



**Master MSI**  
**Année universitaire 2005/2006**

# **Modélisation markovienne en imagerie**

**Vincent Barra**

**Institut Supérieur d'Informatique, de Modélisation et de leurs  
Applications**  
**Campus des Cézeaux - B.P. 1025 - 63173 AUBIERE CEDEX**

# Table des matières

<b>1</b>	<b>L'image</b>	<b>2</b>
<b>2</b>	<b>Modélisation probabiliste de l'image</b>	<b>3</b>
<b>3</b>	<b>Champs de Markov et de Gibbs</b>	<b>3</b>
<b>4</b>	<b>Echantillonnage des processus de Markov</b>	<b>5</b>
4.1	Première approche : échantillonneur de Gibbs . . . . .	5
4.2	Seconde approche : algorithme de Métropolis . . . . .	6
<b>5</b>	<b>L'algorithme du recuit simulé</b>	<b>6</b>
5.1	Distribution de Gibbs avec température . . . . .	7
5.2	Algorithme du recuit simulé . . . . .	7
5.3	Algorithme Iterative Conditional Mode . . . . .	8
<b>6</b>	<b>Exemples de champs de Markov</b>	<b>9</b>
6.1	Modèle d'Ising . . . . .	9
6.2	Modèle de Potts . . . . .	9
6.3	Modèle gaussien . . . . .	9
<b>7</b>	<b>Quelques applications en traitement d'images</b>	<b>10</b>
7.1	Spécification de $\mathcal{U}(x   y)$ pour le problème de restauration d'images	11
7.2	Spécification de $\mathcal{U}(x   y)$ pour le problème de segmentation d'images	12
<b>8</b>	<b>Estimateurs dans un cadre markovien</b>	<b>13</b>
8.1	Introduction . . . . .	13
8.2	Modélisation bayésienne et fonction de coût . . . . .	13
8.3	Estimateur MAP . . . . .	14
8.4	Estimateur MPM . . . . .	14
8.5	Estimateur TPM . . . . .	15
8.6	Comparaison . . . . .	15
<b>9</b>	<b>Estimation de paramètres</b>	<b>16</b>
9.1	Introduction . . . . .	16
9.2	Données complètes . . . . .	17
9.2.1	Méthode des codages . . . . .	17
9.2.2	Pseudo vraisemblance . . . . .	18
9.2.3	Algorithme du gradient stochastique . . . . .	19
9.3	Données incomplètes . . . . .	20
9.3.1	Position du problème . . . . .	20

9.3.2	Estimation des paramètres d'attache aux données . . . . .	20
9.3.3	Algorithme EM . . . . .	21
<b>10</b>	<b>Processus de bords</b>	<b>22</b>
10.1	Processus implicites et explicites . . . . .	23
10.2	Algorithmes de minimisation associés . . . . .	24
<b>11</b>	<b>Quelques illustrations</b>	<b>24</b>
11.1	Segmentation . . . . .	24
11.1.1	Imagerie radar . . . . .	24
11.1.2	Imagerie sismique . . . . .	25
11.1.3	Imagerie de test . . . . .	26
11.2	Restauration . . . . .	27
11.2.1	Imagerie de test . . . . .	27
11.2.2	Restauration avec prise en compte de processus de bords .	28

L'information véhiculée par une image est portée par bien d'autres données que les seuls niveaux de gris. En effet, une image représente une suite d'objets dans un espace 3D et des relations spatiales évidentes existent entre chaque site de résolution de l'image (pixel). Le niveau de gris d'un pixel n'est donc souvent pas significatif en lui-même, mais dans ses relations avec ses voisins. Cette notion de relation de voisinage va nous permettre d'introduire un contexte markovien pour l'analyse d'images, dans des domaines qui vont de la segmentation d'objets à la restauration d'images bruitées. Dans la suite, un processus markovien sera indifféremment dénoté comme tel, ou comme un **champ de Markov**, ou par son abréviation anglaise **MRF** (Markov Random Field).

## 1 L'image

L'image est formée d'un ensemble fini  $S$  de sites  $s$  correspondant aux pixels, aux groupes de pixels, aux régions dans l'image ou à tout autre attribut haut niveau.  $S$  est donc essentiellement un réseau discret, fini de  $\mathbb{Z}^2$  ou  $\mathbb{Z}^3$  (même  $\mathbb{Z}^4$  si l'on considère des volumes évoluant dans le temps), muni d'une topologie  $V$  définie par :

$$V_s = t \text{ tels que : } \begin{cases} s \notin V_s \\ t \in V_s \Leftrightarrow s \in V_t \end{cases}$$

A partir de cette topologie, un système de **cliques** peut être défini : une clique est soit un singleton de  $S$ , soit un ensemble de sites tous voisins au sens de  $V$ . L'ensemble des cliques relatif à  $V$  est noté  $C$ , et l'ensemble des cliques de cardinal  $k$ ,

$C_k$ .

Enfin, à chaque site  $s$  est associé un **descripteur**  $x_s$ , représentant l'état du site : si dans la plupart des cas il s'agit du niveau de gris, un pixel peut néanmoins être caractérisé par un vecteur quelconque, prenant ses valeurs dans un espace  $E$ . Les interactions locales entre descripteurs de sites voisins s'expriment alors comme un **potentiel de clique**. Pour  $c \in C$ , un potentiel  $U_c$  est associé à  $c$ , dont la valeur dépend des descripteurs des pixels constituant la clique. De même, le **potentiel de l'image** est défini comme la somme des potentiels de toutes les cliques :

$$U = \sum_{c \in C} U_c$$

et le **potentiel local** en un site est calculé comme la somme des potentiels de toutes les cliques auxquelles il appartient :

$$U_s = \sum_{c \in C, s \in c} U_c$$

## 2 Modélisation probabiliste de l'image

A chaque site  $s$  on associe une variable aléatoire  $X_s$  prenant ses valeurs dans  $E$ . Le descripteur  $x_s$  est une réalisation de  $X_s$ . On définit alors un processus aléatoire  $X = (X_1, \dots, X_{Card(S)})$  prenant ses valeurs dans  $\Omega = E^{Card(S)}$ . L'image est alors une réalisation de  $X$ , et la probabilité  $P(X = x)$  quantifie la vraisemblance de l'image, les probabilités conditionnelles locales d'une valeur en un site permettant quant à elles de mesurer le lien statistique entre un descripteur et le reste de l'image. L'hypothèse markovienne va permettre d'évaluer ces quantités.

## 3 Champs de Markov et de Gibbs

Sous l'hypothèse markovienne, on a

$$P(X_s = x_s \mid x_r, r \neq s) = P(X_s = x_s \mid x_r, r \in V_s)$$

Autrement dit, la probabilité d'observer un descripteur en  $s$  ne dépend pas de toutes les valeurs des descripteurs de l'image, mais seulement de ceux situés dans le voisinage de  $s$ , ce qui intuitivement est souvent satisfaisant lorsqu'on parle de formation d'une image numérique. Cette hypothèse va permettre, *via* le théorème d'Hammersley-Clifford, d'accéder aux expressions de ces probabilités conditionnelles locales. Il faut pour cela introduire la notion de Champ de Gibbs.

**Définition 1.** La mesure de Gibbs d'hamiltonien  $U : \Omega \rightarrow \mathbb{R}$  est la probabilité  $P$  définie sur  $\Omega$  par :

$$P(X = x) = \frac{e^{-U(x)}}{\sum_{y \in \Omega} e^{-U(y)}}$$

avec

$$U(x) = \sum_{c \in C} U_c(x)$$

Le dénominateur est appelé **fonction de partition** de Gibbs, et permet de normaliser la probabilité. Il est cependant quasi impossible à calculer, même pour des configurations simples :  $\text{Card}(\Omega) = 2^{262144}$  par exemple pour une image simple  $512 \times 512$  à deux niveaux de gris !

**Définition 2.** Le **champ de Gibbs** de potentiel associé au système de voisinage  $V$  est le processus aléatoire  $X$  dont la probabilité est une mesure de Gibbs associée à  $V$

L'énergie totale d'un champ de Gibbs se décompose donc sous forme d'une somme d'énergies locales.

