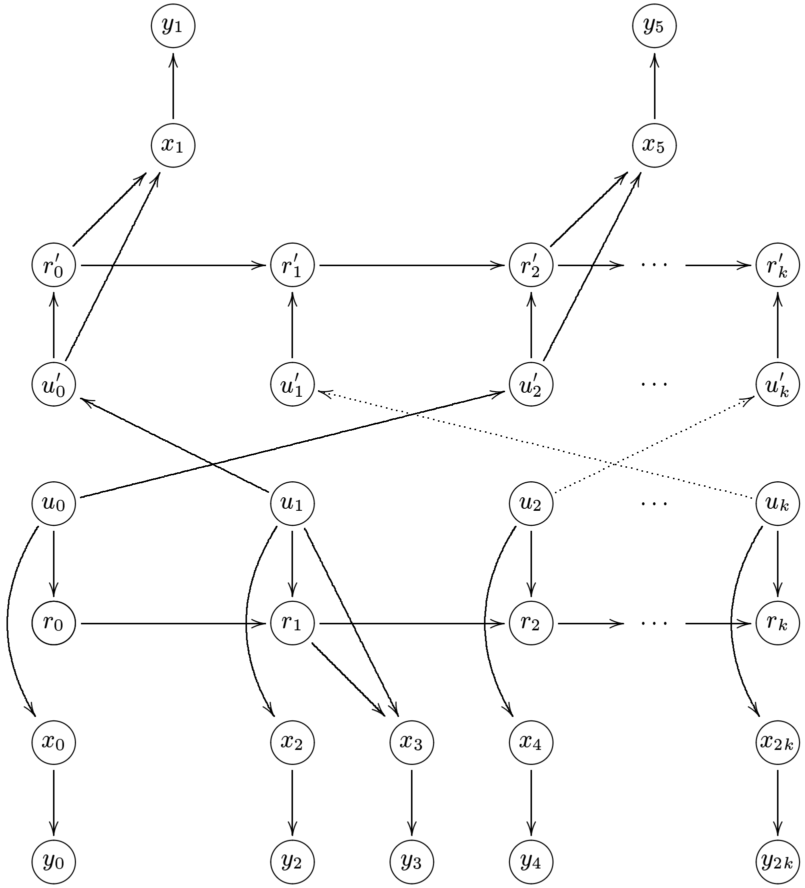
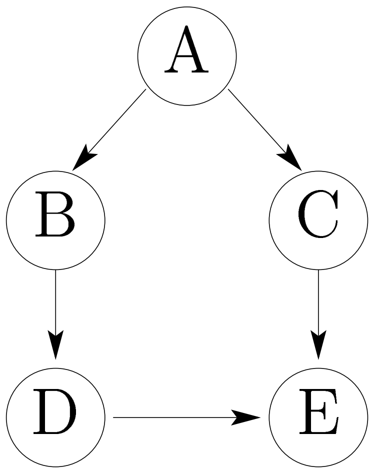


FIG. 6. Bayesian network for a rate 1/2 turbocode with blocksize k. The sequence $(u'_1, \ldots, u'_k)$ is obtained from $(u_1, \ldots, u_k)$ by a known permutation (scrambler).

- Stochastic simulation (forward sampling, Gibbs sampling)
- Pearl's polytree propagation applied to networks with loops, Pearl (1988),

---

- Variational methods for inference, Jordan et al. (1991),
- Penniless propagation, Cano et al. (2000),
- QuickScore Heckerman (1990), and

