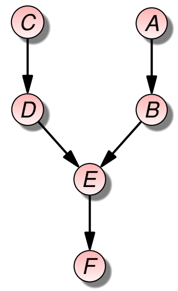
cours 5
Algorithme de Shafer-Shenoy

Aix*Marseille université Initiative d'excellence Master SID — Raisonnement dans l'incertain

Exploitation d'un réseau bayésien

diagnostic

- *diagnostic de panne*
- *sûreté de fonctionnement*
- *filtrage de spams*



prédiction

- *modélisation de joueurs*
- *prévisions boursières*

Une présentation unifiée des algorithmes d'inférence

Définition

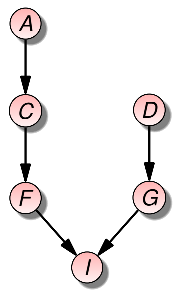
Inférence : calculer des probabilités :
- marginales ($P(X)$), jointes ($P(X, Y)$)
- *a priori* ($P(X)$, $P(X, Y)$), *a posteriori* ($P(X|e)$, $P(X, Y|e)$)

- **Applications :** prédiction, diagnostic
- **Les algorithmes d'inférence :**

Weighted Model Counting	} inférence logique/compilation
Lazy propagation	} méthodes non orientées
Shafer-Shenoy	
Jensen (HUGIN)	
Pearl	} méthode orientée

Osons le calcul des probabilités a priori

$$P(A, C, D, F, G, I) = P(A)P(C|A)P(F|C)P(D)P(G|D)P(I|F, G)$$



Calcul de $P(I)$?

Shafer-Shenoy brut de fonderie

$$P(A,C,D,F,G,I) = P(A)P(C|A)P(F|C)P(D)P(G|D)P(I|F,G)$$

$$P(I) = \sum_G \left(\sum_F \left(\sum_D \left(\sum_C \left(\sum_A P(A,C,D,F,G,I)\right)\right)\right)\right)$$

$$\sum_A P(A,C,D,F,G,I) = \underbrace{\left(\sum_A P(A)P(C|A)\right)}_{P(C)} P(F|C)P(D)P(G|D)P(I|F,G)$$

$$\sum_A P(A,C,D,F,G,I) = P(C)P(F|C)P(D)P(G|D)P(I|F,G)$$

$$\sum_C \sum_A P(A,C,D,F,G,I) = \underbrace{\left(\sum_C P(C)P(F|C)\right)}_{P(F)} P(D)P(G|D)P(I|F,G)$$

Shafer-Shenoy graphique (1/6)

Séquence d'élimination

$\boxed{A}$ C D F G

$$P(A,C,D,F,G,I) = \boxed{P(A)P(C|A)}\, P(F|C)P(D)P(G|D)P(I|F,G)$$

A	AC	FC	D	GD	IFG
$P(A)$	$P(C\|A)$	$P(F\|C)$	$P(D)$	$P(G\|D)$	$P(I\|F,G)$

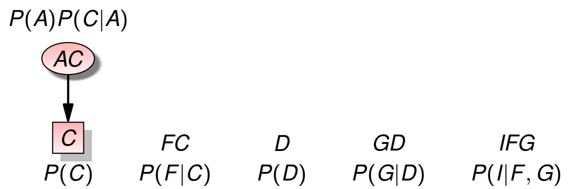
somme sur $A \Longrightarrow P(C) = \sum_A P(A)P(C|A)$

Shafer-Shenoy graphique (2/6)

Séquence d'élimination

A $\boxed{C}$ D F G

$$P(C,D,F,G,I) = \boxed{P(C)P(F|C)}\, P(D)P(G|D)P(I|F,G)$$



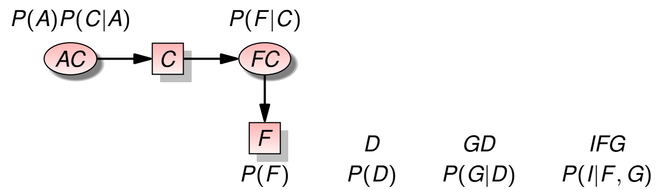
somme sur $C \Longrightarrow P(F) = \sum_C P(C)P(F|C)$

Shafer-Shenoy graphique (3/6)