Le lien entre champ de Gibbs et processus Markovien est effectué par le théorème fondamental suivant :

**Théorème 1.** : *théorème d'Hammersley-Clifford*

Soient  $S$  un ensemble fini ou dénombrable,  $V$  une topologie sur  $S$  et  $E$  un espace d'états discret. Soit  $X$  un processus aléatoire à valeurs dans  $E^{\text{Card}(S)}$  alors :

$X$  est un champ de Markov relativement à  $V$  et  $P(X = x) > 0$  pour tout  $x \in \Omega$  si et seulement si  $X$  est un champ de Gibbs de potentiel associé à  $V$

Ainsi, la probabilité conditionnelle  $P(x_s \mid X_r = x_r, r \neq s)$  s'écrit :

$$P(X_s = x_s \mid X_r = x_r, r \neq s) = \frac{P(X = x)}{P(X_r = x_r, r \neq s)} = \frac{e^{-U(x_s, x_r, r \neq s)}}{\sum_{e \in E} e^{-U(e, x_r, r \neq s)}}$$

En définissant l'énergie locale du site  $s$  par :

$$U_s(X_s = x_s \mid X_r = x_r, r \in V_s) = \sum_{c \in C/s \in C} U_c(x_s, X_r = x_r, r \in V_s)$$

on remarque que  $U_s$  ne fait intervenir que les voisins de  $s$ , et l'énergie globale  $U(x)$  s'écrit alors :

$$\begin{aligned} U(x) &= \sum_{c \in C/s \notin C} U_c(x) + \sum_{c \in C/s \in C} U_c(x) \\ &= \sum_{c \in C/s \notin C} U_c(x) + U_s(X_s = x_s \mid X_r = x_r, r \in V_s) \end{aligned}$$

et en passant aux champs de Gibbs :

$$P(X_s = x_s \mid X_r = x_r, r \neq s) = \frac{e^{-U_s(X_s=x_s \mid X_r=x_r, r \in V_s)}}{\sum_{e \in E} e^{-U_s(X_s=e \mid X_r=x_r, r \in V_s)}}$$

L'expression obtenue ne fait donc intervenir que les potentiels de cliques dans lesquelles est impliqué  $s$  (où au passage on retrouve l'hypothèse markovienne), et le calcul du dénominateur de la fraction devient maintenant possible, contrairement à l'expression générale de la fonction de partition de Gibbs.

## 4 Echantillonnage des processus de Markov

Ainsi, il suffit de définir séquentiellement une topologie, un système de cliques, des potentiels de clique  $U_c$  et une fonction d'énergie  $U$  pour préciser le processus de Markov attaché à l'image. L'ensemble de ces données permet, comme précisé précédemment, d'accéder à la probabilité d'une configuration (*i.e.* pour le traitement d'images, le fait d'observer l'image que l'on voit), et aux probabilités conditionnelles locales.

Se pose donc maintenant le problème de la réalisation du tirage d'une configuration donnée suivant la loi de Gibbs caractéristique (*via* le théorème d'Hammersley-Clifford) du champ de Markov ainsi défini.

### 4.1 Première approche : échantillonneur de Gibbs

**L'échantillonneur de Gibbs** est un algorithme de relaxation probabiliste, qui se propose de réaliser ce tirage par une méthode itérative, convergeant vers une configuration suivant la loi de Gibbs.

A l'itération  $n$ , la construction de la réalisation courante se fait à l'aide des étapes suivantes :

- (i) Choix d'un site  $s$  (par tirage selon une loi uniforme sur  $E$ )
- (ii) Calcul de la probabilité conditionnelle

$$P(X_s = x_s \mid X_r = x_r, r \in V_s) = \frac{e^{-U(x_s, x_r, r \in V_s)}}{\sum_{e \in E} e^{-U(e, x_r, r \in V_s)}}$$

suivant la configuration des sites  $r \in V_s$  à l'itération précédente

- (iii) mise à jour du site  $s$  par tirage aléatoire selon  $P(X_s = x_s \mid X_r = x_r, r \neq s)$

**Algorithme 1:** Echantillonneur de Gibbs

Dans le cas où chaque site  $s$  est visité une infinité de fois, on peut montrer que les réalisations générées sont des réalisations de la loi de Gibbs, indépendamment de la configuration initiale utilisée pour l'algorithme. En pratique, on fera converger l'algorithme en itérant le procédé "un nombre suffisant de fois", ou en détectant le faible nombre de changement dans les mises à jour de sites d'une itération à l'autre.

## 4.2 Seconde approche : algorithme de Métropolis

L'algorithme de Métropolis, issu de la physique statistique, est également un algorithme de relaxation probabiliste. Le principe est donc itératif, et repose sur les étapes suivantes :

- (i) Choix d'un site  $s$  (par tirage selon une loi uniforme sur  $E$ )
- (ii) Tirage aléatoire d'un descripteur  $\delta \in E$  selon une loi uniforme
- (iii) Calcul de la variation d'énergie induite par le changement d'état du site  $s$  (de son état à l'itération précédente à l'état  $\delta$ ) :

$$\Delta U = U_s(X_s = \delta \mid X_r = x_r^{n-1}, r \in V_s) - U_s(X_s = x_s^{n-1} \mid X_r = x_r^{n-1}, r \in V_s)$$

**Si** ( $\Delta U < 0$ ) **Alors**

    | Le changement d'état est accepté pour le site  $s$

**Sinon**

    | Le changement est accepté avec une probabilité  $e^{-\Delta U}$

**Fin Si**

### Algorithme 2: Algorithme de Metropolis

Les critères de convergence sont similaires à ceux de l'échantillonneur de Gibbs, mais cette dernière intervient cependant plus tardivement car les transitions ne sont ici pas toujours acceptées. En revanche, l'itération est plus rapide que celle de l'échantillonneur de Gibbs, car la variation d'énergie ne se calcule qu'entre deux configurations données.

## 5 L'algorithme du recuit simulé

Dans les approches précédentes, chaque itération donnait accès à une réalisation, qui convergeait vers une configuration suivant la loi de Gibbs. Dans ce paragraphe, nous nous intéressons à la recherche des configurations les plus probables qui correspondent à des états d'énergie minimale.

## 5.1 Distribution de Gibbs avec température

**Définition 3.** Une probabilité de Gibbs avec paramètre de température  $T > 0$  est une loi de probabilité de la forme :

$$P_T(X = x) = \frac{e^{-\frac{U(x)}{T}}}{\sum_{y \in \Omega} e^{-\frac{U(y)}{T}}}$$

Lorsque la température tend vers l'infini, il est facile de voir que  $P_T$  converge vers la loi uniforme sur  $\Omega$ , et donc toutes les configurations sont équiprobables.

Lorsque  $T$  tend vers 0, on note  $\Omega^* = \{x_1 \cdots x_k\}$  l'ensemble des configurations atteignant l'énergie minimale  $U^*$ . Alors :

$$\begin{aligned} P_T(X = x) &= \frac{e^{-\frac{U(x)}{T}}}{\sum_{y \in \Omega} e^{-\frac{U(y)}{T}}} \\ &= \frac{e^{-\frac{U(x) - U^*}{T}}}{\sum_{y \in \Omega} e^{-\frac{U(y) - U^*}{T}}} \\ &= \frac{e^{-\frac{U(x) - U^*}{T}}}{\sum_{y \notin \Omega^*} e^{-\frac{U(y) - U^*}{T}} + \sum_{y \in \Omega^*} 1} \end{aligned}$$

et deux cas se présentent alors :

- s'il existe  $i \in \{1 \cdots k\}$  tel que  $x = x_i$ ,  $P_T(x) = \frac{1}{k}$  ( il y a une infinité de termes qui tendent vers 0 au dénominateur)
- sinon,  $U(x) - U^* > 0$ , l'exponentielle tend vers 0 lorsque  $T$  tend vers 0 et  $P_T$  aussi.