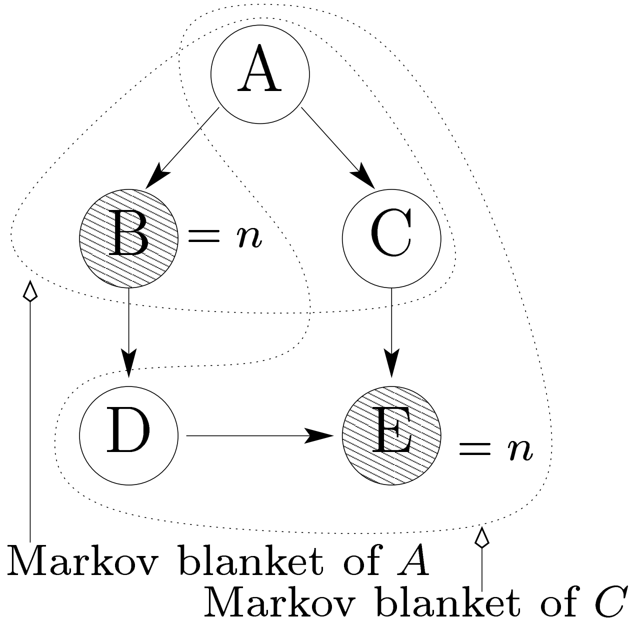


- $P(A) = (0.4, 0.6)$. Random generator gives a number $rnd \in [0, 1]$. If $rnd < 0.4$ then $A = y$ else $A = n$. Assume $A = y$.
- $P(B|A = y) = (0.3, 0.7)$. Assume $rnd > 0.3$ therefore $B = n$.
- $P(C|A = y) = (0.4, 0.6)$. Assume $rnd < 0.4$ therefore $C = y$.
- $P(D|B = n) = (0.1, 0.9)$. Assume $rnd < 0.1$ therefore $D = y$.
- $P(D|C = y, D = y) = (0.9, 0.1)$. Assume $rnd > 0.9$ therefore $E = n$.
- The sample is $(A = y, B = n, C = y, D = y, E = n)$.

:-) Each configuration is determined in time linear to the number of variables

:-(More accurate beliefs requires more time.

:-(If some variables are initiated then often many samples must be disregarded.



- $(A = y, B = n, C = y, D = y, E = n)$.
- $P(A|B = n, C = y, D = y, E = n)$
  $= P(A|B = n, C = y)$
  $\sim P(B = n|A) \cdot P(C = n|A)$
  Assume $rnd < 0.803$, i.e. $A = y$.
- $P(C|A = y, B = n, D = y, E = n) =$
  $= P(C|A = y, D = y, E = n)$
  $\sim P(C|A = y) \cdot P(E = n|C, D = y)$
  Assume $rnd < 0.996$, i.e. $C = y$.
- $P(D|A = y, B = n, C = y, E = n)$
  $\sim P(B = n|D) \cdot P(E = n|C = y, D)$.
  Assume $D = y$.
- $(A = y, B = n, C = y, D = y, E = n)$.

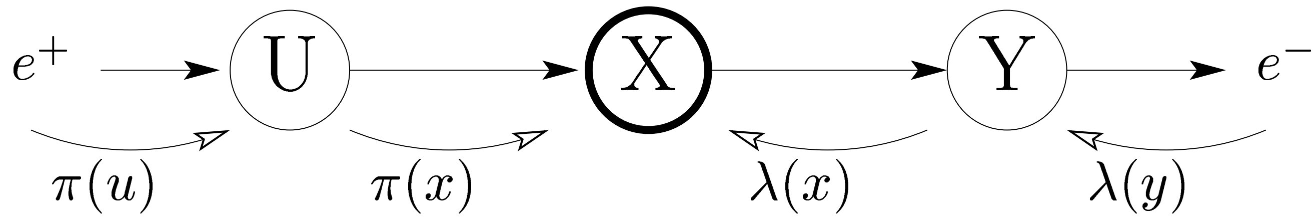
:-) Usually much faster than Forward sampling.

Is the sample representative of the probability distribution?
Not always!

:-(If you start with an improbable initial configuration you may stay too long in an "area" of variables' values.

:-(Sometimes it may be very difficult to switch from one "area" to another.

