

4.1 Rappels de la Notion de Modèles Graphiques non-orientés

(Rappels des notions se trouvant dans le cours du 22 oct, 2008.)

Définition 4.1 Soit $G = (\{1, \dots, n\}, E)$ un graphe non orienté.

Soit \mathcal{C} l'ensemble des cliques maximales de G .

Alors la famille des lois associées à G , $\mathcal{L}(G)$ s'écrit :

$$\mathcal{L}(G) = \left\{ p \mid p(x) = \frac{1}{Z} \prod_{C \in \mathcal{C}} \psi_C(x_C), \psi_C \geq 0 \right\} \quad (4.1)$$

où Z est une constante de normalisation :

$$Z = \sum_x \prod_{C \in \mathcal{C}} \psi_C(x_C) \quad (4.2)$$

4.2 Séparation et Indépendance

Proposition 4.2 Soit $G = (V, E)$ un graphe non orienté.

Si $p(x) \in \mathcal{L}(G)$, alors pour tout $A, B, C \subset V$ tels que C sépare (figure 4.1) A et B alors $X_A \perp\!\!\!\perp X_B \mid X_C$.

Démonstration On se ramène à A et B dans 1 seule composante connexe. Soit S séparant A et B . Pour toutes les composantes connexes de $V \setminus S$, A est inclus dans une seule composante connexe \tilde{A} , B est inclus dans \tilde{B} . Soit c une de clique du graphe, alors il y a 3 possibilités:

- $c \subset S$
- $c \subset \tilde{A} \cup S$
- $c \subset \tilde{B} \cup S$.

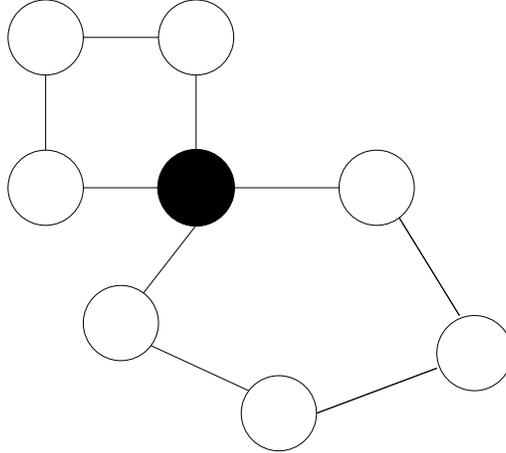


Figure 4.1. Le nœud grisé sépare les deux ensembles de nœuds

(Il est impossible que $c \cap \tilde{A} \neq \emptyset$ et $c \cap \tilde{B} \neq \emptyset$, sinon, il aura un chemin connectant \tilde{A} et \tilde{B}).

$$\begin{aligned}
 p(\mathbf{x}) &= \frac{1}{Z} \prod_{c \in \tilde{A} \cup S} \psi_c(x_c) \prod_{c \notin \tilde{A} \cup S} \psi_c(x_c) \\
 &= \frac{1}{Z} \prod_{c \in \tilde{A} \cup S} \psi_c(x_c) \prod_{c \in \tilde{B} \cup S} \psi_c(x_c) \\
 &= \frac{1}{Z} f(x_{\tilde{A} \cup S}) g(x_{\tilde{B} \cup S})
 \end{aligned}$$

$$\begin{aligned}
 p(\mathbf{x}_S) &= \sum_{\mathbf{x}_{\tilde{A}}} \sum_{\mathbf{x}_{\tilde{B}}} p(\mathbf{x}) \\
 &= \frac{1}{Z} \sum_{\mathbf{x}_{\tilde{A}}} f(x_{\tilde{A} \cup S}) \sum_{\mathbf{x}_{\tilde{B}}} g(x_{\tilde{B} \cup S})
 \end{aligned}$$

$$\begin{aligned}
 p(\mathbf{x}_A, \mathbf{x}_B, \mathbf{x}_S) &= \frac{1}{Z} \sum_{x_{\tilde{A}-A}} f(x_{\tilde{A} \cup S}) \sum_{x_{\tilde{B}-B}} g(x_{\tilde{B} \cup S}) \\
 &= \frac{1}{Z} f'(x_A, x_S) g'(x_B, x_S)
 \end{aligned}$$