Séquence d'élimination

A C $\boxed{D}$ F G

$$P(D,F,G,I) = P(F)\,\boxed{P(D)P(G|D)}\, P(I|F,G)$$



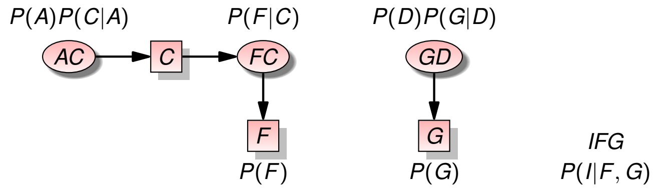
somme sur $D \Longrightarrow P(G) = \sum_D P(D)P(G|D)$

Shafer-Shenoy graphique (4/6)

Séquence d'élimination

$A\ C\ D\ \boxed{F}\ G$

$P(D, F, G, I) = \boxed{P(F)P(I|F, G)}\, P(G)$



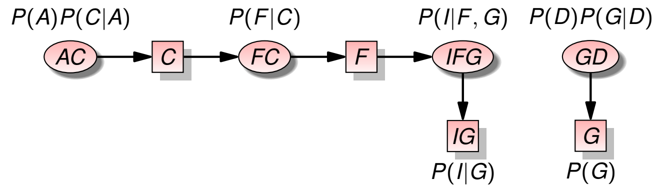
somme sur $F \Longrightarrow P(I|G) = \sum_F P(F)P(I|F, G)$

Shafer-Shenoy graphique (5/6)

Séquence d'élimination

$A\ C\ D\ F\ \boxed{G}$

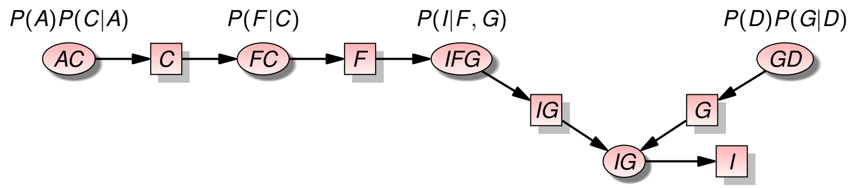
$P(D, F, G, I) = \boxed{P(I|G)P(G)}$



somme sur $G \Longrightarrow P(I) = \sum_G P(G)P(I|G)$

Shafer-Shenoy graphique (6/6)

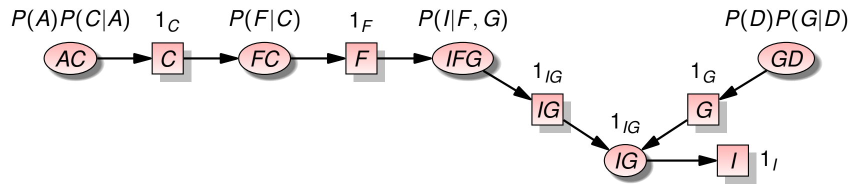
Le graphe final obtenu par Shafer-Shenoy



Algorithme de Shafer-Shenoy

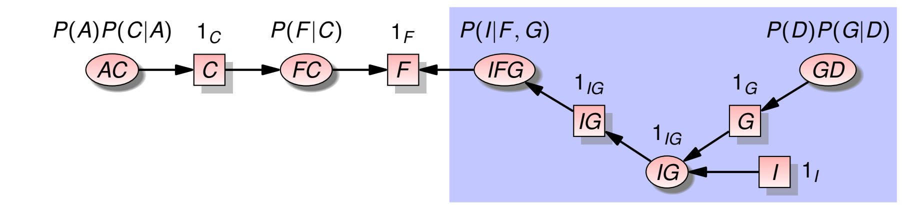
1. Se donner une séquence d'élimination des nœuds $\Longrightarrow$ *join tree, junction tree*
2. propager les impacts dans le sens des flèches :
   - dans les ellipses (cliques), on effectue des multiplications (combinaisons),
   - dans les rectangles (séparateurs), on effectue des additions (projections).

Quelques remarques sur Shafer-Shenoy (1/3)

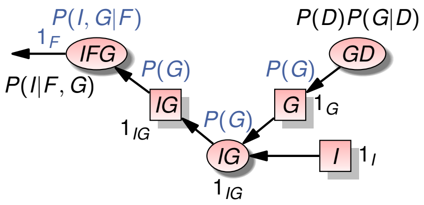


1. Loi jointe des variables = produit de fonctions des cliques et des séparateurs :
   $P(A, C, D, F, G, I) =$
   $P(A)P(C|A)P(F|C)P(D)P(G|D)P(I|F, G)$.
2. Si l'on élimine récursivement des cliques et séparateurs « externes », c'est-à-dire n'ayant qu'un voisin, le produit des fonctions des cliques et des séparateurs restants est la loi jointe des variables restantes.