Lorsque la température est nulle,  $P_T$  est donc uniformément distribuée sur les configurations d'énergie minimale, et donc sur les configurations les plus probables.

## 5.2 Algorithme du recuit simulé

L'algorithme du **recuit simulé** exploite ce résultat. C'est une méthode itérative qui procède, après avoir choisi une température initiale  $T^0$  et une configuration initiale  $x^0$ , selon les étapes suivantes :



- (i) Simulation d'une configuration  $x^n$  pour la loi de Gibbs d'énergie  $\frac{U(x)}{T}$  à partir de la configuration  $(n - 1)$ .  
(ii) Mise à jour de la température, (e.g. théorème d'Hajek :

$$T^n > \frac{c}{\log(2 + n)}$$

,  $c$  constante dépendant de la variation énergétique globale maximale sur  $\Omega$ .

**Algorithme 3:** Algorithme du recuit simulé

La convergence est réalisée lorsque le nombre de changements entre deux configurations est faible. Contrairement aux algorithmes de l'échantillonneur de Gibbs et de Métropolis qui échantillonnent selon la loi de Gibbs et qui sont en mesure de donner toutes les configurations possibles, les images obtenues par recuit simulé sont uniques et doivent en théorie correspondre aux minima globaux de l'énergie.

### 5.3 Algorithme Iterative Conditional Mode

Si l'algorithme du recuit simulé permet d'atteindre un optimum global, il peut être coûteux en temps de calcul car il nécessite la génération d'un grand nombre de configurations au fur et à mesure des itérations. Des algorithmes sous-optimaux ont donc été proposés pour accélérer la convergence, et c'est le cas par exemple de l'algorithme des **modes conditionnels itérés**.

Le principe est toujours itératif, mais cette fois-ci la modification du descripteur  $x_s$  du  $s$  se fait de manière déterministe.

On recherche ici une approximation du maximum *a posteriori*  $\hat{x}$ , résultat de la convergence des itérations. Pour une itération  $k$  donnée, le déroulement de la mise à jour de l'ensemble d'une configuration se fait, pour tout site  $s$ , en mettant à jour  $x_s$  par  $\text{Arg max}_{e \in E} P(X_s = e \mid \hat{x}_r(k), r \in V_s)$ . L'algorithme converge lorsque le nombre de changements d'états des sites est faible (par exemple en pourcentage de  $\text{Card}(S)$ ).

Si on montre que l'énergie globale de  $\hat{x}$  diminue, on ne peut affirmer que la convergence aboutit à un optimum global, l'algorithme dépendant fortement de son initialisation.

## 6 Exemples de champs de Markov

### 6.1 Modèle d'Ising

L'espace d'état est un espace binaire  $\{-1, 1\}$ , et la topologie est indifféremment la 4- ou la 8-connexité dans une image 2D. Les potentiels de cliques d'ordre 1 (un seul site) sont de la forme  $-Bx_s$ , et les potentiels d'ordre 2 sont définis par :

$$U_{c=\{s,t\}}(x_s, x_t) = -\beta x_s x_t = \begin{cases} -\beta & \text{si } x_s = x_t \\ \beta & \text{sinon} \end{cases}$$

, où  $\beta$  est la constante de couplage entre  $s$  et  $t$ . L'énergie globale d'une configuration est donc

$$U(x) = - \sum_{c=\{s,t\}} \beta x_s x_t - \sum_{s \in S} Bx_s$$

$\beta$  régularise le modèle : lorsque  $\beta > 0$ , les configurations d'énergie minimale sont celles pour lesquelles les sites ont le même état, et si  $\beta < 0$  les configurations privilégiées sont celles où les sites sont d'états opposés.

### 6.2 Modèle de Potts

L'espace d'état est un espace de  $n$  valeurs  $\{1, \dots, n\}$ , et la topologie est indifféremment la 4- ou la 8-connexité dans une image 2D. Seuls les potentiels d'ordre 2 sont définis par exemple par :

$$U_{c=\{s,t\}}(x_s, x_t) = -\beta x_s x_t = \begin{cases} -\beta & \text{si } x_s = x_t \\ \beta & \text{sinon} \end{cases}$$

Lorsque  $\beta > 0$ , les configurations les plus probables correspondent à des sites voisins ayant des descripteurs égaux, ce qui dans le cas du traitement d'image donne des réalisations avec de grandes zones homogènes (au sens de la similarité entre descripteurs). La taille de ces régions est gouvernée par la valeur de la constante de couplage.

Il est possible de raffiner ce modèle en considérant des valeurs de  $\beta$  différentes suivant les directions explorées dans la clique (vertical/horizontal en 4-connexité par exemple), et donc de privilégier la détection de zones homogènes directionnelles.

### 6.3 Modèle gaussien

Là encore, l'espace d'état est un espace de  $n$  valeurs  $\{1, \dots, n\}$ , et la topologie est indifféremment la 4- ou la 8-connexité dans une image 2D. Les potentiels d'ordre

1 sont de la forme  $\alpha(x_s - \mu_s)$ , où  $\mu_s$  est une donnée extérieure qui, dans le cas du traitement d'images, peut être une moyenne de niveaux de gris attendue en  $s$  lorsque les descripteurs sont les niveaux de gris. Les potentiels d'ordre 2 sont quant à eux donnés par  $\beta(x_s - x_t)$ , et régularisent les configurations, favorisant les faibles différences de descripteurs entre  $s$  et  $t$  lorsque  $\beta > 0$ . Le rapport  $\frac{\alpha}{\beta}$  pondère les influences respectives des termes de régularisation et d'attache aux données, les valeurs absolues de ces paramètres décrivant le caractère équiréparti ou localisé de la distribution.

## 7 Quelques applications en traitement d'images

Dans la suite,  $y$  est l'image observée, considérée comme la réalisation d'un processus aléatoire  $Y$ . Le problème posé ici est de trouver une réalisation  $x$ , modélisée par un processus markovien  $X$ , correspondant à :

- une image correspondant à  $y$ , dans laquelle le bruit a été corrigé : c'est le problème de **restauration** d'images
- une image d'étiquettes, chaque étiquette correspondant à un objet présent dans  $y$  : c'est le problème de **segmentation** d'images

Etant donné que l'on cherche à remonter à une image idéale de la scène (étiquetée ou débruitée) connaissant seulement une observation  $y$  de cette scène, le problème se résout à l'aide de **chaînes de Markov cachées**.

Dans un cadre bayésien, on peut rechercher la configuration  $\hat{x}$  maximisant la probabilité  $P(X = x \mid Y = y)$ , qui s'écrit :

$$P(X = x \mid Y = y) = \frac{P(Y = y \mid X = x)P(X = x)}{P(Y = y)}$$

- Le premier terme du numérateur décrit la probabilité d'observer  $y$ , sachant que l'image idéale est  $x$  : il modélise donc l'acquisition de l'image. Sous l'hypothèse (pas toujours vérifiée) d'indépendance des pixels,

$$P(Y = y \mid X = x) = \prod_{s \in S} P(Y_s = y_s \mid X_s = x_s)$$

- le second terme du numérateur décrit la probabilité d'existence de l'image idéale  $x$ , qui dans le cadre qui nous intéresse, répond à l'hypothèse markovienne selon un système de voisinage  $V$  et un modèle  $\phi$  dépendant de l'application (restauration ou segmentation) :

$$P(X = x) = \frac{e^{-U(x)}}{\sum_{z \in \Omega} e^{-U(z)}}$$

– le dénominateur est constant, et en particulier indépendant de  $x$   
Ainsi,

$$\begin{aligned} P(X = x \mid Y = y) &= KP(Y = y \mid X = x)P(X = x) \\ &= Ke^{\ln(P(Y=y|X=x))-U(x)} \\ &= Ke^{-\mathcal{U}(x|y)} \end{aligned}$$

où

$$\mathcal{U}(x \mid y) = - \sum_{s \in S} \ln(P(Y_s = y_s \mid X_s = x_s)) + \sum_{c \in C} U_c(x)$$

La distribution  $P(X = x \mid Y = y)$  est donc une distribution de Gibbs, et par le théorème d’Hammersley-Clifford, le champ  $X$  conditionnellement à  $y$  est un champ de Markov. Il est donc possible de simuler ce processus, par exemple à l’aide du recuit simulé qui permet d’aboutir à  $\hat{x}$ , qui maximise la probabilité *a posteriori* (ou de manière équivalente qui minimise  $\mathcal{U}(x \mid y)$ ).