On a ainsi conclu que $X_A \perp\!\!\!\perp X_B | X_S$. ■

4.3 Comparaison

	Modèles graphiques orientés	Modèles graphiques non-orientés
définition	$p(x) = \prod_{i=1}^n p(x_i x_{\pi_i})$	$p(x) = \frac{1}{Z} \prod_{c \in \mathcal{C}} \psi_c(x_c)$
interprétation	$p(x_i x_{\pi_i})$ probabilité	ψ_c n'est pas une probabilité
séparation	d-séparation	séparation
marginalisation	non close	close

Néanmoins, on a aussi

$$\mathcal{L}(G_{\text{orienté}}) = \mathcal{L}(G_{\text{non-orienté}})$$

Définition 4.3 Si G est un DAG le graphe symétrisé est \tilde{G} t.q. $\forall (u, v) \in \tilde{E}$ ssi. $(u, v) \in E$ ou $(v, u) \in E$.

Propriété 4.4 Si G n'a pas de v -structure, alors $\mathcal{L}(G) = \mathcal{L}(\tilde{G})$.

Définition 4.5 Si G est un DAG, le graphe moralisé \bar{G} est le graphe symétrisé dont on a connecté les parents entre eux.

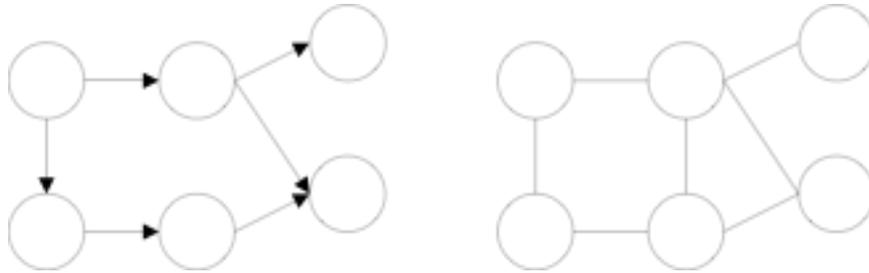


Figure 4.2. Exemple de moralisation d'un graphe

Propriété 4.6 $p(\mathbf{x}) \in \mathcal{L}(G) \Rightarrow p(\mathbf{x}) \in \mathcal{L}(\bar{G})$

Démonstration

$$\begin{aligned} p(\mathbf{x}) &= \prod_{i=1}^n p(x_i | x_{\pi_i}) \\ &= \prod_{i=1}^n \psi_{c_i}(x_{c_i}) \end{aligned}$$

ici, $c_i = i \cup \Pi_i$ est un clique dans \tilde{G}^1 ■

4.4 Algorithme d'élimination

4.4.1 Calcul de la constante de normalisation Z

$$\begin{aligned}
 Z &= \sum_{x_1, x_2, x_3, x_4, x_5, x_6} \psi_{12}(x_1, x_2) \psi_{14}(x_1, x_4) \psi_{45}(x_4, x_5) \psi_{256}(x_2, x_5, x_6) \psi_{23}(x_2, x_3) \\
 &= \sum_{x_1, x_2, x_4, x_5, x_6} \psi_{12}(x_1, x_2) \psi_{14}(x_1, x_4) \psi_{45}(x_4, x_5) \psi_{256}(x_2, x_5, x_6) \sum_{x_3} \psi_{23}(x_2, x_3) \\
 &= \sum_{x_1 x_2 x_4 x_5 x_6} \psi_{12}(x_1, x_2) \psi_{14}(x_1, x_4) \psi_{45}(x_4, x_5) \psi_{256}(x_2, x_5, x_6) m_2(x_2) \\
 &= \sum_{x_1 x_2 x_4 x_5} \psi_{12}(x_1, x_2) \psi_{14}(x_1, x_4) \psi_{45}(x_4, x_5) \sum_{x_6} \psi_{256}(x_2, x_5, x_6) m_2(x_2) \\
 &= \sum_{x_1 x_2 x_4} \psi_{12}(x_1, x_2) \psi_{14}(x_1, x_4) \sum_{x_5} \psi_{45}(x_4, x_5) m_{25}(x_2, x_5) m_2(x_2) \\
 &\dots
 \end{aligned}$$

Principe: sommer de proche en proche n fois, au lieu de sommer 2^n termes à la fois.