Quelques remarques sur Shafer-Shenoy (2/3)



$$P(A) \times P(C|A) \times P(F|C) \times \mathbb{1}_C \times \mathbb{1}_F = P(A, C, F).$$



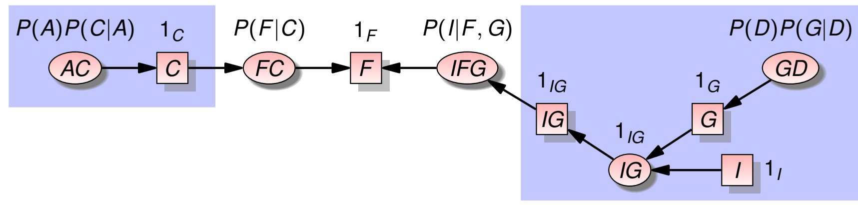
séparateur G : $\sum_D P(D)P(G|D)$

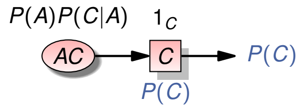
clique IG : $\boxed{P(G)} \times \mathbb{1}_{IG} \times \mathbb{1}_G \times \mathbb{1}_I$

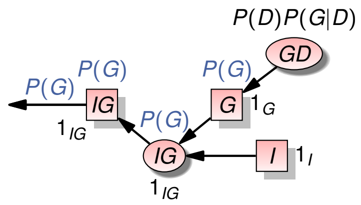
clique IFG : $\sum_{IG} P(I|F, G) \times \boxed{P(G)}$

$\Longrightarrow$ en sortie de IFG : $\boxed{\mathbb{1}_F}$

Quelques remarques sur Shafer-Shenoy (3/3)







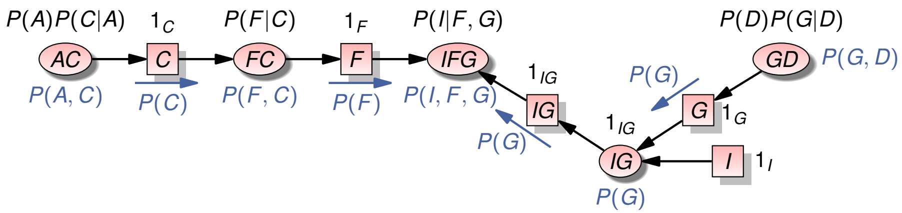
Produit des cliques et séparateurs restants :

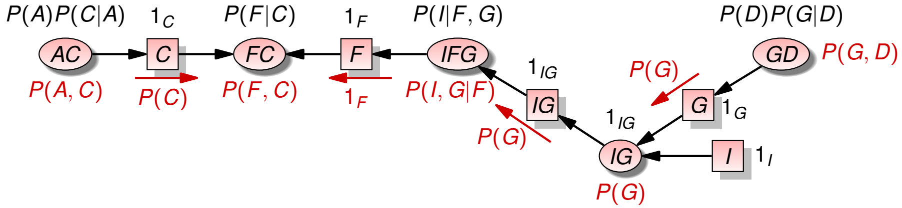
$\boxed{P(C)}\, P(F|C)\mathbb{1}_F P(I|F, G)\, \boxed{P(G)}$
$= P(I, C, F, G)$

Conclusion : on peut utiliser le même graphe pour calculer toutes les probabilités marginales

C'est pas le deux en un, mais le tout en deux (1/3)

En bleu : les calculs de $P(I, F, G)$, en rouge, ceux de $P(F, C)$





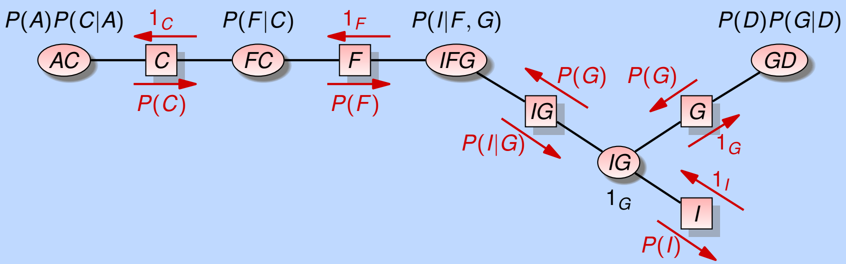
C'est pas le deux en un, mais le tout en deux (2/3)