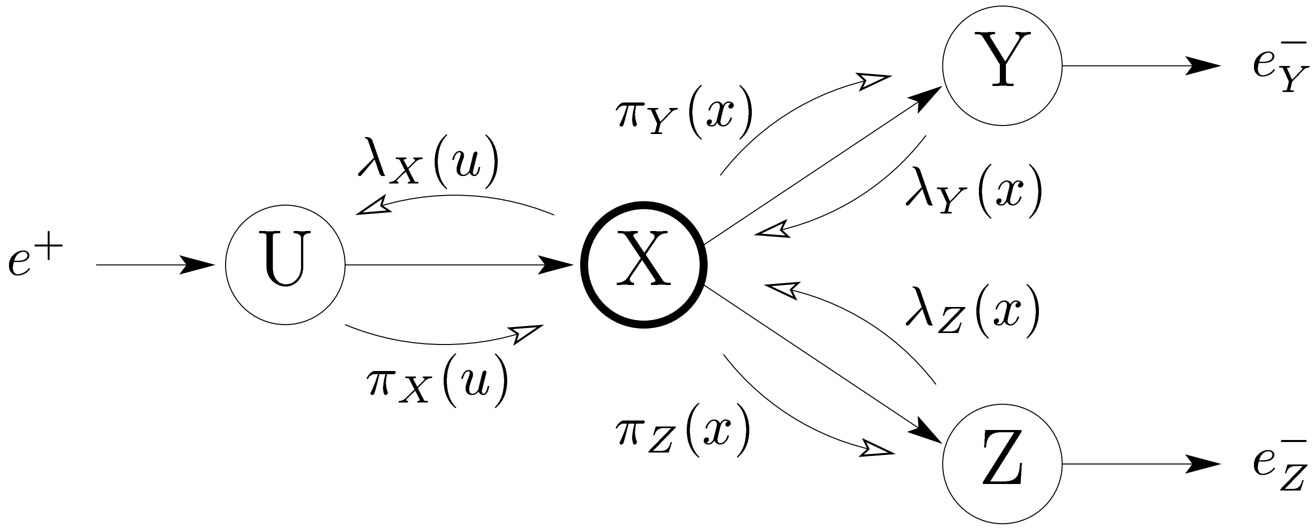
:-(Search for a starting configuration is generally NP-hard.



$$
\begin{aligned}
BEL(x) &\stackrel{df}{=} P(x|e^+, e^-) \;=\; \frac{P(e^-|x, e^+) \cdot P(x|e^+) \cdot P(e^+)}{P(e^+, e^-)} \\
&= \alpha \cdot P(e^-|x) \cdot P(x|e^+) \;=\; \alpha \cdot \lambda(x) \cdot \pi(x)
\end{aligned}
$$

$$
\begin{aligned}
\pi(x) &= P(x|e^+) \;=\; \sum_u P(x|u, e^+) \cdot P(u|e^+) \\
&= \sum_u P(x|u) \cdot \pi(u)
\end{aligned}
$$

$$
\begin{aligned}
\lambda(x) &= P(e^-|x) \;=\; \sum_y P(e^-|y, x) \cdot P(y|x) \\
&= \sum_y \lambda(y) \cdot P(y|x)
\end{aligned}
$$



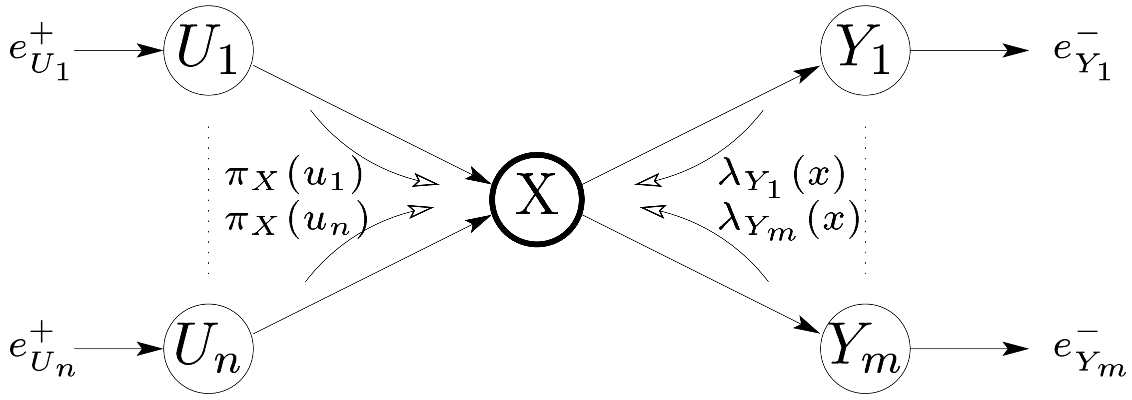
$$BEL(x) \stackrel{df}{=} P(x|e^+, e^-) = \alpha \cdot P(e_Y^-, e_Z^-|x) \cdot P(x|e^+) = \alpha \cdot \lambda(x) \cdot \pi(x)$$