## 7.1 Spécification de $\mathcal{U}(x \mid y)$ pour le problème de restauration d’images

En supposant que la chaîne d’acquisition de l’image introduit un bruit blanc gaussien de variance  $\sigma^2$ , la probabilité d’observer en chaque site  $s$  le descripteur  $y_s$  (supposé ici être le niveau de gris), sachant que le descripteur idéal est  $x_s$  est :

$$P(Y_s = y_s \mid X_s = x_s) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x_s - y_s)^2}{2\sigma^2}}$$

En faisant l’hypothèse que  $X$  est un processus markovien, la probabilité  $P(X = x)$  peut être contrainte par des hypothèses de régularité, et avec un modèle de potentiel d’ordre 2 :

$$P(X = x) = \frac{e^{-\beta \sum_{s,t \in C_2} \phi(x_s, x_t)}}{e^{-\beta \sum_{z \in \Omega, s,t \in C_2} \phi(z_s, z_t)}}$$

$\mathcal{U}(x \mid y)$  vaut alors :

$$\mathcal{U}(x \mid y) = \sum_{s \in S} \frac{(x_s - y_s)^2}{2\sigma^2} + \beta \sum_{s,t \in C_2} \phi(x_s, x_t)$$

Le champ  $X$  conditionné par  $y$  est un champ de Gibbs, pour le même système de voisinage que  $X$ . La constante  $\beta$  pondère le terme d’attache aux données qui

impose des descripteurs  $x_s$  proches de  $y_s$  par rapport au terme regularisation (utilisation des cliques d'ordre 2) qui impose lui une image constituée de larges zones homogènes.

## 7.2 Spécification de $\mathcal{U}(x \mid y)$ pour le problème de segmentation d'images

Dans ce cas,  $X$  est défini sur l'espace  $E$  des étiquettes, et  $Y$  sur l'espace de ses descripteurs. Le terme  $P(Y = y \mid X = x)$  traduit donc la probabilité d'observation de l'image  $y$  connaissant l'appartenance de chaque pixel à un objet présent dans la scène. En supposant l'indépendance de chaque site, et en supposant que  $y_s$  ne dépend que de  $x_s$ , on peut écrire :

$$P(Y = y \mid X = x) = \prod_{s \in S} P(Y_s = y_s \mid X_s = x_s)$$

Les valeurs de probabilité conditionnelles sont par exemple données par la fréquence d'observation des niveaux de gris pour une classe donnée, et si chaque classe  $k$  suit une distribution gaussienne de moyenne  $m_k$  et d'écart-type  $\sigma_k$  :

$$P(Y_s = y_s \mid x_s = k) = \frac{1}{\sqrt{2\pi\sigma_k^2}} e^{-\frac{(y_s - m_k)^2}{2\sigma_k^2}}$$

Si on fait encore une fois l'hypothèse markovienne sur  $X$  et si on se restreint aux cliques d'ordre 2

$$P(X = x) = \frac{e^{-\beta \sum_{s,t \in C_2} \phi(x_s, x_t)}}{e^{-\beta \sum_{z \in \Omega, s,t \in C_2} \phi(z_s, z_t)}}$$

et donc :

$$\mathcal{U}(x \mid y) = \sum_{s \in S} \frac{(x_s - m_{x_s})^2}{2\sigma_{x_s}^2} + \log(2\pi) \sigma_{x_s} + \beta \sum_{s,t \in C_2} \phi(x_s, x_t)$$

Le terme d'ordre 1 exprime l'attache aux données, et le terme d'ordre 2 régularise l'image. Souvent  $\phi$  est le modèle de Potts, pour favoriser les grandes régions homogènes dans l'image.

## 8 Estimateurs dans un cadre markovien

### 8.1 Introduction

Nous avons vu précédemment comment il était possible d'utiliser le formalisme markovien à des fins de restauration et de segmentation. On se situe alors dans le cadre de données incomplètes (on parle aussi de champs de Markov cachés) car la réalisation dont on dispose est une réalisation bruitée (ou plus généralement vue au travers du système d'acquisition) du champ de Markov originel. En notant  $Y$  le champ dont on observe une réalisation, et  $X$  le champ initial, l'objectif est alors d'obtenir la meilleure réalisation  $\hat{x}$  de  $X$  connaissant l'observation  $y$ , autrement dit, reconstruire  $x$  de manière optimale vis-à-vis d'un certain critère. Dans le précédent paragraphe, nous nous étions intéressés à la réalisation maximisant la probabilité *a posteriori*  $P(X = x|Y = y)$  et nous avons vu que le recuit simulé permettait d'accéder à cette réalisation. D'autres choix sont possibles, auxquels correspondent d'autres méthodes de résolution.

### 8.2 Modélisation bayésienne et fonction de coût

En appliquant la règle de Bayes,

$$P(X = x|Y = y) = \frac{P(Y = y|X = x)P(X = x)}{P(Y = y)}$$

Sous certaines hypothèses, la distribution *a posteriori* est une distribution de Gibbs et le champ  $X$  conditionnellement à la donnée  $y$  est markovien. Cette propriété n'est pas nécessaire pour les notions suivantes, mais elle sera primordiale pour les algorithmes de résolution.

Le problème est alors de déterminer une estimation  $\hat{x} = \Phi(y)$  optimisant un certain critère. L'estimation bayésienne construit une fonction  $L$ , estimant le coût de remplacer  $x$  par  $\Phi(y)$ , qui vérifie :

- $(\forall x, x') L(x, x') \geq 0$
- $L(x, x') = 0 \Rightarrow x = x'$

L'estimateur optimal, *i.e.* la fonction  $\Phi$  optimale est alors la fonction minimisant

$$\mathbb{E}(L(X, \Phi(y))|Y = y) = \sum L(x, \Phi(y))P(x|y)$$

Suivant les fonctions de coût envisagées, on obtient différents estimateurs et différentes méthodes de résolution associées.

### 8.3 Estimateur MAP

Si  $L(x, x') = 1_{\{x \neq x'\}}$ ,  $L$  pénalise toute différence entre deux configurations, et ce, quel que soit le nombre de sites en lesquels elles diffèrent. Ainsi

$$\begin{aligned}\mathbb{E}(L(X, \Phi(y))|Y = y) &= \sum L(x, \Phi(y))P(x|y) \\ &= 1 - P(X = \Phi(y)|y)\end{aligned}$$

et la fonction  $\Phi$  minimisant l'espérance de la fonction de coût est celle qui maximise la probabilité *a posteriori*. Il faut donc trouver  $\hat{x}$ , fonction de  $y$ , maximisant  $P(X|y)$ . On parle de l'estimateur MAP (maximum *a posteriori*) ou de maximum de vraisemblance *a posteriori*. Les solutions algorithmiques associées à cet estimateur sont le recuit simulé et l'ICM, utilisés avec la distribution *a posteriori* avec paramètre de température.

### 8.4 Estimateur MPM

Si  $L(x, x') = \sum_{s \in S} L(x_s, x'_s) = \sum_{s \in S} 1_{\{x_s \neq x'_s\}}$ ,  $L$  pénalise une configuration proportionnellement au nombre de différences entre deux configurations. Elle paraît donc plus naturelle que la fonction précédente. On montre alors que :

$$\begin{aligned}\mathbb{E}(L(X, \Phi(y))|Y = y) &= \sum L(x, \Phi(y))P(x|y) \\ &= \sum_{s \in S} \mathbb{E}(L(X_s, \Phi(y)_s|y))\end{aligned}$$

On passe donc de la probabilité conditionnelle globale d'une configuration à la probabilité conditionnelle en un site. Il s'agit d'une somme de termes positifs, et par conséquent la fonction  $\Phi$  optimale minimise en chaque site l'espérance conditionnelle du coût local  $\mathbb{E}(L(X_s, \Phi(y)_s|y))$ . Ce résultat est valable pour toutes les fonctions de coût définies par une somme de coûts en chaque site.

