cours 6
Algorithme de Shafer-Shenoy

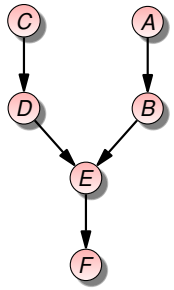
Aix*Marseille université Initiative d'excellence — Master SID — Raisonnement dans l'incertain

©CG(2026)

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 :

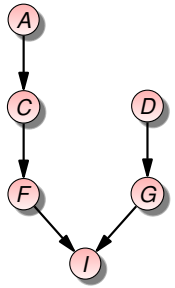
- marginales ($P(X)$), jointes ($P(X, Y)$)
- *a priori* ($P(X)$, $P(X, Y)$), *a posteriori* ($P(X|e)$, $P(X, Y|e)$)

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

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

A C D F G

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

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

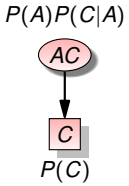
somme sur $A \Longrightarrow P(C) = \sum_A P(A)P(C|A)$

Shafer-Shenoy graphique (2/6)

Séquence d'élimination

A C D F G

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



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

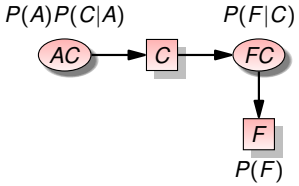
somme sur $C \Longrightarrow P(F) = \sum_C P(C)P(F|C)$

Shafer-Shenoy graphique (3/6)

Séquence d'élimination

A C D F G

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



D	GD	IFG
$P(D)$	$P(G\|D)$	$P(I\|F,G)$

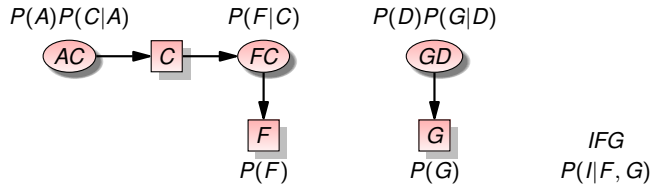
somme sur $D \Longrightarrow P(G) = \sum_D P(D)P(G|D)$

Shafer-Shenoy graphique (4/6)

Séquence d'élimination

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

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



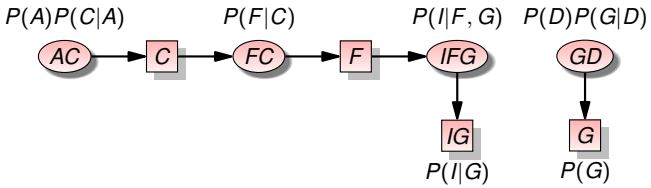
somme sur $F \Longrightarrow P(I|G) = \sum_F P(F)P(I|F,G)$

Shafer-Shenoy graphique (5/6)

Séquence d'élimination

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

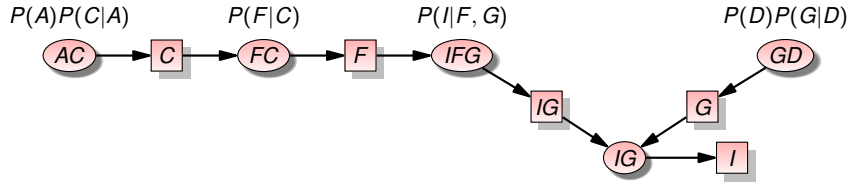
$P(D,F,G,I) = \boxed{P(I|G)P(G)}$



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

Shafer-Shenoy graphique (6/6)

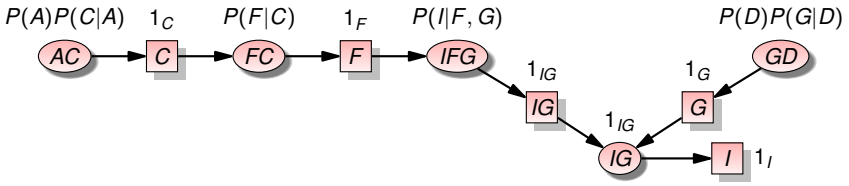
Le graphe final obtenu par Shafer-Shenoy



Algorithme de Shafer-Shenoy

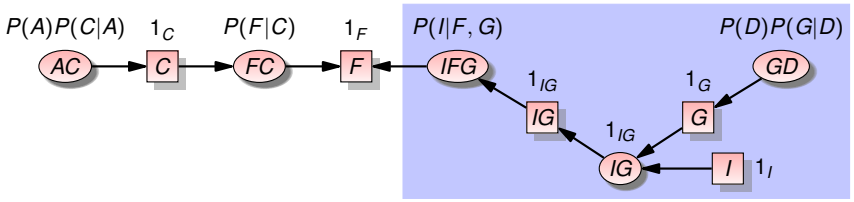
1. Se donner une séquence d'élimination des nœuds
   $\Longrightarrow$ *join tree*, *junction tree*
2. propager les impacts dans le sens des flèches :
   - dans les ellipses (cliques), on effectue des multiplications (combinaisons),
   - dans les rectangles (séparateurs), on effectue des additions (projections).