4.4.2 Algorithme d'élimination

Soit $G = (V, E)$ un graphe non orienté avec $V = \{1, \dots, n\}$. On considère l'ordre d'élimination $I = (n, n-1, \dots, 1)$. A chaque étape, l'algorithme élimine le noeud qui suit dans l'ordre I , c'est-à-dire, supprime le noeud du graphe et connecte les voisins restants, i.e., cet algorithme construit une séquence de graphes $G_n = G, G_{n-1}, \dots, G_0 = \emptyset$ tels que :

$$V_k = \{1, \dots, k\}$$

$$E_{k-1} = (E_k \cap V_{k-1} \times V_{k-1}) \cup N_{G_k}(k) \times N_{G_k}(k)$$

(où $N_{G_k}(k)$ désigne l'ensemble des voisins du sommet k dans G_k).

Définition 4.7 *Le graphe reconstitué est le graphe $\tilde{G} = (V, \tilde{E} = \cup_k E_k)$, dont l'ensemble des arêtes contient l'ensemble E d'origine ainsi que les nouvelles arêtes créées au cours de l'exécution de l'algorithme.*

¹dans la définition de la L , on demande la maximalité de la clique parce que ceci permet de réduire la redondance d'expression. En fait avec petit changement d'écriture il peut même exprimer le produit des cliques non-maximales.



Ce graphe a des propriétés importantes étudiées plus tard dans le cours. La principale est qu'il est triangulé (cette procédure d'élimination est en fait un algorithme de triangularisation de graphe).

Définition 4.8 On appelle *clique d'élimination* du noeud k l'ensemble des voisins d'un noeud après son élimination, le noeud lui-même y compris, i.e., l'ensemble $\{k\} \cup N_{G_k}(k)$.

4.4.3 Elimination et marginalisation

On considère une fonction $u(x)$ qui se factorise dans le graphe G , i.e., qui s'écrit

$$u(x) = \prod_{C \in \mathcal{C}} \psi_C(x_C)$$

où $\mathcal{C}(G)$ désigne l'ensemble des cliques du graphe G .

Proposition 4.9 La marginale de $u(x)$ sur $\{x_1, \dots, x_k\}$

$$u(x_1, \dots, x_k) = \sum_{x_i, i > k} u(x) = \prod_{C \in \mathcal{C}(\tilde{E}_k)} \tilde{\Psi}_C(x_C)$$

se factorise dans G_k .

Proposition 4.10 Passer de $u(x_1, \dots, x_k)$ à $u(x_1, \dots, x_{k-1})$ correspond à une somme sur $\#\{\{k\} \cup N_{G_k}(k)\}$ éléments.

4.4.4 Complexité finale de l'algorithme

La complexité finale de l'algorithme d'élimination est :

$$n \times r^{TM}$$

où TM est la taille maximale des cliques d'élimination. On a donc remplacé n par TM , ce qui permet d'obtenir la **linéarité en n** .

Selon l'ordre que l'on choisit, la complexité sera différente. On introduit donc la notion de **meilleur ordre** pour lequel la taille maximale sera la largeur arborescente, ou encore **tree-width** en anglais. On la note TW et on la définit par:

$$TW = \min_{\text{ordre}} \text{taille maximale des cliques d'élimination}$$

Cependant, on peut faire deux remarques:

1. TW est NP-difficile à calculer

2. Quand bien même on saurait le trouver, TW peut être ‘grand’, ce qui pose problème car la complexité finale est proportionnelle à $\exp(TW)$.

Cas d'école

1. Les arbres

Un arbre est un graphe acyclique. Donc, quand on élimine, on ne crée aucune arête. Dans ce cas, on a donc $TW = 1$.

2. Les grilles

Les grilles ont peu d'arêtes. Pourtant, TW peut être grand: $TW \geq C\sqrt{n}$. On dit des grilles que ce sont des **graphes creux**.

3. Les chaînes de Markov

$$TW = 1$$

4. Les chaînes de Markov cachées

On se ramène au cas de l'arbre.



r peut être très grand! Par exemple, dans le cas de la parole, r est le nombre de mots du dictionnaire. Il est de l'ordre de 10^5 . Il est donc très important de pouvoir passer de r^3 à r^2 .

4.5 Algorithme somme-produit

Sur un arbre non orienté (i.e., un graphe non orienté sans cycles), on considère une fonction $u(x)$ se factorisant dans G :

$$u(x) = \prod_{(i,j) \in E} \psi_{ij}(x_i x_j) \prod_{i \in V} \psi_i(x_i)$$

Le but est de calculer les marginales de $u(x)$.