Comme précédemment, on obtient des estimateurs du maximum *a posteriori* locaux, à calculer en chaque pixel contrairement à la recherche précédente qui était globale. L'estimateur est appelé maximum de vraisemblance *a posteriori* local ou maximum posterior marginal (pour maximum *a posteriori* de la marginale) abrégé en MPM.

D'un point de vue algorithmique, la taille de l'espace des configurations ne permet pas un calcul direct des quantités  $P(x_s|y)$ . Aussi réalise-t-on en pratique des approximations de type Monte-Carlo. En effet, supposons que l'on soit capable de tirer des réalisations de  $X$  selon sa loi conditionnelle à  $y$ , et notons les  $x_1 \cdots x_n$ . Il est alors possible de calculer une approximation de l'estimateur MPM. Le tirage des réalisations ne pose quant à lui pas de problème particulier (échantillonneur de Gibbs et algorithme de Métropolis).

## 8.5 Estimateur TPM

Si  $L(x, x') = \sum_{s \in S} (x_s - x'_s)^2$ ,  $L$  est l'erreur quadratique et pénalise la somme

des carrés des différences entre les deux configurations. Elle peut donc être plus adaptée dans certains cas que les précédentes, puisqu'elle tient compte non seulement du nombre de différences comme le MPM, mais aussi de leurs valeurs.

On montre alors que le minimum de  $L$  est atteint pour  $\Phi$  telle que  $\Phi(y)_s = \mathbb{E}(X_s|y)$ , Cet estimateur consistant alors à prendre en chaque site la moyenne conditionnelle locale donnée par la loi *a posteriori*, d'où le nom de TPM (Thresholded Posteriori Mean).

D'un point de vue algorithmique, la démarche est similaire à celle effectuée dans le paragraphe précédent pour le MPM. On approche l'espérance conditionnelle en chaque site par la moyenne empirique en ce site de  $N$  échantillons tirés selon la loi *a posteriori*.

## 8.6 Comparaison

Ces trois estimateurs sont ici comparés dans le cadre de la restauration. Dans le cas du MAP, les résultats sont obtenus par ICM et par recuit simulé. L'image à restaurer est une image bruitée par un bruit blanc gaussien. L'énergie *a posteriori* utilisée s'écrit :

$$\mathcal{U}(x | y) = \sum_{s \in S} \frac{(x_s - y_s)^2}{\sigma^2} + \beta \sum_{s, t \in C_2} \phi(x_s - x_t)$$

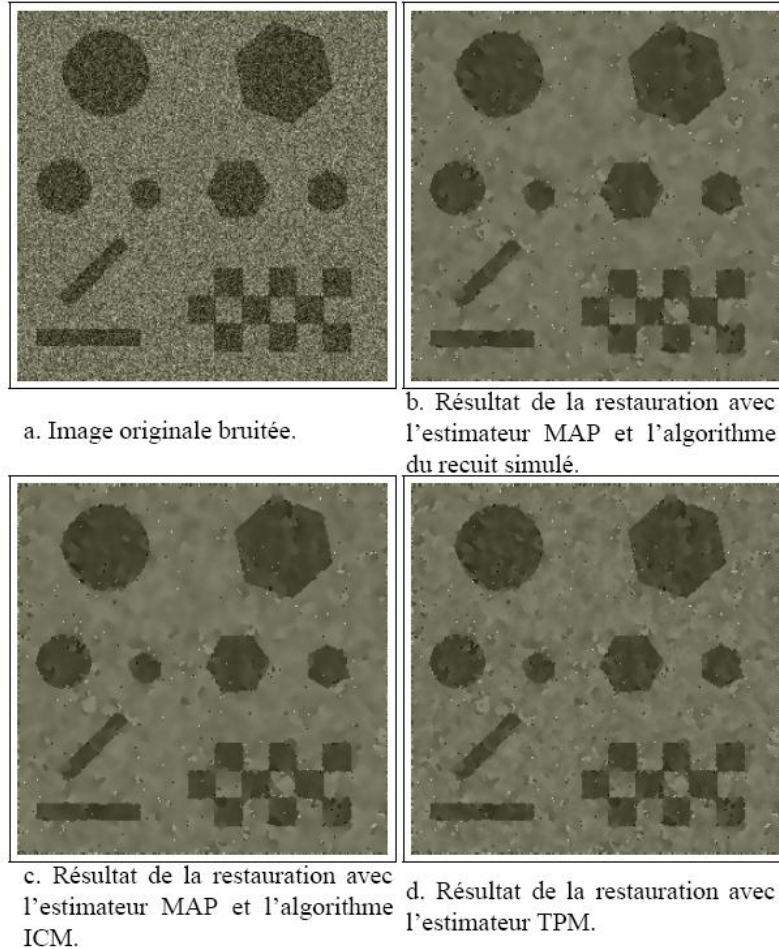
avec

$$\Phi(u) = \frac{u^2}{1 + u^2}$$

L'initialisation est donnée par l'image à restaurer. Pour tous les estimateurs, on réalise 600 itérations. Dans le cas du recuit simulé, la température initiale est de 6.

Visuellement, le meilleur résultat est obtenu par le MAP du recuit simulé. Comme l'image originale est une assez bonne initialisation, il n'y a pas de grandes différences entre les algorithmes de recuit simulé et d'ICM pour l'estimateur du MAP. Ce n'est pas vrai dès que l'initialisation s'écarte du résultat à obtenir et les différences avec le recuit simulé peuvent être très importantes. Par ailleurs, on constate que l'estimateur TPM, qui est par définition plus local, donne un résultat plus bruité et moins régularisé. Cette analyse visuelle est confirmée par l'étude statistique qui peut être effectuée sur des zones homogènes de l'image.





## 9 Estimation de paramètres

### 9.1 Introduction

Le problème de l'estimation de paramètres (encore appelés hyperparamètres dans la littérature), revient très fréquemment en traitement d'image par champs de Markov, et par exemple :

- on se donne une réalisation d'un champ de Markov associé au modèle d'Ising, mais on ne connaît pas ses paramètres.
- On veut généraliser ceci à une image de texture donnée dont on connaît le modèle sous-jacent (par exemple un modèle gaussien en 4-connexité) mais pas les paramètres, qui sont du type : moyenne locale, variance locale, poids de la régularisation locale. Quels sont-ils ? Leur connaissance pourrait bien en effet servir à la classification d'images composées de zones texturées en se

basant sur l'estimation locale de tels paramètres. On classifierait alors selon les valeurs de ces attributs locaux.

- On veut segmenter une image, et pour cela apprendre les paramètres de chaque classe, ainsi que le coefficient optimal du modèle de régularisation adapté à cette tâche. On sait en effet que le résultat d'une segmentation par estimateur MAP par exemple dépend fondamentalement du poids respectif de la régularisation par rapport à celui de l'attache aux données. Il faut donc là aussi estimer ce poids d'une façon optimale dans un sens à définir.

L'ensemble forme un problème réputé difficile. Les solutions les plus utilisées se décomposent en deux classes fondamentales :

- le cas des données dites complètes, correspondant aux deux premiers problèmes cités plus haut : un échantillon d'une distribution de Gibbs est connu. Il s'agit de remonter aux paramètres de cette distribution.
- le cas des données dites incomplètes. Là non seulement le résultat de traitement est inconnu, mais les paramètres sont également à estimer.