Algorithme de Shafer-Shenoy

1. Chaque séparateur contient deux messages initialisés à $\mathbb{1}$, un en direction de chaque clique voisine.
2. Chaque nœud du join tree envoie des messages vers ses voisins en respectant les deux règles suivantes :
   1. avant d'envoyer un message vers son voisin X, le nœud Y attend que tous ses autres voisins lui aient envoyé leur message.
   2. le message d'un nœud Y vers son voisin X est le produit de tous les messages reçus par Y, à l'exception de celui envoyé par X, et de la table stockée par Y, le tout marginalisé sur X (c'est-à-dire sommé sur les variables de $Y \backslash X$).

C'est pas le deux en un, mais le tout en deux (3/3)

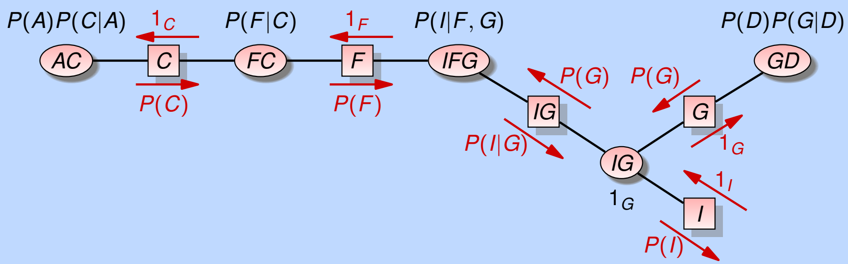
Algorithme de Shafer-Shenoy



À la fin de l'algorithme, pour tout nœud X, le produit de la table stockée en X par l'ensemble des messages envoyés à X est la probabilité jointe des variables de X.

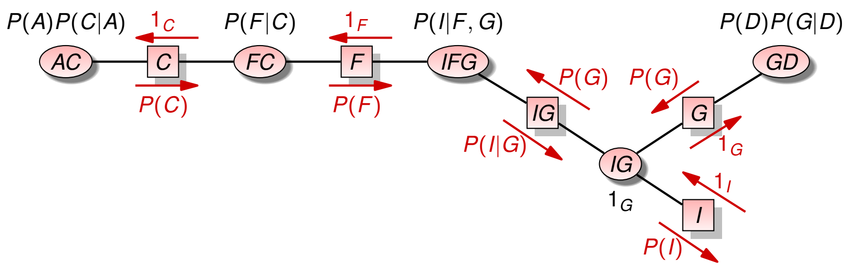
Shafer-Shenoy : ce qu'il faut retenir

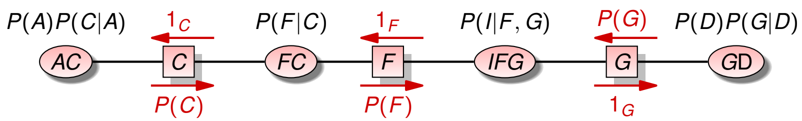
Algorithme de Shafer-Shenoy



- cliques = ellipses, séparateurs = rectangles
- algorithme par envoi de messages
- opérateurs = $+$ sur les séparateurs, $\times$ sur les cliques

Les arbres de jonction





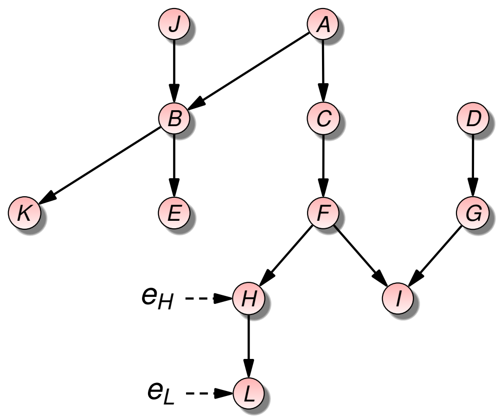
arbre de jonction : suppression des cliques incluses dans d'autres cliques

Soyons observateur : les probabilités a posteriori

Nous observons des informations e_H sur H et e_L sur L.

On veut connaître $P(I|e_H, e_L)$

$\Longrightarrow$ on va calculer $P(I, e_H, e_L)$ car c'est plus simple



Quelles sont ces informations ?