Définition 4.11 *Les lois marginales sont données par:*

$$u(x_i) = \sum_{x_{i,j} \neq i} u(x)$$

$$u(x_i, x_j) = \sum_{x_k, k \neq i, k \neq j} u(x)$$

Le principe sous-jacent de cet algorithme est de remarquer que ce qui a été fait pour x_i n'a pas besoin d'être fait à nouveau pour $x_j, j < i$. On considère donc les messages suivants, auxquels on va attribuer un protocole de passage particulier:

Définition 4.12 Pour un couple $(i, j) \in E$, le message m_{ij} de $i \rightarrow j$ (fonction de x_j) est égal à:

$$m_{ij}(x_j) = \sum_{x_i} \psi_i(x_i) \psi_{ij}(x_i, x_j) \prod_{k \in \mathcal{N}(i) \setminus j} m_{ki}(x_i)$$

Pour calculer le message de i vers j , on prend donc en compte le potentiel lié à l'arête (i, j) et les messages envoyés par les voisins de i qui ne sont pas j . Il faut donc avoir défini tous les messages précédents. D'où le protocole de passage suivant, **valable dans le cas de l'arbre**:

Définition 4.13 Le message de $i \rightarrow j$ est passé après avoir reçu les messages de tous les voisins de i .

Corollaire 4.14 Noter que ceci implique de commencer le passage des messages par les feuilles de l'arbre (i.e., des sommets avec un seul voisin).

Remarque 4.15 Pour un arbre à n nœuds, on compte $2(n - 1)$ messages. En effet, il y a $n - 1$ arêtes et on passe dans les deux sens.

Construction d'un ordre de passage

Lorsque les messages sont passés en série, les ordres de passage respectant le protocole sont tous obtenus comme suit:

1. Définition d'une "racine" r (n'importe quel sommet)
2. Orientation du graphe
3. Choix d'un ordre topologique pour l'arbre orienté (sans perte de généralité, on considère que cet ordre est $(1, \dots, n)$).

Nous pouvons alors distinguer 2 phases:

- La collecte (flèche pointillée)

$$\begin{array}{rcl} n & \rightarrow & \pi_n \\ n - 1 & \rightarrow & \pi_{n-1} \\ & \vdots & \\ 2 & \rightarrow & \pi_2 = 1 \end{array}$$

- la distribution (flèche en tirets)

$$\begin{array}{rcl} \pi_2 & \rightarrow & 2 \\ & \vdots & \\ \pi_n & \rightarrow & n \end{array}$$

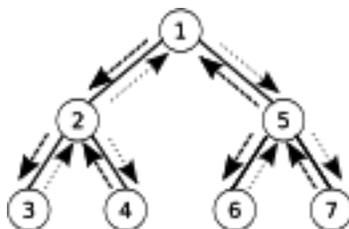


Figure 4.3. Collecte et Distribution des messages

Théorème 4.16 Si le “protocole” de passage des messages est respecté, une fois les $2(n-1)$ messages passés, nous avons:

- $\forall i, \quad u(x_i) = \psi_i(x_i) \prod_{k \in \mathcal{N}(i)} m_{kj}(x_i)$
- $\forall i, j \in E, \quad u(x_i, x_j) = \psi_i(x_i) \psi_j(x_j) \psi_{ij}(x_i, x_j) \prod_{k \in \mathcal{N}(i) \setminus j} m_{ki}(x_i) \prod_{k \in \mathcal{N}(j) \setminus i} m_{kj}(x_j)$

⚠ L’algorithme somme-produit n’est exact que pour les arbres. L’application aux graphes avec cycles (souvent appelée “loopy belief propagation”) pose beaucoup de problèmes de convergence même si elle est souvent utilisée (voir chapitre sur les méthodes approchées d’inférence).

4.5.1 Complexité totale

La complexité de l’algorithme de passage d’un message de $i \rightarrow j$ avec des variables prenant r valeurs vaut:

$$\mathcal{O}(r^2(\underbrace{\#\mathcal{N}(i)}_{d_i = \text{degré de } i} - 1))$$

On en déduit que la complexité totale est $O(r^2n)$, i.e., linéaire en n .

4.5.2 Algorithme max-produit

Dans le contexte du cours, celui d’un semi-anneau commutatif $(\mathbb{R}, +, \times)$ la distributivité $a.b + a.c = a(b + c)$ peut être ré-écrite $\max(a.b, a.c) = a \max(b, c)$ sur le semi-anneau commutatif $(\mathbb{R}_+, \max, \times)$.