## 9.2 Données complètes

Soit  $x$  une configuration observée relativement à une distribution de Gibbs donnée  $P_\theta$ , dont l'énergie associée peut s'écrire sous la forme d'une fonction linéaire d'un paramètre  $\theta$ , par exemple  $U(x) = \theta\Phi(x)$ ,  $\Phi$  étant un potentiel donné. Un principe naturel en vue de la recherche de  $\theta$  est d'écrire la vraisemblance de la donnée  $x$  :

$$L(\theta) = P_\theta(x) = \frac{\exp^{-\theta\Phi(x)}}{Z_\theta}$$

et de chercher par exemple la valeur de l'hyperparamètre  $\hat{\theta}$  maximisant cette vraisemblance. Le problème essentiel est que l'on ne sait en général pas calculer exactement la fonction de partition  $Z_\theta$ . Même pour des modèles aussi simples et fondamentaux que ceux d'Ising et de Potts, le résultat (analytique) est obtenu après des calculs excessivement compliqués. Dans les autres cas, on est amené :

- soit à effectuer des approximations de la fonction de partition globale au moyen des fonctions de partition conditionnelles locales
- soit à employer des algorithmes itératifs (gradient stochastique) à partir de la vraisemblance exacte, mais dont il s'agit alors de prouver la convergence ainsi que le type d'optimum trouvé

### 9.2.1 Méthode des codages

Le principe de la méthode des codages est le suivant. Une fois défini un système de voisinage pour un champ de Markov, on est capable de définir un certain nombre de sous réseaux, chacun formé de sites/pixels indépendants les uns des

autres : chacun de ces sous réseaux est appelé un codage. Par exemple avec un voisinage en 4 connexité il existe deux codages, et 4 codages différents dans le cas de la 8-connexité.

					3	4	3	4	3
2	1	2	1	2	2	1	2	1	2
1	2	1	2	1	3	4	3	4	3
2	1	2	1	2	2	1	2	1	2
					3	4	3	4	3

L'estimation est alors posée dans le cadre de chaque codage. Pour un codage donné, les différents sites/pixels le constituant sont indépendants les uns des autres puisqu'ils ne sont pas des voisins pour le champ de Markov de départ. La probabilité globale d'un codage se trouvera donc être le produit des probabilités individuelles de chacun des sites du codage. Or du fait de la structure de Markov du champ de départ cette probabilité individuelle se trouve être la probabilité conditionnelle locale du site/pixel dans le champ de Markov. Pour un codage  $Cod_n$  donné, on a alors :

$$P_{\theta}(\{X_s = x_s\}_{s \in Cod_n} | \{X_r = x_r\}_{s \notin Cod_n}) = \prod_{s \in Cod_n} P_{\theta}(X_s = x_s | V_s)$$

Etant donnée la structure des probabilités locales, la fonction de vraisemblance devient calculable, puisque les fonctions de partition conditionnelles locales le sont. Dans le cas où la dépendance des énergies locales est linéaire vis à vis des paramètres, la log-vraisemblance associée,

$$\log(P_{\theta}(\{X_s = x_s\}_{s \in Cod_n} | \{X_r = x_r\}_{s \notin Cod_n})) = -\theta \sum_{c \in C} U_c(x) - \sum_{s \in Cod_n} \log(Z_s)$$

est une fonction concave du (des) paramètre(s), comme somme de fonctions concaves. Elle se prête donc bien à la recherche d'un optimum par une méthode classique de type gradient. On peut également montrer qu'il s'agit dans ce cas d'un simple problème de moindres carrés.

### 9.2.2 Pseudo vraisemblance

Expérimentalement, la méthode des codages n'est pas fiable. La méthode du maximum de vraisemblance vrai paraît quant à elle incalculable. Des algorithmes ont cependant été étudiés pour tenter de résoudre ce problème. En fait on utilise souvent une méthode intermédiaire qui a de bonnes propriétés : la méthode du pseudo-maximum de vraisemblance.

Du maximum de vraisemblance vrai, cette méthode conserve l'idée de travailler sur l'ensemble de l'image et non séparément sur des réseaux indépendants. De la méthode de codage, elle garde l'idée de manipuler une fonction de vraisemblance produit des probabilités locales de chacun des sites/pixels. Cette fonction est appelée pseudo-maximum de vraisemblance, et elle s'écrit

$$PL_\theta(X = x) = \prod_{s \in S} P(X_s = x_s | V_s)$$

ou, encore, en considérant le logarithme de cette fonction, et l'expression de la probabilité locale de chaque site/pixel :

$$\log PL_\theta(X = x) = -\theta \sum_{c \in C} U_c(x) - \sum_{s \in S} \log(Z_s)$$

$\log(Z_s)$  devient calculable, puisque relié à la fonction de normalisation de la probabilité conditionnelle locale. Donc un raisonnement identique à celui du paragraphe précédent conduit au fait que la log-pseudo-vraisemblance est une fonction concave des paramètres lorsque l'énergie en dépend de façon linéaire. Les algorithmes usuels de type gradient ou gradient conjugué s'appliquent donc aussi ici naturellement à la recherche de l'optimum unique. Il s'agit alors de qualifier la valeur des paramètres obtenus par cette méthode par rapport à la valeur vraie, et des auteurs ont montré que la méthode de la pseudovraisemblance est consistante et convergente.

### 9.2.3 Algorithme du gradient stochastique

La vraisemblance exacte du paramètre  $\theta$  s'écrit  $L(\theta) = P_\theta(x) = \frac{e^{-\theta\Phi(x)}}{Z_\theta}$ . La valeur de l'hyperparamètre  $\hat{\theta}$  satisfaisant au principe du maximum de vraisemblance doit donc vérifier :

$$\frac{\partial \log P_\theta(x)}{\partial \theta} = -\Phi(x) - \frac{\partial \log Z_\theta}{\partial \theta} = 0$$

On peut montrer que  $\hat{\theta}$  est unique et satisfait à l'équation stochastique

$$\mathbb{E}_{\hat{\theta}}(\Phi) = \Phi(x)$$

La résolution est itérative, fondée sur des méthodes de type Newton-Raphson. L'idée est de remplacer les grandeurs statistiques mises en jeu par leurs valeurs empiriques approchées. Ainsi pour l'espérance du potentiel de régularisation  $\Phi$ , on prend sa moyenne empirique au cours d'une seule itération (valeur effective obtenue !) d'un échantillonneur de Gibbs ou de Metropolis mené avec la valeur

courante du paramètre. Quant à la variance de ce potentiel, on l'estime encore plus crûment par une grandeur positive fixée  $V$ . On peut montrer que le prix à payer pour cette approximation est l'introduction d'un terme correctif supplémentaire en  $1/n + 1$  dans le schéma itératif.

On peut alors montrer que cet algorithme stochastique converge presque sûrement, en termes de probabilité, vers la valeur optimale  $\hat{\theta}$  lorsque  $V$  est suffisamment grand

## 9.3 Données incomplètes

### 9.3.1 Position du problème

Dans ce cas on connaît une observation  $y$ , échantillon de la variable aléatoire  $Y$ , appelée incomplète ou dégradée, car reliée à une scène originale  $x$ , non-dégradée, dont le champ aléatoire correspondant sera noté  $X$ . La relation entre  $y$  et  $x$  s'effectue *via* une loi de probabilité conditionnelle représentant l'attache aux données explicitée la dépendance par rapport à un paramètre  $\lambda > 0$  :

$$P_\lambda(Y = y|X = x) = \frac{e^{-U_\lambda(y|x)}}{Z_\lambda}$$

On suppose également disposer d'une connaissance *a priori* sur la scène à retrouver  $x$ , que ce soit en segmentation ou restauration, *via* la distribution de Gibbs :

$$P_\theta(x) = \frac{e^{-\theta\Phi(x)}}{Z_\theta}$$

et on connaît la forme de la distribution de Gibbs *a posteriori*

$$P_{\theta,\lambda}(X = x|Y = y) = \frac{e^{-U_\lambda(y|x) - \theta\Phi(x)}}{Z_{\theta,\lambda}}$$

L'estimation du meilleur paramètre de régularisation  $\theta$  peut être réalisée, lorsque la loi d'observation (attache aux données) est complètement connue, par une méthode type gradient stochastique généralisé. Lorsque la loi d'observation est paramétrique, il est nécessaire d'estimer les paramètres de l'attache aux données. Nous détaillons ce cas particulier, car important, dans le cas du problème de la segmentation d'images.