$$\lambda(x) = P(e_Y^-, e_Z^-|x) = P(e_Y^-|x) \cdot P(e_Z^-|x) = \lambda_Y(x) \cdot \lambda_Z(x)$$

$$\pi(x) = \sum_u P(x|u) \cdot \pi_X(u)$$

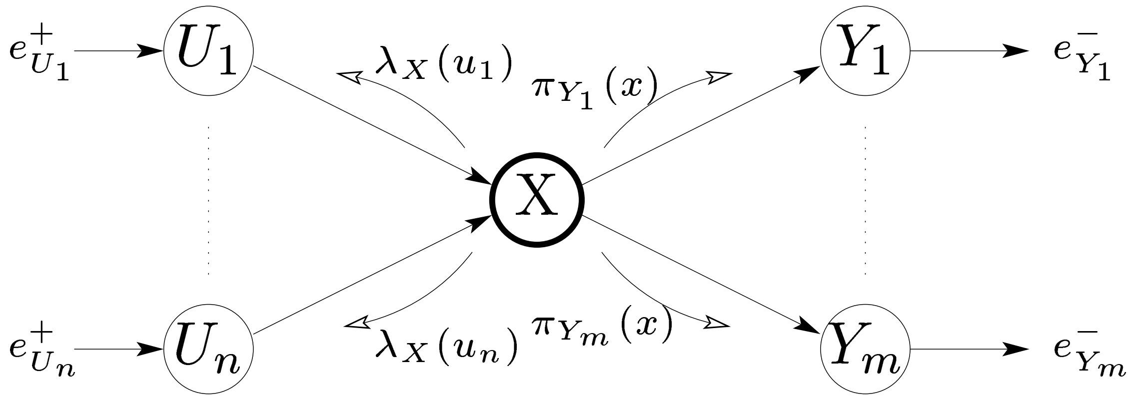
$$\lambda_X(u) = \sum_x \lambda(x) \cdot P(x|u)$$

$$\pi_Y(x) = \alpha \cdot \pi(x) \cdot \lambda_Z(x) \qquad \pi_Z(x) = \alpha \cdot \pi(x) \cdot \lambda_Y(x)$$