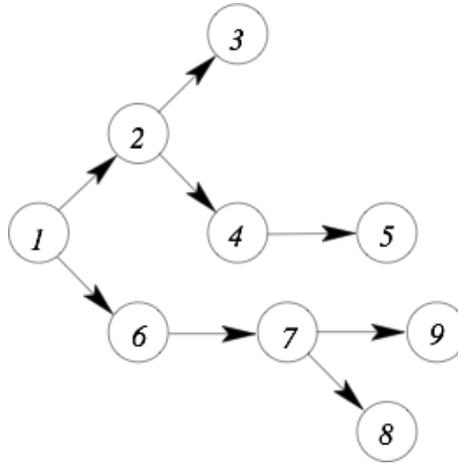
Du coup, toutes les procédures d’élimination et d’algorithmes somme-produit, qui ne sont basées que sur la distributivité s’appliquent. Par exemple, pour calculer $\max_x u(x)$, on fait passer les messages

$$m_{ij}(x_j) = \max_{x_i} \left(\psi_i(x_i) \psi_{ij}(x_i, x_j) \prod_{k \in \mathcal{N}(i) \setminus j} m_{ki}(x_i) \right)$$

4.6 Algorithme somme-produit - Preuve

On considère la procédure suivante (dite “en série”) pour l’ordre de passage des messages: on définit une racine en choisissant un des sommet (au hasard); cette racine permet de définir un unique arbre orienté.

On considère un ordre topologique quelconque (i.e., tel qu’un noeud apparaît toujours après ses parents), et on suppose que les noeuds sont étiquetés en suivant l’ordre topologique, i.e., $V = \{1, \dots, n\}$. Voir figure.



Les $2(n - 1)$ messages sont passés dans l’ordre suivant:

$$n \rightarrow \pi_n, n - 1 \rightarrow \pi_{n-1}, \dots, 2 \rightarrow \pi_2, \pi_2 \rightarrow 2, \dots, \pi_n \rightarrow n$$

Cette procédure respecte bien le protocole de passage des messages, i.e., un sommet n’envoie un message que quand il a reçu un message de tous ses autres voisins.

Nous allons démontrer qu’une fois les $2(n - 1)$ messages passés, les marginales sont égales à

$$\forall i \in V, u(x_i) = \psi_i(x_i) \prod_{k \in \mathcal{N}(i)} m_{k,i}(x_i), \quad (4.3)$$

$$\forall (i, j) \in E, u(x_i, x_j) = \psi_i(x_i) \psi_j(x_j) \psi_{ij}(x_i, x_j) \prod_{k \in \mathcal{N}(i) \setminus j} m_{k,i}(x_i) \prod_{k \in \mathcal{N}(j) \setminus i} m_{k,j}(x_j), \quad (4.4)$$

Le résultat se démontre par récurrence sur le nombre de sommets. L’hypothèse de récurrence est que pour tout arbre de taille n , toute famille de potentiels, toute racine, et tout ordre topologique correspondant, les équations (4.3) et (4.4) sont vraies.

4.6.1 $n = 2$

Si T a deux sommets 1 et 2, alors deux messages sont passés, de 1 vers 2: $m_{12}(x_2) = \sum_{x_1} \psi_1(x_1) \psi_{12}(x_1, x_2)$, puis de 2 vers 1: $m_{21}(x_1) = \sum_{x_2} \psi_2(x_2) \psi_{12}(x_1, x_2)$. D’autre part, on

a par définition

$$\begin{aligned} u(x_1) &= \sum_{x_2} \psi_1(x_1)\psi_2(x_2)\psi_{12}(x_1, x_2) \\ &= \psi_1(x_1) \sum_{x_2} \psi_2(x_2)\psi_{12}(x_1, x_2) \\ u(x_1) &= \psi_1(x_1)m_{21}(x_1), \end{aligned}$$

et de façon analogue, $u(x_2) = \psi(x_2)m_{12}(x_2)$. Ainsi, (4.3) est vérifiée.

Enfin, on a directement par définition $u(x_1, x_2) = \psi(x_1)\psi(x_2)\psi(x_1, x_2)$. D'où (4.4) est également vérifiée.

On a donc montré que le résultat est vrai pour $n = 2$.