Quelques remarques sur Shafer-Shenoy (1/3)

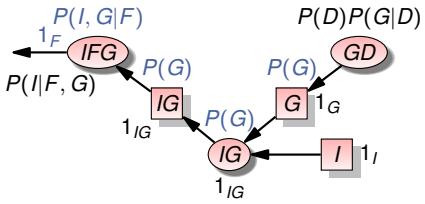


1. Loi jointe des variables = produit de fonctions des cliques et des séparateurs :
   $P(A,C,D,F,G,I) =$
   $P(A)P(C|A)P(F|C)P(D)P(G|D)P(I|F,G)$.
2. Si l'on élimine récursivement des cliques et séparateurs « externes », c'est-à-dire n'ayant qu'un voisin, le produit des fonctions des cliques et des séparateurs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

Quelques remarques sur Shafer-Shenoy (2/3)



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



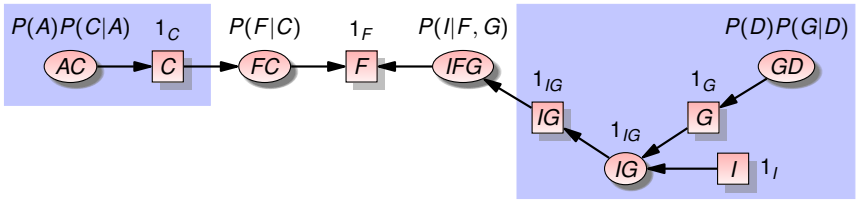
séparateur G : $\sum_D P(D)P(G|D)$

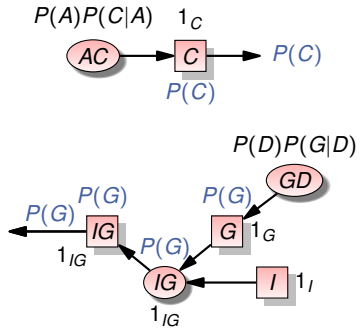
clique IG : $P(G) \times \mathbb{1}_{IG} \times \mathbb{1}_G \times \mathbb{1}_I$

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

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

Quelques remarques sur Shafer-Shenoy (3/3)





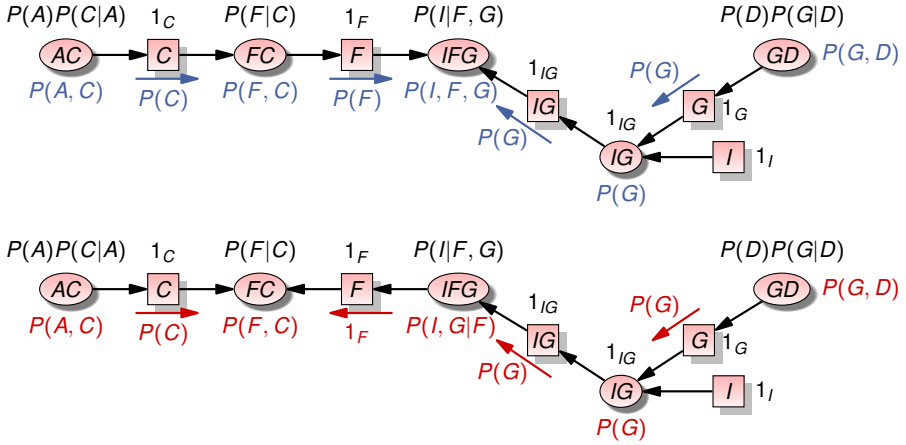
Produit des cliques et séparateurs restants :

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

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 $P(F, C)$



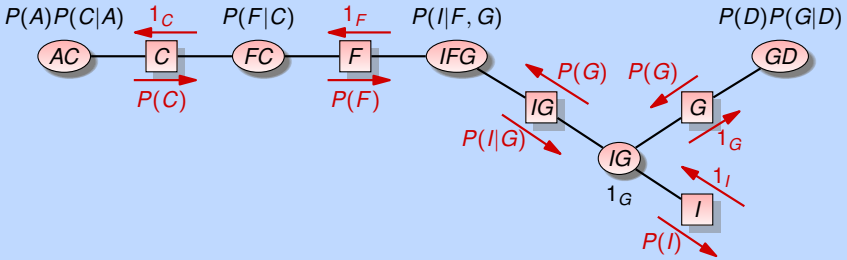
C'est pas le deux en un, mais le tout en deux (2/3)

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)

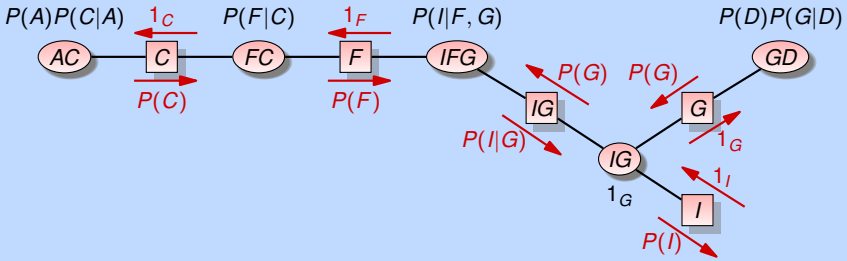
Algorithme de Shafer-Shenoy



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.

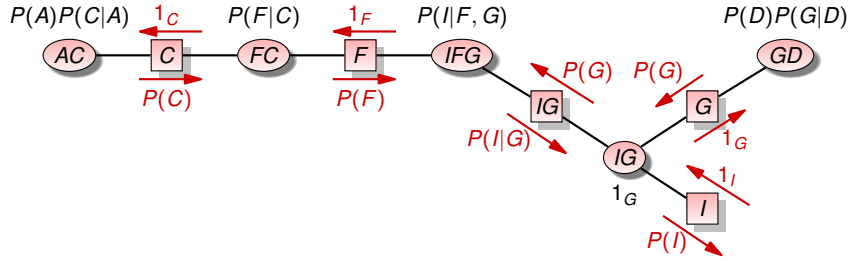
Shafer-Shenoy : ce qu'il faut retenir

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

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