Teneur des informations

informations = observations :

e_L = « L ne peut plus prendre les valeurs l_1 et l_4 »

Entrée de ces observations : $P(e_L|L)$

Vous avez dit calcul de $P(e_L|L)$? (1/2)

observation : e_L = « Le capteur m'indique que L ne peut plus prendre les valeurs l_1 et l_4 »

$$P(e_L|L) = \begin{array}{|c|l} \hline 0 & l_1 \\ \hline 1 & l_2 \\ \hline 1 & l_3 \\ \hline 0 & l_4 \\ \hline \end{array}$$

⚠ dimension de $P(e_L|L) = |L|$

Vous avez dit calcul de $P(e_L|L)$? (2/2)

observation : e_L = « Le capteur m'indique que L a pris la valeur l_2. Mais je n'ai pas totalement confiance en ce capteur. Je pense qu'il y a 90% de chances pour que $L = l_2$, mais il y a également 10% de chances que $L = l_1$ ou que $L = l_3$. »

$$P(e_L|L) = \begin{array}{|c|l} \hline 0.1 & l_1 \\ \hline 0.9 & l_2 \\ \hline 0.1 & l_3 \\ \hline 0 & l_4 \\ \hline \end{array}$$

Hypothèses et conséquences (1/3)

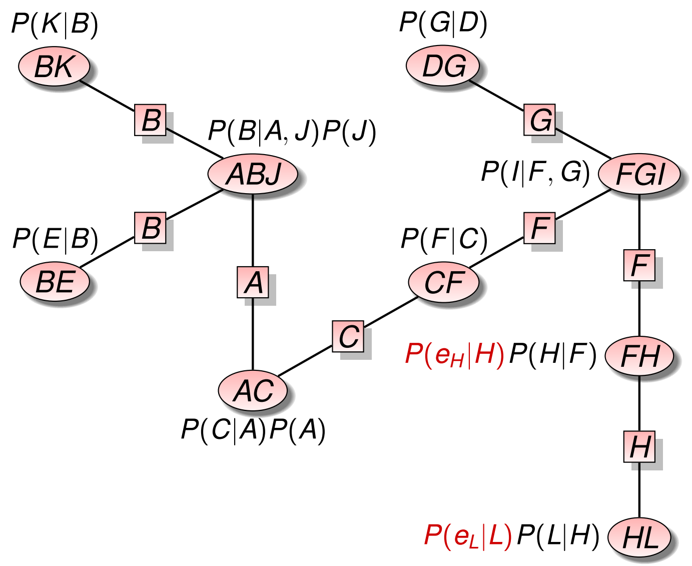
Hypothèse

Toute information e_X sur un nœud X est indépendante du reste du réseau bayésien conditionnellement à X.

Conséquence

$$\begin{aligned} P(A,B,C,D,E,F,G,H,I,J,K,L,e_H,e_L) = \\ & P(e_H|A,B,C,D,E,F,G,H,I,J,K,L,e_L) \times \\ & P(e_L|A,B,C,D,E,F,G,H,I,J,K,L) \times \\ & P(A,B,C,D,E,F,G,H,I,J,K,L) \\ = P(e_H|H)P(e_L|L)P(A,B,C,D,E,F,G,H,I,J,K,L) \end{aligned}$$

Hypothèses et conséquences (2/3)



Hypothèses et conséquences (3/3)

L'algorithme de Shafer-Shenoy permet donc de calculer dans la clique FGI : $P(F, G, I, e_H, e_L)$

$$\Longrightarrow P(I, e_H, e_L) = \sum_{F,G} P(F, G, I, e_H, e_L)$$

$$\Longrightarrow P(I|e_H, e_L) = \frac{P(I, e_H, e_L)}{P(e_H, e_L)}$$

Or $P(e_H, e_L) = \sum_I P(I, e_H, e_L)$

Donc $P(I|e_H, e_L) = \dfrac{P(I, e_H, e_L)}{\sum_I P(I, e_H, e_L)}$.