4.6.2 $n - 1 \rightarrow n$

On suppose que le résultat est vrai pour les arbres de taille $n - 1$, et on considère un arbre de taille n , avec racine et ordre correspondant.

Comme le sommet n (dernier dans l'ordre topologique) n'a qu'un seul voisin π_n , le premier message passé, de n vers π_n est

$$m_{n\pi_n}(x_{\pi_n}) = \sum_{x_n} \psi_n(x_n)\psi_{n\pi_n}(x_n, x_{\pi_n})$$

Le dernier message passé, de π_n vers n est égal à:

$$m_{\pi_n n}(x_n) = \sum_{x_{\pi_n}} \psi_{\pi_n}(x_{\pi_n})\psi_{n\pi_n}(x_n, x_{\pi_n}) \prod_{k \in \mathcal{N}(\pi_n) \setminus \{n\}} m_{k\pi_n}(x_{\pi_n})$$

Nous allons construire un arbre \tilde{T} de taille $n - 1$, ainsi qu'une famille de potentiels, de telle sorte que les $2(n - 2)$ messages passés dans T (i.e., tous les messages exceptés le premier et le dernier) soient égaux aux $2(n - 2)$ messages passés dans \tilde{T} . On définit l'arbre et les potentiels comme suit:

- $\tilde{T} = (\tilde{V}, \tilde{E})$ avec $\tilde{V} = \{1, \dots, n - 1\}$ et $\tilde{E} = E \setminus \{n, \pi_n\}$ (i.e., c'est le sous-arbre correspondant aux $n - 1$ premiers sommets).
- Les potentiels sont tous les mêmes que ceux de T , excepté le potentiel $\tilde{\psi}_{\pi_n}(x_{\pi_n}) = \psi_{\pi_n}(x_{\pi_n})m_{n\pi_n}(x_{\pi_n})$.
- La racine est inchangée et l'ordre topologique est aussi conservé.

Le produit des potentiels de l'arbre de taille $n - 1$, est égale à:

$$\begin{aligned}
\tilde{u}(x_1, \dots, x_{n-1}) &= \prod_{i=1}^{n-1} \psi_i(x_i) \left(\prod_{(i,j) \in E \setminus \{n, \pi_n\}} \psi_{ij}(x_i, x_j) \right) m_{n\pi_n}(x_{\pi_n}) \\
&= \prod_{i=1}^{n-1} \psi_i(x_i) \left(\prod_{(i,j) \in E \setminus \{n, \pi_n\}} \psi_{ij}(x_i, x_j) \right) \sum_{x_n} \psi_n(x_n) \psi_{n\pi_n}(x_n, x_{\pi_n}) \\
&= \sum_{x_n} \prod_{i=1}^n \psi_i(x_i) \prod_{(i,j) \in E} \psi_{ij}(x_i, x_j) \\
&= \sum_{x_n} u(x)
\end{aligned}$$

et est donc égal à la marginalisation de $u(x)$ aux $n - 1$ premiers sommets.

Par construction, tous les messages passés dans \tilde{T} correspondent aux messages passés dans T (hormis le premier et le dernier). Nous montrons maintenant que les équations (4.3) et (4.4) sont vraies.

Cas 1 : $i \neq n, i \neq \pi_n$

Les messages correspondant à ces noeuds sont les mêmes dans les deux arbres T et \tilde{T} , et comme les marginales sont les mêmes (car \tilde{u} est elle-même la marginalisation de u sur les $n - 1$ premiers sommets), les équations (4.3) et (4.4) sont vraies par hypothèse de récurrence. En particulier, (4.3) est vraie pour tout $i \notin \{n, \pi_n\}$ et (4.4) est vraie pour tout $i \notin \{n, \pi_n\}$ et j voisin de i .

Cas 2 : $i = \pi_n$

Dans ce cas, par hypothèse de récurrence,

$$\begin{aligned}
\tilde{u}(x_{\pi_n}) &= \tilde{\psi}_{\pi_n}(x_{\pi_n}) \prod_{k \in \tilde{\mathcal{N}}(\pi_n)} m_{k\pi_n}(x_{\pi_n}) \quad (\text{produit sur les voisins de } \pi_n \text{ dans } \tilde{T}) \\
&= \tilde{\psi}_{\pi_n}(x_{\pi_n}) \prod_{k \in \mathcal{N}(\pi_n) \setminus \{n\}} m_{k\pi_n}(x_{\pi_n}) \\
&= \psi(x_{\pi_n}) m_{n\pi_n}(x_{\pi_n}) \prod_{k \in \mathcal{N}(\pi_n) \setminus \{n\}} m_{k\pi_n}(x_{\pi_n}) \quad \text{par définition de } \tilde{\psi}_{\pi_n} \\
&= \psi(x_{\pi_n}) \prod_{k \in \mathcal{N}(\pi_n)} m_{k\pi_n}(x_{\pi_n})
\end{aligned}$$

Comme $\tilde{u}(x_{\pi_n}) = u(x_{\pi_n})$, l'équation (4.3) est vérifiée.