### 9.3.2 Estimation des paramètres d'attache aux données

On cherche ici à estimer  $\lambda$ . Dans le cas de la segmentation d'images, supposant par exemple que le niveau de gris (descripteur) de chaque région  $1 \leq i \leq m$  (site)

de l'image suit une loi gaussienne de paramètre  $\mu_i, \sigma_i^2$ , alors  $\lambda = (\mu_i, \sigma_i^2)_{1 \leq i \leq m}$ . Pour l'exemple présenté ici, nous étudierons l'estimation de  $\lambda$  dans le cas gaussien par raison de commodité. La probabilité des observations conditionnellement aux labels s'écrit donc :

$$P_\lambda(Y = y|X = x) = \prod_{s \in S} \frac{1}{\sqrt{2\pi}\sigma_{x_s}} e^{-\frac{(y_s - \mu_{x_s})^2}{2\sigma_{x_s}^2}}$$

La distribution *a posteriori* s'écrit alors :

$$P_{\theta,\lambda}(X = x|Y = y) = \frac{e^{-\mathcal{E}_{\theta,\lambda}(x|y)}}{Z_{\theta,\lambda}}$$

où

$$\mathcal{E}_{\theta,\lambda}(x|y) = \sum_{s \in S} \left( \frac{(y_s - \mu_{x_s})^2}{2\sigma_{x_s}^2} + \log \sigma_{x_s} \right) + \theta \Phi(x)$$

La vraisemblance des paramètres d'attache aux données s'écrit donc

$$L(\lambda) = P_{\theta,\lambda}(Y = y) = \frac{Z_{\theta,\lambda}}{\sqrt{2\pi^{|S|}} Z_\theta}$$

Si on souhaite maximiser  $L(\theta)$  par rapport à l'un des paramètres de  $\lambda$ , on montre en dérivant que

$$(\forall i \in \{1 \dots m\}) \mu_i = \frac{\sum_{s \in S} y_s P_{\theta,\lambda}(X_s = i)}{\sum_{s \in S} P_{\theta,\lambda}(X_s = i)}$$

et

$$(\forall i \in \{1 \dots m\}) \sigma_i = \frac{\sum_{s \in S} (y_s - \mu_i)^2 P_{\theta,\lambda}(X_s = i)}{\sum_{s \in S} P_{\theta,\lambda}(X_s = i)}$$

ces solutions sont complexes, et peuvent avantageusement être remplacées par des formes itératives simples, en utilisant un algorithme de type EM (Expectation-Maximisation).

### 9.3.3 Algorithme EM

L'algorithme EM est une méthode itérative. L'idée fondamentale est de considérer que les données observées ne correspondent qu'à une connaissance partielle des

données complétées, et que la maximisation de la vraisemblance associée à ces données est simple.

Supposons connus les paramètres  $\theta_n$  et  $\lambda_n$  à l'étape  $n$ . Supposons avoir également accès aux statistiques liées à la loi *a posteriori* des données cachées courantes (étape E - Expectation). On définit alors

$$Q(\theta, \lambda, \theta_n, \lambda_n) = \mathbb{E}_{\theta_n, \lambda_n} (\log(P_{\theta, \lambda}(X = x|Y = y)))$$

La partie optimisation (étape M - Maximisation) consiste à rechercher

$$(\theta_{n+1}, \lambda_{n+1}) = \arg \max_{\theta, \lambda} Q(\theta, \lambda, \theta_n, \lambda_n)$$

ce qui peut être réalisé de manière séparée sur  $\theta$  et  $\lambda$ , grâce à la séparabilité de la loi jointe :

$$\begin{cases} \theta_{n+1} = \arg \max_{\theta} \mathbb{E}_{\theta_n, \lambda_n} (\log(P_{\theta}(X = x))) \\ \lambda_{n+1} = \arg \max_{\lambda} \mathbb{E}_{\theta_n, \lambda_n} (\log(P_{\lambda}(Y = y|X = x))) \end{cases}$$

On montre alors que la vraisemblance des paramètres croît en fonction de  $n$ .

Le principe de l'algorithme EM est donc d'alterner une phase E et une phase M, avec à chaque itération  $n$  :

$$(\forall i \in \{1 \dots m\}) \mu_i^{n+1} = \frac{\sum_{s \in S} y_s P_{\theta_n, \lambda_n}(X_s = i)}{\sum_{s \in S} P_{\theta_n, \lambda_n}(X_s = i)}$$

et

$$(\forall i \in \{1 \dots m\}) \sigma_i^{n+1} = \frac{\sum_{s \in S} (y_s - \mu_i^{n+1})^2 P_{\theta_n, \lambda_n}(X_s = i)}{\sum_{s \in S} P_{\theta_n, \lambda_n}(X_s = i)}$$

Ce sont bien les distributions *a posteriori* qui sont mises en jeu dans ces estimations itératives. Elles sont donc à re-estimer à chaque itération  $n$ . Une manière bien naturelle est de remplacer ces probabilités à l'étape courante du processus EM par la fréquence empirique d'apparition des labels lors d'une série d'échantillonnages de la distribution de Gibbs *a posteriori* courante menés à l'aide d'un échantillonneur (Gibbs, Metropolis).

## 10 Processus de bords

Les modèles markoviens peuvent être utilisés, comme nous l'avons indiqué, en restauration d'images : connaissant une réalisation  $y$ , on cherche une solution  $x$

minimisant  $\mathcal{U}(x|y) = U_1(x, y) + \beta U_2(x, y)$ . Le terme de régularisation  $U_2$  permet d'agir sur les dérivées de la solution recherchée. Dans le cas d'un potentiel quadratique, cela entraîne un fort lissage de la solution au détriment des discontinuités naturelles présentes dans l'image (bord des objets).

Il est possible de contraindre la dérivée première de la solution pour préserver ces discontinuités naturelles, et gérer ainsi ces processus de bords.

## 10.1 Processus implicites et explicites

On s'intéresse ici aux cliques d'ordre 2 et donc à la prise en compte de discontinuités naturelles entre deux sites voisins.

Un processus de bord, noté  $\mathcal{B}$  est défini sur une grille duale de celle des sites et est décomposé en deux processus, l'un,  $\mathcal{B}_h$  modélisant les effets de bords horizontaux, l'autre,  $\mathcal{B}_v$  les effets de bords verticaux. Le terme de régularisation s'écrit alors pour un processus booléen ( $b=1$  si discontinuité, 0 sinon) selon le modèle weak membrane :

$$U_2(x, b) = \sum_{(s,t) \in C_v} [(x_s - x_t)^2(1 - b_{st}^v) + \gamma b_{st}^v] + \sum_{(s,t) \in C_h} [(x_s - x_t)^2(1 - b_{st}^h) + \gamma b_{st}^h]$$

En l'absence de discontinuités, on retrouve un potentiel quadratique qui lisse la solution. En présence de discontinuités, la pénalité infligée à la discontinuité vaut  $\gamma$  (qui peut prendre des valeurs différentes suivant  $\mathcal{B}_h$  et  $\mathcal{B}_v$ ).

En réalité, il n'est pas nécessaire d'introduire explicitement un processus de bords et un choix de fonction judicieux peut procurer le même résultat. Si  $\phi^v(s_x, s_t, b_{st}^v) = (x_s - x_t)^2(1 - b_{st}^v) + \gamma b_{st}^v$  (idem pour  $\phi^h(s_x, s_t, b_{st}^h)$ ), on cherche  $(x, b)$  qui minimise  $U_2(x, b)$ , ce qui conduit à :

$$\min_{x,b} U_2(x, b) = \min_x \sum_{(s,t) \in C_v} \psi(x_s - x_t) + \sum_{(s,t) \in C_h} \psi(x_s - x_t)$$

avec

$$\psi(z) = \min(z^2, \gamma)$$

La quadratique est remplacée dans ce modèle par une quadratique tronquée. L'utilisation d'un processus de bords explicite est dans ce cas équivalente à l'utilisation de la fonction  $\psi$  (on parle de processus de bords implicite). La valeur de  $\gamma$  détermine alors à partir de quelle valeur du gradient on introduit une discontinuité dans l'image.