Résumé sur l'algorithme de Shafer-Shenoy

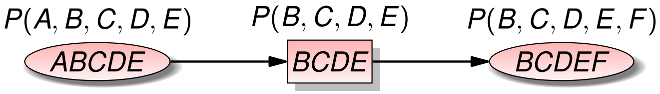
Algorithme de Shafer-Shenoy

Pour calculer une proba *a posteriori* $P(Y|e)$:

1. construire l'arbre de jonction
2. insérer les probas conditionnelles du réseau bayésien dans les cliques
3. insérer des tables contenant uniquement des 1 dans les séparateurs
4. pour chaque information e_X, calculer $P(e_X|X)$ et insérer cette proba dans une clique contenant X
5. envoyer les messages dans l'arbre de jonction
6. choisir une clique contenant Y, faire le produit de sa table de proba par tous les messages qui lui ont été envoyés, puis sommer sur toutes les variables $\neq Y \Longrightarrow P(Y, e)$
7. normaliser $P(Y, e) \Longrightarrow P(Y|e)$

2. Existe-t-il d'autres algorithmes d'inférence ?

De Shafer-Shenoy à Lazy propagation



si $P(A, B, C, D, E) = P(A)P(B)P(C|A, B)P(D)P(E|C, D)$

alors $P(B, C, D, E) = \sum_A P(A)P(B)P(C|A, B)P(D)P(E|C, D)$

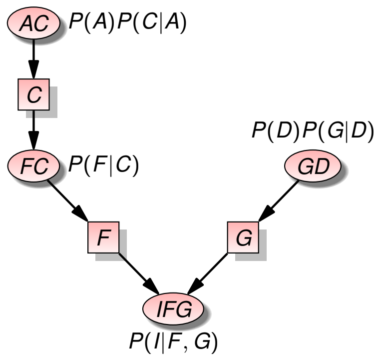
$$= \left(\sum_A P(A)P(C|A, B)\right) P(B)P(D)P(E|C, D)$$

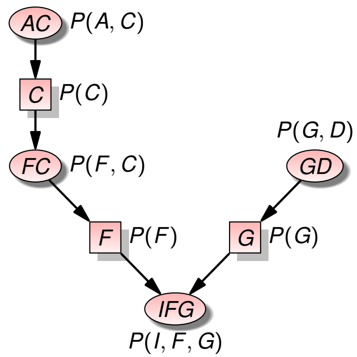
$\Longrightarrow$ on a intérêt à ne pas effectuer les produits avant les sommes

Lazy propagation

Principe : garder les produits sous forme de listes et n'effectuer les multiplications que lorsque c'est nécessaire.

De Shafer-Shenoy à Jensen





Shafer-Shenoy : élimination de $A \Longrightarrow P(C, e_A) = \sum_A P(A, e_A)P(C|A)$

élimination de $C \Longrightarrow P(F, C, e_A) = P(F|C)P(C, e_A)$

Jensen : élimination de $C \Longrightarrow P(F, C, e_A) = \dfrac{P(F, C)}{P(C)} P(C, e_A)$

Bibliographie

- Chavira M. et Darwiche A. (2008) « On probabilistic inference by weighted model counting », Artificial Intelligence 172 :772–799
- Dudek J.M., Phan V. et Vardi M.Y. (2020) « ADDMC : Weighted Model Counting with Algebraic Decision Diagrams », Proceedings of the International Conference on Artificial Intelligence, 1468–1476
- Jensen F.V., Lauritzen S.L. et Olesen K.G. (1990) « Bayesian Updating in Causal Probabilistic Networks by Local Computations », Computational Statistics Quarterly, 4 :269–282
- Lauritzen S.L. et Spiegelhalter D.J. (1988) « Local computations with probabilities on graphical structures and their application to expert systems(with discussion) », Journal of the Royal Statistical Society, *Series B*, 50 :157–224

Bibliographie

- Madsen A.L. et Jensen F.V. (1998) « Lazy Propagation in Junction Trees », Proceedings of Uncertainty in Artificial Intelligence
- Madsen A.L. et Jensen F.V. (1999) « Lazy Propagation : A Junction Tree Inference Algorithm Based on Lazy Evaluation », Artificial Intelligence, 113 :203–245
- Marquis P. (2015) « Compile ! », Proceedings of the International Conference on Artificial Intelligence
- Shafer G. (1996) Probabilistic expert systems, Society for Industrial and Applied Mathematics
- Shenoy P.P. et Shafer G. (1990) « Axioms for probability and belief-function propagation », Proceedings of Uncertainty in Artificial Intelligence, 169–198
- Shenoy P.P. (1997) « Binary join trees for computing marginals in the Shenoy-Shafer architecture », International Journal of Approximate Reasoning 17 :1–25