Cas 3 : $i = n$

Il reste à montrer que les marginales sur n et (n, π_n) sont effectivement correctes. On a :

$$\begin{aligned} u(x_n, x_{\pi_n}) &= \sum_{\substack{x_i \\ i \neq n, i \neq \pi_n}} u(x) \\ &= \psi_n(x_n) \psi_{\pi_n}(x_{\pi_n}) \psi_{n\pi_n}(x_n, x_{\pi_n}) \underbrace{\sum_{\substack{x_i \\ i \neq n, i \neq \pi_n}} \frac{u(x)}{\psi_n(x_n) \psi_{\pi_n}(x_{\pi_n}) \psi_{n\pi_n}(x_n, x_{\pi_n})}}_{\alpha(x_{\pi_n})}, \end{aligned}$$

et par conséquent

$$\begin{aligned} u(x_{\pi_n}) &= \sum_{x_n} u(x_n, x_{\pi_n}) \\ &= \left(\sum_{x_n} \psi_n(x_n) \psi_{n\pi_n}(x_n, x_{\pi_n}) \right) \psi_{\pi_n}(x_{\pi_n}) \alpha(x_{\pi_n}) \\ &= m_{n\pi_n}(x_{\pi_n}) \psi_{\pi_n}(x_{\pi_n}) \alpha(x_{\pi_n}). \end{aligned}$$

On en déduit alors

$$\alpha(x_{\pi_n}) = \frac{1}{\psi_{\pi_n}(x_{\pi_n})} \frac{u(x_{\pi_n})}{m_{n\pi_n}(x_{\pi_n})}.$$

et

$$u(x_n, x_{\pi_n}) = \psi_n(x_n) \psi_{n\pi_n}(x_n, x_{\pi_n}) \frac{u(x_{\pi_n})}{m_{n\pi_n}(x_{\pi_n})}.$$

En utilisant le résultat montré pour le cas 2, on obtient :

$$\begin{aligned} u(x_n, x_{\pi_n}) &= \psi_n(x_n) \psi_{n\pi_n}(x_n, x_{\pi_n}) \frac{\psi_{\pi_n}(x_{\pi_n}) \prod_{k \in \mathcal{N}(\pi_n)} m_{k\pi_n}(x_{\pi_n})}{m_{n\pi_n}(x_{\pi_n})} \\ &= \psi_n(x_n) \psi_{\pi_n}(x_{\pi_n}) \psi_{n\pi_n}(x_n, x_{\pi_n}) \prod_{k \in \mathcal{N}(\pi_n) \setminus n} m_{k\pi_n}(x_{\pi_n}) \end{aligned}$$

ce qui montre le résultat pour la probabilité jointe sur x_n, x_{π_n} . En sommant par rapport à x_{π_n} , on obtient immédiatement le résultat pour $u(x_n)$:

$$\begin{aligned} u(x_n) &= \sum_{x_{\pi_n}} u(x_n, x_{\pi_n}) \\ &= \sum_{x_{\pi_n}} \psi_n(x_n) \psi_{\pi_n}(x_{\pi_n}) \psi_{n\pi_n}(x_n, x_{\pi_n}) \prod_{k \in \mathcal{N}(\pi_n) \setminus n} m_{k\pi_n}(x_{\pi_n}) \\ &= \psi_n(x_n) \sum_{x_{\pi_n}} \psi_{\pi_n}(x_{\pi_n}) \psi_{n\pi_n}(x_n, x_{\pi_n}) \prod_{k \in \mathcal{N}(\pi_n) \setminus n} m_{k\pi_n}(x_{\pi_n}) \\ &= \psi_n(x_n) m_{\pi_n n}(x_n) \text{ par définition de } m_{\pi_n n} \end{aligned}$$