Le choix d'un processus de bords implicite ou explicite va donner lieu à différents algorithmes de minimisation. En effet, le recuit simulé permettant d'accéder au minimum global de l'énergie peut être long et coûteux. Il peut dans de nombreux cas être remplacé par un algorithme déterministe pour accélérer la recherche de la solution.



## 10.2 Algorithmes de minimisation associés

Parmi les algorithmes de minimisation associés aux processus de bords, citons l'algorithme Graduated Non Convexity (GNC) qui minimise le critère sous sa forme implicite. Le principe de l'algorithme consiste à approcher le critère par une fonction convexe qui permet de définir une bonne solution initiale. Puis le critère est graduellement modifié perdant sa propriété de convexité pour se rapprocher du critère initial. A chaque étape, une solution est trouvée de manière déterministe en utilisant pour initialisation la solution donnée par l'étape précédente. Si des preuves de convergence peuvent être obtenues dans certains cas particuliers, cette démarche n'assure cependant pas de trouver le minimum global pour toutes les fonctions d'énergie.

D'autres approches utilisent un recuit par champ moyen MFA (Mean Field Annealing) sur le processus de bords sous sa forme explicite. L'algorithme met en oeuvre une descente de température au cours de laquelle à chaque étape une solution (image restaurée et processus de bords) est estimée au sens du champ moyen. Des expressions explicites sont déterminées pour le processus de bords, tandis que  $x$  est estimée de manière itérative.

## 11 Quelques illustrations

Dans cette section, nous donnons quelques résultats en images des champs de Markov en segmentation et restauration.

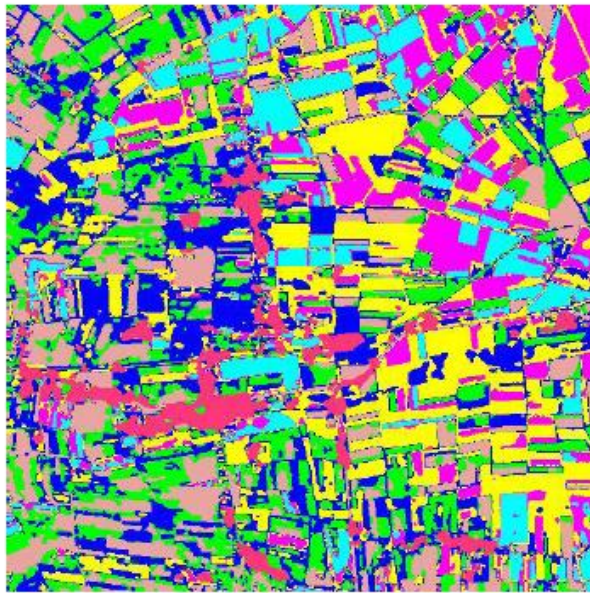
### 11.1 Segmentation

#### 11.1.1 Imagerie radar

L'image suivante (a) (source : INRIA, Sophia-Antipolis) représente une zone rurale prise par le satellite SPOT. Sa segmentation en une dizaine de régions de nature différente est donnée en (b).



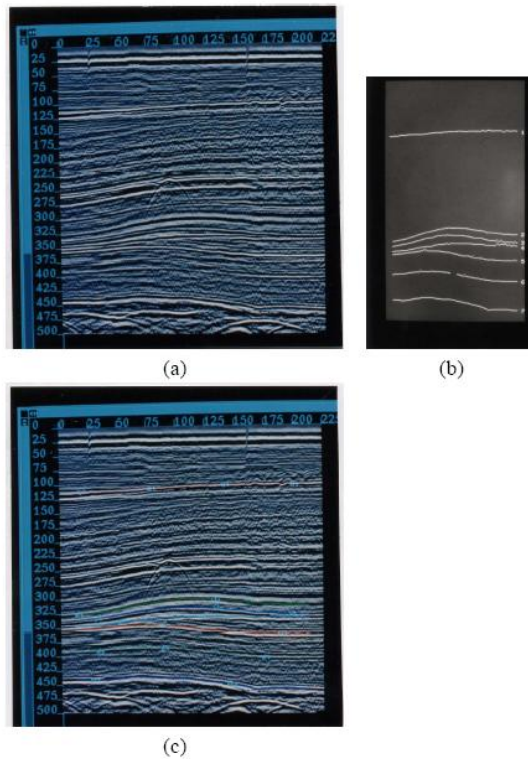
(a)



(b)

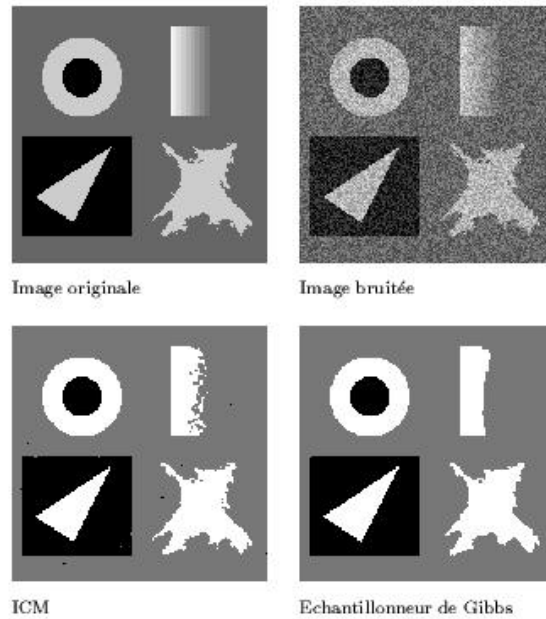
### 11.1.2 Imagerie sismique

L'image suivante (a) est une image de réflexion sismique. Le but est ici de segmenter les couches géologiques, les champs de Markov fournissant les séparateurs optimaux (b) reportés sur l'image originale (c).



### 11.1.3 Imagerie de test

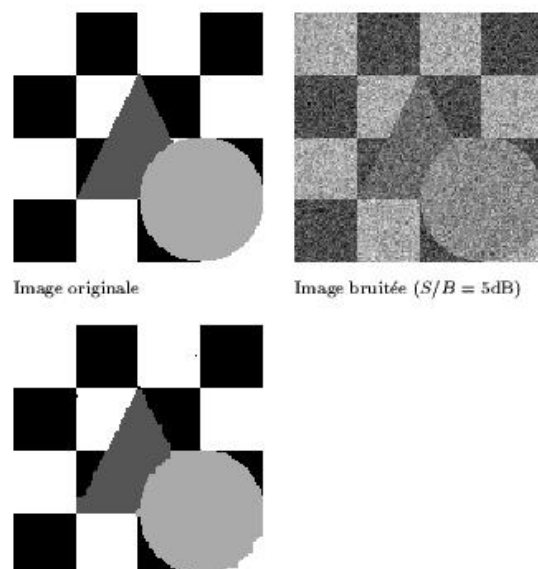
L'exemple quivant montre, sur une image test classique, l'influence de la méthode de recherche des configurations les plus probables qui correspondent à des états d'énergie minimale. A partir d'une image originale bruitée, l'ICM et l'échantillonneur de Gibbs fournissent deux résultats de segmentation bien distincts (problème de minimum local pour ICM)



## 11.2 Restauration

### 11.2.1 Imagerie de test

La première ligne de l'image suivante montre une image originale, bruitée par un bruit gaussien. L'image restaurée est donnée sur la seconde ligne, sans prise en compte des effets de bords .



### 11.2.2 Restauration avec prise en compte de processus de bords

L'image suivante montre la restauration effectuée sur l'image (a) avec (d) et sans (c) prise en compte de processus de bords. L'image (b) est un lissage de (a) qui peut être effectué pour diminuer l'influence du bruit dans l'image