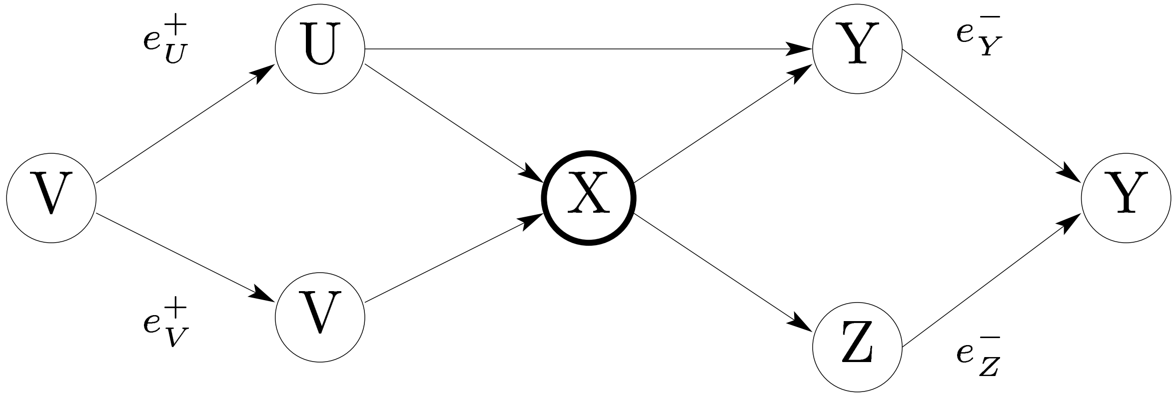


$$
\begin{aligned}
BEL(x) &\stackrel{df}{=} P(x|e^+, e^-) \\
&= \alpha \cdot P(e^-_{Y_1}, \ldots, e^-_{Y_m}|x) \cdot P(x|e^+_{U_1}, \ldots, e^+_{U_n}) \;=\; \alpha \cdot \lambda(x) \cdot \pi(x)
\end{aligned}
$$

$$
\begin{aligned}
\pi(x) &= \sum_{u_1,\ldots,u_n} P(x|u_1, \ldots, u_n) \cdot \prod_{i=1}^{n} p(u_i|e^+_{U_i}) \\
&= \sum_{u_1,\ldots,u_n} P(x|u_1, \ldots, u_n) \cdot \prod_{i=1}^{n} \pi_X(u_i) \\
\lambda(x) &= \prod_{j=1}^{m} P(e^-_{Y_j}|x) \;=\; \prod_{j=1}^{m} \lambda_{Y_j}(x)
\end{aligned}
$$



$$
\begin{aligned}
\lambda_X(u_i) &= P(x|e^+_{U_1},\ldots,e^+_{U_{i-1}},e^+_{U_{i+1}},\ldots,e^+_{U_n},e^-_{Y_1},\ldots,e^-_{Y_m}) \\
&= \sum_x \sum_{\ell\neq i} \lambda(x)\cdot P(x|u_1,\ldots,u_n)\cdot \prod_{\ell\neq i} P(u_\ell|e^-_{U_\ell}) \\
&= \alpha\cdot \sum_x \lambda(x)\cdot \sum_{\ell\neq i} P(x|u_1,\ldots,u_n)\cdot \prod_{\ell\neq i}\pi_X(u_\ell) \\
\pi_{Y_j}(x) &= P(x|e^+_{U_1},\ldots,e^+_{U_n},e^-_{Y_1},\ldots,e^-_{Y_{j-1}},e^-_{Y_{j+1}},\ldots,e^-_{Y_m}) \\
&= \alpha\cdot\pi(x)\cdot\prod_{k\neq j}\lambda_{Y_k}(x)
\end{aligned}
$$



Any of the following independence assumptions need not hold:

- $X \perp\!\!\!\perp e_U^+ | U, \quad X \perp\!\!\!\perp e_V^+ | V,$
- $X \perp\!\!\!\perp e_Y^- | Y, \quad X \perp\!\!\!\perp e_Z^- | Z,$
- $U \perp\!\!\!\perp V, \quad Y \perp\!\!\!\perp Z$

However, the algorithm can be used as an approximative one.

:-) In some applications it provides surprisingly good results - Turbo-Codes.

:-(Sometimes it oscillates - QMR-DT.

:-(It seems that the behavior depends on both the structure and numerical values of a Bayesian network.

:-) However, oscillations can be recognized quite soon. This may help to check whether the method is appropriate for a given problem.

References

Cano, A., S. Moral, and A. Salmerón (2000). Penniless propagation in join trees. *International Journal of Intelligent Systems 15*(1027–1095).

Heckerman, D. (1990). A tractable inference algorithm for diagnosing multiple diseases. In M. Henrion, R. Shachter, L. Kanal, and J. Lemmer (Eds.), *Uncertainty in Artificial Intelligence 5*, pp. 163–171. Elsevier Science Publishers B.V. (North Holland).

Jordan, M. I., Z. Ghahramani, T. S. Jaakkola, and L. K. Saul (1991). An introduction to variational methods for graphical models. *Machine Learning 37*, 183–233.

Pearl, J. (1988). *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Infrence.* San Mateo, California: Morgan Kaufmann Publishers.